

Covid-19 - Modelagem Causal

Parte I: Aprendizado Bayesiano de estrutura Causal na Síndrome Respiratória Aguda Grave - SRAG

17/04/2021

SRAG^{causal modeling} & COVID - 19^{simulations}

Dataset = <https://dados.gov.br/dataset/bd-srag-2021>

Banco de Dados de Síndrome Respiratória Aguda Grave - incluindo dados da COVID-19

Vigilância de Síndrome Respiratória Aguda Grave (SRAG)

O Ministério da Saúde (MS), por meio da Secretaria de Vigilância em Saúde (SVS), desenvolve a vigilância da Síndrome Respiratória Aguda Grave (SRAG) no Brasil, desde a pandemia de Influenza A(H1N1)pdm09 em 2009. A partir disso, esta vigilância foi implantada na rede de Influenza e outros vírus respiratórios, que anteriormente atuava apenas com a vigilância sentinela de Síndrome Gripal (SG).

Recentemente (2020), a vigilância da COVID-19, a infecção humana causada pelo novo Coronavírus, foi incorporada na rede de vigilância da Influenza e outros vírus respiratórios.

Esta página tem como finalidade disponibilizar o legado dos bancos de dados (BD) epidemiológicos de SRAG, da rede de vigilância da Influenza e outros vírus respiratórios, desde o início da sua implantação (2009) até os dias atuais (2021), com a incorporação da vigilância da COVID-19. Atualmente, o sistema oficial para o registro dos casos e óbitos de SRAG é o Sistema de Informação da Vigilância Epidemiológica da Gripe (SIVEP-Gripe).

No Guia de Vigilância Epidemiológica Emergência de Saúde Pública de Importância Nacional pela Doença pelo Coronavírus 2019 estão disponíveis informações sobre definições de casos, critérios de confirmação e encerramento dos casos, dentre outros.

Ressaltamos que os dados da vigilância de SRAG no Brasil disponibilizados nesta página, estão sujeitos a alterações decorrentes da investigação pelas equipes de vigilância epidemiológica que desenvolvem o serviço nas três esferas de gestão.

As bases de dados de SRAG disponibilizadas neste portal passam por tratamento que envolve a anonimização, em cumprimento a legislação.

Os dados de 2021 são disponibilizados semanalmente, às quartas-feiras, podendo, excepcionalmente, a atualização ocorrer em outro dia

Para mais informações, acessar:

Gripe/Influenza - <https://saude.gov.br/saude-de-a-z/gripe>

COVID-19 - <https://coronavirus.saude.gov.br/>

Guia Nacional de Vigilância da COVID-19 - https://portalarquivos.saude.gov.br/images/af_gvs_coronavirus_6ago20_ajustes-finais-2.pdf

Nota

Este documento é ainda tão somente experimentação e análise exploratória em fase absolutamente incipiente. Por isso a falta de estrutura, gráficos não explicativos, *typos*, etc

Inferência da estrutura da Rede Causal usando o Algoritmo mmhc

Variáveis selecionadas

```
## Rows: 76,666
## Columns: 9
## $ 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,~
## $ 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 (*whitelisting*)

```
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)
```

Descrição do DAG (*Directed Acyclic Graph*) sem *whitelisting*

```
##
## Bayesian network learned via Hybrid methods
##
## model:
```

```

##      [UTI] [SUPOORT_VEN|UTI] [EVOLUCAO|UTI:SUPOORT_VEN] [OBESIDADE|SUPOORT_VEN]
##      [ANTIVIRAL|UTI:SUPOORT_VEN] [RENAL|EVOLUCAO:OBESIDADE]
##      [PNEUMOPATI|RENAL:OBESIDADE] [DIABETES|RENAL:PNEUMOPATI]
##      [CARDIOPATI|DIABETES:OBESIDADE:PNEUMOPATI]
## nodes:                                     9
## arcs:                                     15
##   undirected arcs:                       0
##   directed arcs:                         15
## average markov blanket size:             3.78
## average neighbourhood size:              3.33
## average branching factor:                1.67
##
## learning algorithm:                      Max-Min Hill-Climbing
## constraint-based method:                 Max-Min Parent Children
## conditional independence test:           Mutual Information (disc.)
## score-based method:                     Hill-Climbing
## score:                                  BIC (disc.)
## alpha threshold:                        0.05
## penalization coefficient:                5.570924
## tests used in the learning procedure:    1365
## optimized:                             TRUE

```

Descrição do DAG com *whitelisting*

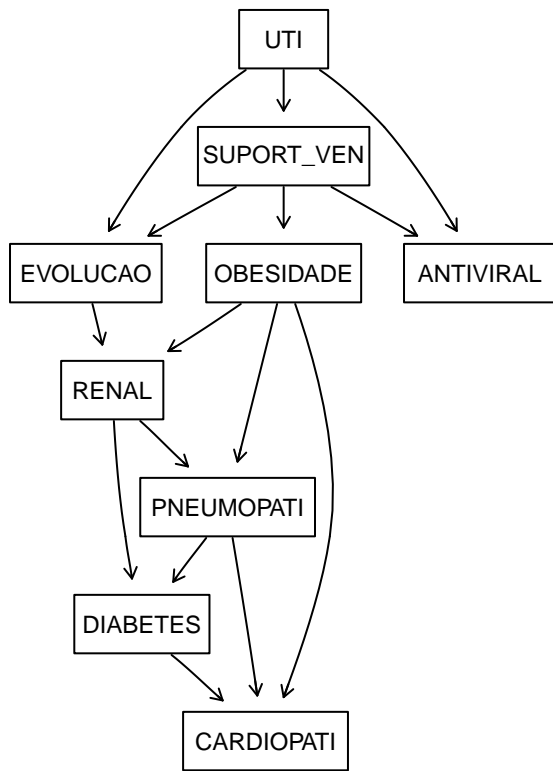
```

##
## Bayesian network learned via Hybrid methods
##
## model:
##      [UTI] [EVOLUCAO|UTI] [SUPOORT_VEN|EVOLUCAO:UTI] [OBESIDADE|SUPOORT_VEN]
##      [ANTIVIRAL|UTI:SUPOORT_VEN] [DIABETES|OBESIDADE] [RENAL|DIABETES:OBESIDADE]
##      [PNEUMOPATI|RENAL:DIABETES:OBESIDADE]
##      [CARDIOPATI|DIABETES:OBESIDADE:PNEUMOPATI]
## nodes:                                     9
## arcs:                                     15
##   undirected arcs:                       0
##   directed arcs:                         15
## average markov blanket size:             3.33
## average neighbourhood size:              3.33
## average branching factor:                1.67
##
## learning algorithm:                      Max-Min Hill-Climbing
## constraint-based method:                 Max-Min Parent Children
## conditional independence test:           Mutual Information (disc.)
## score-based method:                     Hill-Climbing
## score:                                  BIC (disc.)
## alpha threshold:                        0.05
## penalization coefficient:                5.570924
## tests used in the learning procedure:    1130
## optimized:                             TRUE

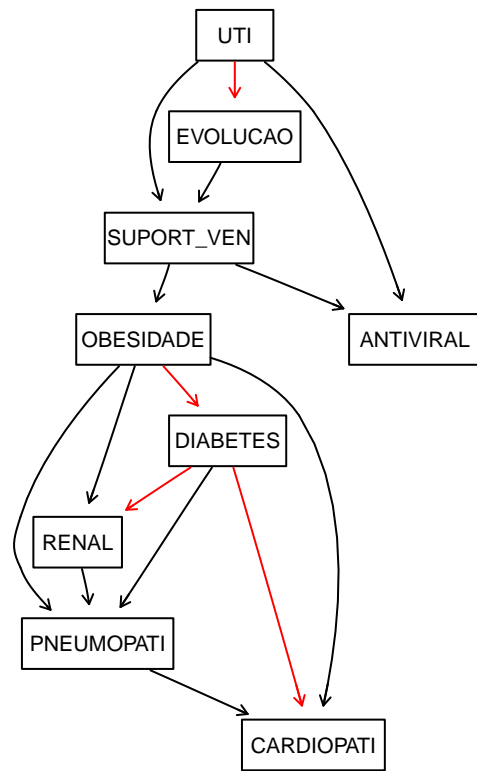
```

DAG usando ou não *whitelisting* no algoritmo de aprendizagem

DAG sem WL

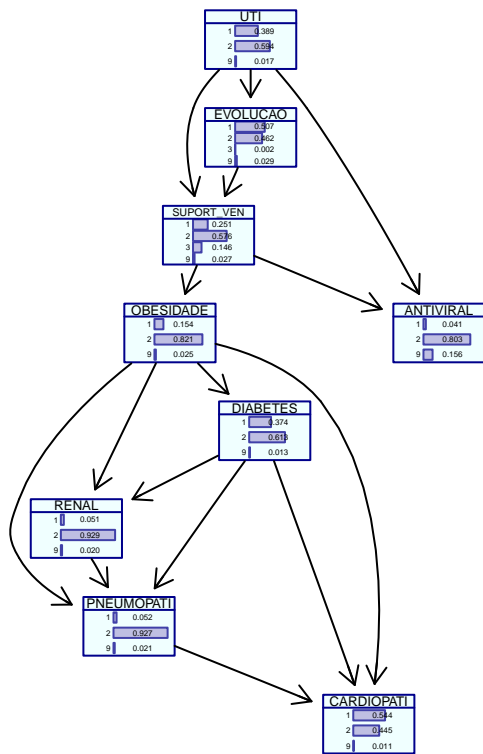


DAG com imposição de uma WL

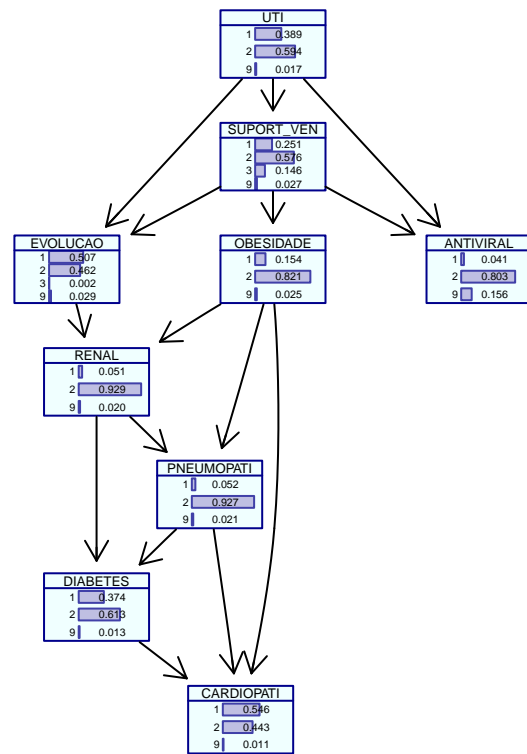


Distribuição marginal de probabilidades

DAG sem WL

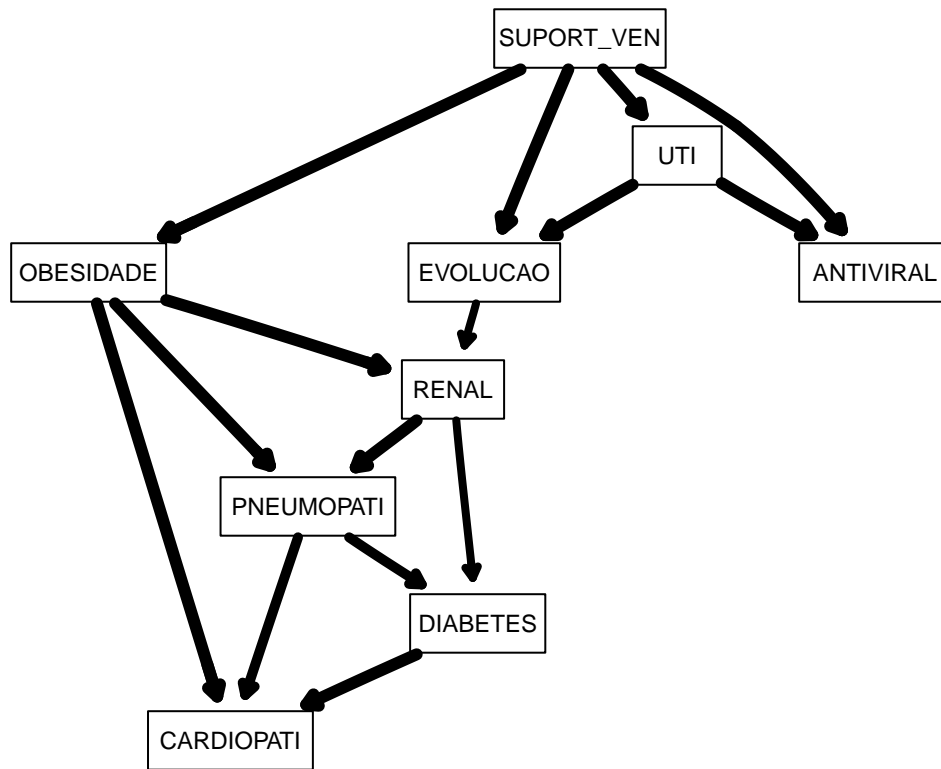


DAG com WL



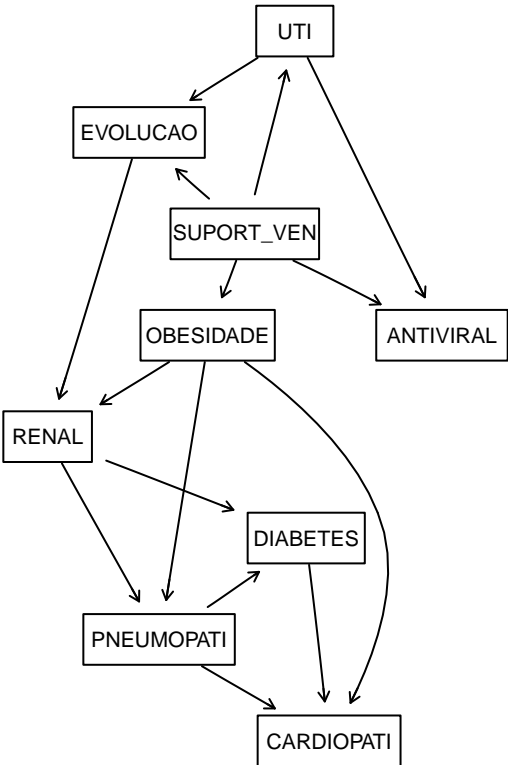
Obtendo o DAG médio: fase de Bootstrap usando 300 ciclos

Iter = 300 Thr: 0.5266666666666667

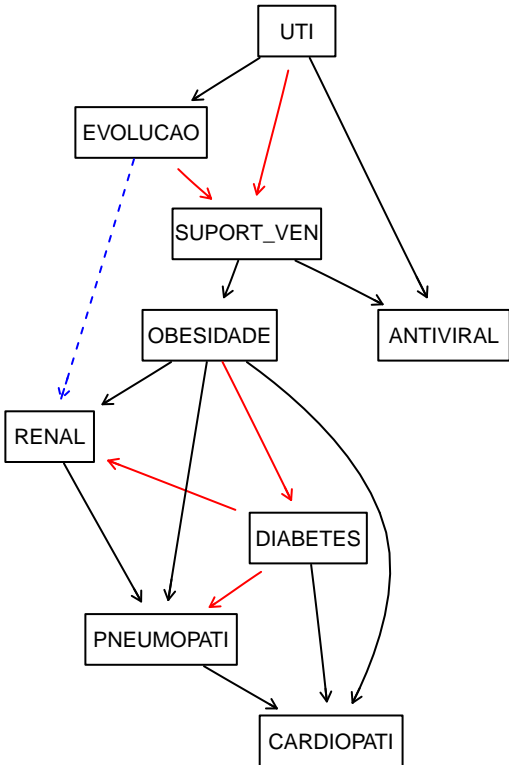


How can we compare the averaged network (avg.diff) with the network we originally learned in from all the data? The most qualitative way is to plot the two networks side by side, with the nodes in the same positions, and highlight the arcs that appear in one network and not in the other, or that appear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

DAG médio sem WL

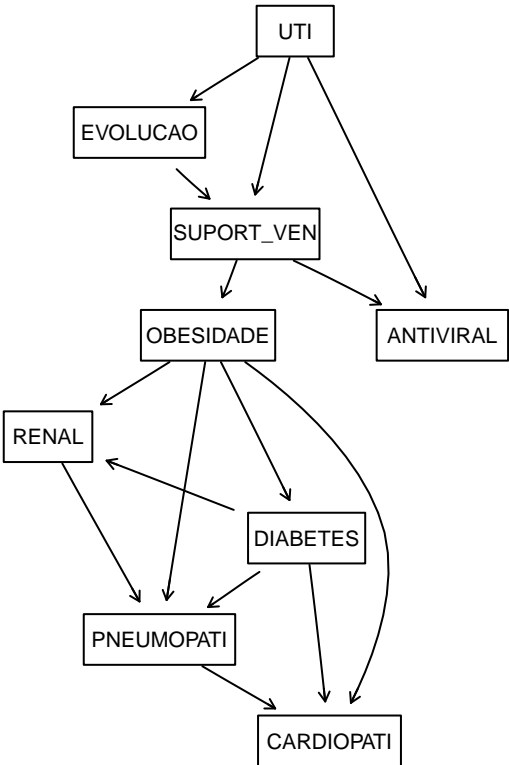


DAG médio com WL

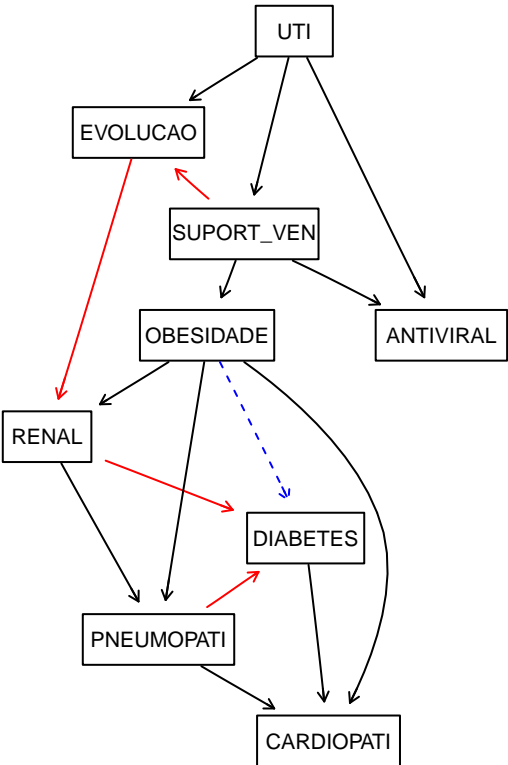


rections.

DAG único sem WL

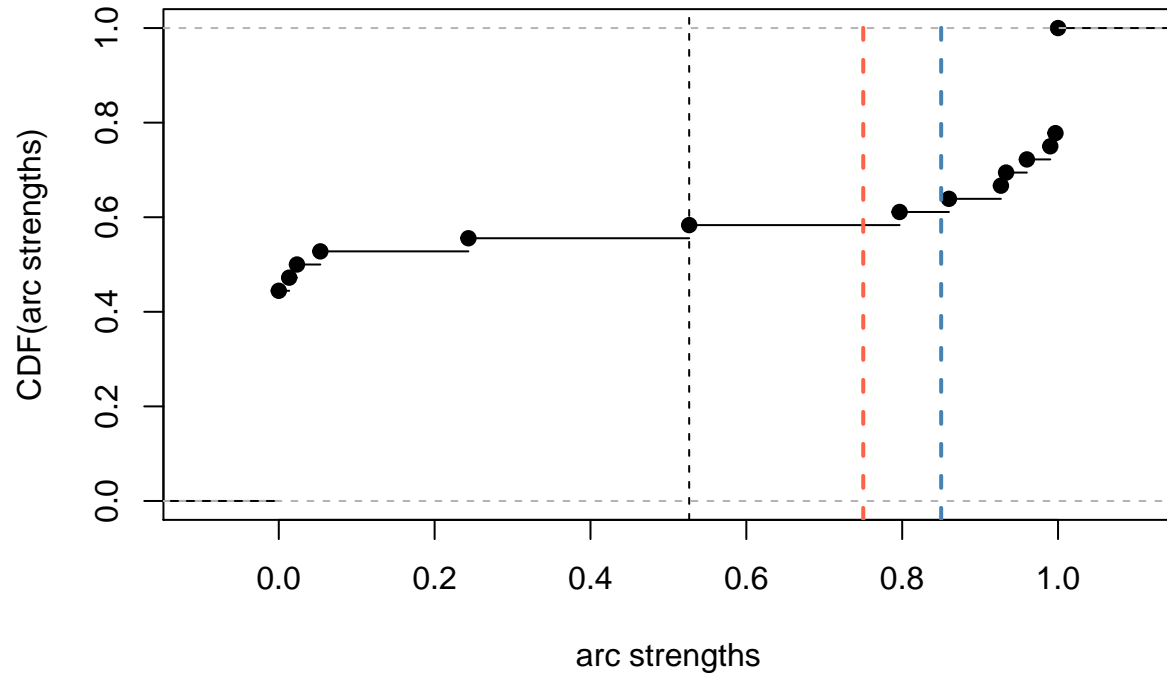


DAG único com WL



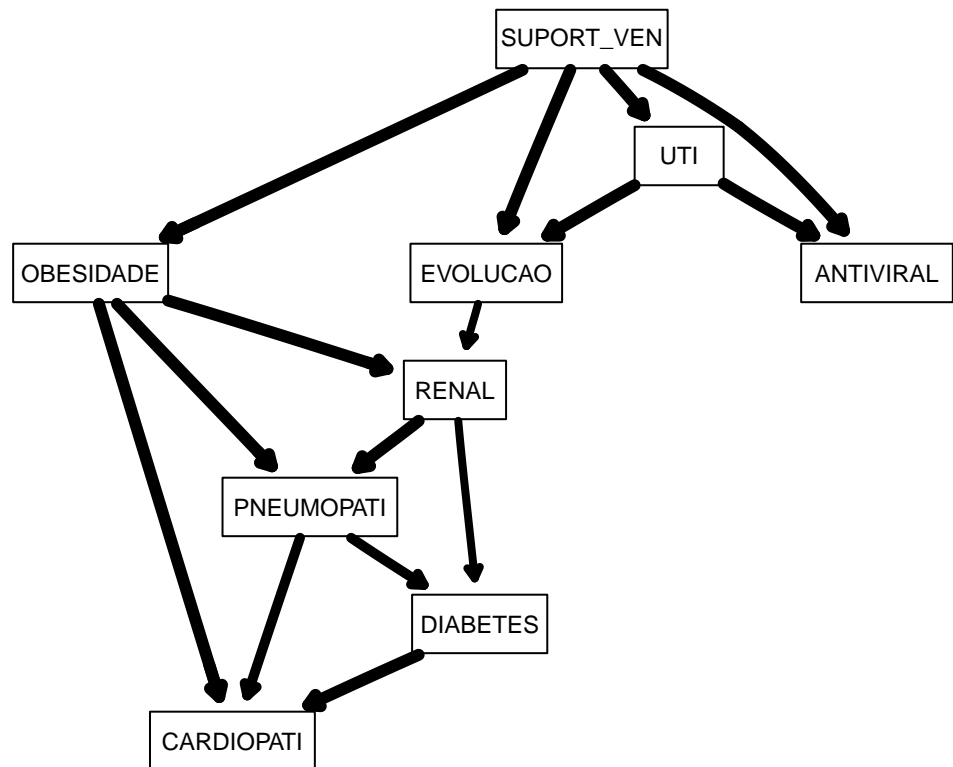
It is also a good idea to look at the threshold with respect to the distribution of the arc strengths

threshold = 0.527



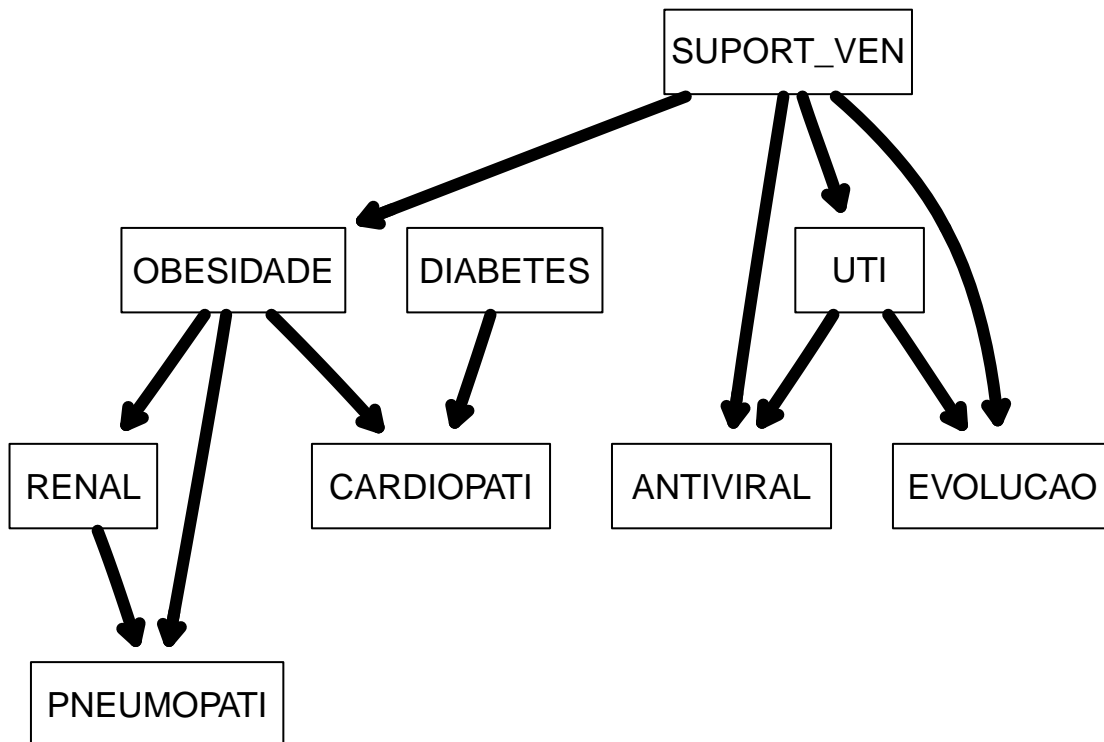
The simpler network we obtain by setting **threshold = 0.95** in `averaged.network()` is shown below; it is certainly easier to reason with from a qualitative point of view. Na figura abaixo R é o número de iterações usadas

R = 300 Thr = 0.5266666666666667



na fase de *bootstrapping*

R = 300 Thr = 0.95



Markov Blanket da variável EVOLUCAO

RENAL OBESIDADE UTI SUPORT_VEN

UTI SUPORT_VEN