

# Validação - Somente COVID-19

José Elvano Moraes

4/15/2021

## Variáveis selecionadas

```
## Rows: 76,666
## Columns: 17
## $ IDADE      <fct> "(37,73]", "(37,73]", "(73,109]", "(37,73]", "(73,109]", "(~
## $ FEBRE      <fct> 2, 2, 2, 1, 2, 1, 1, 1, 1, 2, 1, 2, 1, 2, 2, 1, 1, 2, 2, 2,~
## $ GARGANTA   <fct> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 1, 2, 2, 2, 1, 2, 2, 2,~
## $ DISPNEIA   <fct> 1, 2, 2, 1, 2, 2, 2, 2, 1, 1, 1, 1, 1, 2, 1, 2, 1, 2, 1, 1, 2,~
## $ SATURACAO  <fct> 1, 1, 2, 2, 2, 2, 2, 1, 2, 2, 1, 1, 1, 1, 1, 2, 2, 2, 1, 2, 2,~
## $ EVOLUCAO   <fct> 2, 1, 1, 1, 2, 1, 2, 2, 1, 2, 1, 2, 2, 2, 1, 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, 2, 1, 2, 2, 2, 1, 2, 2, 9, 2,~
## $ PERD_OLFT  <fct> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2,~
## $ PERD_PALA  <fct> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,~
## $ NEUROLOGIC <fct> 2, 2, 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, .8, FALSE)
cat(nrow(ddf) - nrow(s1))

## 15333

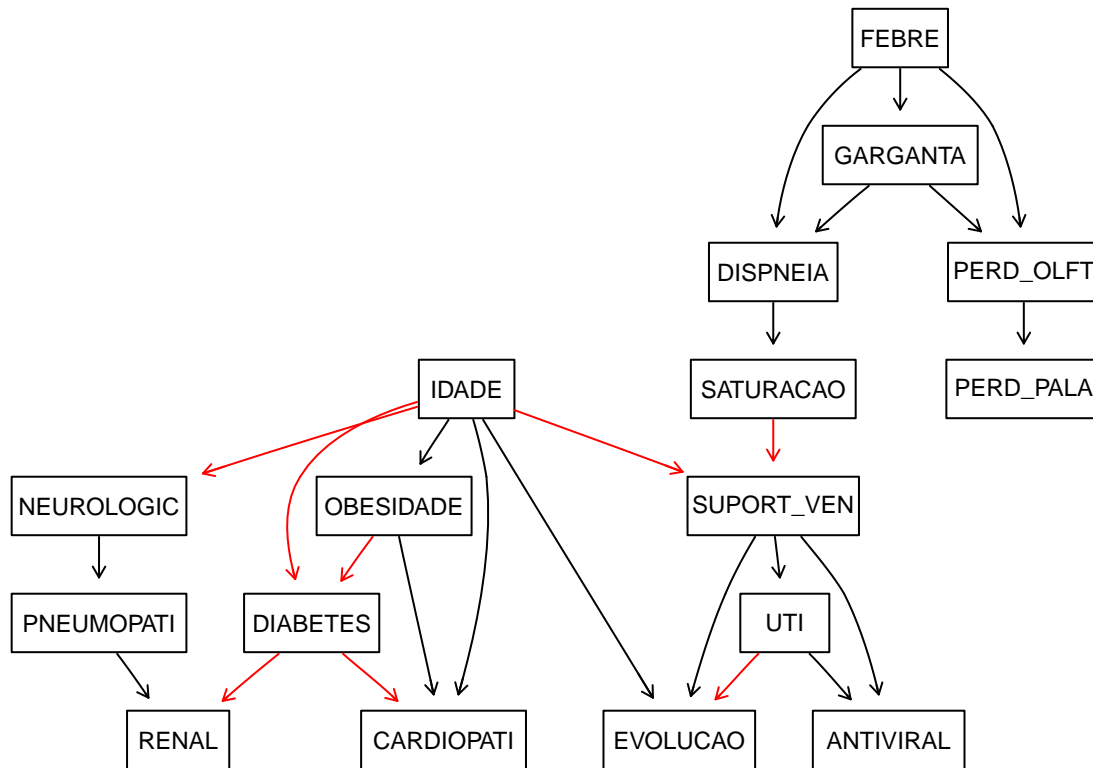
wl = matrix(c("OBESIDADE", "DIABETES",
              "IDADE", "DIABETES",
              "SATURACAO", "SUPORT_VEN",
              "IDADE", "SUPORT_VEN",
              "IDADE", "NEUROLOGIC",
              "DIABETES", "RENAL",
              "DIABETES", "CARDIOPATI",
              "UTI", "EVOLUCAO"),
  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

bn1 <- mmhc(s1, whitelist = wl)
```

## Rede Causal (DAG, *Directed Acyclic Graph*)

```
#par(mfrow=c(2,2))  
graphviz.plot(bn1, shape='rectangle', highlight = list(arcs = w1), main = '...')
```

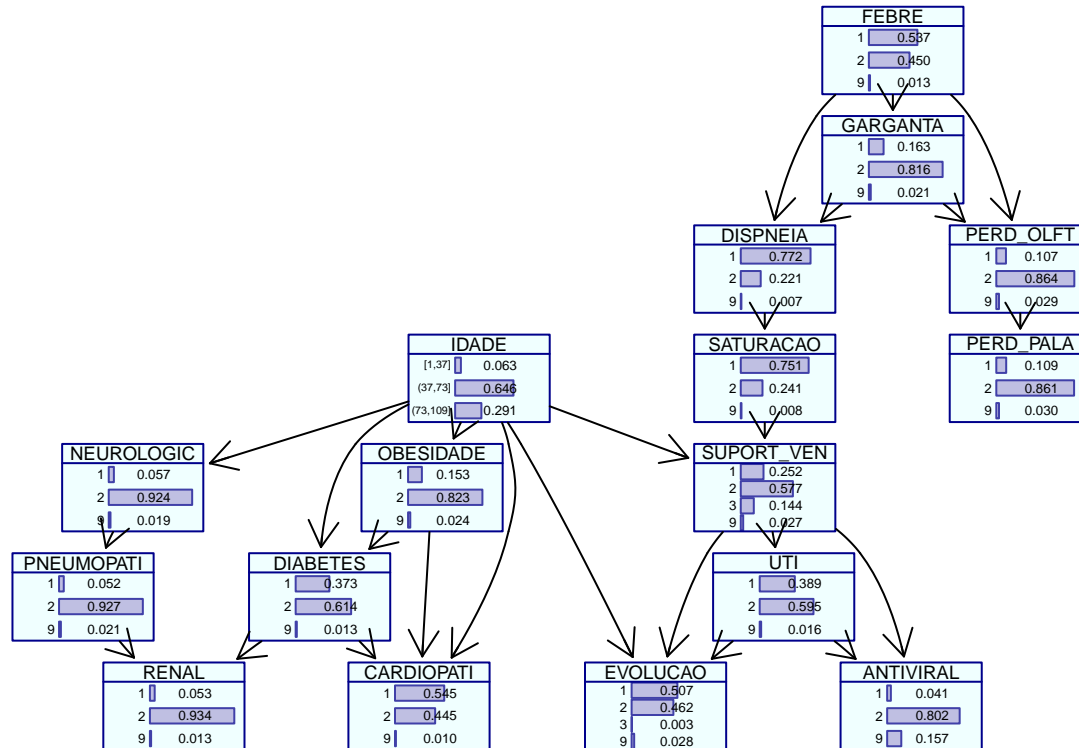
...



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

```
#par(mfrow=c(2,2))  
graphviz.chart(fitted.1, scale = c(2, 3), type = "barprob", col = "darkblue", bg = "azure", bar.col = ...)
```

## Rede de probabilidades



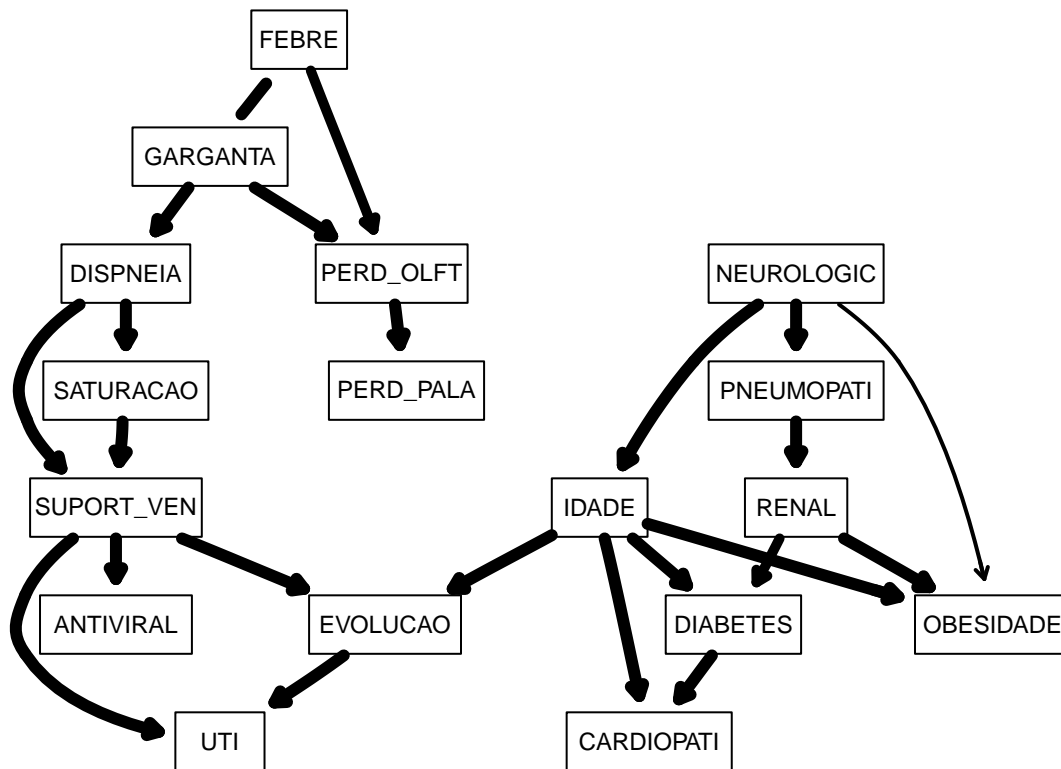
## Bootstrapping

```
boots.trap <- 400
str.diff = suppressMessages(boot.strength(s1, R = boots.trap, algorithm = "mmhc"))
cat(paste('Threshold: ', attr(str.diff, "threshold")))
```

## Threshold: 0.5

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

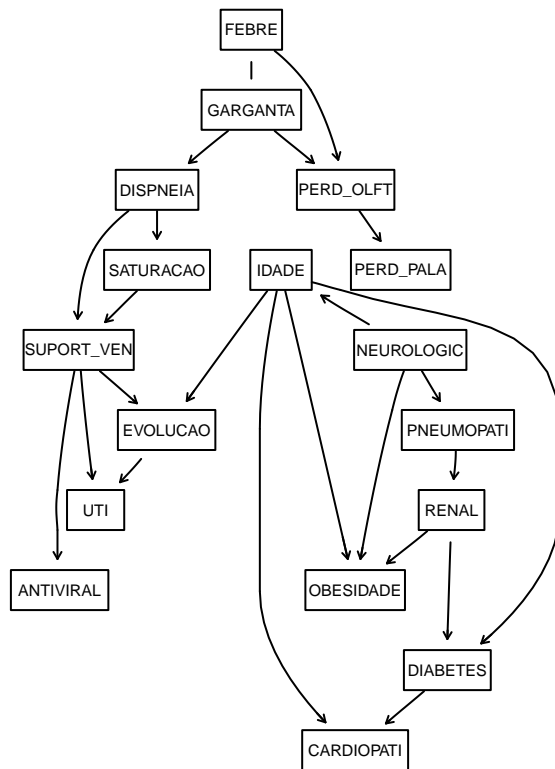
Iterações = 400



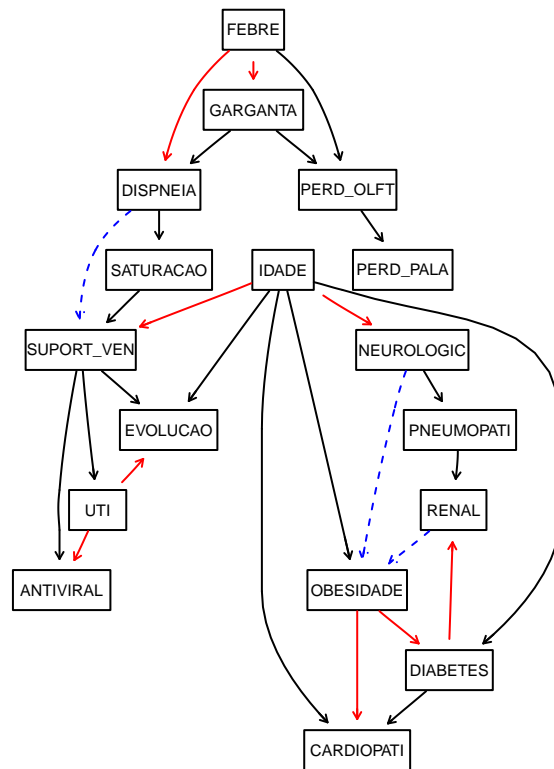
How can we compare the averaged network (avg.diff) with the network we originally learned in from all the data? The most qualitative way is to plot the two networks side by side, with the nodes in the same positions, and highlight the arcs that appear in one network and not in the other, or that appear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

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

DAG médio



DAG único

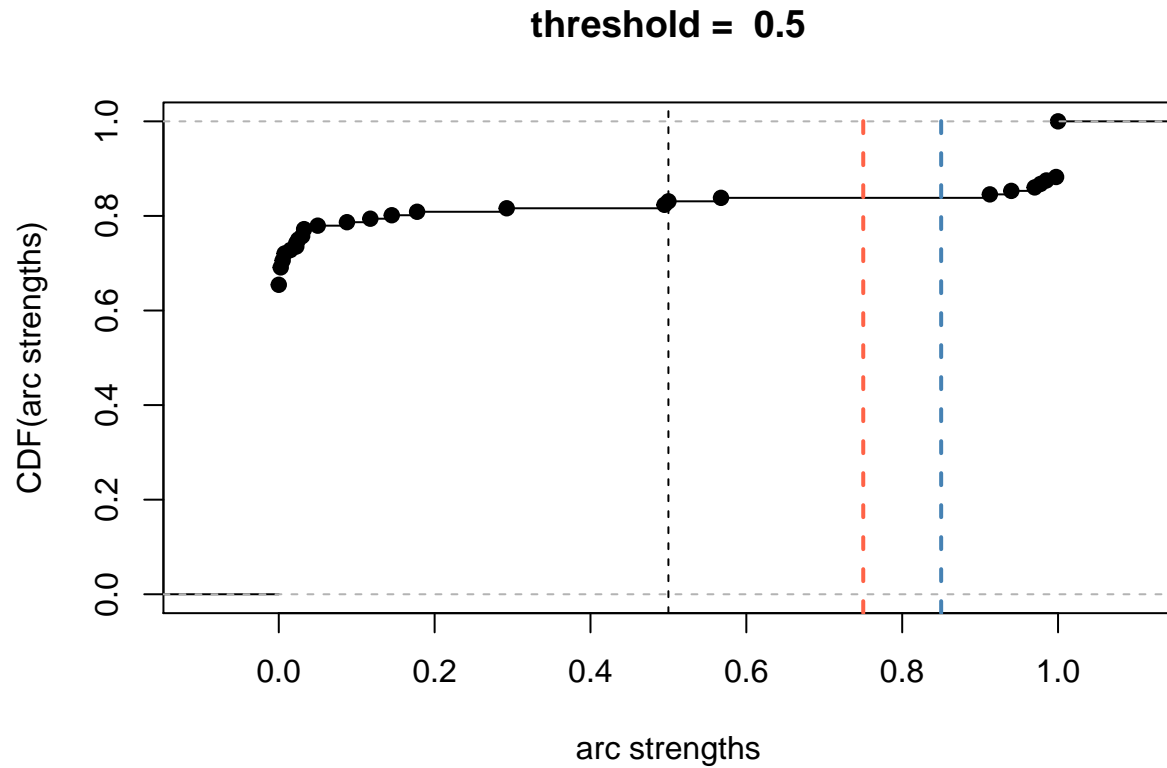


It is also a good idea to look at the threshold with respect to the distribution of the arc strengths

```

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

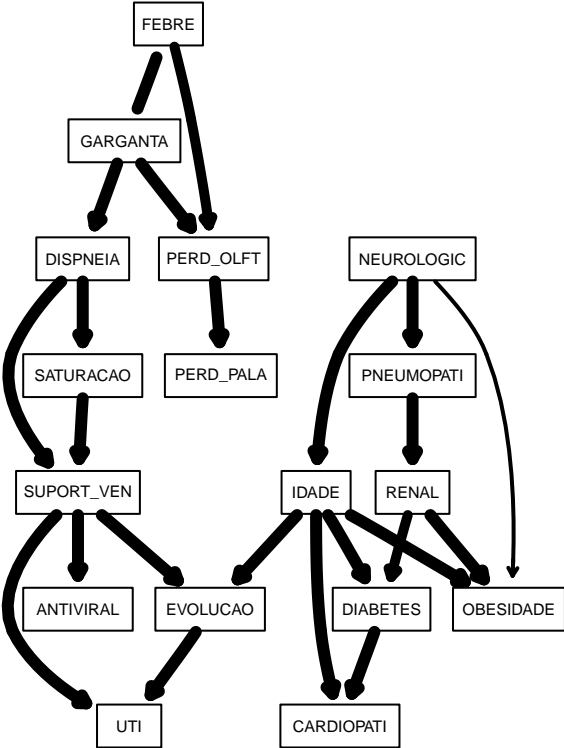
```



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 = 0.75)
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 = 0.75')
```

Iterações = 400 Thr = 0.5



Iterações = 100 Thr = 0.75

