

Manson_version_1.2.7

F. Fuhrmann, E. McGowan, T. Nolte, A. Stete, R. Trslic, A. Veyhl

04 6 2020

```
knitr::opts_chunk$set(echo = TRUE)
options(max.print = 999999)
```

Vorwort / Zusammenfassung (Abstract)

Verfolgen kriminelle Gruppierungen ihre Ziele systematisch, so spricht man von organisierter Kriminalität. Um der Organisation in der jeweiligen Gruppe gerecht zu werden, braucht es klare Hierarchien und Strukturen. Diese lassen sich in Netzwerken abbilden und analysieren. Der nachfolgende Forschungsbericht untersucht das Beziehungsnetzwerk von Charles Manson, einem Massenmörder aus der USA. Hierbei spielen die Manson-Family, in der er als Anführer agierte, sowie die Opfer eine wichtige Rolle. Es zeigt sich ... erst am Ende wichtigste Ergebnisse verfassen

Keywords: Netzwerkanalyse, Teilnetzwerke, Serienmörder, Kriminalitätsmustertheorie?

.....

Beschreibung des Themenfeldes

Im Jahr 1969 kam es in Kalifornien innerhalb von zwei Tagen zum siebenfachen Mord. Diese sind bis heute unter den Namen LaBianca- und Tate-Morde bekannt. Unter der Führung von Charles Manson wurden die Morde von der Manson-Family, eine sektenähnliche Kommune, begangen. In unserer Netzwerkforschung soll Charles Manson als Ego-Netzwerk untersucht werden und damit seine Verbindungen zur Manson-Family und zu den Opfern. (1967-1969)

.....

Einleitung

In unseren Augen sind die Netzwerke von Kriminellen sehr interessant. Bei Charles Manson ist eine Netzwerkanalyse besonders spannend, da sämtliche Handlungen der Manson-Family von ihm aus gesteuert wurden. Wir untersuchen Charles Manson als Hauptakteur und bilden ein Ego-Netzwerk ab. Dabei setzen wir einen klaren Fokus auf die Beziehungsebene. Wir sind motiviert, die verschiedenen Stärken der Beziehungen zwischen Charles Manson und den Mitgliedern der Manson-Family herauszuarbeiten. Es gilt herauszufinden,

welche Mitglieder besonders eng mit ihm in Verbindung standen, da die Annahme besteht, dass Mitglieder stark durch Manson beeinflusst und durch ihn zum morden animiert wurden. Hierbei ist interessant, ob es auch unter den Mitgliedern zentrale Akteure gab, die eng miteinander verbunden waren. Ebenso möchten wir analysieren, wie Charles Manson und die Manson-Family in Verbindung mit ihren Opfern stand.

.....

Vorarbeiten und vergleichbare Studien

Zeigen Sie kurz auf, ob es vergleichbare empirische Studien gibt oder welche theoretischen Konzepte für die Forschung besonders interessant sind. Ziehen Sie hier ein Zwischenfazit in dem klar die Forschungsfrage der Arbeit benannt wird. Diese können in Form von (falsifizierbaren) Hypothesen oder auch explorativ beschrieben werden.

Es wird auf die Studie “Tactical Social Network Analysis” von Bichler, Lim und Larin (2017) zurückgegriffen, die eine Netzwerkanalyse anhand des Serienmörders Green River durchführte. Für das weitere inhaltliche Verständnis, wie in Kriminalitätsanalysen vorgegangen wird, war das Buch Encyclopedia of Criminological Theory von Cullen und Wilcox (2009) von großem Nutzen.

.....

Forschungsstand

.....

Arbeitshypothesen

Wir gehen von folgenden Arbeitshypothesen aus: Wir gehen davon aus, dass Beziehungen zwischen den Opfern der Tatemorden und La Bianca gab. Wir gehen davon aus, dass die Mansonfamilie ein sehr enges Verhältnis hatte. Wir gehen davon aus, dass Charles Manson der Alleinige Anführer der Mansonfamilie war.

.....

Datenerhebung

.....

Zugang

Die Materialien für unsere Netzwerkanalyse haben wir breit gefächert ausgewählt, sodass wir eine möglichst große Überschneidung der Ergebnisse erzielen können. Dies gewährleistet eine Kontinuität in der subjektiv dokumentierten Thematik.

.....

Bereinigung

Der Datensatz ist unter (<https://github.com/thomas5nolte/Manson>) verfügbar.

.....

Codebuch

Das Codebuch (<https://raw.githubusercontent.com/thomas5nolte/Manson/master/Codebuch.md>) beschreibt die Variablen, Relationen und Gewichte des Netzwerks und ist ebenfalls auf Github hinterlegt.

.....

Vorbereitung der IDE

In der ersten Chunkzeile können verschiedenen Befehle eingefügt werden, ob der Chunk, wie der Chunk ausgeführt werden soll. Die Installationspackages sind mit dem Befehl `eval= FALSE` gekennzeichnet. Dies bedeutet, dass der Chunk nicht ausgeführt wird. Sollten die Packages installiert werden müssen, so muss lediglich das “FALSE” mit einem “TRUE” ersetzt werden.

```
library(igraph)
library(igraphdata)
library(ggraph)
library(graphlayouts)
library(dplyr)
library(knitr)
```

Gesamtnetzwerk

.....

Einlesen des Datensatzes & Erstellung Igraph-Objekt

Die Edge- und Nodelisten werden über `read.csv` von Github geladen und mit dem Packet `igraph` zu einem Objekt zusammengeführt.

```
el_manson <-
  read.csv(
    "https://raw.githubusercontent.com/thomas5nolte/Manson/master/el_manson.csv",
    header = T,
    as.is = T,
    sep = ",",
  )
nl_manson <-
  read.csv(
    "https://raw.githubusercontent.com/thomas5nolte/Manson/master/nl_manson.csv",
    header = T,
    as.is = T,
    sep = ",",
  )
```

```

# Matrix erstellen
manson_matrix <- as.matrix(el_manson)
# Die Daten werden im Dataframe gespeichert
manson <-
  graph_from_data_frame(d = manson_matrix,
                        vertices = nl_manson,
                        directed = T)

```

Das Gesamtnetzwerk umfasst 195 Knoten und 634 Beziehungen (siehe igraph-Objekt). Es ist gerichtet und gewichtet.

```

el_hollywood <-
  read.csv(
    "https://raw.githubusercontent.com/thomas5nolte/Manson/master/el_film.csv",
    header = T,
    as.is = T,
    sep = ",",
  )
nl_hollywood <-
  read.csv(
    "https://raw.githubusercontent.com/thomas5nolte/Manson/master/nl_film.csv",
    header = T,
    as.is = T,
    sep = ",",
  )
# Matrix erstellen
hollywood_matrix <- as.matrix(el_hollywood)
# Die Daten werden im Dataframe gespeichert
hollywood <-
  graph_from_data_frame(d = hollywood_matrix,
                        vertices = nl_hollywood,
                        directed = T)

```

Das Gesamtnetzwerk umfasst 23 Knoten und 106 Beziehungen (siehe igraph-Objekt). Es ist gerichtet und gewichtet.

.....

Werte Überprüfen

Da es zu Beginn der Arbeiten mit dem igraph-Objekt zu Unstimmigkeiten zwischen der Darstellung und den hinterlegten Daten in der Edge- und Nodelist gab, mussten wir im ersten Schritt die Daten, die R-Studio in der Matrix speichert überprüfen.

```

list.vertex.attributes(manson)
list.edge.attributes(manson)

list.vertex.attributes(hollywood)
list.edge.attributes(hollywood)

```

Die Kategorie des Objektes *manson* "X" sind von uns getroffene Bearbeitungshinweise, welche bei einzelnen Knoten und Kanten ausgefüllt sind. Diese sind für das Plotten oder Auswerten des Netzwerkes irrelevant, deshalb werden sie im nächsten Schritt herausgelöscht.

```
manson <- delete_edge_attr(manson, "X")
manson <- delete_vertex_attr(manson, "X")
```

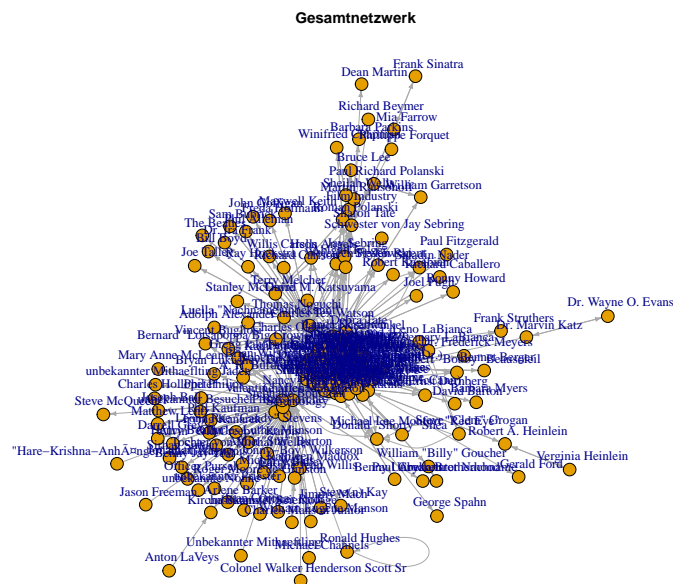
Desweiteren überprüfen wir die hinterlegten Nodedaten. Dazu muss im Chunk include und message auf “TRUE” gesetzt werden.

.....

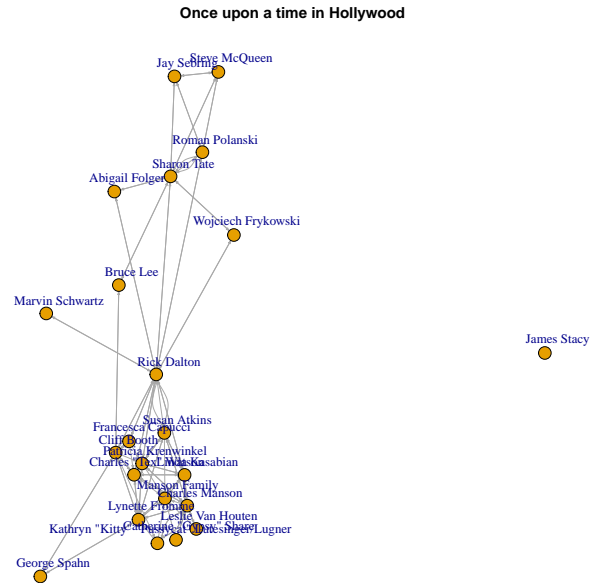
Plotten der Rohdaten

In diesem Schritt plotten wir das Gesamtnetzwerk um einen Eindruck von der Größe des Netzwerks zu gewinnen.

```
plot(manson, aps=0, main = "Gesamtnetzwerk", vertex.size = 5, vertex.label.dist=1, edge.arrow.size=.4)
```



```
plot(vertex.label.dist = 1, hollywood, edge.arrow.size = 0.2, main = "Once upon a time in Hollywood", v
```



Erkennbar ist, dass der Film nur einen kleinen Teil der Realität widerspiegelt.

.....

Analyse der Netzwerkdaten

Netzwerkmaße im Überblick

Bei der Untersuchung des Gesamtnetzwerks werden *generelle Netzwerkmaße* berechnet. Die wichtigsten sind * Dichte (density) * Durchmesser (diameter) * Pfaddistanz (path_distance)

Positionale Maße geben eine Auskunft über die Bedeutung der einzelnen Knoten innerhalb des Netzwerks. Die wichtigsten postionalen oder akteursbezogenen Maße sind * Degree (indegree/outdegree) * Closeness * Betweenness

.....

Zentralitätsmaße

```
# Berechnung der Dichte des Gesamtnetzwerks
edge_density(manson)
```

```
## [1] 0.01675919
```

```
# Berechnung der Dichte des Filmnetzwerks
edge_density(hollywood)
```

```
## [1] 0.2094862
```

Im **Gesamtnetzwerk** unserer Erhebung sind nur **1,6 %** der Beziehungen zwischen den Knoten realisiert. Dies bedeutet, dass viele Knoten untereinander nicht in Verbindung stehen. Gewisse Cluster können aber dennoch eine weitaus höhere Dichte aufweisen, deswegen ist es wichtig die Teilnetzwerke genauer zu betrachten. Im **Netzwerk**, dass im **Film “Once upon a time in Hollywood”** dargestellt wird, ist die liegt die Dichte bei **20,9%**. Dies liegt daran, dass alle “unwichtigen” Charaktere aus dem Film herausgelassen wurden.

```
# https://igraph.org/r/doc/diameter.html
```

```
# Was ist der längste Pfad in einem Netzwerk?  
get.diameter(manson)
```

```
## + 5/195 vertices, named, from da31853:  
## [1] Stephane Bourgoin Charles Manson      Terry Melcher      Roman Polanski  
## [5] Film Industry
```

```
# Welche Knoten sind am weitesten voneinander entfernt?  
farthest_vertices(manson)
```

```
## $vertices  
## + 2/195 vertices, named, from da31853:  
## [1] Stephane Bourgoin Film Industry  
##  
## $distance  
## [1] 201
```

Der längste Pfad durch das Netzwerk ist: “Stephane Bourgoin” “Charles Manson” “Terry Melcher” “Roman Polanski” “Film Industry” Dementsprechend sind Stephane Bourgoin und die Film Industry am weitesten voneinander entfernt, mit einer Distanz von 201 (Noch herausuchen, ob das weit oder nicht ist)

```
# https://igraph.org/r/doc/diameter.html
```

```
# Was ist der längste Pfad in einem Netzwerk?  
get.diameter(hollywood)
```

```
## + 5/23 vertices, named, from da6b0bf:  
## [1] Jay Sebring      Roman Polanski      Rick Dalton  
## [4] Patricia Krenwinkel Manson Family
```

```
# Welche Knoten sind am weitesten voneinander entfernt?  
farthest_vertices(hollywood)
```

```
## $vertices  
## + 2/23 vertices, named, from da6b0bf:  
## [1] Jay Sebring      Manson Family  
##  
## $distance  
## [1] 103
```

Der längste Pfad durch das Netzwerk ist: “Jay Sebring” -> “Roman Polanski” -> “Rick Dalton” -> “Patricia Krenwinkel” -> “Manson Family”

Dementsprechend sind Jay Sebring und die Manson Family am weitesten voneinander entfernt, mit einer Distanz von 103 (Noch herausuchen, ob das weit oder nicht ist).

#Indegree = Anzahl der Kanten, die auf einen Knoten eingehen. (Popularität)
`degree(manson, mode="in")`

##	Allen Delisle	Alan Leroy Springer
##	0	1
##	Barbara Hoyt	Beach Boys
##	4	4
##	William Joseph "Bill" Vance	Robert "Bobby" Beausoleil
##	5	2
##	Bruce Davis	Bruce Hall
##	4	1
##	Bryan Lukashevsky	Catherine Gillies
##	1	6
##	Carol Loveless	Catherine "Gypsy" Share
##	0	5
##	Charles Allen Beard	Charlee Griffin
##	0	0
##	Charles Manson	Charles Tex Watson
##	96	39
##	Claudia Smith	Colleen Sinclair
##	0	0
##	David Baker	Danny DeCarlo
##	0	3
##	David Hannum	Dianne Lake
##	0	5
##	Diane Von Ahn	Ella Jo Bailey
##	2	4
##	Harold Irving True	Jack Gordon
##	1	0
##	Johnny Harold Swartz	Juan Flynn
##	0	0
##	Kathryn "Kitty" Lutesinger/Lugner	Kenneth Bell
##	3	0
##	Larry Bailey	Larry Craven
##	1	0
##	Laura Shepard	Leslie Van Houten
##	0	11
##	Linda Kasabian	Lynette Fromme
##	15	2
##	Maria Alonzo	Marcus Arneson
##	3	0
##	Mary Brunner	Madeleine Joan Cottage
##	5	5
##	Patricia Krenwinkel	Paul Alan Watkins
##	12	3
##	Phil Philips	Raymond Petrizzo
##	0	0
##	Randy Starr	Robert Murray
##	2	0
##	Robert Reinhard	Ruth Gordon
##	0	0
##	Ruth Ann Moorehouse	Sandra Good
##	7	5

##	Sherry Ann Cooper	Steve "Clem" Grogan
##	4	0
##	Stephen Palazzo	Stephanie Rowe
##	0	0
##	Stephanie Schram	Susan Atkins
##	0	15
##	Susan Bartell	Thomas Galella
##	3	0
##	Thomas "TJ" Walleman	Vern Plumlee
##	3	0
##	Aryan Brotherhood	Abigail Folger
##	2	8
##	Adolph Alexander	Afton "Star" Burton
##	2	1
##	A. H. Burdick	Alvin "Old Creepy" Karpis
##	2	1
##	Alan Rose	Anton LaVeys
##	0	0
##	Arlene Barker	Barbara Myers
##	1	1
##	Barbara Parkins	Barbara Rosenberg
##	1	3
##	Bennet Berger	Bernard "Lotsapoppa/Big Crow" Crowe
##	1	4
##	Benny Jay Teal	Benny Unbekannter Nachname
##	1	1
##	Bill Boyd	Bruce Lee
##	1	2
##	Brooks Poston	Brian Wilson
##	0	2
##	Charles Hollopeter	Charles Manson Junior
##	1	2
##	Charles Luther Manson	Charles Older
##	1	6
##	Colonel Walker Henderson Scott Sr	David Barton
##	1	1
##	Darrell Grey	David M. Katsuyama
##	1	5
##	David Smith (Dr.)	Dean Martin
##	5	1
##	Dean Moorhouse	Debra Tate
##	2	2
##	Dennis Wilson	Donald "Shorty" Shea
##	8	7
##	Edward Davis	Dr. Ernst Dernberg
##	0	2
##	Film Industry	Freda Hofmann
##	4	1
##	Dr. Frederick Meyers	Frank Sinatra
##	1	1
##	Frank Struthers	Gary Allen Hinman
##	2	6
##	Gerald Ford	George Spahn
##	1	1

##	Gregg Jakobson	Hells Angels
##	2	2
##	"Hare-Krishna-Anh�nger" aus Gefaengnis	Henry Beatly
##	0	1
##	Dr. Ira Frank	Irving Kanarek
##	1	1
##	Jason Freeman	Jay Sebring
##	0	11
##	Jimmy Mach	Joseph Ball
##	1	1
##	John Goffigan	John "Zero" Haught
##	1	5
##	Dr. Joel Hochmann	Joel Pugh
##	1	1
##	Joan Svelte	Joe Talley
##	1	1
##	Jonathan Wayne "Jonny-Boy" Wilkerson	Juan Corona
##	1	1
##	Kathleen Maddox	Kirche Satans
##	3	2
##	Leno LaBianca	Leona Rae "Candy" Stevens
##	7	2
##	Luella "Nachname unbekannt"	Dr. Marvin Katz
##	3	1
##	Maxwell Keith	Matthew Lentz
##	1	1
##	Mary Anne McLean	Martin Ransohoff
##	1	1
##	Manson Family	Michael Channels
##	72	0
##	Mia Farrow	Michael Lee Monfort "Red Eye"
##	2	2
##	Michal Welles	Nancy Pitman (alias Brenda McCann)
##	1	3
##	Officer Pursel	Officer Rudi
##	1	1
##	Paul Crockett	Paul Fitzgerald
##	1	1
##	Paul Richard Polanski	Phil Alleman
##	3	1
##	Phillippe Forquet	Phil Kaufman
##	1	1
##	Prozesskirche	Ray Hoekstra
##	1	1
##	Richard Beymer	Richard Caballero
##	1	1
##	Richard Carlson	Robert Kenneth "Bobby" Beusoleil
##	1	4
##	Robert A. Heinlein	Ronny Howard
##	2	1
##	Ronald Hughes	Robert Kasabian
##	2	1
##	Rosemary LaBianca	Ronald Markman
##	7	0

##	Roger Moore de Gimston	Roman Polanski
##	1	8
##	Roger Smith	Rosalie Jean Willis
##	2	2
##	Sam Bubrick	Saladin Nader
##	1	2
##	Schwester von Jay Sebring	Sharon Tate
##	2	14
##	Sheilah Wells	Sirhan Sirhan
##	1	1
##	Scientology	Stephane Bourgoin
##	1	1
##	Steve(n) Kay	Stanley McGuire
##	1	1
##	Steve McQueen	Steven Parent
##	1	5
##	The Beatles	Terry Melcher
##	1	5
##	Thomas Noguchi	Tochter von Michal Welles
##	6	1
##	The Straight Satans	unbekannter Besucher "Rex"
##	6	1
##	unbekannter Mithaeftling Jaden	Unbekannter Mithaeftling
##	1	0
##	unbekannte Nonne	unbekannter Priester
##	1	1
##	unbekannter Scientologe	Valentine Michael Manson
##	1	3
##	Vincent Bugliosi	Verginia Heinlein
##	2	1
##	Dr. Wayne O. Evans	Willis Carson
##	1	1
##	Winifried Chapman	William Garretson
##	2	2
##	William "Billy" Goucher	William Eugene Manson
##	1	1
##	Wojciech Frykowski	
##	8	

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

```
## $res
## [1] 0 1 4 4 5 2 4 1 1 6 0 5 0 0 96 39 0 0 0 3 0 5 2 4 1
## [26] 0 0 0 3 0 1 0 0 11 15 2 3 0 5 5 12 3 0 0 2 0 0 0 7 5
## [51] 4 0 0 0 0 15 3 0 3 0 2 8 2 1 2 1 0 0 1 1 1 3 1 4 1
## [76] 1 1 2 0 2 1 2 1 6 1 1 1 5 5 1 2 2 8 7 0 2 4 1 1 1
## [101] 2 6 1 1 2 2 0 1 1 1 0 11 1 1 1 5 1 1 1 1 1 3 2 7
## [126] 2 3 1 1 1 1 1 72 0 2 2 1 3 1 1 1 1 3 1 1 1 1 1 1
## [151] 1 4 2 1 2 1 7 0 1 8 2 2 1 2 2 14 1 1 1 1 1 1 5 1
## [176] 5 6 1 6 1 1 0 1 1 1 3 2 1 1 1 2 2 1 1 8
##
## $centralization
## [1] 0.4780862
##
```

```
## $theoretical_max
## [1] 37830
```

```
#Outdegree = Anzahl der Kanten, die ein Knoten zu anderen Knoten hat. (Aktivität)
degree(manson, mode="out")
```

##	Allen Delisle	Alan Leroy Springer
##	1	3
##	Barbara Hoyt	Beach Boys
##	5	1
##	William Joseph "Bill" Vance	Robert "Bobby" Beausoleil
##	7	1
##	Bruce Davis	Bruce Hall
##	6	2
##	Bryan Lukashevsky	Catherine Gillies
##	2	8
##	Carol Loveless	Catherine "Gypsy" Share
##	1	6
##	Charles Allen Beard	Charlee Griffin
##	1	1
##	Charles Manson	Charles Tex Watson
##	102	44
##	Claudia Smith	Colleen Sinclair
##	1	1
##	David Baker	Danny DeCarlo
##	1	6
##	David Hannum	Dianne Lake
##	1	6
##	Diane Von Ahn	Ella Jo Bailey
##	3	6
##	Harold Irving True	Jack Gordon
##	2	1
##	Johnny Harold Swartz	Juan Flynn
##	1	1
##	Kathryn "Kitty" Lutesinger/Lugner	Kenneth Bell
##	5	1
##	Larry Bailey	Larry Craven
##	1	1
##	Laura Shepard	Leslie Van Houten
##	1	14
##	Linda Kasabian	Lynette Fromme
##	21	5
##	Maria Alonzo	Marcus Arneson
##	4	1
##	Mary Brunner	Madeleine Joan Cottage
##	6	6
##	Patricia Krenwinkel	Paul Alan Watkins
##	21	5
##	Phil Philips	Raymond Petrizzo
##	2	1
##	Randy Starr	Robert Murray
##	3	1
##	Robert Reinhard	Ruth Gordon
##	1	1

##	Ruth Ann Moorehouse	Sandra Good
##	8	6
##	Sherry Ann Cooper	Steve "Clem" Grogan
##	5	4
##	Stephen Palazzo	Stephanie Rowe
##	1	1
##	Stephanie Schram	Susan Atkins
##	1	23
##	Susan Bartell	Thomas Galella
##	4	1
##	Thomas "TJ" Walleman	Vern Plumlee
##	5	1
##	Aryan Brotherhood	Abigail Folger
##	0	4
##	Adolph Alexander	Afton "Star" Burton
##	2	1
##	A. H. Burdick	Alvin "Old Creepy" Karpis
##	2	1
##	Alan Rose	Anton LaVeys
##	1	1
##	Arlene Barker	Barbara Myers
##	1	1
##	Barbara Parkins	Barbara Rosenberg
##	0	3
##	Bennet Berger	Bernard "Lotsapoppa/Big Crow" Crowe
##	1	3
##	Benny Jay Teal	Benny Unbekannter Nachname
##	1	2
##	Bill Boyd	Bruce Lee
##	1	1
##	Brooks Poston	Brian Wilson
##	1	3
##	Charles Hollopeter	Charles Manson Junior
##	1	1
##	Charles Luther Manson	Charles Older
##	2	6
##	Colonel Walker Henderson Scott Sr	David Barton
##	0	1
##	Darrell Grey	David M. Katsuyama
##	1	6
##	David Smith (Dr.)	Dean Martin
##	5	0
##	Dean Moorhouse	Debra Tate
##	3	3
##	Dennis Wilson	Donald "Shorty" Shea
##	9	2
##	Edward Davis	Dr. Ernst Dernberg
##	1	2
##	Film Industry	Freda Hofmann
##	0	1
##	Dr. Frederick Meyers	Frank Sinatra
##	1	1
##	Frank Struthers	Gary Allen Hinman
##	0	3

##	Gerald Ford	George Spahn
##	0	0
##	Gregg Jakobson	Hells Angels
##	3	2
##	"Hare-Krishna-Anhänger" aus Gefaengnis	Henry Beatly
##	1	1
##	Dr. Ira Frank	Irving Kanarek
##	1	1
##	Jason Freeman	Jay Sebring
##	1	7
##	Jimmy Mach	Joseph Ball
##	1	1
##	John Goffigan	John "Zero" Haught
##	1	6
##	Dr. Joel Hochmann	Joel Pugh
##	2	1
##	Joan Svelte	Joe Talley
##	2	1
##	Jonathan Wayne "Jonny-Boy" Wilkerson	Juan Corona
##	1	1
##	Kathleen Maddox	Kirche Satans
##	4	0
##	Leno LaBianca	Leona Rae "Candy" Stevens
##	3	1
##	Luella "Nachname unbekannt"	Dr. Marvin Katz
##	3	2
##	Maxwell Keith	Matthew Lentz
##	1	1
##	Mary Anne McLean	Martin Ransohoff
##	1	1
##	Manson Family	Michael Channels
##	2	1
##	Mia Farrow	Michael Lee Monfort "Red Eye"
##	2	2
##	Michal Welles	Nancy Pitman (alias Brenda McCann)
##	1	4
##	Officer Pursel	Officer Rudi
##	1	1
##	Paul Crockett	Paul Fitzgerald
##	1	1
##	Paul Richard Polanski	Phil Alleman
##	3	1
##	Phillippe Forquet	Phil Kaufman
##	0	1
##	Prozesskirche	Ray Hoekstra
##	0	1
##	Richard Beymer	Richard Caballero
##	0	1
##	Richard Carlson	Robert Kenneth "Bobby" Beusoleil
##	1	4
##	Robert A. Heinlein	Ronny Howard
##	2	1
##	Ronald Hughes	Robert Kasabian
##	1	1

##	Rosemary LaBianca	Ronald Markman
##	2	1
##	Roger Moore de Gimston	Roman Polanski
##	1	10
##	Roger Smith	Rosalie Jean Willis
##	3	2
##	Sam Bubrick	Saladin Nader
##	1	1
##	Schwester von Jay Sebring	Sharon Tate
##	2	17
##	Sheilah Wells	Sirhan Sirhan
##	1	1
##	Scientology	Stephane Bourgoin
##	0	1
##	Steve(n) Kay	Stanley McGuire
##	1	2
##	Steve McQueen	Steven Parent
##	0	1
##	The Beatles	Terry Melcher
##	1	5
##	Thomas Noguchi	Tochter von Michal Welles
##	6	1
##	The Straight Satans	unbekannter Besucher "Rex"
##	1	1
##	unbekannter Mithaeftling Jaden	Unbekannter Mithaeftling
##	1	1
##	unbekannte Nonne	unbekannter Priester
##	1	1
##	unbekannter Scientologe	Valentine Michael Manson
##	1	4
##	Vincent Bugliosi	Verginia Heinlein
##	2	1
##	Dr. Wayne O. Evans	Willis Carson
##	1	1
##	Winifried Chapman	William Garretson
##	2	2
##	William "Billy" Goucher	William Eugene Manson
##	2	2
##	Wojciech Frykowski	
##	5	

```
centr_degree(manson, mode="out", normalized=T)
```

```
## $res
## [1] 1 3 5 1 7 1 6 2 2 8 1 6 1 1 102 44 1 1
## [19] 1 6 1 6 3 6 2 1 1 1 5 1 1 1 1 14 21 5
## [37] 4 1 6 6 21 5 2 1 3 1 1 1 8 6 5 4 1 1
## [55] 1 23 4 1 5 1 0 4 2 1 2 1 1 1 1 1 0 3
## [73] 1 3 1 2 1 1 1 3 1 1 2 6 0 1 1 6 5 0
## [91] 3 3 9 2 1 2 0 1 1 1 0 3 0 0 3 2 1 1
## [109] 1 1 1 7 1 1 1 6 2 1 2 1 1 1 4 0 3 1
## [127] 3 2 1 1 1 1 2 1 2 2 1 4 1 1 1 1 3 1
## [145] 0 1 0 1 0 1 1 4 2 1 1 1 2 1 1 10 3 2
## [163] 1 1 2 17 1 1 0 1 1 2 0 1 1 5 6 1 1 1
```

```
## [181] 1 1 1 1 1 4 2 1 1 1 2 2 2 2 5
##
## $centralization
## [1] 0.509014
##
## $theoretical_max
## [1] 37830
```

#Components zeigt die Anzahl der Teilnetzwerke und deren Größe
`components(manson)`

```
## $membership
##
## Allen Delisle Alan Leroy Springer
## 1 1
## Barbara Hoyt Beach Boys
## 1 1
## William Joseph "Bill" Vance Robert "Bobby" Beausoleil
## 1 1
## Bruce Davis Bruce Hall
## 1 1
## Bryan Lukashevsky Catherine Gillies
## 1 1
## Carol Loveless Catherine "Gypsy" Share
## 1 1
## Charles Allen Beard Charlee Griffin
## 1 1
## Charles Manson Charles Tex Watson
## 1 1
## Claudia Smith Colleen Sinclair
## 1 1
## David Baker Danny DeCarlo
## 1 1
## David Hannum Dianne Lake
## 1 1
## Diane Von Ahn Ella Jo Bailey
## 1 1
## Harold Irving True Jack Gordon
## 1 1
## Johnny Harold Swartz Juan Flynn
## 1 1
## Kathryn "Kitty" Lutesinger/Lugner Kenneth Bell
## 1 1
## Larry Bailey Larry Craven
## 1 1
## Laura Shepard Leslie Van Houten
## 1 1
## Linda Kasabian Lynette Fromme
## 1 1
## Maria Alonzo Marcus Arneson
## 1 1
## Mary Brunner Madeleine Joan Cottage
## 1 1
## Patricia Krenwinkel Paul Alan Watkins
## 1 1
```


##	Phil Philips	Raymond Petrizzo
##	1	1
##	Randy Starr	Robert Murray
##	1	1
##	Robert Reinhard	Ruth Gordon
##	1	1
##	Ruth Ann Moorehouse	Sandra Good
##	1	1
##	Sherry Ann Cooper	Steve "Clem" Grogan
##	1	1
##	Stephen Palazzo	Stephanie Rowe
##	1	1
##	Stephanie Schram	Susan Atkins
##	1	1
##	Susan Bartell	Thomas Galella
##	1	1
##	Thomas "TJ" Walleman	Vern Plumlee
##	1	1
##	Aryan Brotherhood	Abigail Folger
##	1	1
##	Adolph Alexander	Afton "Star" Burton
##	1	1
##	A. H. Burdick	Alvin "Old Creepy" Karpis
##	1	1
##	Alan Rose	Anton LaVeys
##	1	1
##	Arlene Barker	Barbara Myers
##	1	1
##	Barbara Parkins	Barbara Rosenberg
##	1	1
##	Bennet Berger	Bernard "Lotsapoppa/Big Crow" Crowe
##	1	1
##	Benny Jay Teal	Benny Unbekannter Nachname
##	1	1
##	Bill Boyd	Bruce Lee
##	1	1
##	Brooks Poston	Brian Wilson
##	1	1
##	Charles Hollopeter	Charles Manson Junior
##	1	1
##	Charles Luther Manson	Charles Older
##	1	1
##	Colonel Walker Henderson Scott Sr	David Barton
##	1	1
##	Darrell Grey	David M. Katsuyama
##	1	1
##	David Smith (Dr.)	Dean Martin
##	1	1
##	Dean Moorhouse	Debra Tate
##	1	1
##	Dennis Wilson	Donald "Shorty" Shea
##	1	1
##	Edward Davis	Dr. Ernst Dernberg
##	1	1

##	Film Industry	Freda Hofmann
##	1	1
##	Dr. Frederick Meyers	Frank Sinatra
##	1	1
##	Frank Struthers	Gary Allen Hinman
##	1	1
##	Gerald Ford	George Spahn
##	1	1
##	Gregg Jakobson	Hells Angels
##	1	1
##	"Hare-Krishna-Anhänger" aus Gefaengnis	Henry Beatly
##	1	1
##	Dr. Ira Frank	Irving Kanarek
##	1	1
##	Jason Freeman	Jay Sebring
##	1	1
##	Jimmy Mach	Joseph Ball
##	1	1
##	John Goffigan	John "Zero" Haught
##	1	1
##	Dr. Joel Hochmann	Joel Pugh
##	1	1
##	Joan Svelte	Joe Talley
##	1	1
##	Jonathan Wayne "Jonny-Boy" Wilkerson	Juan Corona
##	1	1
##	Kathleen Maddox	Kirche Satans
##	1	1
##	Leno LaBianca	Leona Rae "Candy" Stevens
##	1	1
##	Luella "Nachname unbekannt"	Dr. Marvin Katz
##	1	1
##	Maxwell Keith	Matthew Lentz
##	1	1
##	Mary Anne McLean	Martin Ransohoff
##	1	1
##	Manson Family	Michael Channels
##	1	1
##	Mia Farrow	Michael Lee Monfort "Red Eye"
##	1	1
##	Michal Welles	Nancy Pitman (alias Brenda McCann)
##	1	1
##	Officer Pursel	Officer Rudi
##	1	1
##	Paul Crockett	Paul Fitzgerald
##	1	1
##	Paul Richard Polanski	Phil Alleman
##	1	1
##	Phillippe Forquet	Phil Kaufman
##	1	1
##	Prozesskirche	Ray Hoekstra
##	1	1
##	Richard Beymer	Richard Caballero
##	1	1

##	Richard Carlson	Robert Kenneth "Bobby" Beusoleil
##	1	1
##	Robert A. Heinlein	Ronny Howard
##	1	1
##	Ronald Hughes	Robert Kasabian
##	1	1
##	Rosemary LaBianca	Ronald Markman
##	1	1
##	Roger Moore de Gimston	Roman Polanski
##	1	1
##	Roger Smith	Rosalie Jean Willis
##	1	1
##	Sam Bublick	Saladin Nader
##	1	1
##	Schwester von Jay Sebring	Sharon Tate
##	1	1
##	Sheilah Wells	Sirhan Sirhan
##	1	1
##	Scientology	Stephane Bourgoin
##	1	1
##	Steve(n) Kay	Stanley McGuire
##	1	1
##	Steve McQueen	Steven Parent
##	1	1
##	The Beatles	Terry Melcher
##	1	1
##	Thomas Noguchi	Tochter von Michal Welles
##	1	1
##	The Straight Satans	unbekannter Besucher "Rex"
##	1	1
##	unbekannter Mithaeftling Jaden	Unbekannter Mithaeftling
##	1	1
##	unbekannte Nonne	unbekannter Priester
##	1	1
##	unbekannter Scientologe	Valentine Michael Manson
##	1	1
##	Vincent Bugliosi	Verginia Heinlein
##	1	1
##	Dr. Wayne O. Evans	Willis Carson
##	1	1
##	Winifried Chapman	William Garretson
##	1	1
##	William "Billy" Goucher	William Eugene Manson
##	1	1
##	Wojciech Frykowski	
##	1	
##		
##	\$csize	
##	[1] 195	
##		
##	\$no	
##	[1] 1	

```
#Gibt die durchschnittliche Lnger, der Verbindung zwischen zwei Knoten aus
mean_distance(manson)
```

```
## [1] 2.823062
```

Indegreewerte (Popularitt) sind: Charles Manson 96 Manson Family 72 Charles Tex Watson 39 Linda Kasabian 15 Susan Atkins 15 Sharon Tate 14 Patricia Krenwinkel 12 Leslie Van Houten 11 Jay Sebring 11

Outdegreewerte (Aktivitt) sind: Charles Manson 102 Charles Tex Watson 44 Susan Atkins 23 Linda Kasabian 21 Patricia Krenwinkel 21 Sharon Tate 17 Leslie Van Houten 14 Roman Polanski 10

Das Gesamtnetzwerk hat nur **eine Komponente**. Das bedeutet, dass alle Knoten in irgendeiner Form miteinander verbunden sind. Wobei die durchschnittliche Lnge die es braucht, um zwei Knoten miteinander zu Verbinden **2.82 Schritte** sind.

```
degree_hollywood <- degree(hollywood, mode="IN") #Hier lsst sich der Knoten mit den meisten Verbindung
degree_hollywood
```

```
##                Abigail Folger
##                2
##                Bruce Lee
##                2
##          Catherine "Gypsy" Share
##                1
##                Charles Manson
##                9
##          Charles "Tex" Watson
##                8
##                Cliff Booth
##                6
##          Francesca Capucci
##                2
##                George Spahn
##                2
##                Jay Sebring
##                3
##                James Stacy
##                0
## Kathryn "Kitty" "Pussycat" Lutesinger/Lugner
##                3
##                Leslie Van Houten
##                1
##                Linda Kasabian
##                6
##                Lynette Fromme
##                8
##                Marvin Schwartz
##                1
##                Manson Family
##                9
##          Patricia Krenwinkel
##                8
##                Rick Dalton
```

```
##                                11
##                                Roman Polanski
##                                4
##                                Sharon Tate
##                                8
##                                Steve McQueen
##                                3
##                                Susan Atkins
##                                7
##                                Wojciech Frykowski
##                                2
```

```
#View(degree_hollywood)
```

#Da die Console die Ausgabe auf eine gewisse Anzahl Ansgaben begrenzt, muss die Tabelle mit view ausgeg

```
components(hollywood)
```

```
## $membership
##                                Abigail Folger
##                                1
##                                Bruce Lee
##                                1
##                                Catherine "Gypsy" Share
##                                1
##                                Charles Manson
##                                1
##                                Charles "Tex" Watson
##                                1
##                                Cliff Booth
##                                1
##                                Francesca Capucci
##                                1
##                                George Spahn
##                                1
##                                Jay Sebring
##                                1
##                                James Stacy
##                                2
## Kathryn "Kitty" "Pussycat" Lutesinger/Lugner
##                                1
##                                Leslie Van Houten
##                                1
##                                Linda Kasabian
##                                1
##                                Lynette Fromme
##                                1
##                                Marvin Schwartz
##                                1
##                                Manson Family
##                                1
##                                Patricia Krenwinkel
##                                1
##                                Rick Dalton
##                                1
##                                Roman Polanski
```

```
## 1
## Sharon Tate
## 1
## Steve McQueen
## 1
## Susan Atkins
## 1
## Wojciech Frykowski
## 1
##
## $csize
## [1] 22 1
##
## $no
## [1] 2
```

*#Components zeigt die Anzahl der Teilnetzwerke und deren Größewhich.max(degree(hollywood, normalized =
#Liefert den Knoten, im Netzwerk hollywood, welcher den größten Degreewert hat
betweenness(hollywood, normalized = T)*

```
## Abigail Folger
## 0.000000000
## Bruce Lee
## 0.003988868
## Catherine "Gypsy" Share
## 0.000000000
## Charles Manson
## 0.202298495
## Charles "Tex" Watson
## 0.041120388
## Cliff Booth
## 0.113079777
## Francesca Capucci
## 0.026479076
## George Spahn
## 0.000000000
## Jay Sebring
## 0.000000000
## James Stacy
## 0.000000000
## Kathryn "Kitty" "Pussycat" Lutesinger/Lugner
## 0.015873016
## Leslie Van Houten
## 0.000000000
## Linda Kasabian
## 0.053890950
## Lynette Fromme
## 0.095516388
## Marvin Schwartz
## 0.000000000
## Manson Family
## 0.000000000
## Patricia Krenwinkel
## 0.053890950
```

```
## Rick Dalton
## 0.501061637
## Roman Polanski
## 0.105689549
## Sharon Tate
## 0.083704391
## Steve McQueen
## 0.000000000
## Susan Atkins
## 0.053890950
## Wojciech Frykowski
## 0.000000000
```

```
#Wie wahrscheinlich ist es, dass dieser Knoten die Verbindung zu anderen Knoten im Netzwerk herstellen
which.max(betweenness(hollywood, normalized = T))
```

```
## Rick Dalton
## 18
```

```
#Liefert den Knoten, im Netzwerk manson, welcher den größten Betweeneswert hat
ego_size(hollywood)
```

```
## [1] 3 3 3 10 10 9 5 3 4 1 5 3 8 10 2 10 10 12 5 8 4 8 3
```

```
#Liefert uns den Knoten, mit den meisten Verbindungen
mean_distance(hollywood)
```

```
## [1] 2.267574
```

```
#Gibt die längste Verbindung zwischen zwei Knoten aus
edge_density(hollywood)
```

```
## [1] 0.2094862
```

```
#Gibt die Kantendichte des Netzwerks aus
```

Aus unseren Recherchen kommt heraus, dass Charles Manson der Akteur mit dem höchsten Degreewert ist. Gemäß dem Film "Once upon a time in Hollywood" ist es Rick Dalton. Dieser Akteur ist ein fiktiver Charakter, welcher von Hollywood für ein besseres Storytelling erfunden wurde.

```
# Visualisierung der Pfaddistanz
#
dia <- get.diameter(manson, directed=T) # ruft die Werte auf
vcol <- rep("gray80", vcount(manson)) # setzt alle Werte der Knoten auf grau
vcol[dia] <- "gold" # setzt alle Vertices des Diameters auf gold
ecol <- rep("gray80", ecount(manson)) # setzt alle Kanten auf grau
ecol[E(manson, path=dia)] <- "orange" # definiert die Farbe des Pfads

# sucht die Kanten entlang des Pfades und färbt diese ein
plot(manson,
```

```

layout=layout_nicely,
vertex.color=vcol,
edge.color=ecol,
edge.arrow.size=.2,
edge.curved=.2,
main="Diameter im Netzwerk",
sub="Durchmesser auf dem kürzesten Weg")

```

Diameter im Netzwerk



Durchmesser auf dem kürzesten Weg

```

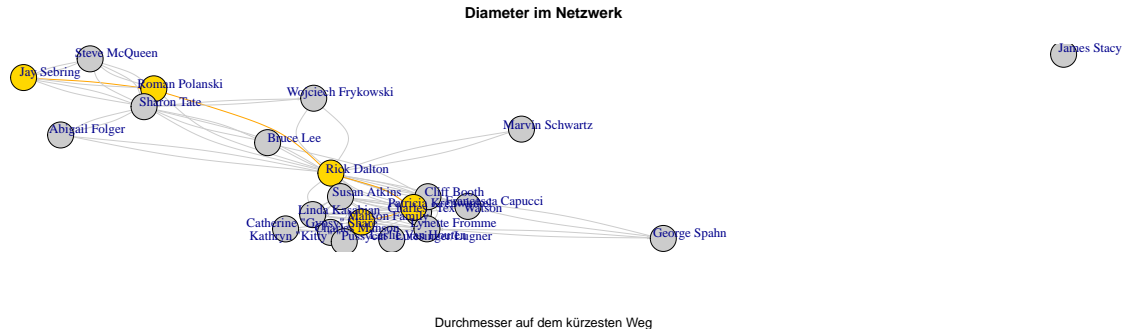
# Visualisierung der Pfaddistanz
#
dia <- get.diameter(hollywood, directed=T) # ruft die Werte auf
vcol <- rep("gray80", vcount(hollywood)) # setzt alle Werte der Knoten auf grau
vcol[dia] <- "gold" # setzt alle Vertices des Diameters auf gold
ecol <- rep("gray80", ecount(hollywood)) # setzt alle Kanten auf grau
ecol[E(hollywood, path=dia)] <- "orange" # definiert die Farbe des Pfads

# sucht die Kanten entlang des Pfades und färbt diese ein
plot(hollywood,
     asp=0,
     layout=layout_nicely,
     vertex.color=vcol,
     edge.color=ecol,
     edge.arrow.size=.2,
     edge.curved=.2,
     vertex.size=5,

```



```
vertex.label.dist=1,
main="Diameter im Netzwerk",
sub="Durchmesser auf dem kürzesten Weg")
```



#Wie viele Komponenten hat das Netzwerk?
`components(manson)`

```
## $membership
```

##	Allen Delisle	Alan Leroy Springer
##	1	1
##	Barbara Hoyt	Beach Boys
##	1	1
##	William Joseph "Bill" Vance	Robert "Bobby" Beausoleil
##	1	1
##	Bruce Davis	Bruce Hall
##	1	1
##	Bryan Lukashevsky	Catherine Gillies
##	1	1
##	Carol Loveless	Catherine "Gypsy" Share
##	1	1
##	Charles Allen Beard	Charlee Griffin
##	1	1
##	Charles Manson	Charles Tex Watson
##	1	1
##	Claudia Smith	Colleen Sinclair
##	1	1
##	David Baker	Danny DeCarlo
##	1	1
##	David Hannum	Dianne Lake
##	1	1
##	Diane Von Ahn	Ella Jo Bailey
##	1	1
##	Harold Irving True	Jack Gordon
##	1	1
##	Johnny Harold Swartz	Juan Flynn
##	1	1
##	Kathryn "Kitty" Lutesinger/Lugner	Kenneth Bell
##	1	1
##	Larry Bailey	Larry Craven
##	1	1
##	Laura Shepard	Leslie Van Houten

##	1	1
##	Linda Kasabian	Lynette Fromme
##	1	1
##	Maria Alonzo	Marcus Arneson
##	1	1
##	Mary Brunner	Madeleine Joan Cottage
##	1	1
##	Patricia Krenwinkel	Paul Alan Watkins
##	1	1
##	Phil Philips	Raymond Petrizzo
##	1	1
##	Randy Starr	Robert Murray
##	1	1
##	Robert Reinhard	Ruth Gordon
##	1	1
##	Ruth Ann Moorehouse	Sandra Good
##	1	1
##	Sherry Ann Cooper	Steve "Clem" Grogan
##	1	1
##	Stephen Palazzo	Stephanie Rowe
##	1	1
##	Stephanie Schram	Susan Atkins
##	1	1
##	Susan Bartell	Thomas Galella
##	1	1
##	Thomas "TJ" Walleman	Vern Plumlee
##	1	1
##	Aryan Brotherhood	Abigail Folger
##	1	1
##	Adolph Alexander	Afton "Star" Burton
##	1	1
##	A. H. Burdick	Alvin "Old Creepy" Karpis
##	1	1
##	Alan Rose	Anton LaVeys
##	1	1
##	Arlene Barker	Barbara Myers
##	1	1
##	Barbara Parkins	Barbara Rosenberg
##	1	1
##	Bennet Berger	Bernard "Lotsapoppa/Big Crow" Crowe
##	1	1
##	Benny Jay Teal	Benny Unbekannter Nachname
##	1	1
##	Bill Boyd	Bruce Lee
##	1	1
##	Brooks Poston	Brian Wilson
##	1	1
##	Charles Hollopeter	Charles Manson Junior
##	1	1
##	Charles Luther Manson	Charles Older
##	1	1
##	Colonel Walker Henderson Scott Sr	David Barton
##	1	1
##	Darrell Grey	David M. Katsuyama

##	1	1
##	David Smith (Dr.)	Dean Martin
##	1	1
##	Dean Moorhouse	Debra Tate
##	1	1
##	Dennis Wilson	Donald "Shorty" Shea
##	1	1
##	Edward Davis	Dr. Ernst Dernberg
##	1	1
##	Film Industry	Freda Hofmann
##	1	1
##	Dr. Frederick Meyers	Frank Sinatra
##	1	1
##	Frank Struthers	Gary Allen Hinman
##	1	1
##	Gerald Ford	George Spahn
##	1	1
##	Gregg Jakobson	Hells Angels
##	1	1
##	"Hare-Krishna-Anh�nger" aus Gefaengnis	Henry Beatly
##	1	1
##	Dr. Ira Frank	Irving Kanarek
##	1	1
##	Jason Freeman	Jay Sebring
##	1	1
##	Jimmy Mach	Joseph Ball
##	1	1
##	John Goffigan	John "Zero" Haught
##	1	1
##	Dr. Joel Hochmann	Joel Pugh
##	1	1
##	Joan Svelte	Joe Talley
##	1	1
##	Jonathan Wayne "Jonny-Boy" Wilkerson	Juan Corona
##	1	1
##	Kathleen Maddox	Kirche Satans
##	1	1
##	Leno LaBianca	Leona Rae "Candy" Stevens
##	1	1
##	Luella "Nachname unbekannt"	Dr. Marvin Katz
##	1	1
##	Maxwell Keith	Matthew Lentz
##	1	1
##	Mary Anne McLean	Martin Ransohoff
##	1	1
##	Manson Family	Michael Channels
##	1	1
##	Mia Farrow	Michael Lee Monfort "Red Eye"
##	1	1
##	Michal Welles	Nancy Pitman (alias Brenda McCann)
##	1	1
##	Officer Pursel	Officer Rudi
##	1	1
##	Paul Crockett	Paul Fitzgerald

##	1	1
##	Paul Richard Polanski	Phil Alleman
##	1	1
##	Phillippe Forquet	Phil Kaufman
##	1	1
##	Prozesskirche	Ray Hoekstra
##	1	1
##	Richard Beymer	Richard Caballero
##	1	1
##	Richard Carlson	Robert Kenneth "Bobby" Beusoleil
##	1	1
##	Robert A. Heinlein	Ronny Howard
##	1	1
##	Ronald Hughes	Robert Kasabian
##	1	1
##	Rosemary LaBianca	Ronald Markman
##	1	1
##	Roger Moore de Gimston	Roman Polanski
##	1	1
##	Roger Smith	Rosalie Jean Willis
##	1	1
##	Sam Bublick	Saladin Nader
##	1	1
##	Schwester von Jay Sebring	Sharon Tate
##	1	1
##	Sheilah Wells	Sirhan Sirhan
##	1	1
##	Scientology	Stephane Bourgoin
##	1	1
##	Steve(n) Kay	Stanley McGuire
##	1	1
##	Steve McQueen	Steven Parent
##	1	1
##	The Beatles	Terry Melcher
##	1	1
##	Thomas Noguchi	Tochter von Michal Welles
##	1	1
##	The Straight Satans	unbekannter Besucher "Rex"
##	1	1
##	unbekannter Mithaeftling Jaden	Unbekannter Mithaeftling
##	1	1
##	unbekannte Nonne	unbekannter Priester
##	1	1
##	unbekannter Scientologe	Valentine Michael Manson
##	1	1
##	Vincent Bugliosi	Verginia Heinlein
##	1	1
##	Dr. Wayne O. Evans	Willis Carson
##	1	1
##	Winifried Chapman	William Garretson
##	1	1
##	William "Billy" Goucher	William Eugene Manson
##	1	1
##	Wojciech Frykowski	

```

##                                     1
##
## $csize
## [1] 195
##
## $no
## [1] 1

is_connected(manson)

## [1] TRUE

#Welchen Durchmesser hat das Netzwerk?
diameter(manson)

## [1] 201

#Wie ist die Dichte des Netzwerks?
edge_density(manson)

## [1] 0.01675919

#Wie ist die Pfad-Distanz im Netzwerk?
mean_distance(manson)

## [1] 2.823062

#Wie viele Cluster hat das Netzwerk?
cluster_walktrap(manson)

## IGRAPH clustering walktrap, groups: 54, mod: 0.34
## + groups:
## $`1`
## [1] "Kathleen Maddox" "William Eugene Manson"
##
## $`2`
## [1] "Charles Tex Watson"
## [2] "Adolph Alexander"
## [3] "Bernard \"Lotsapoppa/Big Crow\" Crowe"
## [4] "Bill Boyd"
## [5] "Freda Hofmann"
## [6] "Hells Angels"
## + ... omitted several groups/vertices

groups(manson)

## NULL

```

```
#Wie viele Komponenten hat das Netzwerk?  
components(hollywood)
```

```
## $membership  
##           Abigail Folger  
##           1  
##           Bruce Lee  
##           1  
##           Catherine "Gypsy" Share  
##           1  
##           Charles Manson  
##           1  
##           Charles "Tex" Watson  
##           1  
##           Cliff Booth  
##           1  
##           Francesca Capucci  
##           1  
##           George Spahn  
##           1  
##           Jay Sebring  
##           1  
##           James Stacy  
##           2  
## Kathryn "Kitty" "Pussycat" Lutesinger/Lugner  
##           1  
##           Leslie Van Houten  
##           1  
##           Linda Kasabian  
##           1  
##           Lynette Fromme  
##           1  
##           Marvin Schwartz  
##           1  
##           Manson Family  
##           1  
##           Patricia Krenwinkel  
##           1  
##           Rick Dalton  
##           1  
##           Roman Polanski  
##           1  
##           Sharon Tate  
##           1  
##           Steve McQueen  
##           1  
##           Susan Atkins  
##           1  
##           Wojciech Frykowski  
##           1  
##  
## $csize  
## [1] 22 1
```

```
##  
## $no  
## [1] 2
```

```
is_connected(hollywood)
```

```
## [1] FALSE
```

```
#Welchen Durchmesser hat das Netzwerk?  
diameter(hollywood)
```

```
## [1] 103
```

```
#Wie ist die Dichte des Netzwerks?  
edge_density(hollywood)
```

```
## [1] 0.2094862
```

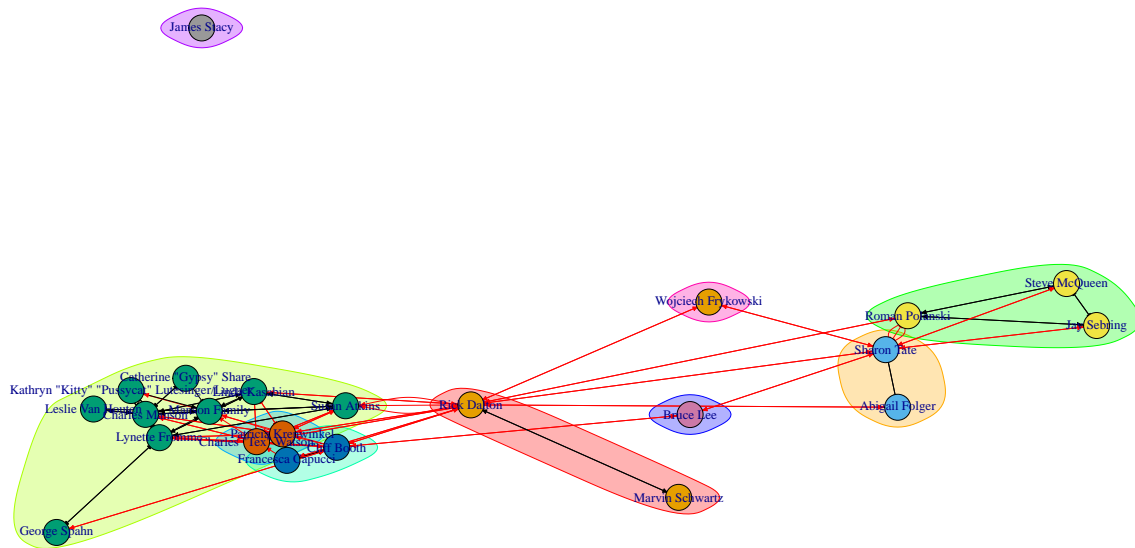
```
#Wie ist die Pfad-Distanz im Netzwerk?  
mean_distance(hollywood)
```

```
## [1] 2.267574
```

```
#Wie viele Cluster hat das Netzwerk?  
cluster_walktrap(hollywood)
```

```
## IGRAPH clustering walktrap, groups: 9, mod: 0.089  
## + groups:  
## $`1`  
## [1] "Marvin Schwartz" "Rick Dalton"  
##  
## $`2`  
## [1] "Abigail Folger" "Sharon Tate"  
##  
## $`3`  
## [1] "Catherine \"Gypsy\" Share"  
## [2] "Charles Manson"  
## [3] "George Spahn"  
## + ... omitted several groups/vertices
```

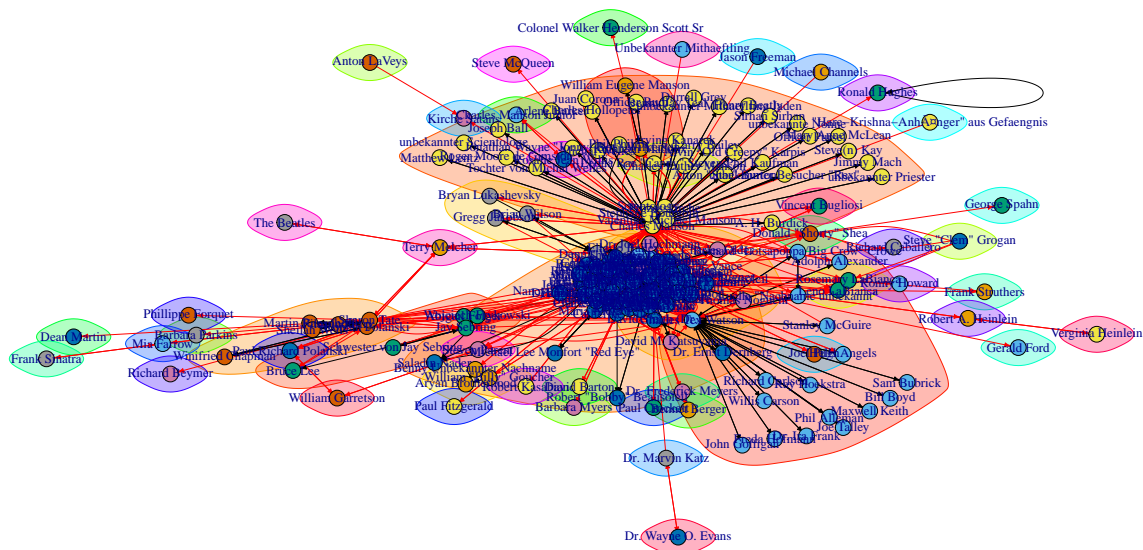
```
cw_hollywood <- cluster_walktrap(hollywood)  
plot(cw_hollywood, hollywood, vertex.size=5, layout=layout_nicely, asp=0, edge.arrow.size=0.4)
```



```

cw_gesamt <- cluster_walktrap(manson)
plot(cw_gesamt, manson, vertex.size=3, layout=layout_nicely, asp=0, edge.arrow.size=0.4)

```



Es gibt noch weitere Clustering-Verfahren, die Cluster nach unterschiedlichen Kriterien bilden. Der Algorithmus von `cluster_edge_betweenness()` geht davon aus, dass sich Cluster vor allem an den "Sollbruchstellen" eines Netzwerks trennen lassen. Diese werden über den Wert der Betweenness berechnet, also die Knoten, die in hohem Maße für die Verbindung zu anderen Knoten beitragen.


```
# erstellt die Berechnung für die Modularität und deren Teilgruppen
eb_manson <- cluster_edge_betweenness(manson)
```

```
## Warning in cluster_edge_betweenness(manson): At community.c:460 :Membership
## vector will be selected based on the lowest modularity score.
```

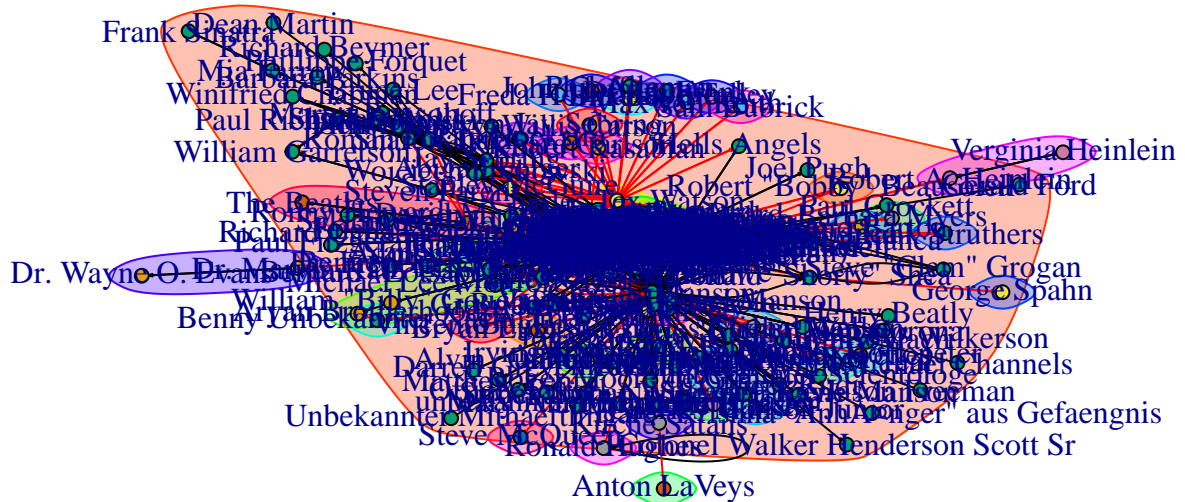
```
## Warning in cluster_edge_betweenness(manson): At community.c:467 :Modularity
## calculation with weighted edge betweenness community detection might not make
## sense -- modularity treats edge weights as similarities while edge betweenness
## treats them as distances
```

```
eb_manson
```

```
## IGRAPH clustering edge betweenness, groups: 56, mod: 0.073
## + groups:
## $`1`
## [1] "Allen Delisle"
##
## $`2`
## [1] "Alan Leroy Springer"
##
## $`3`
## [1] "Barbara Hoyt"
## [2] "Beach Boys"
## [3] "William Joseph \"Bill\" Vance"
## + ... omitted several groups/vertices
```

```
plot(eb_manson, manson, vertex.size=3, layout=layout_nicely, asp=0, edge.arrow.size=0.1, main= "Edge-Be
```

Edge-Betweenness-Cluster Gesamtnetzwerk



```
eb_hollywood <- cluster_edge_betweenness(hollywood)
```

```
## Warning in cluster_edge_betweenness(hollywood): At community.c:460 :Membership
## vector will be selected based on the lowest modularity score.
```

```
## Warning in cluster_edge_betweenness(hollywood): At community.c:467 :Modularity
## calculation with weighted edge betweenness community detection might not make
## sense -- modularity treats edge weights as similarities while edge betweenness
## treats them as distances
```

eb_hollywood

```
## IGRAPH clustering edge betweenness, groups: 3, mod: 0.056
```

```
## + groups:
```

\$`1`

```
##      [1] "Abigail Folger"      "Bruce Lee"           "Jay Sebring"
```

```
##      [4] "Roman Polanski"      "Sharon Tate"         "Steve McQueen"
```

```
## [7] "Wojciech Frykowski"
```

##

\$`2`

```
##      [1] "Catherine \"Gypsy\" Share"
```

```
## [2] "Charles Manson"
```

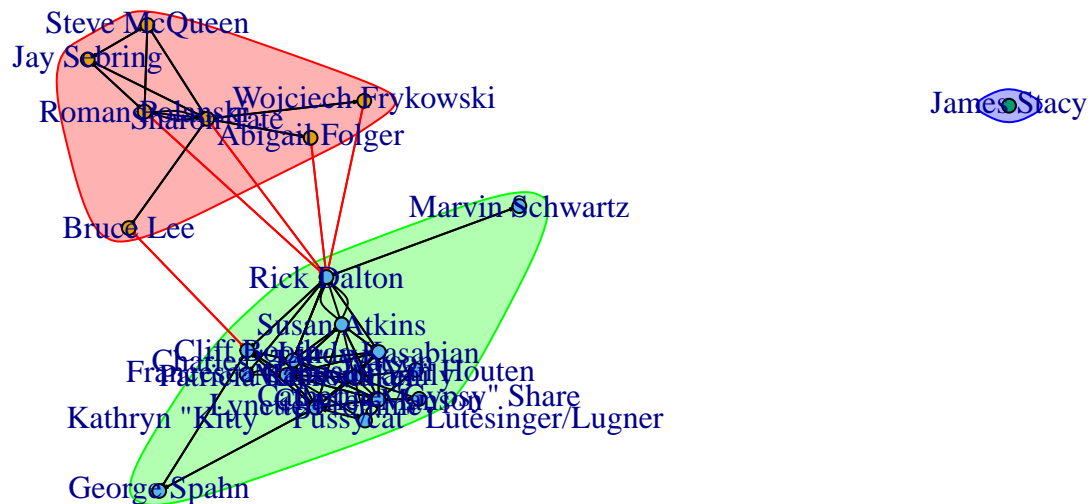
```
##      [3] "Charles \"Tex\" Watson"
```

```
## [4] "Cliff Booth"
```

```
## + ... omitted several groups/vertices
```

```
plot(eb_hollywood, hollywood, vertex.size=3, layout=layout_nicely, asp=0, edge.arrow.size=0.1, main= "E
```

Edge-Betweenness-Cluster Gesamtnetzwerk



Es gibt 56 Gruppen. Der Cluster macht im Gesamtnetzwerk noch keinen Sinn... Teilnetzwerk beachten!!!

Um Netzwerke in einzelne Komponenten oder Cluster zu zerlegen verwenden wir den Befehl `decompose.graph(g)`. Wir wissen, dass das Netzwerk zwei Cluster oder Komponenten hat, die ausgegeben werden. Danach lassen sich die Cluster getrennt auftrennen, als neue `igraph`-Objekte umschreiben und visualisieren.

Teilnetzwerke

Mansonfamilie

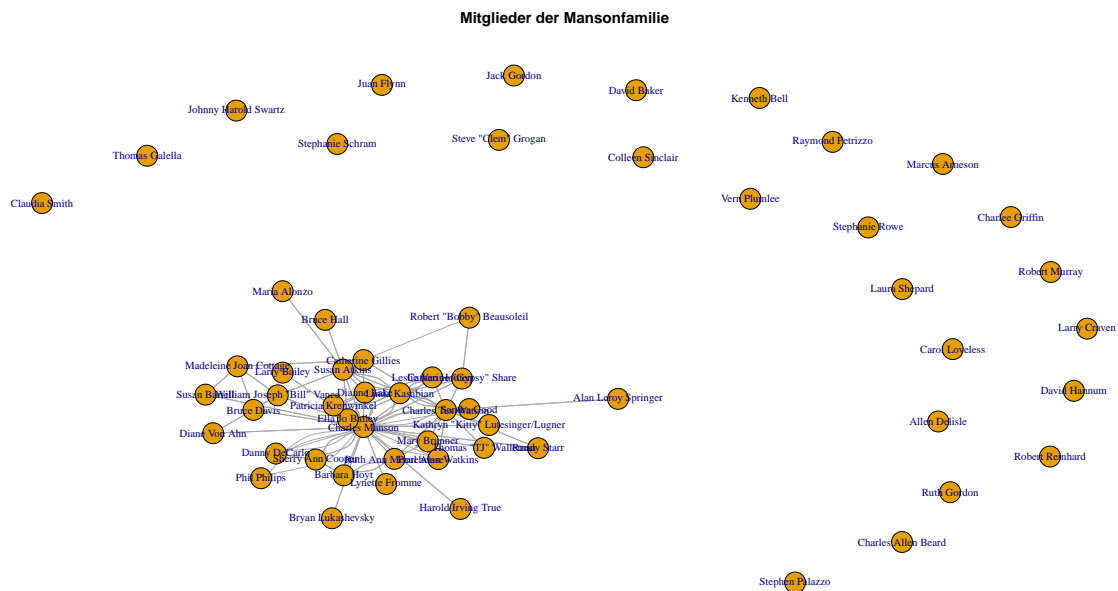
```
#Löscht alle Knoten mit Member gleich 1. Also alle Knoten, welche nicht in der Mansonfamilie sind.
member <- delete.vertices(manson, V(manson)[member != "2"])

# Version 1
plot (
  member,
  asp = 0,
  rescale = T,
  vertex.size = 4,
```

```

vertex.frame.width = 0.01,
edge.width = 0.3,
vertex.label.cex = 0.8,
edge.arrow.size = .1,
edge.curved = curve_multiple(member),
main = "Mitglieder der Mansonfamilie"
)

```



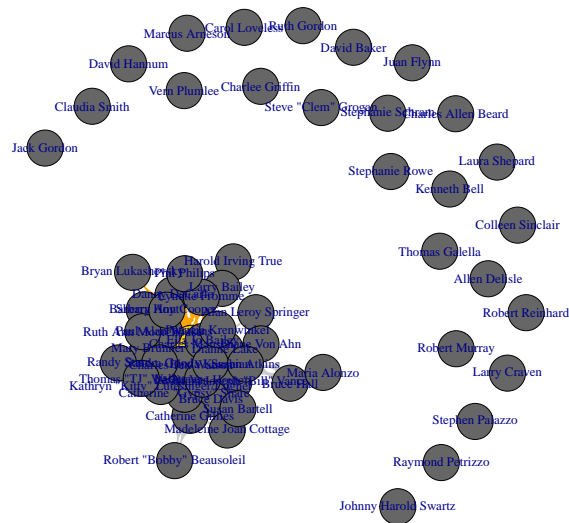
```

#Version 2- Mit Manson hervorgehoben
inc.edges <- incident(member, V(member)[name=="Charles Manson"], mode="all")

# Set colors to plot the selected edges.
ecol <- rep("gray80", ecount(member))
ecol[inc.edges] <- "orange"
vcol <- rep("grey40", vcount(member))
vcol[V(member)$name=="Charles Manson"] <- "gold"

plot(member, vertex.color=vcol, edge.color=ecol)

```



```
plot (
  member,
  asp = 0,
  rescale = T,
  vertex.size = 4,
  vertex.frame.width = 0.01,
  edge.width = 0.3,
  vertex.label.cex = 0.8,
  edge.arrow.size = .1,
  edge.curved = curve_multiple(member),
  main = "Mitglieder der Mansonfamilie",
  sub = "Hilight",
  vertex.color=vcol,
  edge.color=ecol
)
```


##	2	4
##	Charlee Griffin	Charles Manson
##	5	2
##	Charles Tex Watson	Claudia Smith
##	2	6
##	Colleen Sinclair	David Baker
##	7	8
##	Danny DeCarlo	David Hannum
##	2	9
##	Dianne Lake	Diane Von Ahn
##	2	2
##	Ella Jo Bailey	Harold Irving True
##	2	2
##	Jack Gordon	Johnny Harold Swartz
##	10	11
##	Juan Flynn Kathryn "Kitty" Lutesinger/Lugner	
##	12	2
##	Kenneth Bell	Larry Bailey
##	13	2
##	Larry Craven	Laura Shepard
##	14	15
##	Leslie Van Houten	Linda Kasabian
##	2	2
##	Lynette Fromme	Maria Alonzo
##	2	2
##	Marcus Arneson	Mary Brunner
##	16	2
##	Madeleine Joan Cottage	Patricia Krenwinkel
##	2	2
##	Paul Alan Watkins	Phil Philips
##	2	2
##	Raymond Petrizzo	Randy Starr
##	17	2
##	Robert Murray	Robert Reinhard
##	18	19
##	Ruth Gordon	Ruth Ann Moorehouse
##	20	2
##	Sandra Good	Sherry Ann Cooper
##	2	2
##	Steve "Clem" Grogan	Stephen Palazzo
##	21	22
##	Stephanie Rowe	Stephanie Schram
##	23	24
##	Susan Atkins	Susan Bartell
##	2	2
##	Thomas Galella	Thomas "TJ" Walleman
##	25	2
##	Vern Plumlee	
##	26	
##		
##	\$csize	
##	[1] 1 34 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
##	[26] 1	
##		

```
## $no
## [1] 26
```

```
#Components zeigt die Anzahl der Teilnetzwerke und deren Größewhich.max(degree(member, normalized = T))
#Liefert den Knoten, im Netzwerk member, welcher den größten Degreewert hat
betweenness(member, normalized = T)
```

##	Allen Delisle	Alan Leroy Springer
##	0.0000000000	0.0000000000
##	Barbara Hoyt	William Joseph "Bill" Vance
##	0.0004537205	0.0040762827
##	Robert "Bobby" Beausoleil	Bruce Davis
##	0.0004537205	0.0000000000
##	Bruce Hall	Bryan Lukashovsky
##	0.0000000000	0.0000000000
##	Catherine Gillies	Carol Loveless
##	0.0131378494	0.0000000000
##	Catherine "Gypsy" Share	Charles Allen Beard
##	0.0174430329	0.0000000000
##	Charlee Griffin	Charles Manson
##	0.0000000000	0.1787556561
##	Charles Tex Watson	Claudia Smith
##	0.0886416447	0.0000000000
##	Colleen Sinclair	David Baker
##	0.0000000000	0.0000000000
##	Danny DeCarlo	David Hannum
##	0.0000000000	0.0000000000
##	Dianne Lake	Diane Von Ahn
##	0.0000000000	0.0074244519
##	Ella Jo Bailey	Harold Irving True
##	0.0000000000	0.0000000000
##	Jack Gordon	Johnny Harold Swartz
##	0.0000000000	0.0000000000
##	Juan Flynn Kathryn "Kitty" Lutesinger/Lugner	
##	0.0000000000	0.0000000000
##	Kenneth Bell	Larry Bailey
##	0.0000000000	0.0000000000
##	Larry Craven	Laura Shepard
##	0.0000000000	0.0000000000
##	Leslie Van Houten	Linda Kasabian
##	0.0098414268	0.0591106883
##	Lynette Fromme	Maria Alonzo
##	0.0000000000	0.0000000000
##	Marcus Arneson	Mary Brunner
##	0.0000000000	0.0000000000
##	Madeleine Joan Cottage	Patricia Krenwinkel
##	0.0027139208	0.0046682515
##	Paul Alan Watkins	Phil Phillips
##	0.0000000000	0.0000000000
##	Raymond Petrizzo	Randy Starr
##	0.0000000000	0.0093378801
##	Robert Murray	Robert Reinhard
##	0.0000000000	0.0000000000
##	Ruth Gordon	Ruth Ann Moorehouse


```
##          0.0000000000          0.0080814464
##          Sandra Good          Sherry Ann Cooper
##          0.0000000000          0.0000000000
##          Steve "Clem" Grogan          Stephen Palazzo
##          0.0000000000          0.0000000000
##          Stephanie Rowe          Stephanie Schram
##          0.0000000000          0.0000000000
##          Susan Atkins          Susan Bartell
##          0.0527624982          0.0050211736
##          Thomas Galella          Thomas "TJ" Wallemann
##          0.0000000000          0.0000000000
##          Vern Plumlee
##          0.0000000000
```

```
#Wie wahrscheinlich ist es, dass dieser Knoten die Verbindung zu anderen Knoten im Netzwerk herstellen
which.max(betweenness(member, normalized = T))
```

```
## Charles Manson
##          14
```

```
#Liefert den Knoten, im Netzwerk member, welcher den größten Betweeneswert hat
ego_size(member)
```

```
## [1] 1 2 4 5 3 3 2 2 5 1 6 1 1 29 15 1 1 1 3 1 5 3 4 2 1
## [26] 1 1 3 1 2 1 1 6 7 2 2 1 4 5 5 3 2 1 3 1 1 1 5 3 4
## [51] 1 1 1 1 9 3 1 3 1
```

```
#Liefert uns den Knoten, mit den meisten Verbindungen
mean_distance(member)
```

```
## [1] 2.10652
```

```
#Gibt die längste Verbindung zwischen zwei Knoten aus
edge_density(member)
```

```
## [1] 0.04412624
```

```
#Gibt die Kantendichte des Netzwerks aus
#Wie viele Komponenten hat das Netzwerk?
components(hollywood)
```

```
## $membership
##          Abigail Folger
##          1
##          Bruce Lee
##          1
##          Catherine "Gypsy" Share
##          1
##          Charles Manson
##          1
##          Charles "Tex" Watson
```

```

##                                1
##                        Cliff Booth
##                                1
##                Francesca Capucci
##                                1
##                        George Spahn
##                                1
##                        Jay Sebring
##                                1
##                James Stacy
##                                2
## Kathryn "Kitty" "Pussycat" Lutesinger/Lugner
##                                1
##                Leslie Van Houten
##                                1
##                Linda Kasabian
##                                1
##                Lynette Fromme
##                                1
##                Marvin Schwartz
##                                1
##                Manson Family
##                                1
##                Patricia Krenwinkel
##                                1
##                Rick Dalton
##                                1
##                Roman Polanski
##                                1
##                Sharon Tate
##                                1
##                Steve McQueen
##                                1
##                Susan Atkins
##                                1
##                Wojciech Frykowski
##                                1
##
## $csize
## [1] 22  1
##
## $no
## [1] 2

```

```

#Welchen Durchmesser hat das Netzwerk?
diameter(hollywood)

```

```
## [1] 103
```

```

#Wie ist die Dichte des Netzwerks?
edge_density(hollywood)

```

```
## [1] 0.2094862
```

```
#Wie ist die Pfad-Distanz im Netzwerk?
mean_distance(hollywood)
```

```
## [1] 2.267574
```

```
#Wie viele Cluster hat das Netzwerk?
cluster_walktrap(hollywood)
```

```
## IGRAPH clustering walktrap, groups: 9, mod: 0.089
## + groups:
## $`1`
## [1] "Marvin Schwartz" "Rick Dalton"
##
## $`2`
## [1] "Abigail Folger" "Sharon Tate"
##
## $`3`
## [1] "Catherine \"Gypsy\" Share"
## [2] "Charles Manson"
## [3] "George Spahn"
## + ... omitted several groups/vertices
```

```
#Da der Pfad nur über verbundene Knoten entlang läuft, blenden wir alle Isolates aus.
member1 <- delete.vertices(member, degree(member) == 0)#Löscht alle Isolates
```

```
# Visualisierung der Pfaddistanz
```

```
dia <- get.diameter(member1, directed=T) # ruft die Werte auf
vcol <- rep("gray80", vcount(member1)) # setzt alle Werte der Knoten auf grau
vcol[dia] <- "gold" # setzt alle Vertices des Diameters auf gold
ecol <- rep("gray80", ecoun(member1)) # setzt alle Kanten auf grau
ecol[E(member1, path=dia)] <- "orange" # definiert die Farbe des Pfads
```

```
# sucht die Kanten entlang des Pfades und färbt diese ein
```

```
plot(member1,
      layout=layout_nicely,
      vertex.color=vcol,
      edge.color=ecol,
      edge.arrow.size=.2,
      edge.curved=.2,
      vertex.size=5,
      main="Diameter im Netzwerk",
      sub="Durchmesser auf dem kürzesten Weg")
```

Diameter im Netzwerk



Durchmesser auf dem kürzesten Weg

#Die Clusterberechnung zeigt die verschiedenen Untergruppen in einem Netzwerk an.

```
member_gc <- cluster_walktrap(member1)
modularity(member_gc)
```

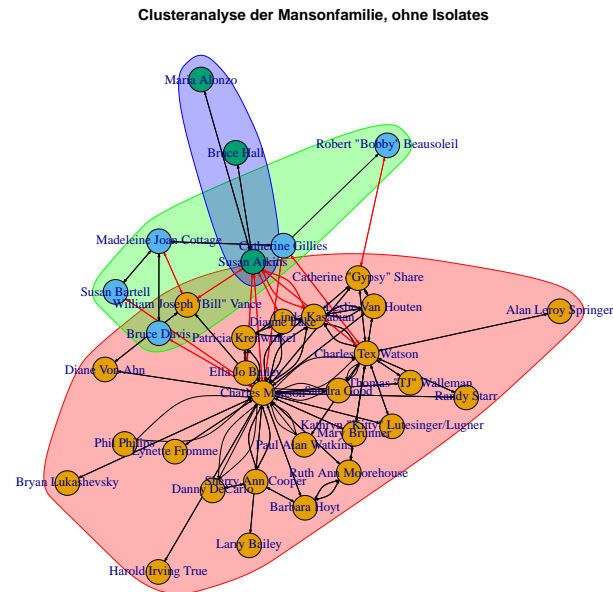
```
## [1] 0.09480457
```

```
membership(member_gc)
```

```
##          Alan Leroy Springer          Barbara Hoyt
##                      1                      1
## William Joseph "Bill" Vance      Robert "Bobby" Beausoleil
##                      1                      2
##          Bruce Davis              Bruce Hall
##                      2                      3
##          Bryan Lukashevsky        Catherine Gillies
##                      1                      2
## Catherine "Gypsy" Share          Charles Manson
##                      1                      1
##          Charles Tex Watson        Danny DeCarlo
##                      1                      1
##          Dianne Lake              Diane Von Ahn
##                      1                      1
##          Ella Jo Bailey            Harold Irving True
##                      1                      1
## Kathryn "Kitty" Lutesinger/Lugner  Larry Bailey
##                      1                      1
##          Leslie Van Houten        Linda Kasabian
##                      1                      1
##          Lynette Fromme          Maria Alonzo
##                      1                      3
##          Mary Brunner            Madeleine Joan Cottage
##                      1                      2
##          Patricia Krenwinkel      Paul Alan Watkins
##                      1                      1
##          Phil Philips              Randy Starr
##                      1                      1
##          Ruth Ann Moorehouse      Sandra Good
##                      1                      1
##          Sherry Ann Cooper        Susan Atkins
##                      1                      3
##          Susan Bartell            Thomas "TJ" Walleman
```

1

```
plot(member_gc, member1, vertex.size=10, edge.arrow.size=.2, main="Clusteranalyse der Mansonfamilie, ohne")
```



Es gibt noch weitere Clustering-Verfahren, die Cluster nach unterschiedlichen Kriterien bilden. Der Algorithmus von `cluster_edge_betweenness()` geht davon aus, dass sich Cluster vor allem an den “Sollbruchstellen” eines Netzwerks trennen lassen. Diese werden über den Wert der Betweenness berechnet, also die Knoten, die in hohem Maße für die Verbindung zu anderen Knoten beitragen.

```
# erstellt die Berechnung für die Modularität und deren Teilgruppen
```

```
eb_member <- cluster_edge_betweenness(member1)
```

```
## Warning in cluster_edge_betweenness(member1): At community.c:460 :Membership
## vector will be selected based on the lowest modularity score.
```

```
## Warning in cluster_edge_betweenness(member1): At community.c:467 :Modularity
## calculation with weighted edge betweenness community detection might not make
## sense -- modularity treats edge weights as similarities while edge betweenness
## treats them as distances
```

eb_member

```
## IGRAPH clustering edge betweenness, groups: 11, mod: 0.088
```

```
## + groups:
```

\$`1`

```
##      [1] "Alan Leroy Springer"
```

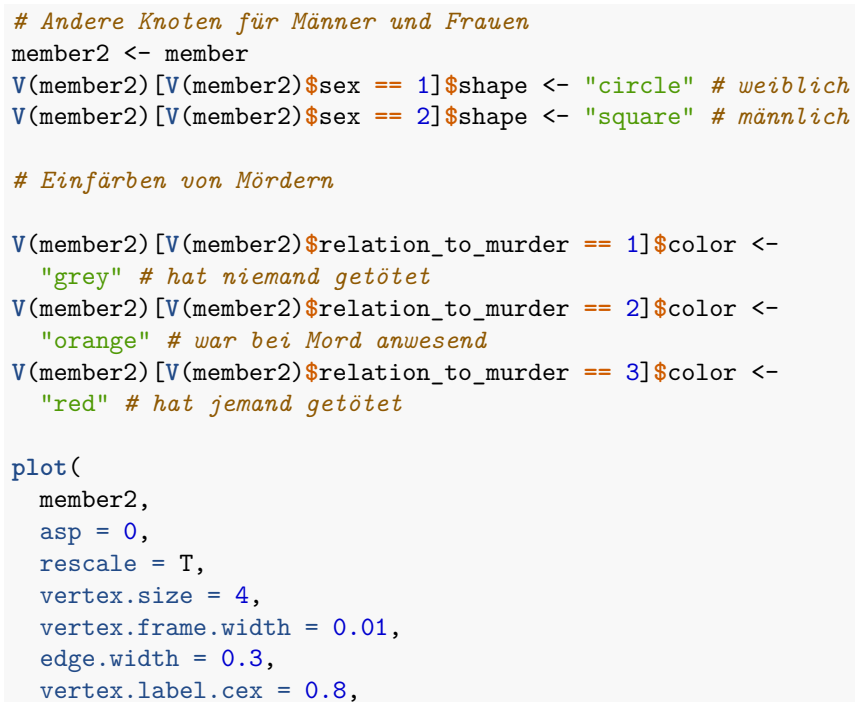
```
## [2] "William Joseph \"Bill\" Vance"
```

```
## [3] "Bruce Hall"
```

```
## [4] "Charles Manson"
```

```
## [5] "Charles Tex Watson"
```

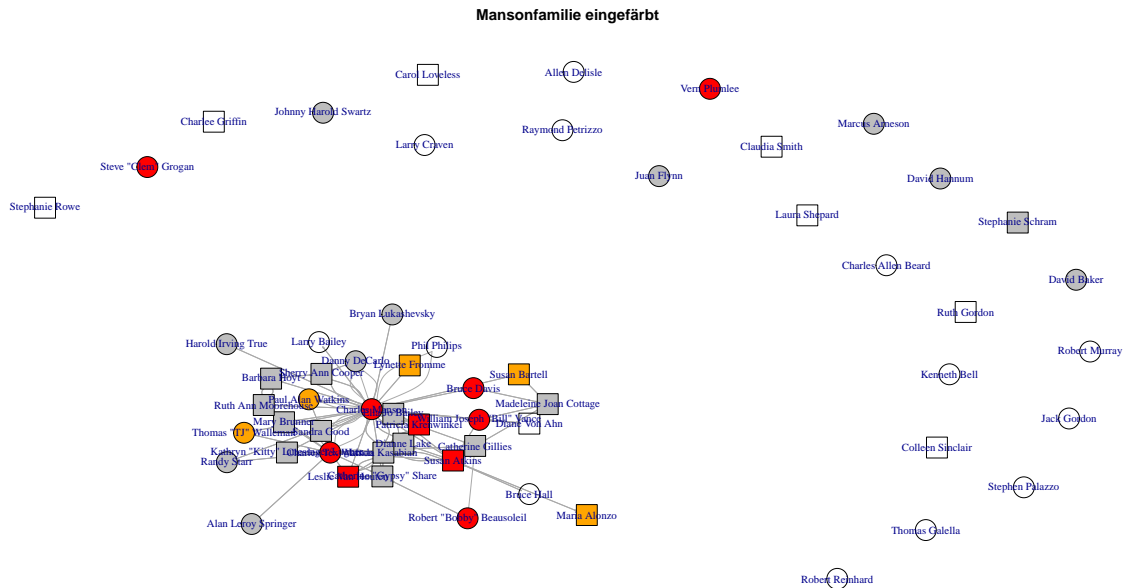
```
plot(eb_member, member1, vertex.size=3, layout=layout_nicely, edge.arrow.size=0.1, main= "Edge-Betweenness Centrality")
```



```

edge.arrow.size = .1,
edge.curved = curve_multiple(member2),
main = "Mansonfamilie eingefärbt"
)

```



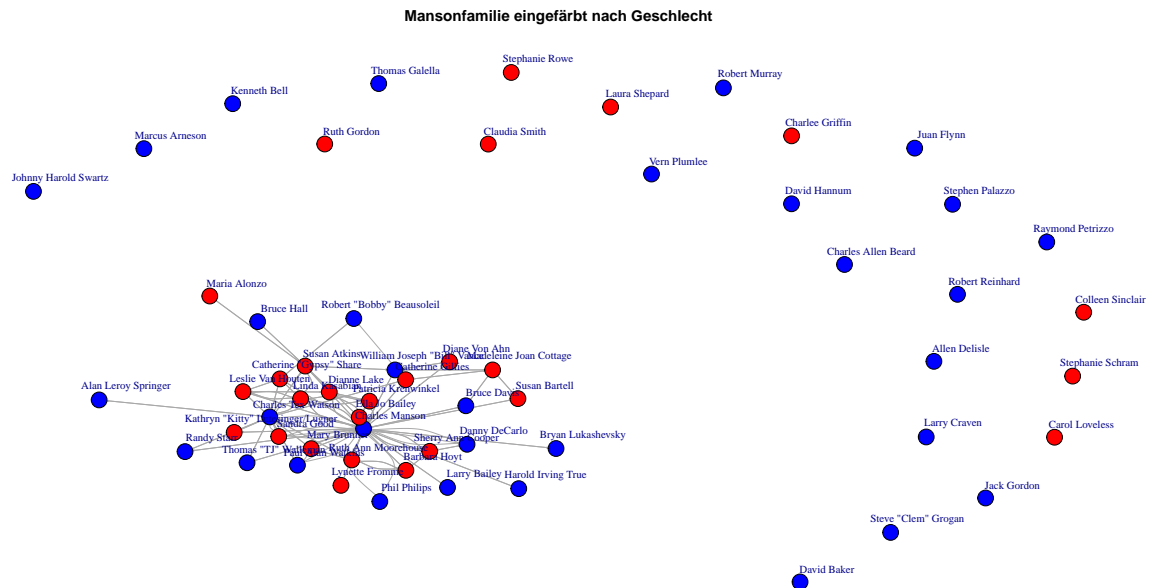
Mansonfamilie nach Geschlecht

```

# Einfärben von Mördern
member3 <- member
V(member3)[V(member3)$sex == 1]$color <- "blue" # Männer
V(member3)[V(member3)$sex == 2]$color <- "red" # Frauen

plot(
  member3,
  asp = 0,
  rescale = T,
  vertex.size = 3,
  vertex.frame.width = 0.01,
  edge.width = 0.3,
  vertex.label.cex = 0.8,
  vertex.label.dist = 1,
  edge.arrow.size = .1,
  edge.curved = curve_multiple(member),
  main = "Mansonfamilie eingefärbt nach Geschlecht"
)

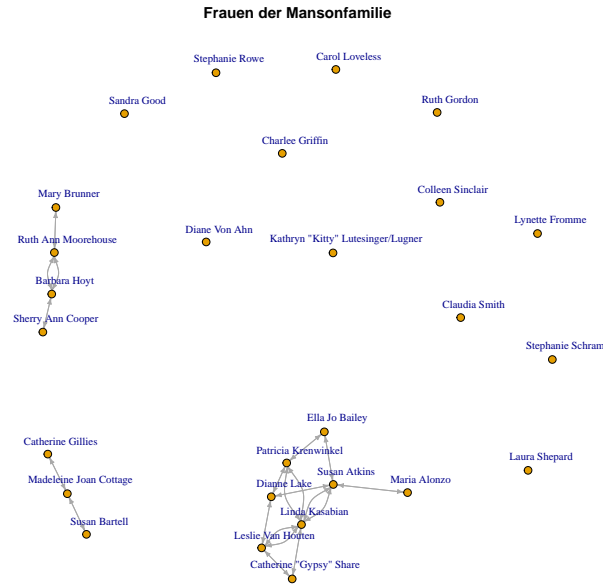
```



```
member_women <- delete.vertices(member, V(member)[(sex != 2)])
member_women
```

```
## IGRAPH dbe2b51 DNWB 27 40 --
## + attr: name (v/c), type (v/n), sex (v/n), date_of_birth (v/c),
## | date_of_death (v/c), type_of_death (v/n), power (v/n),
## | relation_to_murder (v/n), member (v/n), relationship (e/c), weight
## | (e/c), year_beginning (e/c), year_end (e/c)
## + edges from dbe2b51 (vertex names):
## [1] Barbara Hoyt      ->Ruth Ann Moorehouse
## [2] Barbara Hoyt      ->Ruth Ann Moorehouse
## [3] Barbara Hoyt      ->Sherry Ann Cooper
## [4] Catherine Gillies  ->Madeleine Joan Cottage
## [5] Catherine "Gypsy" Share->Leslie Van Houten
## + ... omitted several edges
```

```
plot(
  member_women,
  vertex.size = 3,
  vertex.frame.width = 0.01,
  edge.width = 0.3,
  vertex.label.cex = 0.8,
  vertex.label.dist = 1,
  edge.arrow.size = .4,
  main = "Frauen der Mansonfamilie"
)
```

#Wie wahrscheinlich ist es, dass dieser Knoten die Verbindung zu anderen Knoten im Netzwerk herstellen
betweenness(member_women, **normalized** = F)

##	Barbara Hoyt	Catherine Gillies
##	4.000000	0.000000
##	Carol Loveless	Catherine "Gypsy" Share
##	0.000000	0.000000
##	Charlee Griffin	Claudia Smith
##	0.000000	0.000000
##	Colleen Sinclair	Dianne Lake
##	0.000000	0.000000
##	Diane Von Ahn	Ella Jo Bailey
##	0.000000	0.000000
##	Kathryn "Kitty" Lutesinger/Lugner	Laura Shepard
##	0.000000	0.000000
##	Leslie Van Houten	Linda Kasabian
##	2.666667	20.000000
##	Lynette Fromme	Maria Alonzo
##	0.000000	0.000000
##	Mary Brunner	Madeleine Joan Cottage
##	0.000000	2.000000
##	Patricia Krenwinkel	Ruth Gordon
##	4.666667	0.000000
##	Ruth Ann Moorehouse	Sandra Good
##	4.000000	0.000000
##	Sherry Ann Cooper	Stephanie Rowe
##	0.000000	0.000000
##	Stephanie Schram	Susan Atkins
##	0.000000	16.666667
##	Susan Bartell	
##	0.000000	

```
#Wie schnell kann dieser Knoten alle anderen Knoten im Netzwerk erreichen? Hub bzw. Verteilerknoten für
close_women <-closeness(member_women, normalized=T)
```

```
## Warning in closeness(member_women, normalized = T): At centrality.c:
## 2617 :closeness centrality is not well-defined for disconnected graphs
```

```
#Der Befehl prüft, wie hoch die Dichte des Netzwerks ist.
edge_density(member_women)
```

```
## [1] 0.05698006
```

Es gibt 27 Frauen in der Mansonfamilie.

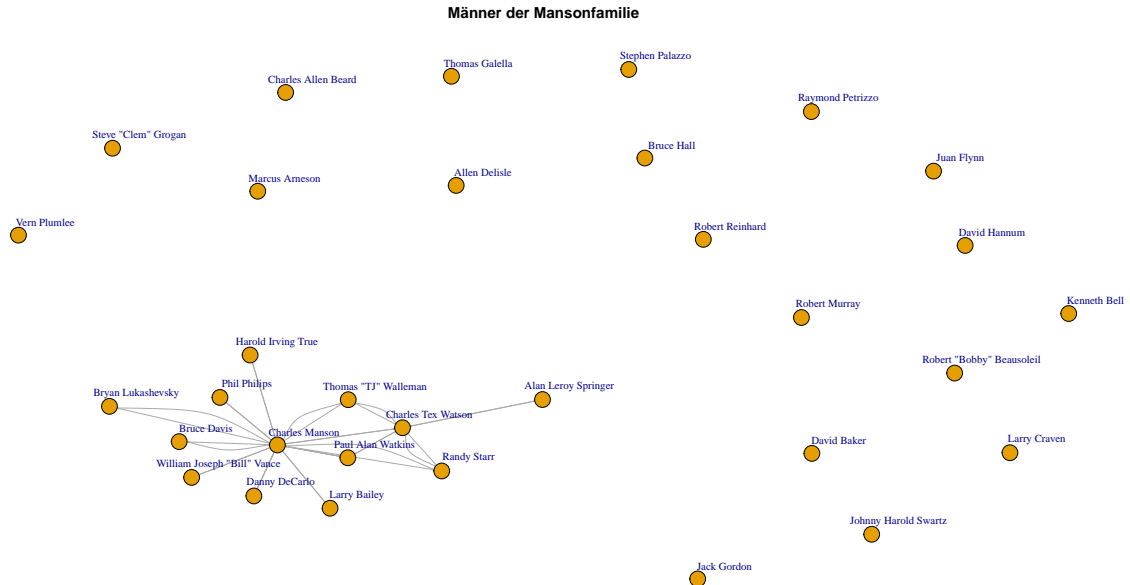
5,6% der Beziehungen zwischen den Knoten sind realisiert.

Eine Clusteranalyse wird hinfällig, da die Communities und Untergruppen gut einsehbar sind.

```
member_men <- delete.vertices(member, V(member)[(sex != 1)])
member_men
```

```
## IGRAPH dbe9194 DNWB 32 32 --
## + attr: name (v/c), type (v/n), sex (v/n), date_of_birth (v/c),
## | date_of_death (v/c), type_of_death (v/n), power (v/n),
## | relation_to_murder (v/n), member (v/n), relationship (e/c), weight
## | (e/c), year_beginning (e/c), year_end (e/c)
## + edges from dbe9194 (vertex names):
## [1] Alan Leroy Springer      ->Charles Tex Watson
## [2] William Joseph "Bill" Vance->Charles Manson
## [3] Bruce Davis              ->Charles Manson
## [4] Bryan Lukashevsky        ->Charles Manson
## [5] Charles Manson           ->Bruce Davis
## + ... omitted several edges
```

```
plot(
  member_men,
  asp = 0,
  rescale = T,
  vertex.size = 3,
  vertex.frame.width = 0.01,
  edge.width = 0.3,
  vertex.label.cex = 0.8,
  vertex.label.dist = 1,
  edge.arrow.size = .1,
  edge.curved = curve_multiple(member),
  main = "Männer der Mansonfamilie"
)
```



#Wie wahrscheinlich ist es, dass dieser Knoten die Verbindung zu anderen Knoten im Netzwerk herstellen.
`betweenness(member_men, normalized = T)`

##	Allen Delisle	Alan Leroy Springer
##	0.00000000	0.00000000
##	William Joseph "Bill" Vance	Robert "Bobby" Beausoleil
##	0.00000000	0.00000000
##	Bruce Davis	Bruce Hall
##	0.00000000	0.00000000
##	Bryan Lukashevsky	Charles Allen Beard
##	0.00000000	0.00000000
##	Charles Manson	Charles Tex Watson
##	0.10913978	0.03422939
##	David Baker	Danny DeCarlo
##	0.00000000	0.00000000
##	David Hannum	Harold Irving True
##	0.00000000	0.00000000
##	Jack Gordon	Johnny Harold Swartz
##	0.00000000	0.00000000
##	Juan Flynn	Kenneth Bell
##	0.00000000	0.00000000
##	Larry Bailey	Larry Craven
##	0.00000000	0.00000000
##	Marcus Arneson	Paul Alan Watkins
##	0.00000000	0.00000000
##	Phil Philips	Raymond Petrizzo
##	0.00000000	0.00000000
##	Randy Starr	Robert Murray
##	0.02652330	0.00000000
##	Robert Reinhard	Steve "Clem" Grogan
##	0.00000000	0.00000000

```
##          Stephen Palazzo          Thomas Galella
##          0.00000000          0.00000000
##      Thomas "TJ" Walleman      Vern Plumlee
##          0.00000000          0.00000000
```

```
#Wie schnell kann dieser Knoten alle anderen Knoten im Netzwerk erreichen? Hub bzw. Verteilerknoten für
test <-closeness(member_men, normalized=T)
```

```
## Warning in closeness(member_men, normalized = T): At centrality.c:
## 2617 :closeness centrality is not well-defined for disconnected graphs
```

```
#Der Befehl edge_density() prüft, wie hoch die Dichte des Netzwerks ist.
edge_density(member_men)
```

```
## [1] 0.03225806
```

Es gibt 32 Männer in der Mansonfamilie 3,2% der Beziehungen zwischen den Knoten sind realisiert.

```
zeit0 <- member
# Überprüfen der hinterlegten Daten
E(member)$year_beginning
```

```
## [1] "1969" "1969" "1968" "1969" "1967" " 99" " 99" " 99" "1969" " 99"
## [11] " 99" " 99" "1967" " 99" "1968" "1968" " 99" "1968" " 99" "1969"
## [21] "1968" " 99" " 99" "1969" " 99" " 99" "1968" "1969" "1968" "1972"
## [31] " 99" " 99" "1967" "1969" " 99" "1968" " 99" " 99" "1968" "1969"
## [41] "1970" "1967" "1967" "1967" "1967" "1968" " 99" "1966" "1968" "1967"
## [51] "1969" "1968" "1969" "1966" " 99" "1969" "1969" "1968" "1968" "1968"
## [61] "1968" "1969" "1970" "1969" "1968" "1968" "1968" "1968" "1972" "1968"
## [71] "1968" " 99" "1967" "1967" "1969" "1968" "1968" "1968" "1969" "1969"
## [81] " 99" " 99" " 99" "1968" " 99" "1968" " 99" " 99" "1968" "1968"
## [91] "1969" "1970" "1968" " 99" "1969" "1970" "1969" "1970" "1969" "1970"
## [101] "1969" "1970" "1969" "1970" "1967" " 99" "1967" "1969" "1966" " 99"
## [111] " 99" " 99" " 99" "1967" "1967" " 99" "1969" "1970" "1968" "1968"
## [121] "1968" "1968" "1961" "1961" " 99" "1968" "1968" "1969" "1966" "1968"
## [131] "1966" "1968" "1972" "1968" "1967" "1967" "1969" "1967" "1968" "1968"
## [141] " 99" "1969" "1970" " 99" "1967" " 99" "1968" "1969" " 99" " 99"
## [151] "1968"
```

```
#Jahre, welche zu betrachten sind 1961, 1966, 1967, 1968, 1969, 1970. Die Betrachtung läuft immer nach
```

```
## 1961
zeit1 <- subgraph.edges(zeit0, E(zeit0)[year_beginning == "1961"])

## 1966
zeit2 <-
  delete.edges(zeit0, E(zeit0)[(year_beginning != "1961") &
                                (year_beginning != "1966")])
zeit2 <- delete.vertices(zeit2, degree(zeit2) == 0)
E(zeit2)$year_beginning #Überprüfen der Daten
```

```
## [1] "1966" "1966" "1966" "1961" "1961" "1966" "1966"
```

```
## 1967
```

```
zeit3 <-  
  delete.edges(zeit0, E(zeit0)[(year_beginning != "1961") &  
                                (year_beginning != "1966") & (year_beginning != "1967")])  
zeit3 <- delete.vertices(zeit3, degree(zeit3) == 0)  
E(zeit3)$year_beginning #Überprüfen der Daten
```

```
## [1] "1967" "1967" "1967" "1967" "1967" "1967" "1967" "1966" "1967" "1966"  
## [11] "1967" "1967" "1967" "1967" "1966" "1967" "1967" "1961" "1961" "1966"  
## [21] "1966" "1967" "1967" "1967" "1967"
```

```
## 1968
```

```
zeit4 <-  
  delete.edges(zeit0, E(zeit0)[(year_beginning != "1961") &  
                                (year_beginning != "1966") &  
                                (year_beginning != "1967") & (year_beginning != "1968")])  
zeit4 <- delete.vertices(zeit4, degree(zeit4) == 0)  
E(zeit4)$year_beginning #Überprüfen der Daten
```

```
## [1] "1968" "1967" "1967" "1968" "1968" "1968" "1968" "1968" "1968" "1967"  
## [11] "1968" "1968" "1967" "1967" "1967" "1967" "1968" "1966" "1968" "1967"  
## [21] "1968" "1966" "1968" "1968" "1968" "1968" "1968" "1968" "1968" "1968"  
## [31] "1968" "1968" "1967" "1967" "1968" "1968" "1968" "1968" "1968" "1968"  
## [41] "1968" "1968" "1967" "1967" "1966" "1967" "1967" "1968" "1968" "1968"  
## [51] "1968" "1961" "1961" "1968" "1968" "1966" "1968" "1966" "1968" "1968"  
## [61] "1967" "1967" "1967" "1968" "1968" "1967" "1968" "1968"
```

```
## 1969
```

```
zeit5 <-  
  delete.edges(zeit0, E(zeit0)[(year_beginning != "1961") &  
                                (year_beginning != "1966") &  
                                (year_beginning != "1967") &  
                                (year_beginning != "1968") & (year_beginning != "1969")])  
zeit5 <- delete.vertices(zeit5, degree(zeit5) == 0)  
E(zeit5)$year_beginning #Überprüfen der Daten
```

```
## [1] "1969" "1969" "1968" "1969" "1967" "1969" "1967" "1968" "1968" "1968"  
## [11] "1969" "1968" "1969" "1968" "1969" "1968" "1967" "1969" "1968" "1968"  
## [21] "1969" "1967" "1967" "1967" "1967" "1968" "1966" "1968" "1967" "1969"  
## [31] "1968" "1969" "1966" "1969" "1969" "1968" "1968" "1968" "1968" "1969"  
## [41] "1969" "1968" "1968" "1968" "1968" "1968" "1968" "1967" "1967" "1969"  
## [51] "1968" "1968" "1968" "1969" "1969" "1968" "1968" "1968" "1968" "1969"  
## [61] "1968" "1969" "1969" "1969" "1969" "1969" "1967" "1967" "1969" "1966"  
## [71] "1967" "1967" "1969" "1968" "1968" "1968" "1968" "1961" "1961" "1968"  
## [81] "1968" "1969" "1966" "1968" "1966" "1968" "1968" "1967" "1967" "1969"  
## [91] "1967" "1968" "1968" "1969" "1967" "1968" "1969" "1968"
```

```
## 1970
```

```
zeit6 <-  
  delete.edges(zeit0, E(zeit0)[(year_beginning != "1961") &
```

```

        (year_beginning != "1966") &
        (year_beginning != "1967") &
        (year_beginning != "1968") &
        (year_beginning != "1969") & (year_beginning != "1970"]])
zeit6 <- delete.vertices(zeit6, degree(zeit6) == 0)
E(zeit6)$year_beginning #Überprüfen der Daten

```

```

## [1] "1969" "1969" "1968" "1969" "1967" "1969" "1967" "1968" "1968" "1968"
## [11] "1969" "1968" "1969" "1968" "1969" "1968" "1967" "1969" "1968" "1968"
## [21] "1969" "1970" "1967" "1967" "1967" "1967" "1968" "1966" "1968" "1967"
## [31] "1969" "1968" "1969" "1966" "1969" "1969" "1968" "1968" "1968" "1968"
## [41] "1969" "1970" "1969" "1968" "1968" "1968" "1968" "1968" "1968" "1967"
## [51] "1967" "1969" "1968" "1968" "1968" "1969" "1969" "1968" "1968" "1968"
## [61] "1968" "1969" "1970" "1968" "1969" "1970" "1969" "1970" "1969" "1970"
## [71] "1969" "1970" "1969" "1970" "1967" "1967" "1969" "1966" "1967" "1967"
## [81] "1969" "1970" "1968" "1968" "1968" "1968" "1961" "1961" "1968" "1968"
## [91] "1969" "1966" "1968" "1966" "1968" "1968" "1967" "1967" "1969" "1967"
## [101] "1968" "1968" "1969" "1970" "1967" "1968" "1969" "1968"

```

```

zeit7 <- delete.vertices(zeit0, degree(zeit0) == 0)

# Plotten der Daten
zoom <- layout_nicely(zeit0)
zoom <- norm_coords(zoom, ymin = -1, ymax = 1, xmin = -1, xmax = 1)

zeitstrahl <- layout.kamada.kawai(zeit0)
zeitstrahl <- ((vertex.size= 88) & (edge.arrow.size = 0.2))
zeitstrahl <- (edge.color= "gray70")

#plot(zeit1, zeitstrahl)

plot(
  zeit1,
  asp = 0,
  layout = layout_with_kk,
  main = 1961,
  vertex.size = 8,
  edge.arrow.size = 0.2,
  edge.color = gray(.8, alpha=.7)
)

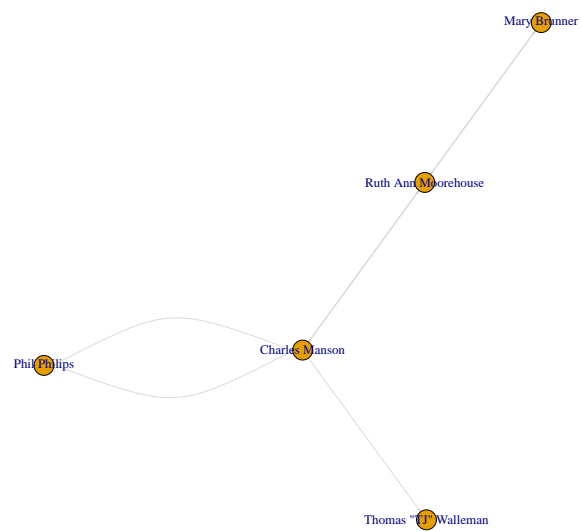
```

1961



```
plot(  
  zeit2,  
  layout = layout_with_kk,  
  main = 1966,  
  vertex.size = 8,  
  edge.arrow.size = 0.2,  
  edge.color = gray(.8, alpha=.7)  
)
```

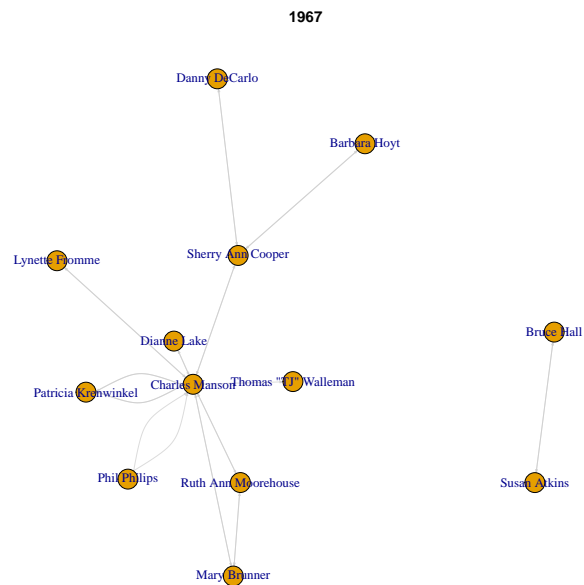
1966



```

plot(
  zeit3,
  layout = layout_with_kk,
  main = 1967,
  vertex.size = 8,
  edge.arrow.size = 0.2,
  edge.color = gray(.8, alpha=.7)
)

```

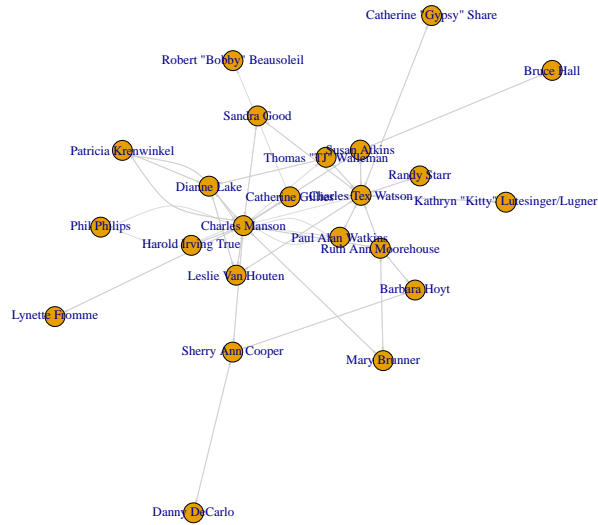


```

plot(
  zeit4,
  layout = layout_with_kk,
  main = 1968,
  vertex.size = 8,
  edge.arrow.size = 0.2,
  edge.color = gray(.8, alpha=.7)
)

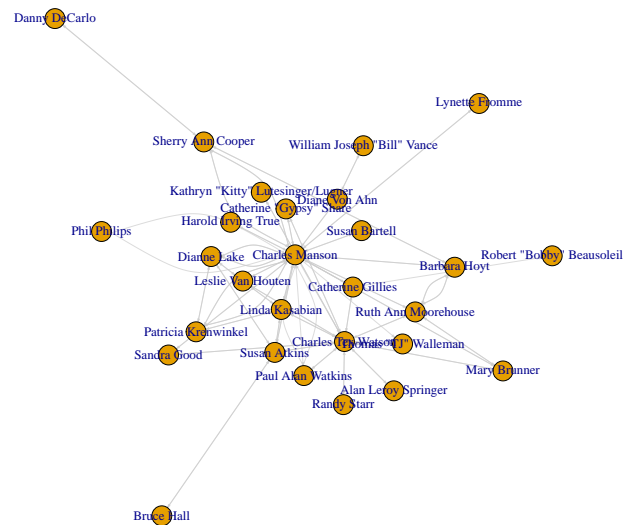
```


1968



```
plot(
  zeit5,
  layout = layout_with_kk,
  main = 1969,
  vertex.size = 8,
  edge.arrow.size = 0.2,
  edge.color = gray(.8, alpha=.7)
)
```

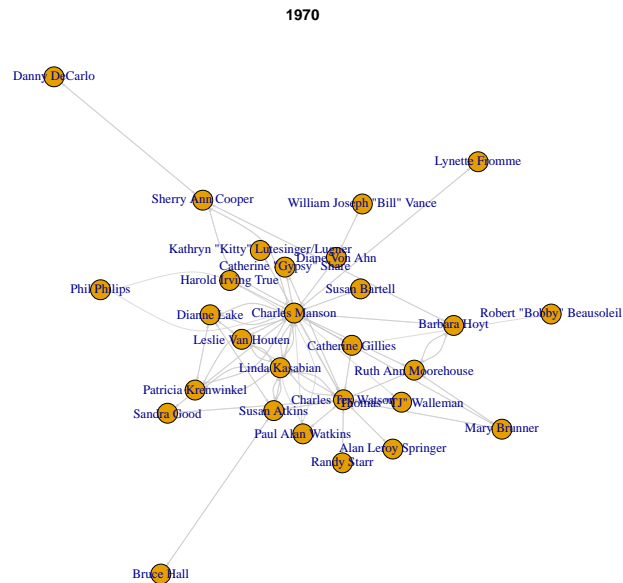
1969



```

plot(
  zeit6,
  layout = layout_with_kk,
  main = 1970,
  vertex.size = 8,
  edge.arrow.size = 0.2,
  edge.color = gray(.8, alpha=.7)
)

```

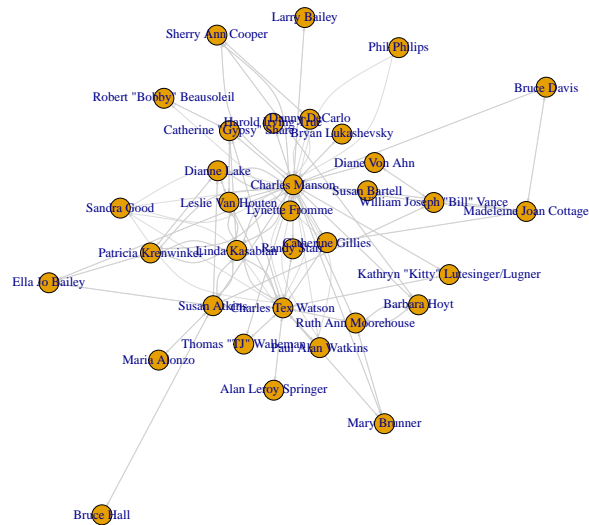


```

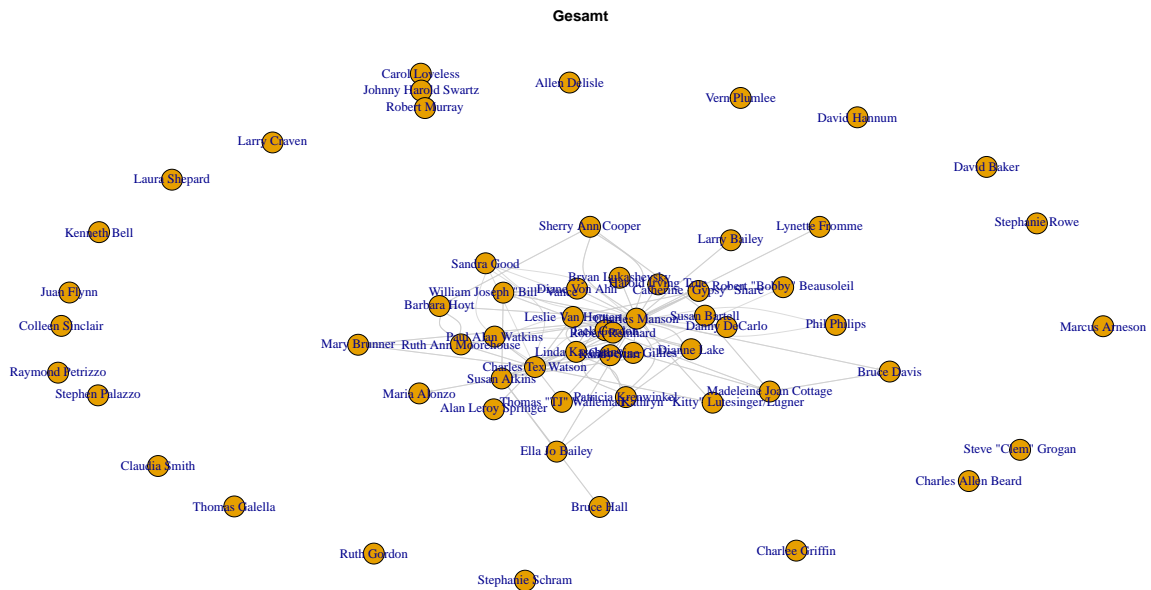
plot(
  zeit7,
  layout = layout_with_kk,
  main = "Ohne Isolates",
  vertex.size = 8,
  edge.arrow.size = 0.2,
  edge.color = gray(.8, alpha=.7)
)

```

Ohne Isolates



```
plot(
  zeit0,
  asp = 0,
  layout = layout_with_kk,
  vertex.size = 4,
  main = "Gesamt",
  edge.arrow.size = 0.2,
  edge.color = gray(.8, alpha=.7)
)
```



```
# Berechnung der Dichte für die Netzwerke
edge_density(zeit0)
```

```
## [1] 0.04412624
```

```
edge_density(zeit1)
```

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

```
edge_density(zeit2)
```

```
## [1] 0.35
```

```
edge_density(zeit3)
```

```
## [1] 0.1602564
```

```
edge_density(zeit4)
```

```
## [1] 0.1343874
```

```
edge_density(zeit5)
```

```
## [1] 0.1296296
```

```
edge_density(zeit6)
```

```
## [1] 0.1428571
```

```
edge_density(zeit7)
```

```
## [1] 0.1345811
```

```
liebe <- delete.vertices(manson, V(manson)[member != "2"])
```

```
liebe
```

```
## IGRAPH dc15d70 DNWB 59 151 --  
## + attr: name (v/c), type (v/n), sex (v/n), date_of_birth (v/c),  
## | date_of_death (v/c), type_of_death (v/n), power (v/n),  
## | relation_to_murder (v/n), member (v/n), relationship (e/c), weight  
## | (e/c), year_beginning (e/c), year_end (e/c)  
## + edges from dc15d70 (vertex names):  
## [1] Alan Leroy Springer      ->Charles Tex Watson  
## [2] Barbara Hoyt              ->Charles Manson  
## [3] Barbara Hoyt              ->Ruth Ann Moorehouse  
## [4] Barbara Hoyt              ->Ruth Ann Moorehouse  
## [5] Barbara Hoyt              ->Sherry Ann Cooper  
## + ... omitted several edges
```

```
edge.attributes(liebe)
```

```
## $relationship  
## [1] " 1" " 3" " 2" " 2" " 2" " 2" " 2" " 2" " 1" " 2" " 2" " 5" " 1" " 2"  
## [16] " 1" " 2" " 2" " 1" " 2" " 2" " 2" " 2" " 7" " 2" " 1" " 2" " 2" " 2" " 3"  
## [31] " 2" " 1" " 2" " 3" "99" " 1" " 3" " 2" " 2" " 2" " 3" " 2" " 5" " 2" " 2"  
## [46] " 2" " 1" " 2" " 5" " 2" " 7" " 5" " 2" " 2" " 2" " 2" " 1" " 1" " 2" " 1"  
## [61] " 2" " 2" " 3" " 5" " 2" " 1" " 1" " 2" " 3" " 5" " 2" " 2" " 5" " 2" " 3"  
## [76] " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 1" " 1" " 1" " 2" " 2" " 2" " 2"  
## [91] " 2" " 3" " 2" " 2" " 2" " 3" " 2" " 3" " 2" " 3" " 2" " 3" " 2" " 3" " 2"  
## [106] " 2" " 5" " 5" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 3" " 2" " 2"  
## [121] " 2" " 2" " 1" " 1" " 1" " 1" " 2" " 3" " 2" " 1" " 2" " 5" " 3" " 2" " 2"  
## [136] " 2" " 3" " 5" " 5" " 5" " 2" " 2" " 3" " 2" " 5" " 2" " 2" " 2" " 2" " 2"  
## [151] " 2"  
##  
## $weight  
## [1] " 1" " 2" " 1" " 1" " 3" " 3" " 2" " 2" " 1" " 1" " 3" " 2" " 3" " 1" " 3"  
## [16] " 1" " 2" " 2" " 1" " 3" " 3" " 2" " 2" " 3" " 3" " 1" " 1" " 1" " 3" " 2"  
## [31] " 3" " 2" " 1" " 3" "99" " 1" " 2" " 2" " 1" " 1" " 1" " 3" " 3" " 2" " 3"  
## [46] " 3" " 1" " 3" " 3" " 2" " 3" " 2" " 1" " 2" " 3" " 1" " 1" " 1" " 3" " 2"  
## [61] " 3" " 1" " 1" " 2" " 1" " 1" " 1" " 2" " 2" " 1" " 1" " 1" " 3" " 3" " 3"  
## [76] " 2" " 2" " 2" " 1" " 2" " 3" " 2" " 2" " 1" " 3" " 2" " 2" " 2" " 1" " 3"  
## [91] " 1" " 1" " 2" " 2" " 3" " 3" " 1" " 1" " 1" " 1" " 1" " 1" " 1" " 1" " 3"  
## [106] " 1" " 3" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 3" " 2" " 1" " 1" " 2" " 2"  
## [121] " 2" " 1" " 2" " 2" " 1" " 1" " 1" " 3" " 2" " 2" " 2" " 2" " 2" " 2" " 3"  
## [136] " 2" " 2" " 3" " 2" " 2" " 2" " 1" " 1" " 1" " 3" " 2" " 2" " 2" " 2" " 2"
```

```

## [151] " 2"
##
## $year_beginning
## [1] "1969" "1969" "1968" "1969" "1967" " 99" " 99" " 99" "1969" " 99"
## [11] " 99" " 99" "1967" " 99" "1968" "1968" " 99" "1968" " 99" "1969"
## [21] "1968" " 99" " 99" "1969" " 99" " 99" "1968" "1969" "1968" "1972"
## [31] " 99" " 99" "1967" "1969" " 99" "1968" " 99" " 99" "1968" "1969"
## [41] "1970" "1967" "1967" "1967" "1967" "1968" " 99" "1966" "1968" "1967"
## [51] "1969" "1968" "1969" "1966" " 99" "1969" "1969" "1968" "1968" "1968"
## [61] "1968" "1969" "1970" "1969" "1968" "1968" "1968" "1968" "1972" "1968"
## [71] "1968" " 99" "1967" "1967" "1969" "1968" "1968" "1968" "1969" "1969"
## [81] " 99" " 99" " 99" "1968" " 99" "1968" " 99" " 99" "1968" "1968"
## [91] "1969" "1970" "1968" " 99" "1969" "1970" "1969" "1970" "1969" "1970"
## [101] "1969" "1970" "1969" "1970" "1967" " 99" "1967" "1969" "1966" " 99"
## [111] " 99" " 99" " 99" "1967" "1967" " 99" "1969" "1970" "1968" "1968"
## [121] "1968" "1968" "1961" "1961" " 99" "1968" "1968" "1969" "1966" "1968"
## [131] "1966" "1968" "1972" "1968" "1967" "1967" "1969" "1967" "1968" "1968"
## [141] " 99" "1969" "1970" " 99" "1967" " 99" "1968" "1969" " 99" " 99"
## [151] "1968"
##
## $year_end
## [1] "1971" " 99" "1969" "1969" " 99" "1969" "1969" " 99" "1969" " 99"
## [11] " 99" "1969" "1969" " 99" " 99" " 99" "1969" "1969" " 99" "1980"
## [21] " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99"
## [31] "1969" " 99" "1969" " 99" " 99" " 99" " 99" " 99" " 99" "1970"
## [41] "1971" " 99" "1977" " 99" " 99" "1969" " 99" " 99" "1971" "1969"
## [51] " 99" " 99" " 99" " 99" " 99" "1969" "1969" "1971" " 99" " 99"
## [61] " 99" "1970" "1971" "1969" "1969" "1969" " 99" "1971" "2011" "1969"
## [71] "1969" "1969" "1975" "1969" "1969" "1969" "1969" "1969" "1969" "1969"
## [81] " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99"
## [91] "1970" "1971" "1969" " 99" "1970" "1971" "1970" "1971" "1970" "1971"
## [101] "1970" "1971" "1970" "1971" " 99" " 99" "1977" "1969" "1966" "1969"
## [111] "1969" "1969" " 99" " 99" " 99" " 99" "1970" "1971" "1969" "1969"
## [121] "1969" "1969" "1965" "1965" " 99" "1969" "1969" "1969" " 99" " 99"
## [131] "1966" "1971" "2011" "1971" " 99" "1969" " 99" "1975" " 99" "1969"
## [141] " 99" "1970" "1971" " 99" "1969" "1969" "1969" " 99" "1969" " 99"
## [151] "1969"

```

```

liebe <- delete_edge_attr(liebe, "year_beginning")
liebe <- delete_edge_attr(liebe, "weight")
liebe <- delete_edge_attr(liebe, "year_end")

```

```

edge.attributes(liebe)

```

```

## $relationship
## [1] " 1" " 3" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 1" " 2" " 2" " 5" " 1" " 2"
## [16] " 1" " 2" " 2" " 1" " 2" " 2" " 2" " 2" " 7" " 2" " 1" " 2" " 2" " 2" " 2" " 3"
## [31] " 2" " 1" " 2" " 3" "99" " 1" " 3" " 2" " 2" " 2" " 3" " 2" " 5" " 2" " 2"
## [46] " 2" " 1" " 2" " 5" " 2" " 7" " 5" " 2" " 2" " 2" " 2" " 1" " 1" " 2" " 1"
## [61] " 2" " 2" " 3" " 5" " 2" " 1" " 1" " 2" " 3" " 5" " 2" " 2" " 5" " 2" " 3"
## [76] " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 1" " 1" " 1" " 2" " 2" " 2" " 2"
## [91] " 2" " 3" " 2" " 2" " 2" " 3" " 2" " 3" " 2" " 3" " 2" " 3" " 2" " 3" " 2"
## [106] " 2" " 5" " 5" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 3" " 2" " 2"
## [121] " 2" " 2" " 1" " 1" " 1" " 1" " 2" " 3" " 2" " 1" " 2" " 5" " 3" " 2" " 2"

```

```
## [136] " 2" " 3" " 5" " 5" " 5" " 2" " 2" " 3" " 2" " 5" " 2" " 2" " 2" " 2" " 2"
## [151] " 2"
```

```
liebe1 <- delete.edges(liebe, E(liebe)[(relationship != " 5")])
edge.attributes(liebe1)
```

```
## $relationship
## [1] " 5" " 5" " 5" " 5" " 5" " 5" " 5" " 5" " 5" " 5" " 5" " 5" " 5" " 5" " 5"
```

```
liebe1 <- delete.vertices(liebe1, degree(liebe1) == 0)
```

```
plot(
  liebe1,
  layout = layout_with_fr,
  edge.arrow.size = .4,
  main = "Liebesbeziehungen",
  vertex.label.dist = 2.5
)
```



```
abneigung <- delete.vertices(manson, V(manson)[member != "2"])
```

```
abneigung
```

```
## IGRAPH dc1c281 DNWB 59 151 --
```

```
## + attr: name (v/c), type (v/n), sex (v/n), date_of_birth (v/c),
## | date_of_death (v/c), type_of_death (v/n), power (v/n),
## | relation_to_murder (v/n), member (v/n), relationship (e/c), weight
## | (e/c), year_beginning (e/c), year_end (e/c)
## + edges from dc1c281 (vertex names):
## [1] Alan Leroy Springer      ->Charles Tex Watson
## [2] Barbara Hoyt             ->Charles Manson
## [3] Barbara Hoyt             ->Ruth Ann Moorehouse
## [4] Barbara Hoyt             ->Ruth Ann Moorehouse
## [5] Barbara Hoyt             ->Sherry Ann Cooper
## + ... omitted several edges
```

```
edge.attributes(abneigung)
```

```
## $relationship
## [1] " 1" " 3" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 1" " 2" " 2" " 5" " 1" " 2"
## [16] " 1" " 2" " 2" " 1" " 2" " 2" " 2" " 2" " 7" " 2" " 1" " 2" " 2" " 2" " 2" " 3"
## [31] " 2" " 1" " 2" " 3" "99" " 1" " 3" " 2" " 2" " 2" " 3" " 2" " 5" " 2" " 2"
## [46] " 2" " 1" " 2" " 5" " 2" " 7" " 5" " 2" " 2" " 2" " 2" " 1" " 1" " 2" " 1"
## [61] " 2" " 2" " 3" " 5" " 2" " 1" " 1" " 2" " 3" " 5" " 2" " 2" " 5" " 2" " 3"
## [76] " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 1" " 1" " 1" " 2" " 2" " 2" " 2"
## [91] " 2" " 3" " 2" " 2" " 2" " 3" " 2" " 3" " 2" " 3" " 2" " 3" " 2" " 3" " 2"
## [106] " 2" " 5" " 5" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 3" " 2" " 2"
## [121] " 2" " 2" " 1" " 1" " 1" " 1" " 2" " 3" " 2" " 1" " 2" " 5" " 3" " 2" " 2"
## [136] " 2" " 3" " 5" " 5" " 5" " 2" " 2" " 3" " 2" " 5" " 2" " 2" " 2" " 2" " 2"
## [151] " 2"
##
## $weight
## [1] " 1" " 2" " 1" " 1" " 3" " 3" " 2" " 2" " 1" " 1" " 3" " 2" " 3" " 1" " 3"
## [16] " 1" " 2" " 2" " 1" " 3" " 3" " 2" " 2" " 3" " 3" " 1" " 1" " 1" " 3" " 2"
## [31] " 3" " 2" " 1" " 3" "99" " 1" " 2" " 2" " 1" " 1" " 1" " 3" " 3" " 2" " 3"
## [46] " 3" " 1" " 3" " 3" " 2" " 3" " 2" " 1" " 2" " 3" " 1" " 1" " 1" " 3" " 2"
## [61] " 3" " 1" " 1" " 2" " 1" " 1" " 1" " 2" " 2" " 1" " 1" " 1" " 3" " 3" " 3"
## [76] " 2" " 2" " 2" " 1" " 2" " 3" " 2" " 2" " 1" " 3" " 2" " 2" " 2" " 1" " 3"
## [91] " 1" " 1" " 2" " 2" " 3" " 3" " 1" " 1" " 1" " 1" " 1" " 1" " 1" " 1" " 3"
## [106] " 1" " 3" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 3" " 2" " 1" " 1" " 2" " 2"
## [121] " 2" " 1" " 2" " 2" " 1" " 1" " 1" " 3" " 2" " 2" " 2" " 2" " 2" " 2" " 3"
## [136] " 2" " 2" " 3" " 2" " 2" " 2" " 1" " 1" " 1" " 3" " 2" " 2" " 2" " 2" " 2"
## [151] " 2"
##
## $year_beginning
## [1] "1969" "1969" "1968" "1969" "1967" " 99" " 99" " 99" "1969" " 99"
## [11] " 99" " 99" "1967" " 99" "1968" "1968" " 99" "1968" " 99" "1969"
## [21] "1968" " 99" " 99" "1969" " 99" " 99" "1968" "1969" "1968" "1972"
## [31] " 99" " 99" "1967" "1969" " 99" "1968" " 99" " 99" "1968" "1969"
## [41] "1970" "1967" "1967" "1967" "1967" "1968" " 99" "1966" "1968" "1967"
## [51] "1969" "1968" "1969" "1966" " 99" "1969" "1969" "1968" "1968" "1968"
## [61] "1968" "1969" "1970" "1969" "1968" "1968" "1968" "1968" "1972" "1968"
## [71] "1968" " 99" "1967" "1967" "1969" "1968" "1968" "1968" "1969" "1969"
## [81] " 99" " 99" " 99" "1968" " 99" "1968" " 99" " 99" "1968" "1968"
## [91] "1969" "1970" "1968" " 99" "1969" "1970" "1969" "1970" "1969" "1970"
## [101] "1969" "1970" "1969" "1970" "1967" " 99" "1967" "1969" "1966" " 99"
## [111] " 99" " 99" " 99" "1967" "1967" " 99" "1969" "1970" "1968" "1968"
## [121] "1968" "1968" "1961" "1961" " 99" "1968" "1968" "1969" "1966" "1968"
```



```
## [131] "1966" "1968" "1972" "1968" "1967" "1967" "1969" "1967" "1968" "1968"
## [141] " 99" "1969" "1970" " 99" "1967" " 99" "1968" "1969" " 99" " 99"
## [151] "1968"
##
## $year_end
## [1] "1971" " 99" "1969" "1969" " 99" "1969" "1969" " 99" "1969" " 99"
## [11] " 99" "1969" "1969" " 99" " 99" " 99" "1969" "1969" " 99" "1980"
## [21] " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99"
## [31] "1969" " 99" "1969" " 99" " 99" " 99" " 99" " 99" " 99" "1970"
## [41] "1971" " 99" "1977" " 99" " 99" "1969" " 99" " 99" "1971" "1969"
## [51] " 99" " 99" " 99" " 99" " 99" "1969" "1969" "1971" " 99" " 99"
## [61] " 99" "1970" "1971" "1969" "1969" "1969" " 99" "1971" "2011" "1969"
## [71] "1969" "1969" "1975" "1969" "1969" "1969" "1969" "1969" "1969" "1969"
## [81] " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99" " 99"
## [91] "1970" "1971" "1969" " 99" "1970" "1971" "1970" "1971" "1970" "1971"
## [101] "1970" "1971" "1970" "1971" " 99" " 99" "1977" "1969" "1966" "1969"
## [111] "1969" "1969" " 99" " 99" " 99" " 99" "1970" "1971" "1969" "1969"
## [121] "1969" "1969" "1965" "1965" " 99" "1969" "1969" "1969" " 99" " 99"
## [131] "1966" "1971" "2011" "1971" " 99" "1969" " 99" "1975" " 99" "1969"
## [141] " 99" "1970" "1971" " 99" "1969" "1969" "1969" " 99" "1969" " 99"
## [151] "1969"
```

```
abneigung <- delete_edge_attr(abneigung, "year_beginning")
abneigung <- delete_edge_attr(abneigung, "weight")
abneigung <- delete_edge_attr(abneigung, "year_end")
```

```
abneigung
```

```
## IGRAPH dc1c281 DN-B 59 151 --
## + attr: name (v/c), type (v/n), sex (v/n), date_of_birth (v/c),
## | date_of_death (v/c), type_of_death (v/n), power (v/n),
## | relation_to_murder (v/n), member (v/n), relationship (e/c)
## + edges from dc1c281 (vertex names):
## [1] Alan Leroy Springer      ->Charles Tex Watson
## [2] Barbara Hoyt             ->Charles Manson
## [3] Barbara Hoyt             ->Ruth Ann Moorehouse
## [4] Barbara Hoyt             ->Ruth Ann Moorehouse
## [5] Barbara Hoyt             ->Sherry Ann Cooper
## [6] William Joseph "Bill" Vance->Charles Manson
## + ... omitted several edges
```

```
edge.attributes(abneigung)
```

```
## $relationship
## [1] " 1" " 3" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 1" " 2" " 2" " 5" " 1" " 2"
## [16] " 1" " 2" " 2" " 1" " 2" " 2" " 2" " 2" " 2" " 7" " 2" " 1" " 2" " 2" " 2" " 3"
## [31] " 2" " 1" " 2" " 3" "99" " 1" " 3" " 2" " 2" " 2" " 3" " 2" " 5" " 2" " 2" " 2"
## [46] " 2" " 1" " 2" " 5" " 2" " 7" " 5" " 2" " 2" " 2" " 2" " 1" " 1" " 2" " 1" " 1"
## [61] " 2" " 2" " 3" " 5" " 2" " 1" " 1" " 2" " 3" " 5" " 2" " 2" " 5" " 2" " 3" " 2"
## [76] " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 1" " 1" " 1" " 2" " 2" " 2" " 2"
## [91] " 2" " 3" " 2" " 2" " 2" " 3" " 2" " 3" " 2" " 3" " 2" " 3" " 2" " 3" " 2" " 2"
## [106] " 2" " 5" " 5" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 2" " 3" " 2" " 2"
## [121] " 2" " 2" " 1" " 1" " 1" " 1" " 2" " 3" " 2" " 1" " 2" " 5" " 3" " 2" " 2" " 2"
```

```
## [136] " 2" " 3" " 5" " 5" " 5" " 2" " 2" " 3" " 2" " 5" " 2" " 2" " 2" " 2" " 2"
## [151] " 2"
```

```
abneigung1 <-
  delete.edges(abneigung, E(abneigung)[(relationship != " 3")])
edge.attributes(abneigung1)
```

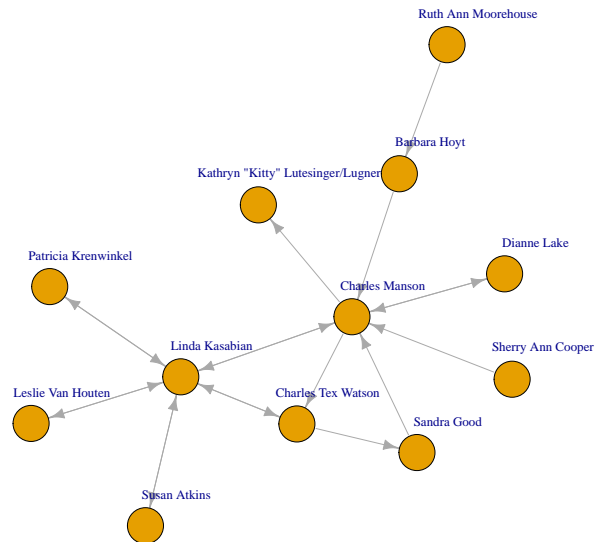
```
## $relationship
## [1] " 3" " 3" " 3" " 3" " 3" " 3" " 3" " 3" " 3" " 3" " 3" " 3" " 3" " 3" " 3"
## [16] " 3" " 3" " 3" " 3"
```

```
abneigung1
```

```
## IGRAPH dc1dfb9 DN-B 59 19 --
## + attr: name (v/c), type (v/n), sex (v/n), date_of_birth (v/c),
## | date_of_death (v/c), type_of_death (v/n), power (v/n),
## | relation_to_murder (v/n), member (v/n), relationship (e/c)
## + edges from dc1dfb9 (vertex names):
## [1] Barbara Hoyt      ->Charles Manson
## [2] Charles Manson   ->Charles Tex Watson
## [3] Charles Manson   ->Dianne Lake
## [4] Charles Manson   ->Kathryn "Kitty" Lutesinger/Lugner
## [5] Charles Manson   ->Linda Kasabian
## [6] Charles Tex Watson->Linda Kasabian
## + ... omitted several edges
```

```
abneigung1 <- delete.vertices(abneigung1, degree(abneigung1) == 0)
plot(
  abneigung1,
  layout = layout_with_kk,
  edge.arrow.size = 1,
  main = "Abneigung",
  vertex.label.dist = 2.5
)
```

Abneigung

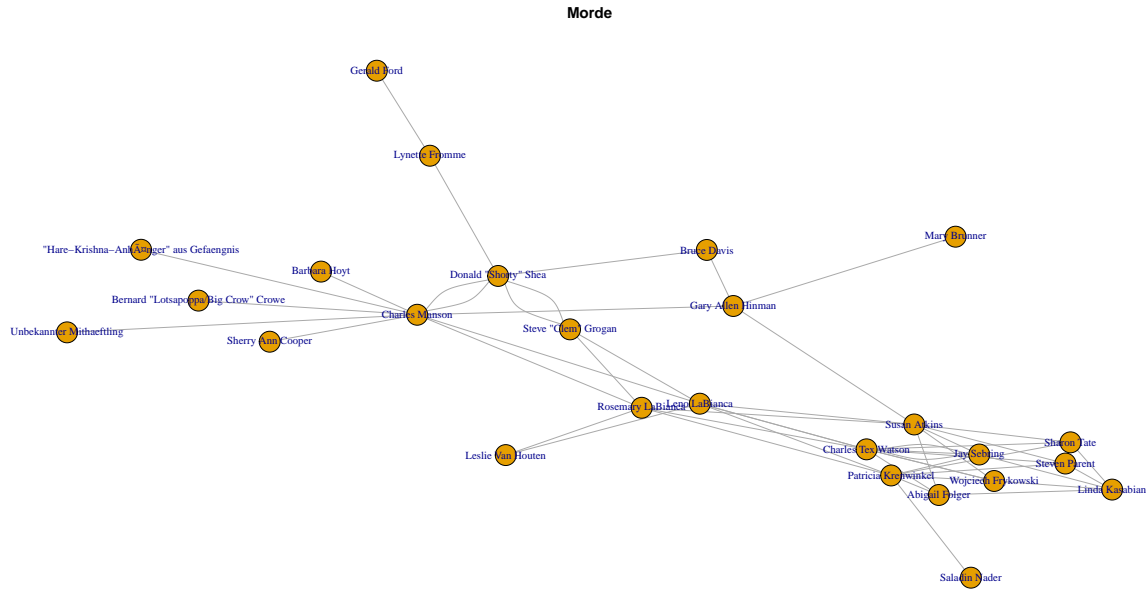


Morde

```

# Überprüfen der hinterlegten Daten
morde <-
  delete.edges(manson, E(manson)[(relationship != " 7") &
                                   (relationship != " 6")])
morde <- delete.vertices(morde, degree(morde) == 0)
plot (
  morde,
  asp = 0,
  rescale = T,
  vertex.size = 4,
  vertex.frame.width = 0.01,
  edge.width = 0.3,
  vertex.label.cex = 0.8,
  edge.arrow.size = .1,
  main = "Morde"
)

```



#Löschen aller Knoten, außer der Mord beteiligten

```
tate <-
  delete.vertices(manson, V(manson)[(name != "Sharon Tate") & (name != "Wojciech Frykowski") &
    (name != "Abigail Folger") &
    (name != "Steven Parent") &
    (name != "Susan Atkins") &
    (name != "Linda Kasabian") &
    (name != "Patricia Krenwinkel") &
    (name != "Charles Tex Watson")])

tate
```

```
## IGRAPH dc27a5b DNWB 8 37 --
## + attr: name (v/c), type (v/n), sex (v/n), date_of_birth (v/c),
## | date_of_death (v/c), type_of_death (v/n), power (v/n),
## | relation_to_murder (v/n), member (v/n), relationship (e/c), weight
## | (e/c), year_beginning (e/c), year_end (e/c)
## + edges from dc27a5b (vertex names):
## [1] Abigail Folger    -> Sharon Tate
## [2] Abigail Folger    -> Wojciech Frykowski
## [3] Charles Tex Watson->Abigail Folger
## [4] Charles Tex Watson->Linda Kasabian
## [5] Charles Tex Watson->Linda Kasabian
## + ... omitted several edges
```

Mansonfamilie rot eingefärbt

```
V(tate)[V(tate)$member == 2]$color <- "red"
V(tate)[V(tate)$type_of_death == 3]$color <- "blue"

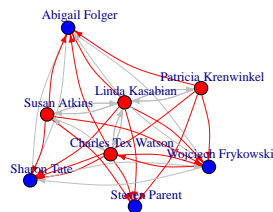
color_node <- V(tate)[V(tate)$type_of_death == 3]$color <- "blue"
```

```

E(tate)$color <- "gray"
E(tate)[E(tate)$relationship == " 6"]$color <- "red"

plot(
  tate,
  layout = layout_with_kk,
  edge.arrow.size = .5,
  vertex.label.dist = 2.5,
  edge.curved=.2
)

```



```

#Löschen aller Knoten, außer der Mordbeteiligten
LaBianca <-
  delete.vertices(manson, V(manson)[(name != "Charles Manson") &
    (name != "Charles Tex Watson") &
    (name != "Patricia Krenwinkel") &
    (name != "Leslie Van Houten") &
    (name != "Leno LaBianca") &
    (name != "Rosemary LaBianca")])

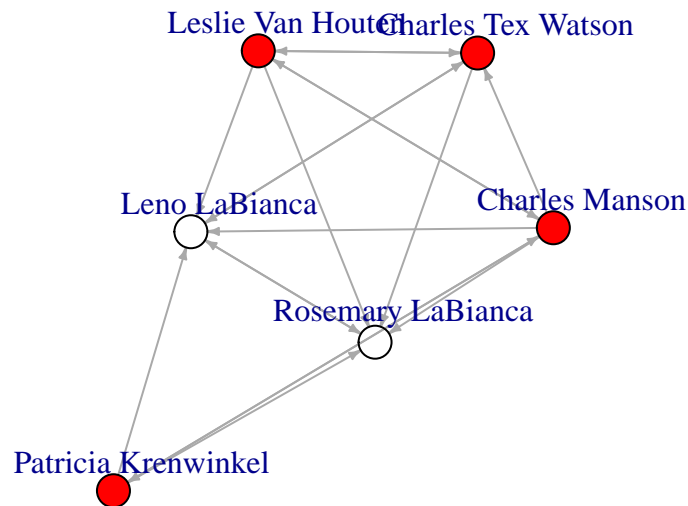
# Mansonfamilie rot eingefärbt
V(LaBianca)[V(LaBianca)$member == 2]$color <- "red"

#Doppelte Kanten herauslöschen
LaBianca <-
  delete.edges(LaBianca, E(LaBianca)[which_multiple(LaBianca, eids = E(LaBianca))])

plot(
  LaBianca,
  layout = layout_with_kk,
  edge.arrow.size = .3,
  vertex.label.dist = 2.5,
  main = "LaBianca Mord",
  sub = "Rot Member"
)

```

LaBianca Mord



Rot Member

Wir schauen uns im genaueren den LaBianca Mord an

Cliquen

```
# Cliquen entsprechen der Triade 300, d.h. einer komplett reziproken Struktur (A<->B<->C,A<->C).
```

```
# Zählt die Cliquen im Netzwerk
```

```
clique_num(member)
```

```
## Warning in clique_num(member): At cliques.c:1087 :directionality of edges is
```

```
## ignored for directed graphs
```

```
## [1] 5
```

```
# listet die Cliquen im Netzwerk auf
```

```
lc <- largest_cliques(member)
```

```
## Warning in largest_cliques(member): At cliques.c:1087 :directionality of edges
```

```
## is ignored for directed graphs
```

```
lc
```

```
## [[1]]
## + 5/59 vertices, named, from db87048:
## [1] Charles Manson          Charles Tex Watson      Linda Kasabian
## [4] Catherine "Gypsy" Share Leslie Van Houten
```

```
# Umwandlung für Visualisierung
```

```
familie_lc <- subgraph(member, lc[[1]])
```

```
## Warning in subgraph(member, lc[[1]]): At structural_properties.c:
## 1984 :igraph_subgraph is deprecated from igraph 0.6, use igraph_induced_subgraph
## instead
```

```
#Doppelte Kanten herauslöschen
```

```
familie_lc <-
  delete.edges(familie_lc, E(familie_lc)[which_multiple(familie_lc, eids = E(familie_lc))])
```

```
vertex.attributes(familie_lc)
```

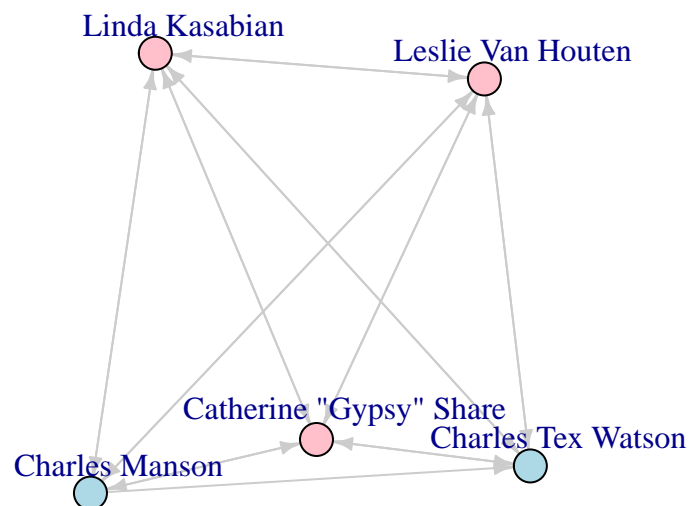
```
## $name
## [1] "Catherine \"Gypsy\" Share" "Charles Manson"
## [3] "Charles Tex Watson"      "Leslie Van Houten"
## [5] "Linda Kasabian"
##
## $type
## [1] 1 1 1 1 1
##
## $sex
## [1] 2 1 1 2 2
##
## $date_of_birth
## [1] "10-12-1942" "12-11-1934" "02-12-1945" "23-08-1949" "21-06-1949"
##
## $date_of_death
## [1] "99"      "19-11-2017" "99"      "99"      "99"
##
## $type_of_death
## [1] 99 2 1 1 1
##
## $power
## [1] 2 5 4 2 2
##
## $relation_to_murder
## [1] 1 3 3 3 1
##
## $member
## [1] 2 2 2 2 2
```

```
V(familie_lc)[V(familie_lc)$sex == 1]$color <- "lightblue" # Mann
V(familie_lc)[V(familie_lc)$sex == 2]$color <- "pink" # Frau
```

```
plot(
```

```
familie_lc,
layout = layout.fruchterman.reingold,
edge.arrow.size = 0.5,
edge.color = "gray80",
main = "Größte Clique in der Manson Familie",
vertex.label.dist = 2.5
)
```

Größte Clique in der Manson Familie



Im Netzwerk der Mansonfamilie gibt es nur eine Clique, mit 5 Knoten.

```
# Triadenzensus
count_triangles(member)
```

```
## [1] 0 0 2 2 0 0 0 0 1 0 6 0 0 28 17 0 0 0 1 0 3 1 2 0 0
## [26] 0 0 1 0 0 0 0 7 9 0 0 0 3 0 3 1 0 0 1 0 0 0 4 1 2
## [51] 0 0 0 0 6 0 0 1 0
```

hier wird ausgegeben, welcher Knoten an wie vielen "Dreiecken" beteiligt ist (Gewichtung ist egal)

```
clique_num(member)
```

```
## Warning in clique_num(member): At cliques.c:1087 :directionality of edges is
## ignored for directed graphs
```

```
## [1] 5
```



```

# Größte Cliques finden
grosseccli <- largest_cliques(member)

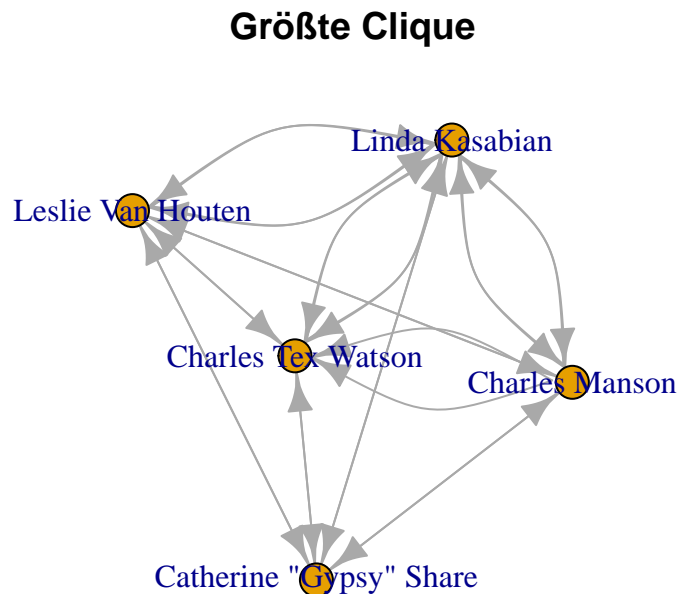
## Warning in largest_cliques(member): At cliques.c:1087 :directionality of edges
## is ignored for directed graphs

ma_cli <- subgraph(member, grosseccli[[1]])

## Warning in subgraph(member, grosseccli[[1]]): At structural_properties.c:
## 1984 :igraph_subgraph is deprecated from igraph 0.6, use igraph_induced_subgraph
## instead

plot(ma_cli,
     main= "Größte Clique")

```



```

# Triadenzensus
count_triangles(manson)

```

```

## [1] 0 2 5 3 7 0 8 1 1 7 0 10 0 0 122 71 0 0
## [19] 0 5 0 7 3 11 1 0 0 0 5 0 0 0 0 28 50 2
## [37] 1 0 9 8 30 3 0 0 3 0 0 0 12 4 5 1 0 0
## [55] 0 47 5 0 5 0 0 22 1 0 1 0 0 0 0 0 0 3
## [73] 0 3 0 0 0 3 0 2 0 1 1 8 0 0 0 11 1 0

```

```
## [91] 3 0 15 3 0 0 3 0 0 0 1 12 0 0 2 1 0 0
## [109] 0 0 0 24 0 0 0 11 1 0 1 0 0 0 1 0 13 1
## [127] 3 0 0 0 0 1 91 0 0 0 0 3 0 0 0 0 1 0
## [145] 0 0 0 0 0 0 0 7 0 0 0 0 13 0 0 12 1 1
## [163] 0 0 0 30 1 0 0 0 0 1 0 4 0 2 11 0 4 0
## [181] 0 0 0 0 0 1 1 0 0 0 1 0 0 1 22
```

hier wird ausgegeben, welcher Knoten an wie vielen "Dreiecken" beteiligt ist (Gewichtung ist egal)

```
clique_num(manson)
```

```
## Warning in clique_num(manson): At cliques.c:1087 :directionality of edges is
## ignored for directed graphs
```

```
## [1] 7
```

Größte Cliques finden

```
grosseccli_ge <- largest_cliques(manson)
```

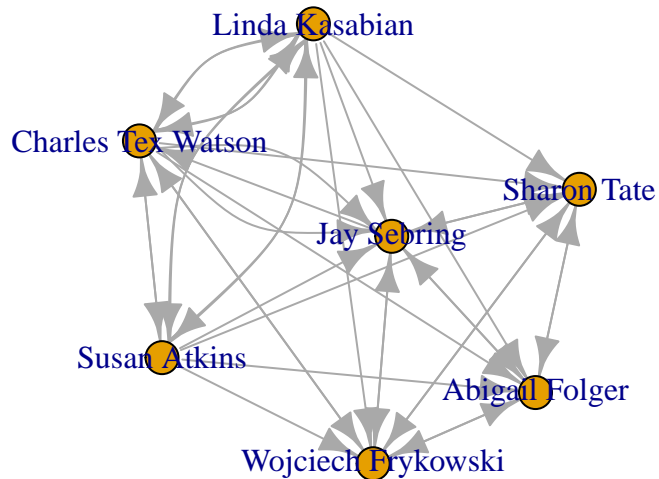
```
## Warning in largest_cliques(manson): At cliques.c:1087 :directionality of edges
## is ignored for directed graphs
```

```
ma_ge_cli <- subgraph(manson, grosseccli_ge[[1]])
```

```
## Warning in subgraph(manson, grosseccli_ge[[1]]): At structural_properties.c:
## 1984 :igraph_subgraph is deprecated from igraph 0.6, use igraph_induced_subgraph
## instead
```

```
plot(ma_ge_cli,
     main= "Größte Clique")
```

Größte Clique



```
# Triadenzensus
count_triangles(hollywood)
```

```
## [1] 1 0 1 19 24 10 6 1 3 0 4 1 18 21 0 19 24 14 4 6 3 18 1
```

```
# hier wird ausgegeben, welcher Knoten an wie vielen "Dreiecken" beteiligt ist (Gewichtung ist egal)
```

```
clique_num(hollywood)
```

```
## Warning in clique_num(hollywood): At cliques.c:1087 :directionality of edges is
## ignored for directed graphs
```

```
## [1] 7
```

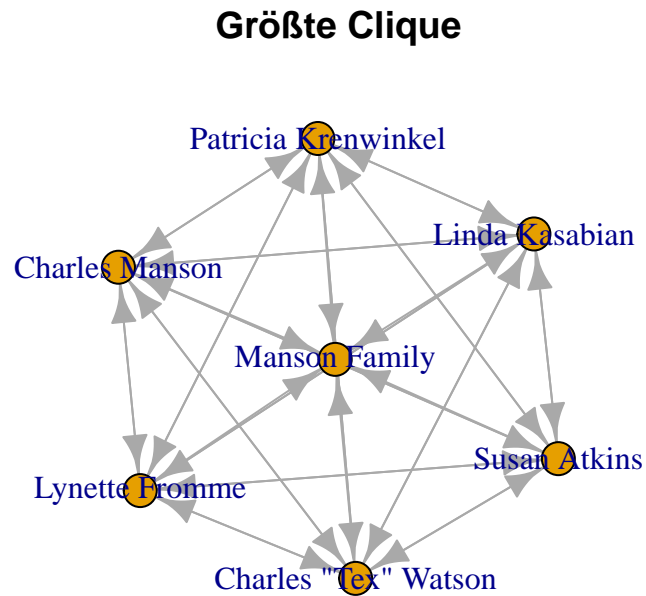
```
# Größte Cliquen finden
hollywood_cli <- largest_cliques(hollywood)
```

```
## Warning in largest_cliques(hollywood): At cliques.c:1087 :directionality of
## edges is ignored for directed graphs
```

```
ho_ge_cli <- subgraph(hollywood, hollywood_cli[[1]])
```

```
## Warning in subgraph(hollywood, hollywood_cli[[1]]): At structural_properties.c:
## 1984 :igraph_subgraph is deprecated from igraph 0.6, use igraph_induced_subgraph
## instead
```

```
plot(ho_ge_cli,
     main= "Größte Clique")
```



Literatur und Anhang

a) verwendete Literatur

.....

Bücher

- Helter Skelter: Der Mordanschlag des Charles Manson
- Ed Sanders: The Family – die Geschichte des Charles Manson
- Carol Green: Der Fall Charles Manson
- Charles Watson: Bekenntnisse eines Mörders
- Christiane von Beller : Berühmte Verbrecher (Stadtbibliothek Stuttgart)

.....

Online-Quellen

- <https://www.biography.com/crime-figure/charles-manson>
 - <https://allthatsinteresting.com/charles-manson-facts>
 - <https://www.merkur.de/welt/charles-manson-ist-tot-moerder-sektenfuehrer-und-perverse-pop-ikone-zr-9378627.html>
 - <https://www.tagesspiegel.de/gesellschaft/medien/charles-manson-doku-das-personifizierte-boese/24958466.html>
 - <https://www.welt.de/kultur/literarischewelt/article10573701/Charles-Manson-und-Roman-Polanski-treffen-sich.html>
 - https://www.sueddeutsche.de/thema/Charles_Manson
-

Dokumentationen

- <https://www.youtube.com/watch?v=iAu1Mc0KqJk>
- <https://www.youtube.com/watch?v=UMaZ3QKz8EQ>
- <https://www.youtube.com/watch?v=v4qZB2ytq10>

b) Codebuch (Link auf Github)

Codebuch Manson Projekt

c) verwendete Datenquellen (Link auf Github Edge- und Nodelist)

- Nodelist Manson
- Edgelist Manson
- Nodelist Film
- Edgelist Film

d) komplettes annotiertes RNotebook (Link auf Github)

Das RNotebook ist unter https://raw.githubusercontent.com/thomas5nolte/Manson/master/Manson_version_1.2.7.rmd erreichbar.

e) TeilnehmerInnen des Projekts und Arbeitsaufwand im Projekt

Name	Matrikelnummer	Arbeitsaufwand
Frederike Fuhrmann	37426	zu viel
Eva McGowan	36957	zu viel
Thomas Nolte	36867	zu viel

Name	Matrikelnummer	Arbeitsaufwand
Annika Stete	37511	zu viel
Rromina Trslic	37510	zu viel
Anna Veyhl	36955	zu viel

Testbereich

####