

# Validação - Somente COVID-19

José Elvano Moraes

4/15/2021

## Variáveis selecionadas

```
## Rows: 76,666
## Columns: 11
## $ IDADE      <fct> "(37,73]", "(37,73]", "(73,109]", "(37,73]", "(73,109]", "(~
## $ EVOLUCAO   <fct> 2, 1, 1, 1, 2, 1, 2, 2, 1, 2, 1, 2, 2, 2, 1, 1, 1, 1, 1, 1,~
## $ RENAL      <fct> 2, 2, 2, 1, 2, 1, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,~
## $ DIABETES   <fct> 2, 1, 2, 1, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2,~
## $ OBESIDADE  <fct> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 9, 2,~
## $ NEUROLOGIC <fct> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 9, 2, 2, 2,~
## $ PNEUMOPATI <fct> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,~
## $ UTI        <fct> 1, 1, 2, 1, 2, 2, 1, 2, 2, 2, 2, 2, 1, 1, 2, 2, 1, 2, 2, 2,~
## $ CARDIOPATI <fct> 2, 2, 2, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 1, 1,~
## $ SUPORT_VEN <fct> 2, 2, 2, 9, 3, 3, 2, 2, 3, 2, 2, 3, 1, 1, 3, 3, 2, 3, 9, 3,~
## $ ANTIVIRAL  <fct> 2, 2, 2, 9, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,~
```

## Imposição de estrutura com arcos que fazem sentido clínico ( *White list* )

```
s1 <- sample_frac(ddf, .9, FALSE)
cat(nrow(ddf) - nrow(s1))

## 7667

wl = matrix(c("OBESIDADE", "DIABETES",
              "IDADE", "DIABETES",
              "IDADE", "SUPORT_VEN",
              "IDADE", "NEUROLOGIC",
              "DIABETES", "RENAL",
              "DIABETES", "CARDIOPATI",
              "UTI", "EVOLUCAO"),
            ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from", "to")))

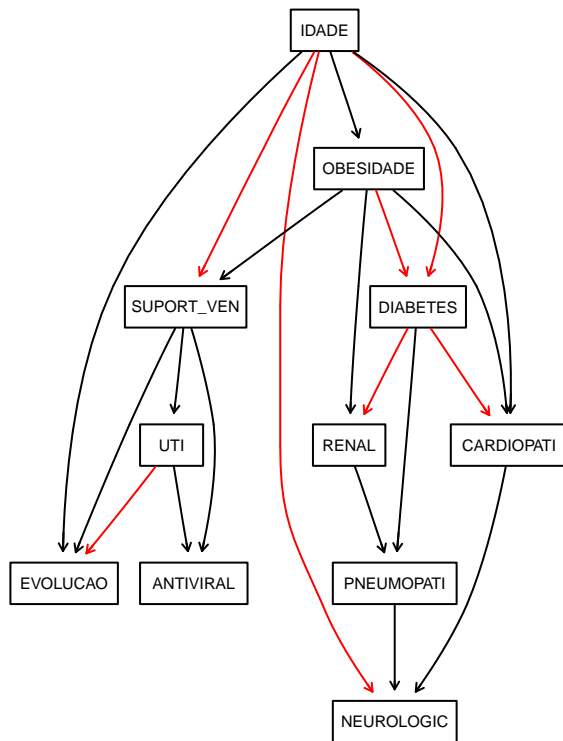
bn1 <- mmhc(s1, whitelist = wl)

#sem WL
bn2 <- mmhc(s1, whitelist = NULL)
```

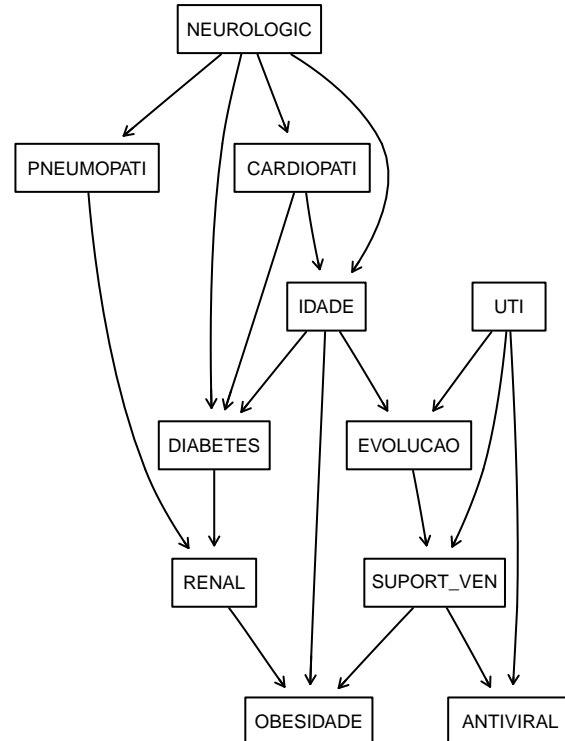
## DAG (*Directed Acyclic Graph*) usando ou não *whitelisting* no algoritmo de aprendizado da estrutura da rede causal

```
par(mfrow = c(1, 2))
graphviz.plot(bn1, shape='rectangle', highlight = list(arcs = wl), main = 'DAG com imposição de uma WL')
graphviz.plot(bn2, shape='rectangle', highlight = NULL, main = 'DAG sem WL')
```

DAG com imposição de uma WL



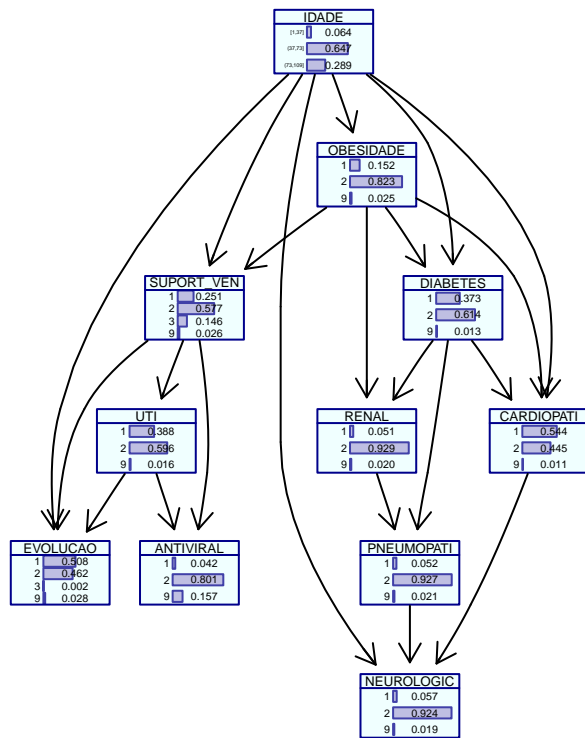
DAG sem WL



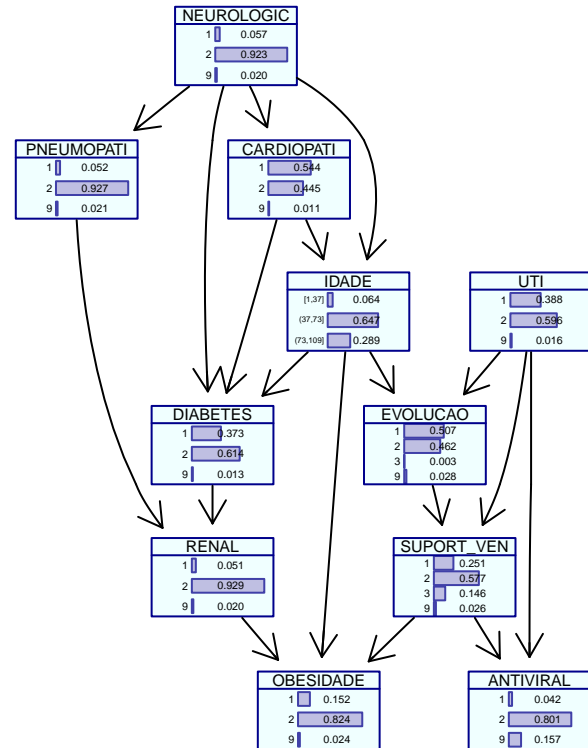
```
fitted.1 <- bn.fit(bn1, s1)
fitted.2 <- bn.fit(bn2, s1)

par(mfrow=c(1,2))
graphviz.chart(fitted.1,type = "barprob", col = "darkblue", bg = "azure", bar.col = "darkblue", main = "DAG com imposição de uma WL")
graphviz.chart(fitted.2, type = "barprob", col = "darkblue", bg = "azure", bar.col = "darkblue", main = "DAG sem WL")
```

## DAG sem WL



## DAG com WL



## Bootstrapping com WL

```
boots.trap <- 300
str.diff = suppressMessages(boot.strength(s1, R = boots.trap, algorithm = "mmhc"))
cat(paste('Threshold: ', attr(str.diff, "threshold")))
```

## Threshold: 0.5033333333333333

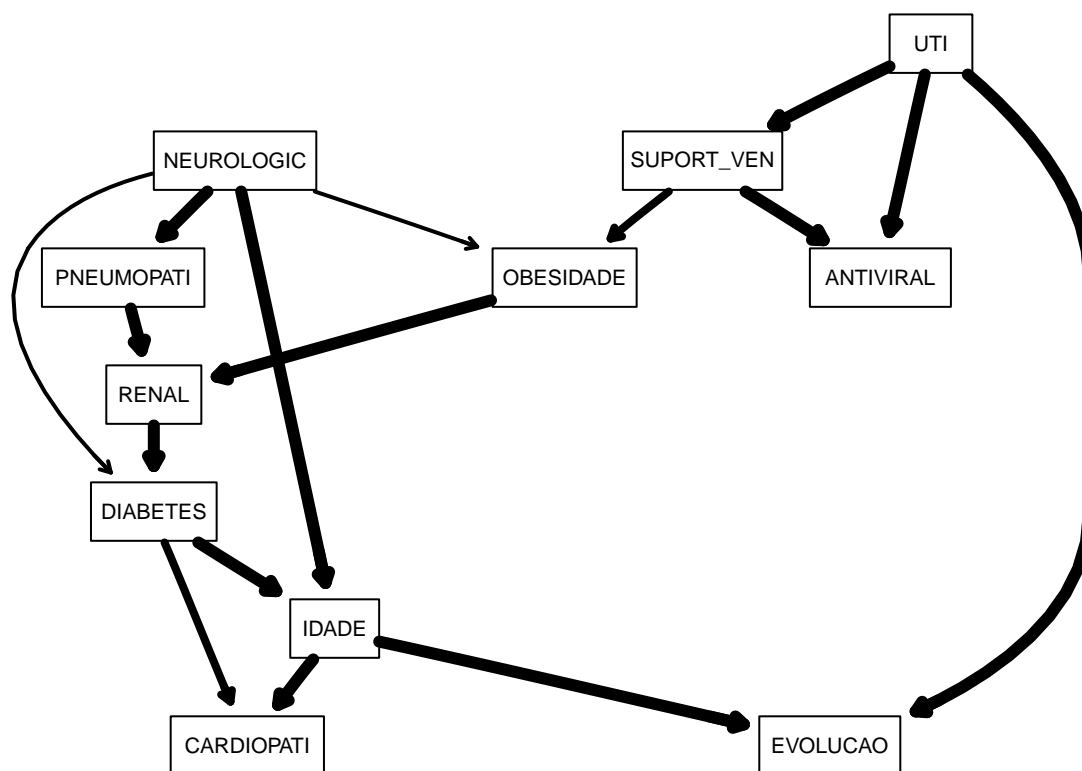
```
avg.diff = averaged.network(str.diff)
```

## Warning in averaged.network.backend(strength = strength, nodes = nodes, : arc  
## EVOLUCAO -> SUPORT\_VEN would introduce cycles in the graph, ignoring.

## Warning in averaged.network.backend(strength = strength, nodes = nodes, : arc  
## IDADE -> OBESIDADE would introduce cycles in the graph, ignoring.

```
strength.plot(avg.diff, str.diff, shape = "rectangle", main = paste("Iterações = ", boots.trap))
```

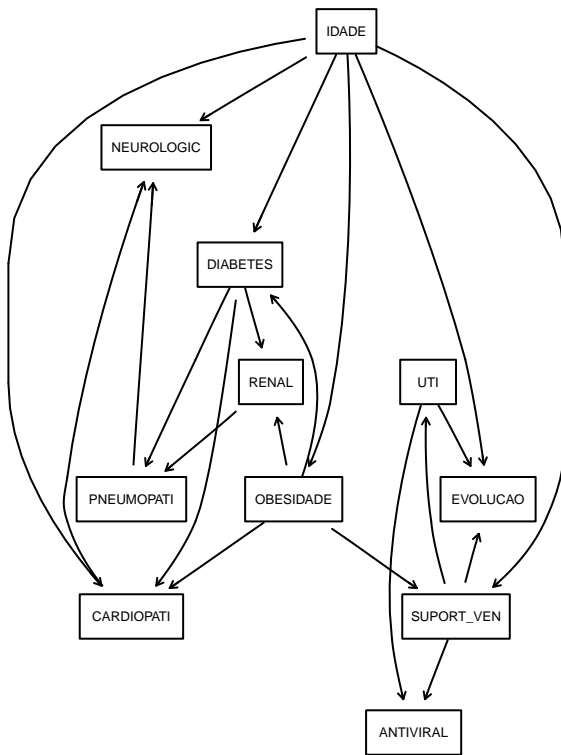
Iterações = 300



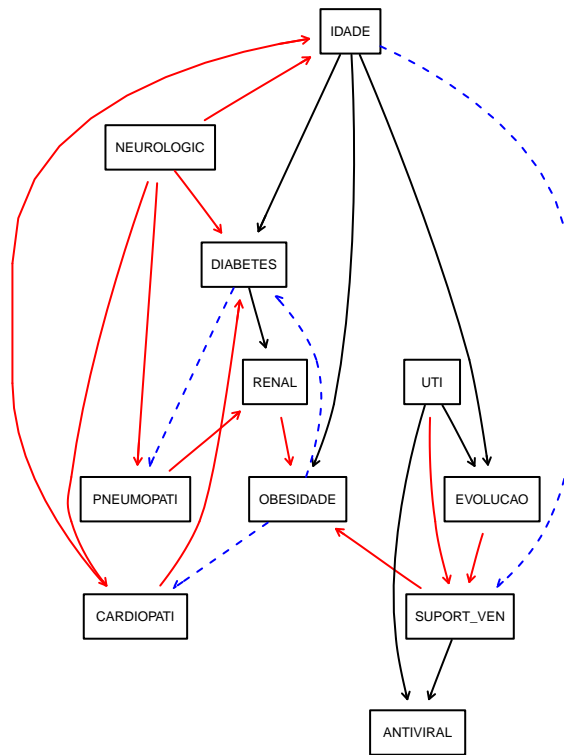
How can we compare the averaged network (avg.diff) with the network we originally learned in from all the data? The most qualitative way is to plot the two networks side by side, with the nodes in the same positions, and highlight the arcs that appear in one network and not in the other, or that appear with different directions.

```
par(mfrow = c(1, 2))
graphviz.compare(bn1, bn2, shape = "rectangle", main = c("DAG único sem WL", "DAG único com WL"))
```

DAG único sem WL

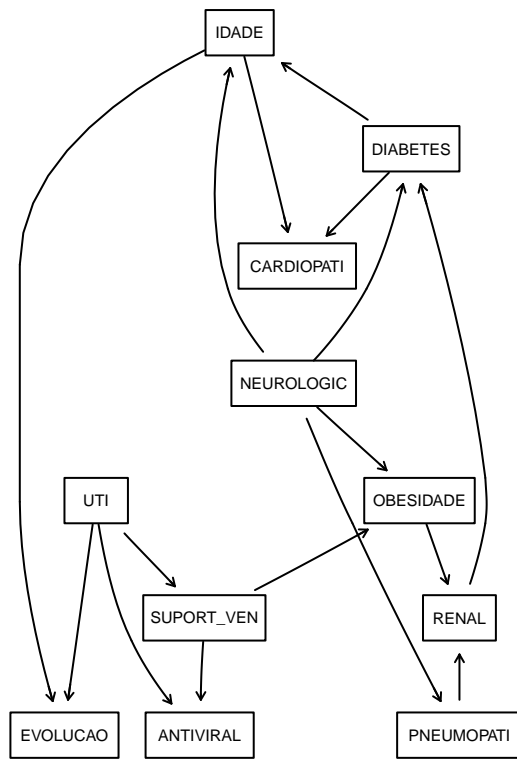


DAG único com WL

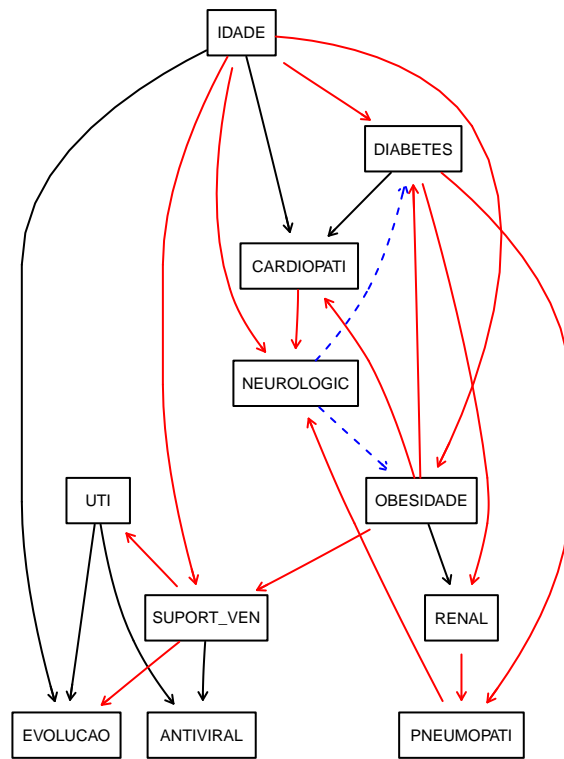


```
par(mfrow = c(1, 2))
graphviz.compare(avg.diff, bn1, shape = "rectangle", main = c("DAG médio sem WL", "DAG médio com WL"))
```

DAG médio sem WL



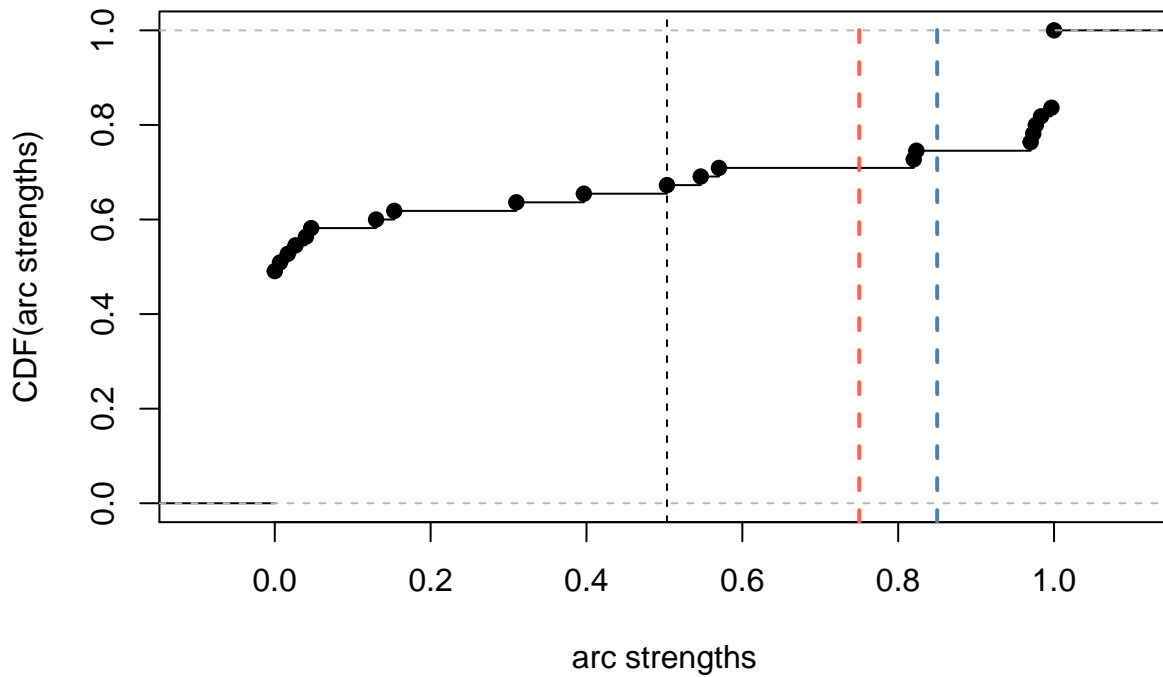
DAG médio com WL



It is also a good idea to look at the threshold with respect to the distribution of the arc strengths

```
plot(str.diff)
abline(v = 0.75, col = "tomato", lty = 2, lwd = 2)
abline(v = 0.85, col = "steelblue", lty = 2, lwd = 2)
```

**threshold = 0.503**



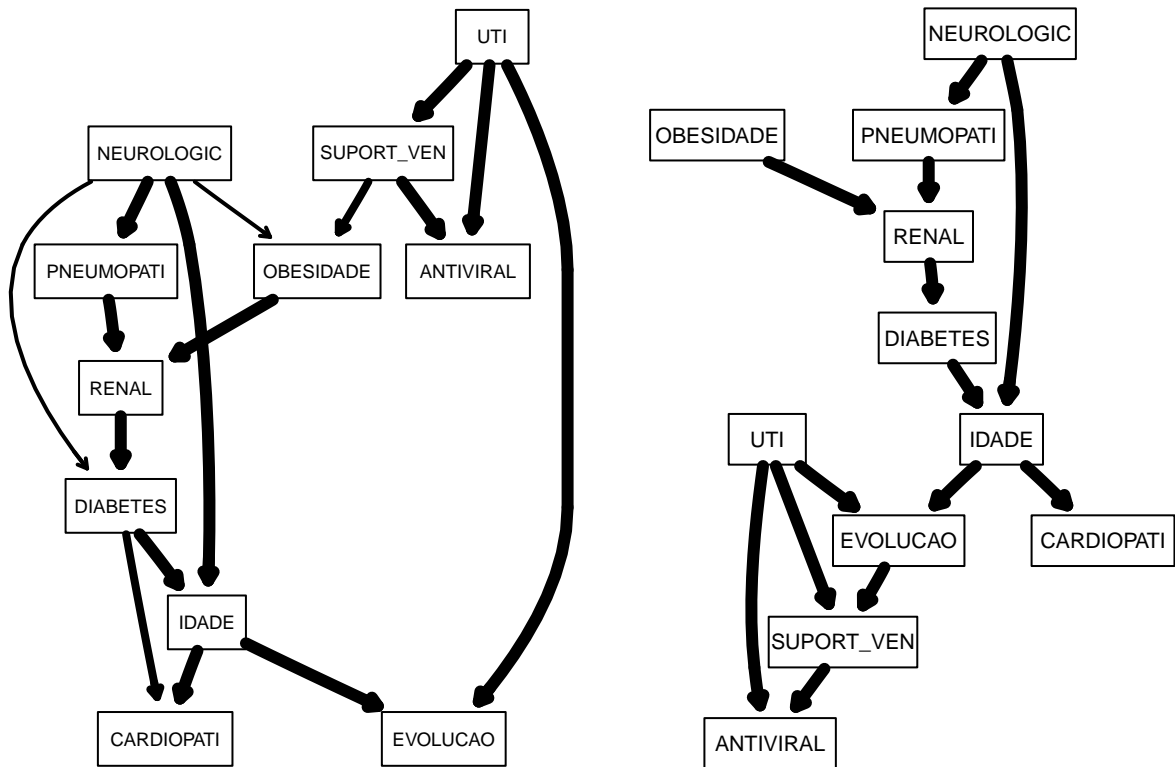
The simpler network we obtain by setting `threshold = 0.8` in `averaged.network()` is shown below; it is certainly easier to reason with from a qualitative point of view.

```
par(mfrow = c(1, 2))
avg.simpler = averaged.network(str.diff, threshold = 0.95)

## Warning in averaged.network.backend(strength = strength, nodes = nodes, : arc
## IDADE -> OBESIDADE would introduce cycles in the graph, ignoring.

strength.plot(avg.diff, str.diff, shape = "rectangle",
              main = paste("Iterações = ", boots.trap, " Thr = ", attr(str.diff, "threshold")))
strength.plot(avg.simpler, str.diff, shape = "rectangle", main = 'Iterações = 100 Thr = 0.95')
```

ações = 300 Thr = 0.5033333333333333 Iterações = 100 Thr = 0.95



## Markov Blanket da variável evolução

```
cat (mb(x = avg.diff, node = 'EVOLUCAO'))
```

```
## IDADE UTI
```

```
##
```

```
cat (mb(x = avg.simpler, node = 'EVOLUCAO'))
```

```
## IDADE UTI SUPORT_VEN
```

```
#save the current workspace
```

```
save(list = ls(all.names = TRUE), file = "kk_04_90pc_300it.RData", envir = .GlobalEnv)
```