test 6 - 10 / 3 / 2023

Script per la preparazione dei dati

```
#read data a
df <- read_csv('dati_6_10_marzo_2023.csv')
```

```
## New names:
## Rows: 117510 Columns: 17
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (5): origin, destination, type, direction, res_trav dbl (11): ...1, n, 00_03,
## 03_06, 06_09, 09_12, 12_15, 15_18, 18_21, 21_24, ... date (1): day
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

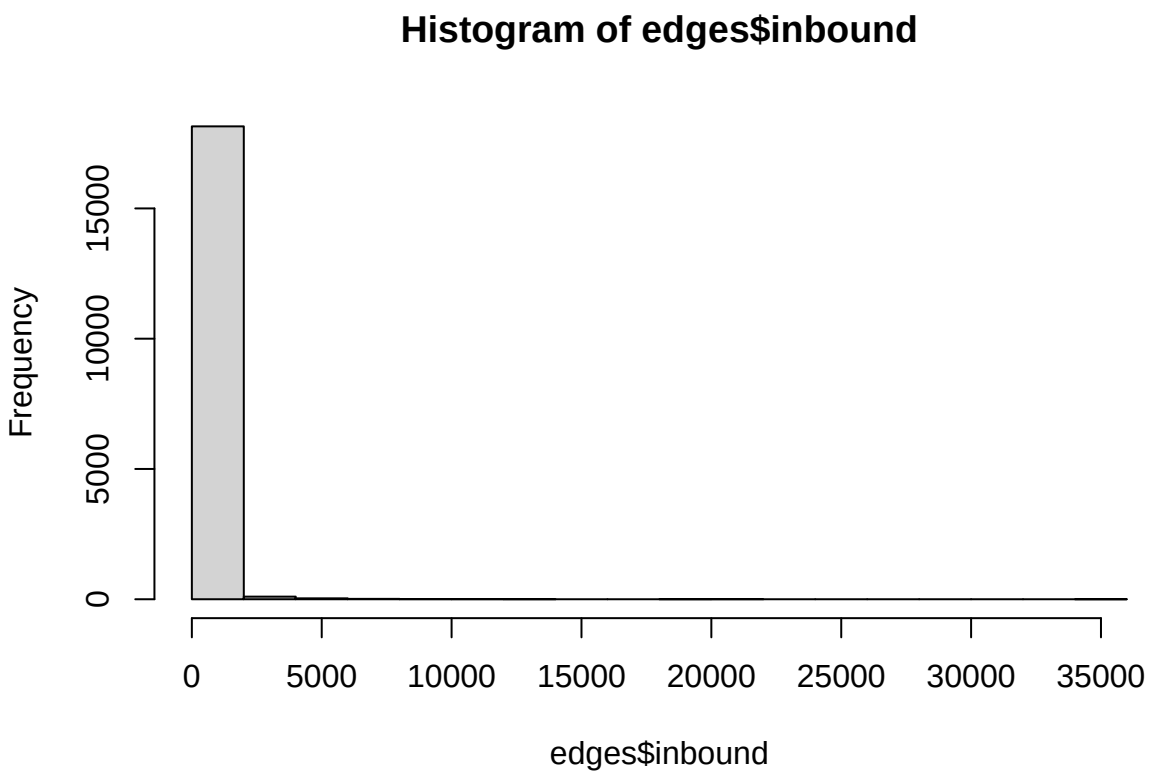
```
colnames(df)
```

```
##  [1] "...1"        "day"         "origin"      "destination" "type"
##  [6] "n"           "00_03"       "03_06"       "06_09"       "09_12"
## [11] "12_15"       "15_18"       "18_21"       "21_24"       "weekday"
## [16] "direction"   "res_trav"
```

```
# Group by departure and destination, then calculate the sum of n_viaggi
edges <- df %>%
  group_by(origin, destination, direction) %>%
  summarise(n = sum(n)) %>%
  ungroup() %>%
  pivot_wider(names_from = direction, values_from = n, values_fill = 0) %>%
  mutate(net = inbound - outbound) %>%
  mutate(ratio = if_else( outbound > 0  , inbound / outbound, 0))
```

```
## `summarise()` has grouped output by 'origin', 'destination'. You can override
## using the `.groups` argument.
```

```
hist(edges$inbound)
```




```r
table(edges$work_holiday)
```

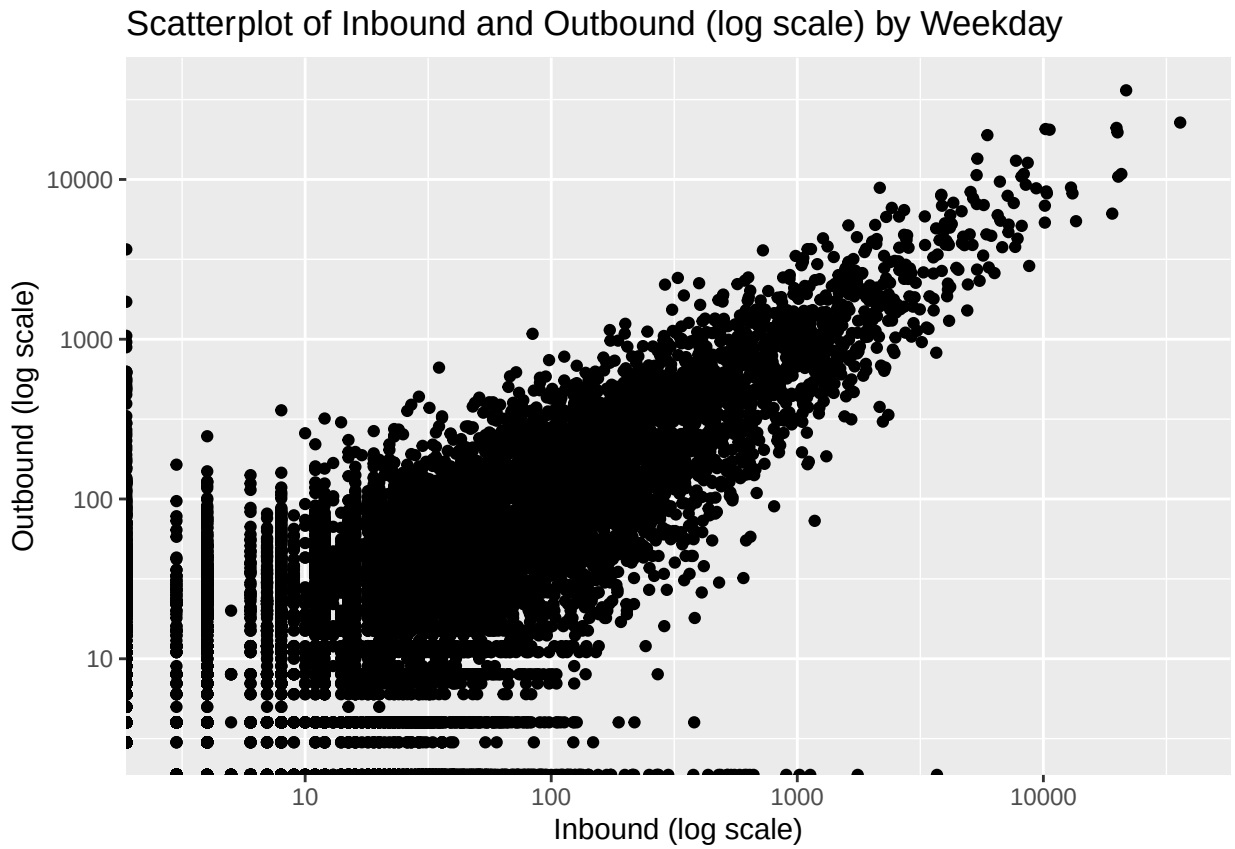
```
## Warning: Unknown or uninitialised column: `work_holiday`.
```

```
## < table of extent 0 >
```

```r
# Create scatterplot
edges %>% ggplot(aes(x = inbound, y = outbound)) +
  geom_point() +
  scale_x_log10() +
  scale_y_log10() +
  labs(x = "Inbound (log scale)", y = "Outbound (log scale)", color = "Weekday") +
  ggtitle("Scatterplot of Inbound and Outbound (log scale) by Weekday")
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
```

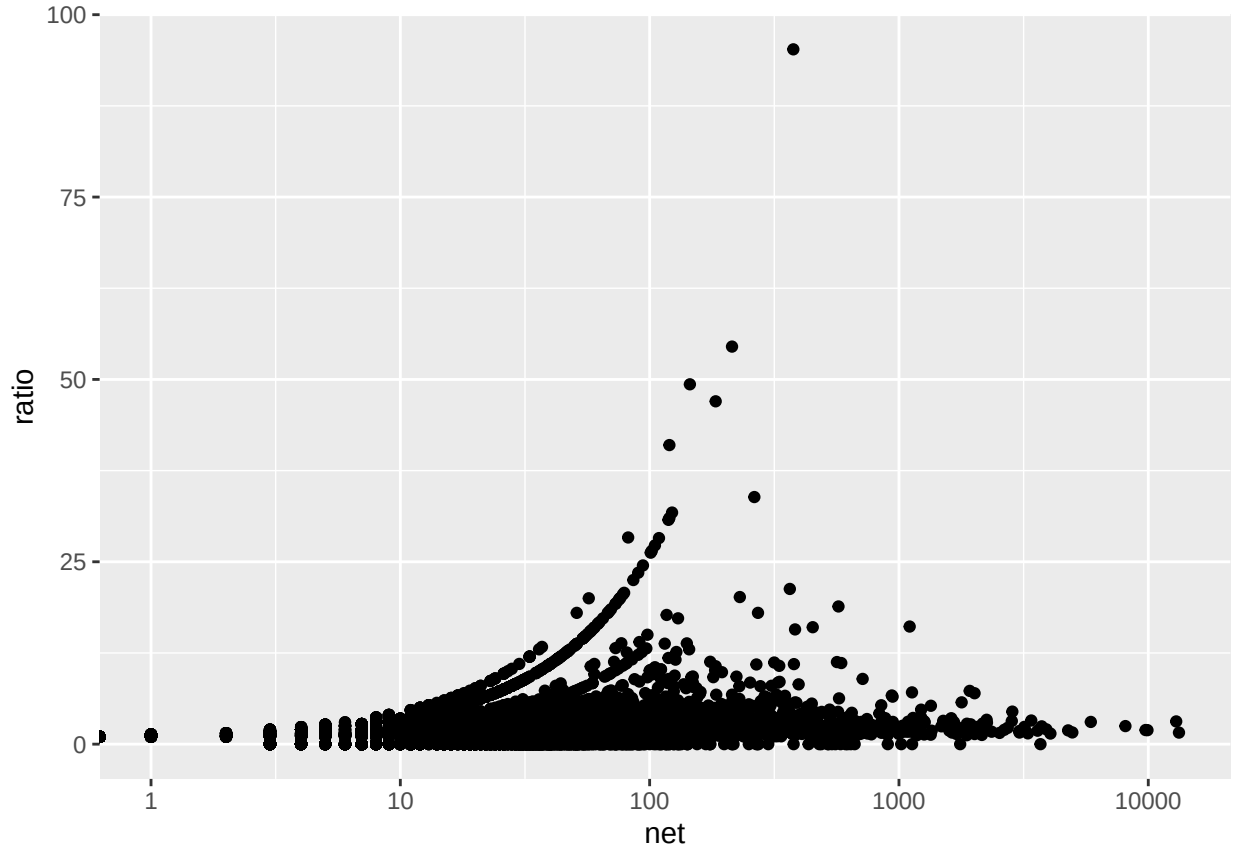



```
# Create scatterplot
edges %>% ggplot(aes(x = net, y = ratio)) +
  geom_point() +
  scale_x_log10()
```

```
## Warning in self$trans$transform(x): Si è prodotto un NaN
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Removed 9179 rows containing missing values (`geom_point()`).
```




```
library(igraph)
```

```
## Warning: il pacchetto 'igraph' è stato creato con R versione 4.1.3
```

```
##
## Caricamento pacchetto: 'igraph'
```

```
## I seguenti oggetti sono mascherati da 'package:lubridate':
##
##     %--%, union
```

```
## I seguenti oggetti sono mascherati da 'package:dplyr':
##
##     as_data_frame, groups, union
```

```
## I seguenti oggetti sono mascherati da 'package:purrr':
##
##     compose, simplify
```

```
## Il seguente oggetto è mascherato da 'package:tidyr':
##
##     crossing
```

```
## Il seguente oggetto è mascherato da 'package:tibble':
##
##     as_data_frame
```

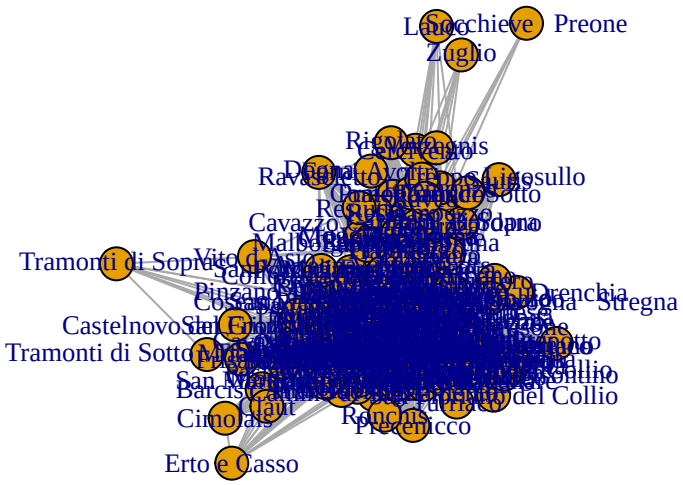
```
## I seguenti oggetti sono mascherati da 'package:stats':
##
##     decompose, spectrum
```

```
## Il seguente oggetto è mascherato da 'package:base':
##
##     union
```

```
g <- igraph::graph_from_data_frame(edges,  directed = FALSE)
g <- set_edge_attr(g, "weight", value = E(g)$outbound)
g <- igraph::simplify(g)
all_positive <- all(E(g)$weight >= 0)

if (all_positive) {
  print("All edge weights are positive.")
} else {
  print("Not all edge weights are positive.")
}
```

```
## [1] "All edge weights are positive."
```

```
# Prune edges with weight equal to 0
g <- delete_edges(g, which(E(g)$weight == 0))

# Plot the network
plot(g, vertex.label.cex = 0.8)
```



check varaibility of resutls

```
g1 <-igraph::as.undirected(g)
for (i in 1:10){
  print(max(cluster_louvain(g1, resolution = 1)$membership))
}
```

```
## [1] 5
## [1] 5
## [1] 5
## [1] 5
## [1] 5
## [1] 6
## [1] 5
## [1] 5
## [1] 5
## [1] 6
```

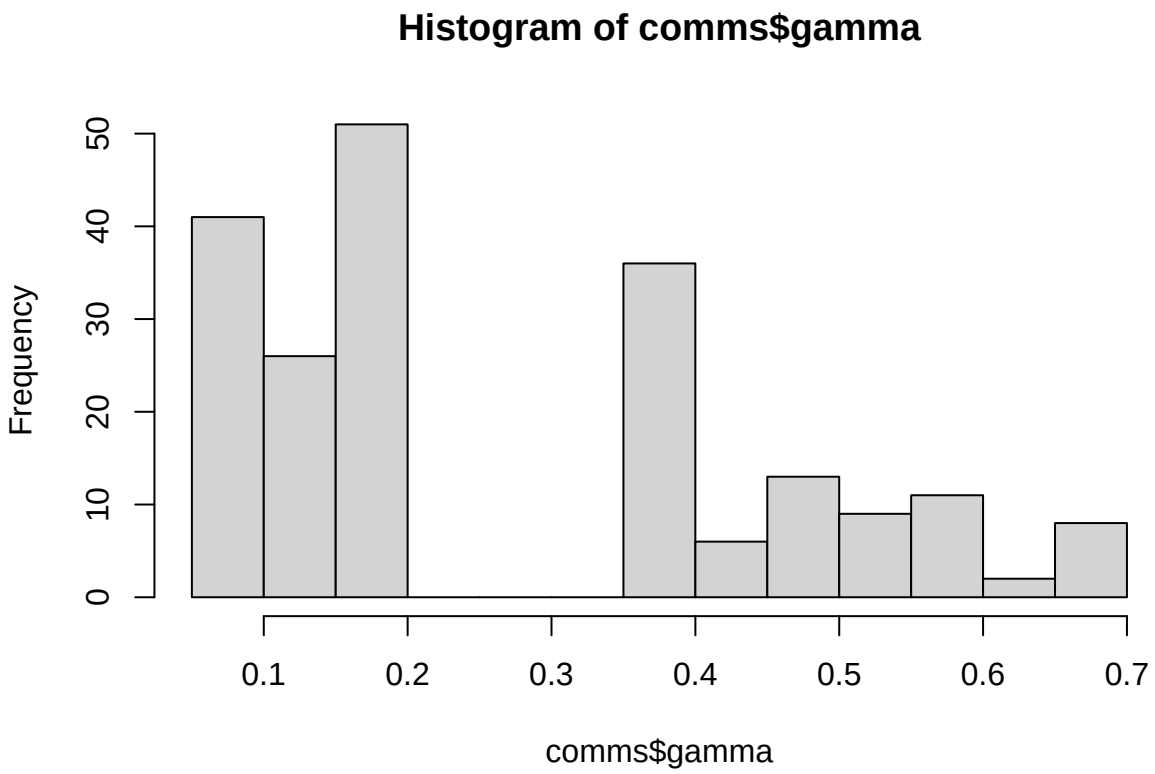
```
library(CCD)
library(aricode)
```

```
## Warning: il pacchetto 'aricode' è stato creato con R versione 4.1.3
```

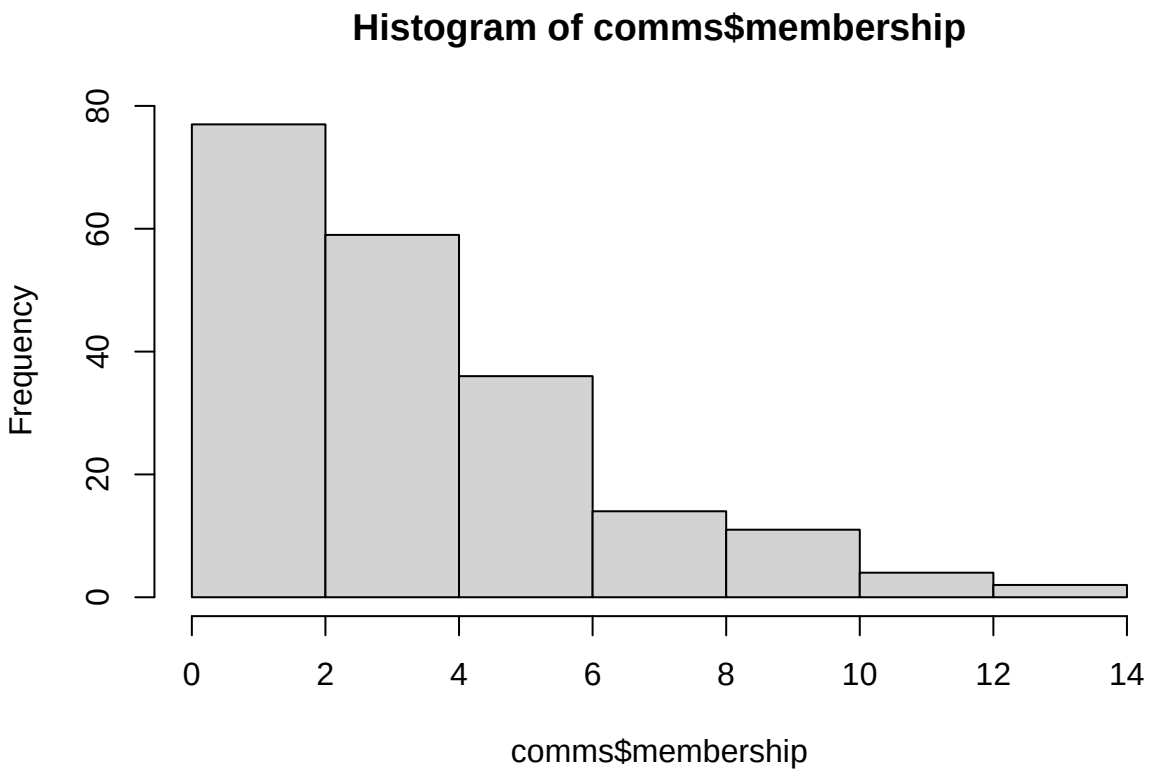
```
comms <- CCD::consensus_community_detection(g,
                                            p = 0.9,
                                            q = 0.5,
                                            t = 100,
                                            method = "LV",
                                            r = c(0.8,1.0,1.5),
                                            group_outliers = FALSE)
V(g)$community <- comms$membership
V(g)$gamma <- comms$gamma
mu = CCD::empirical_mu(g)
print(mu)
```

```
## [1] 0.2842343
```

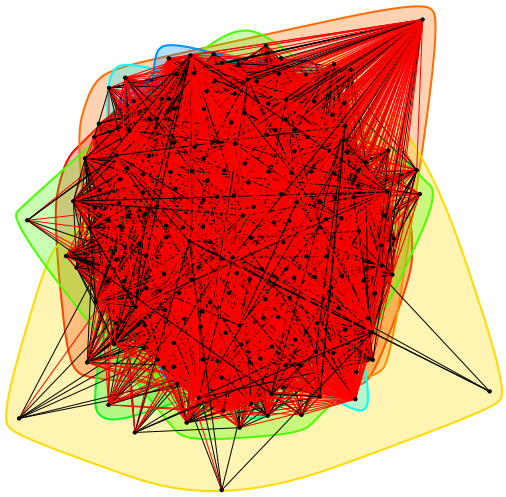
```
hist(comms$gamma)
```




```
hist(comms$membership)
```

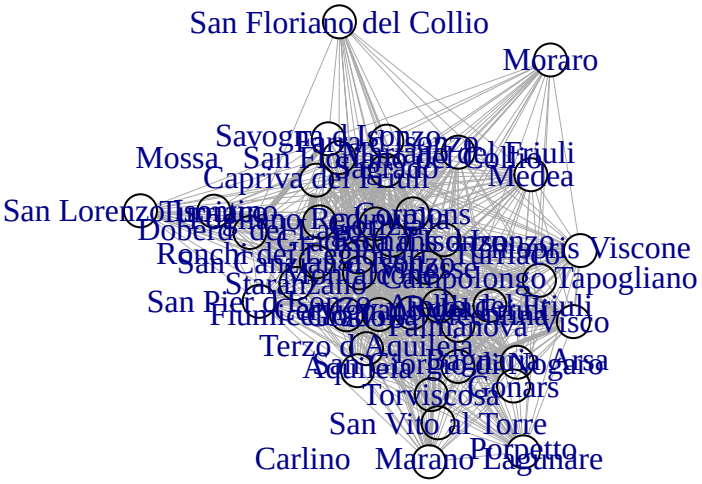



```
plot(comms, g, vertex.label = NA, vertex.size = 1, edge.width = 0.1, layout = layout.kamada.kawai(g), v
```

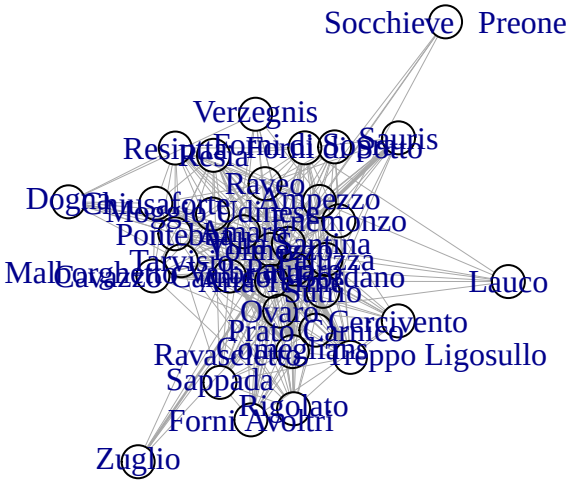


```r
#comms$membership <- as.factor(comms$membership)

# Create a list to store subgraphs
subgraphs <- list()

# Loop through each community
for (i in unique(comms$membership)) {
  # Extract vertices belonging to the current community
  vertices_in_community <- which(comms$membership == i)
  # Create subgraph for the current community
  subgraphs[[i]] <- subgraph(g, vertices_in_community)
  # Plot the subgraph
  plot(subgraphs[[i]], main = paste("Community", i),vertex.color = V(g)$gamma, edge.width = E(g)$weight,
}
```

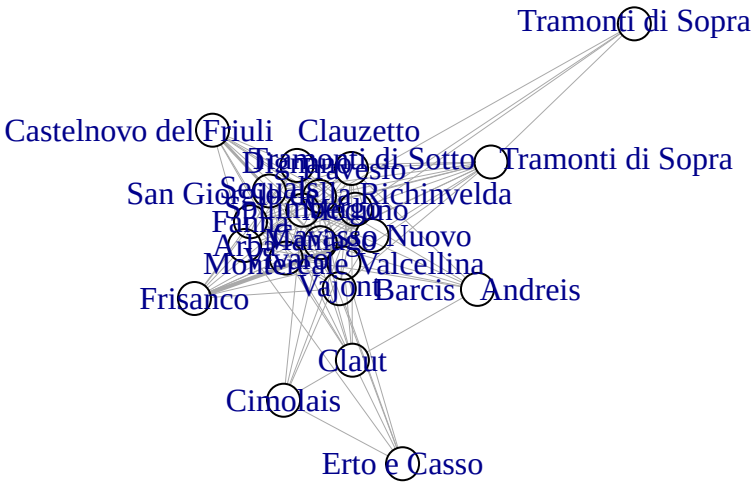
Community 1


San Floriano del Collio
Moraro
Mossa
Capriva del Friuli
Medea
San Lorenzo Isontino
Viscone
Tapogliano
Fiumicello
Terzo d'Aquileia
Torviscosa
San Vito al Torre
Carlino
Marano Lagunare

Community 3


Socchieve
Preone
Verzegnis
Resia
Raveo
Lauco
Ovaro
Prato Carnico
Treppo Ligosullo
Sappada
Forni Avoltri
Zuglio

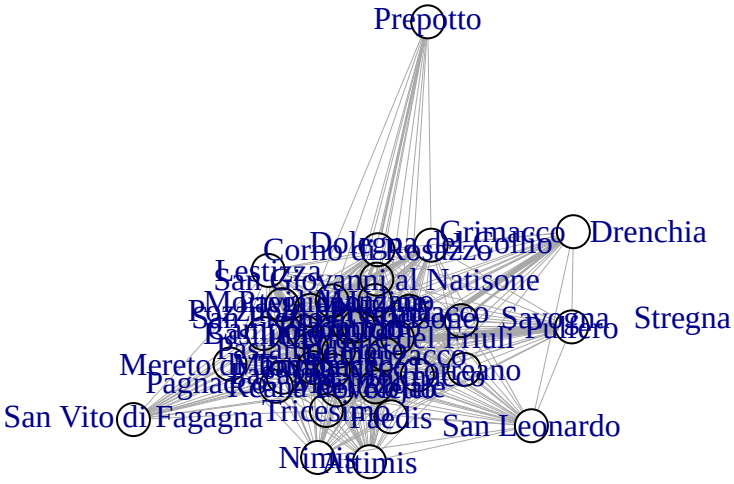
Community 5


Tramonti di Sopra
Castelnovo del Friuli
Clauzetto
Tramonti di Sotto
Tramonti di Sopra
Nuovo
Barcis
Andreis
Frisanco
Claut
Cimolais
Erto e Casso

Community 6


Rive d Arcano
Colloredo di Monte Albano
Pinzano al Tagliamento
San Daniele del Friuli
Artegna
Buja
Osoppo
Gemona del Friuli
Trasaghis
Forgaria nel Friuli
Vito d Asio

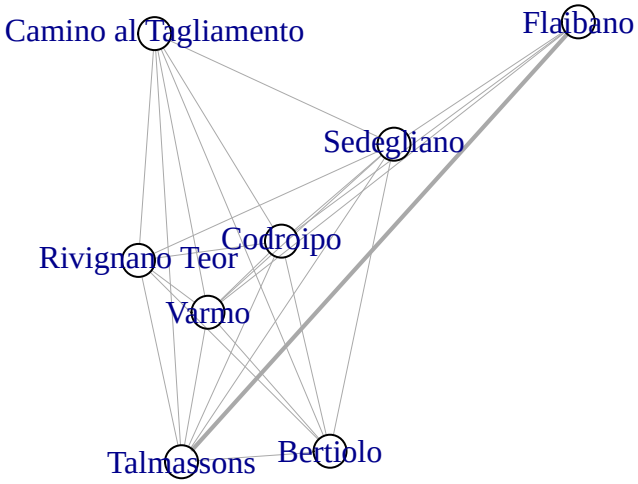
Community 2


Prepotto
Grimacco
Drenchia
Stregna
Savogna
Pulfero
Meretodi
Pagnacco
San Vito di Fagagna
Tricesimo
Paderno
San Leonardo
Nimis
Attimis

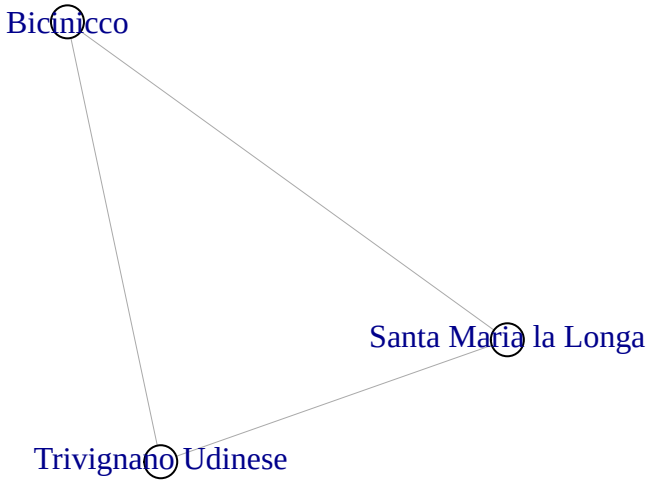
Community 4


Polcenigo
Budoia
San Quirino
Valvasone Arzene
Aviano
Caneva
Fontanafredda
Sacile
Zoppola
Casarsa della Delizia
Prugnera
Roveredo in Piano
San Vito al Tagliamento
Azzano Decimo
Morsano al Tagliamento
Pasiano di Pordenone
Sesto al Reghena
Cordovado
Pravisdomini

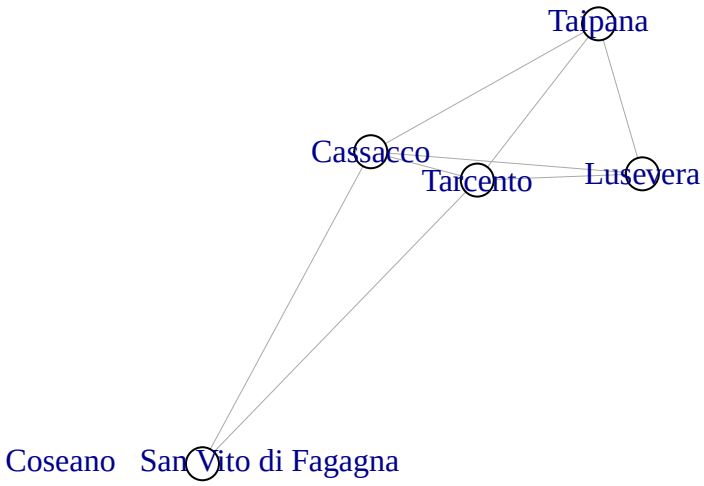
Community 7


Camino al Tagliamento
Flaibano
Sedegliano
Codroipo
Rivignano Teor
Varmo
Talmassons
Bertiolo

Community 11

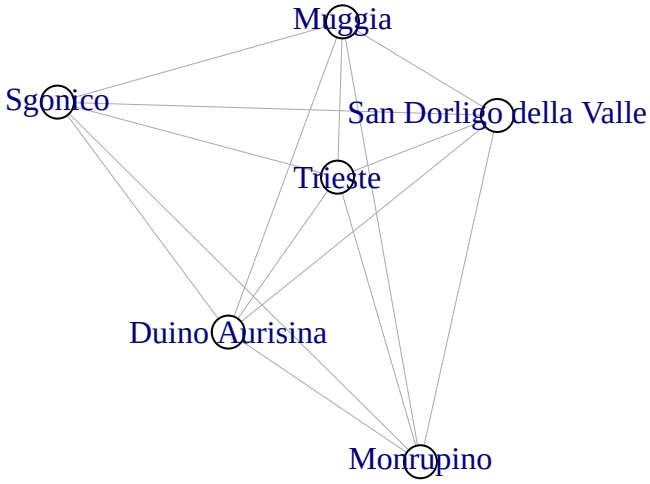


Community 10

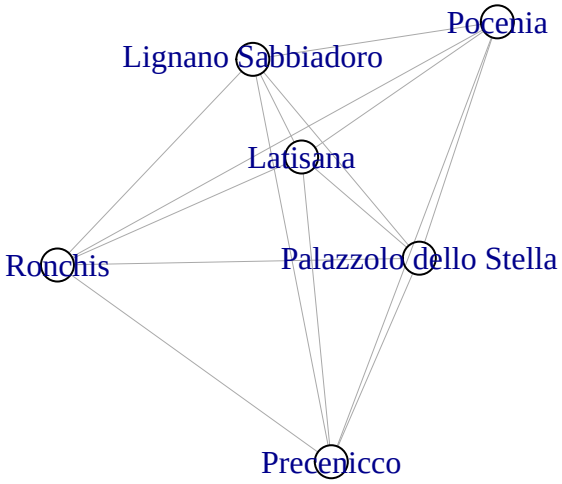


Community 12

Community 8


Muggia
Sgonico
San Dorligo della Valle
Trieste
Duino Aurisina
Monrupino

Community 9


Pocenia
Lignano Sabbiadoro
Latisana
Ronchis
Palazzolo dello Stella
Precenicco

Community 13

Muzzana (de) Turgnano

Community 14

San Martino al Tagliamento