Inferencia. Fase I

José Elvano Moraes

17/04/2021

Redes Bayesianas

Probabilistic reasoning on BNs works in the framework of Bayesian statistics and focuses on the computation of posterior probabilities or densities. For example, suppose we have learned a BN B with DAG G and parameters Θ. We want to use B to investigate the effects of a new piece of evidence E using the knowledge encoded in B, that is, to investigate the posterior distribution

$$P(\mathbf{X}|\mathbf{E}, \mathcal{B}) = P(\mathbf{X}|\mathbf{E}, \mathbf{G}, \Theta)$$

The first step of fitting a Bayesian network is called structure learning and consists in identifying the graph structure of the Bayesian network. Ideally, it should be the minimal I-map of the dependence structure of the data or, failing that, it should at least result in a dis- tribution as close as possible to the correct one in the probability space. Several algorithms have been proposed in the literature for structure learning. Despite the variety of theoretical backgrounds and terminology, they fall under three broad cate- gories: constraint-based, score-based, and hybrid algorithms. As an alternative, the network structure can be built manually from the domain knowledge of a human expert and prior information available on the data.

he second step is called parameter learning. As the name suggests, it imple- ments the estimation of the parameters of the global distribution. This task can be performed efficiently by estimating the parameters of the local distributions implied by the structure obtained in the previous step.

Questions that can be asked are called queries and are typically an event of interest. The two most common queries are conditional probability (CPQ) and maximum a posteriori (MAP) queries, also known as most probable explanation (MPE) queries

some content here

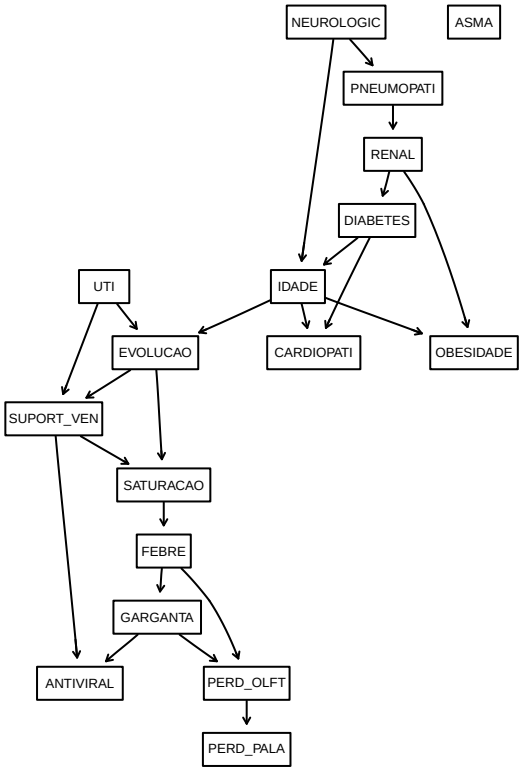
```
glimpse(ddf)
```

```
## 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, 1, 2, 1, 2, 2, 2, 1, 2, 2, 2, 2,~
## $ SATURACAO  <fct> 1, 1, 2, 2, 2, 2, 2, 1, 2, 2, 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,~
## $ 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,~
## $ PERD_OLFT  <fct> 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,~
```

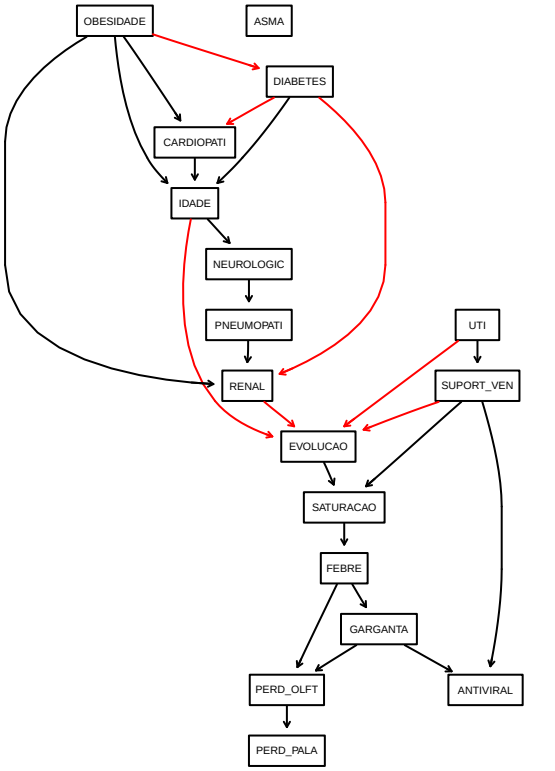
```
## $ NEUROLOGIC <fct> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 9, 2, 2, 2,~
## $ PNEUMOPATI <fct> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,~
## $ UTI        <fct> 1, 1, 2, 1, 2, 2, 1, 2, 2, 2, 2, 2, 1, 1, 2, 2, 1, 2, 2, 2,~
## $ CARDIOPATI <fct> 2, 2, 2, 1, 1, 2, 2, 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,~
## $ ASMA       <fct> 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,~
## $ ANTIVIRAL  <fct> 2, 2, 2, 9, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,~
```

some content here

DAG sem WL



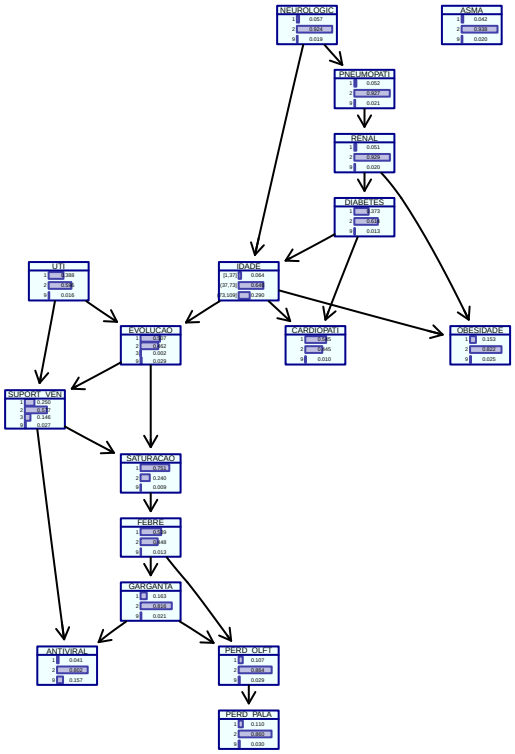
DAG com imposição de uma WL



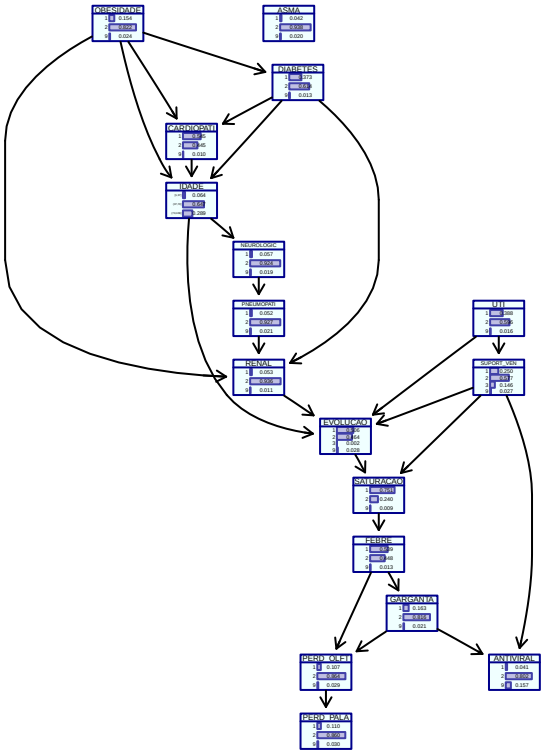
some content here

```
## Warning in from.bn.fit.to.grain(x): NaN conditional probabilities in EVOLUCAO,
## replaced with a uniform distribution.
```

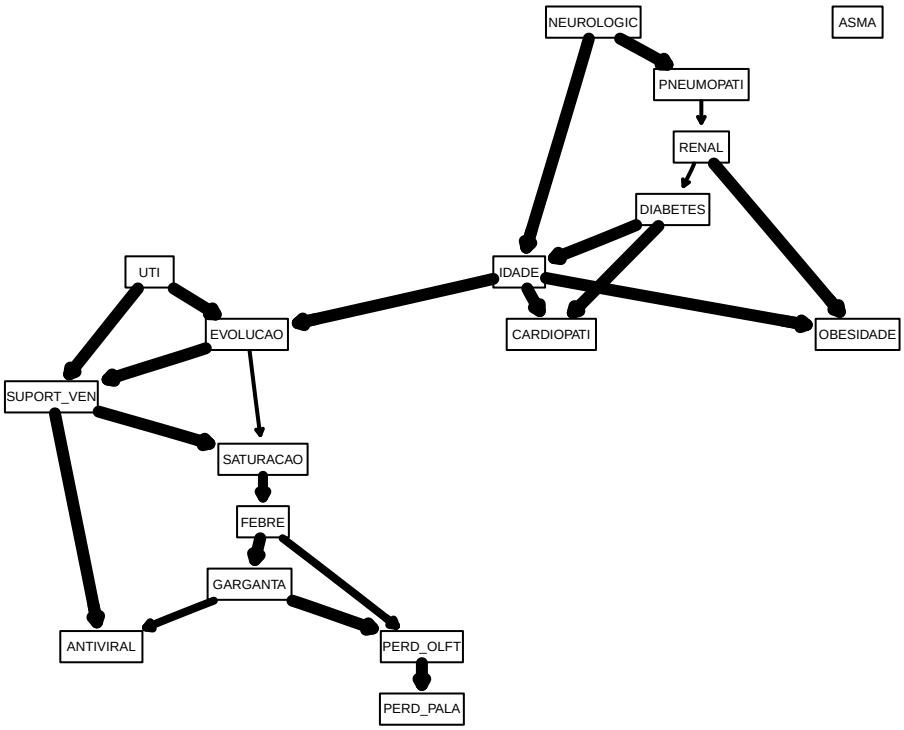
DAG sem WL



DAG com WL



DAG médio e *força da correlação* entre pares de variáveis



A espessura dos arcos representa a correlação de Pearson entre variáveis

Descrição das redes

```
avg.diff
```

```
##
##   Random/Generated Bayesian network
##
##   model:
##    [NEUROLOGIC][UTI][ASMA][PNEUMOPATI|NEUROLOGIC][RENAL|PNEUMOPATI]
##    [DIABETES|RENAL][IDADE|DIABETES:NEUROLOGIC][EVOLUCAO|IDADE:UTI]
##    [OBESIDADE|IDADE:RENAL][CARDIOPATI|IDADE:DIABETES][SUPORT_VEN|EVOLUCAO:UTI]
##    [SATURACAO|EVOLUCAO:SUPORT_VEN][FEBRE|SATURACAO][GARGANTA|FEBRE]
##    [PERD_OLFT|FEBRE:GARGANTA][ANTIVIRAL|GARGANTA:SUPORT_VEN]
##    [PERD_PALA|PERD_OLFT]
##   nodes:                                 17
##   arcs:                                  22
##     undirected arcs:                     0
##     directed arcs:                       22
##   average markov blanket size:           3.06
##   average neighbourhood size:            2.59
##   average branching factor:              1.29
##
##   generation algorithm:                  Model Averaging
```

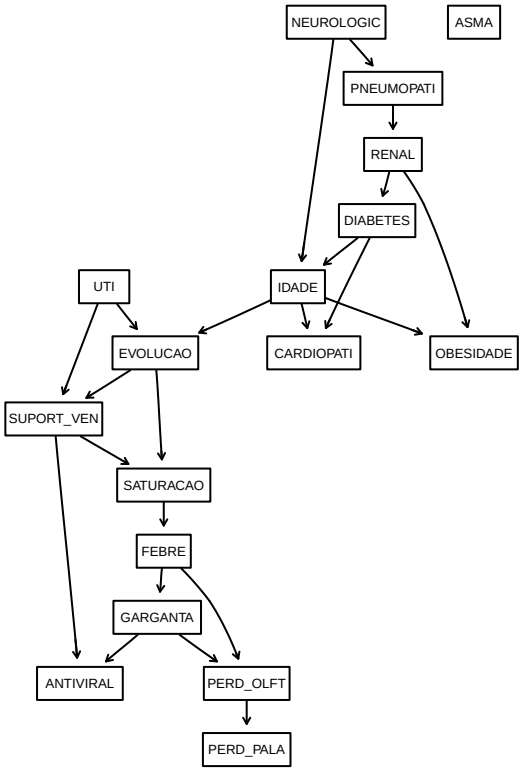
```
##   significance threshold:               0.45
```

```
avg.simpler
```

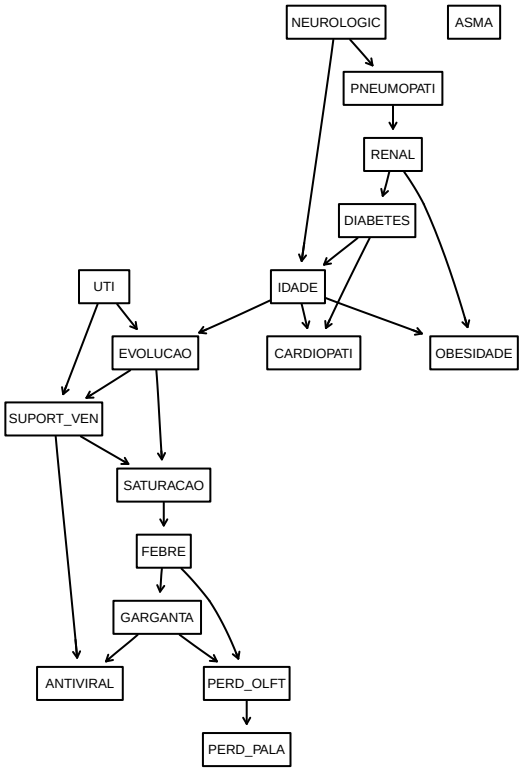
```
##
##   Random/Generated Bayesian network
##
##   model:
##    [FEBRE][RENAL][DIABETES][NEUROLOGIC][UTI][ASMA][IDADE|DIABETES:NEUROLOGIC]
##    [GARGANTA|FEBRE][PNEUMOPATI|NEUROLOGIC][EVOLUCAO|IDADE:UTI]
##    [OBESIDADE|IDADE:RENAL][PERD_OLFT|GARGANTA][CARDIOPATI|IDADE:DIABETES]
##    [PERD_PALA|PERD_OLFT][SUPORT_VEN|EVOLUCAO:UTI][SATURACAO|SUPORT_VEN]
##    [ANTIVIRAL|SUPORT_VEN]
##   nodes:                                17
##   arcs:                                 16
##     undirected arcs:                    0
##     directed arcs:                      16
##   average markov blanket size:          2.24
##   average neighbourhood size:           1.88
##   average branching factor:             0.94
##
##   generation algorithm:                 Model Averaging
##   significance threshold:               0.95
```

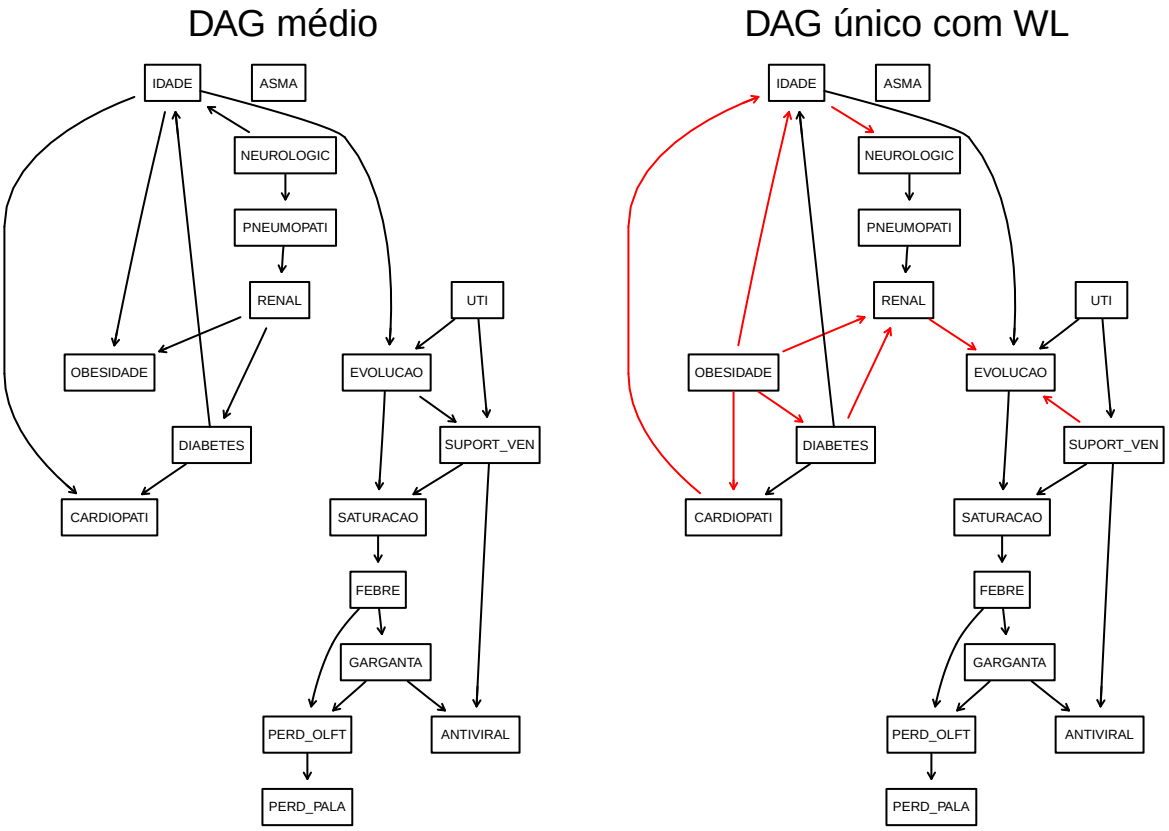
Dag médio *versus* DAG único

DAG médio



DAG único sem WL



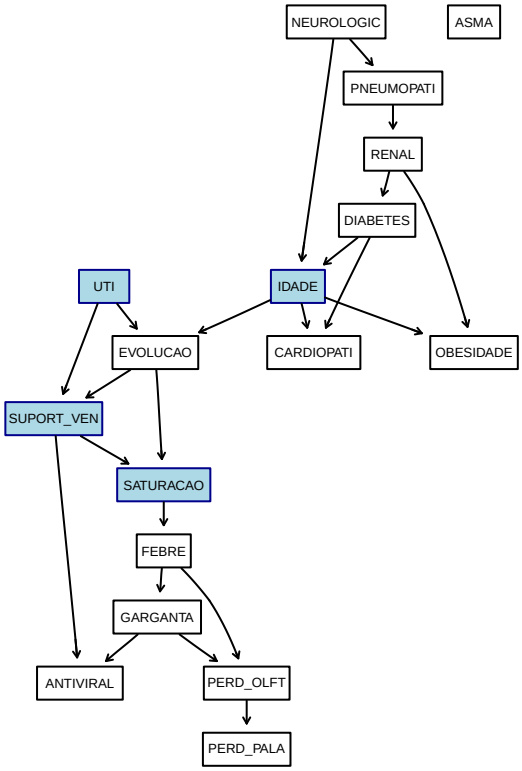


DAG médio *versus* simplificado

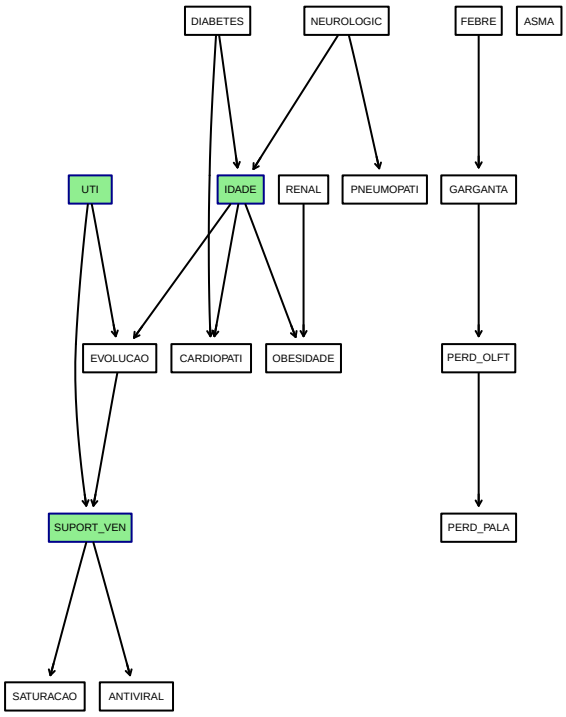
DAG médio obtido pelo processo de bootstrapping e DAG *simplicada* no qual desenhou-se somente os arcos com *strenght*, (correlação de Pearson) acima de 0.95.

Os nodos coloridos formam o *markov blanket* da variável EVOLUCAO, ou seja, os nodos suficientes para descrevere completamente a distribuição estatistica da variável

Rede Média



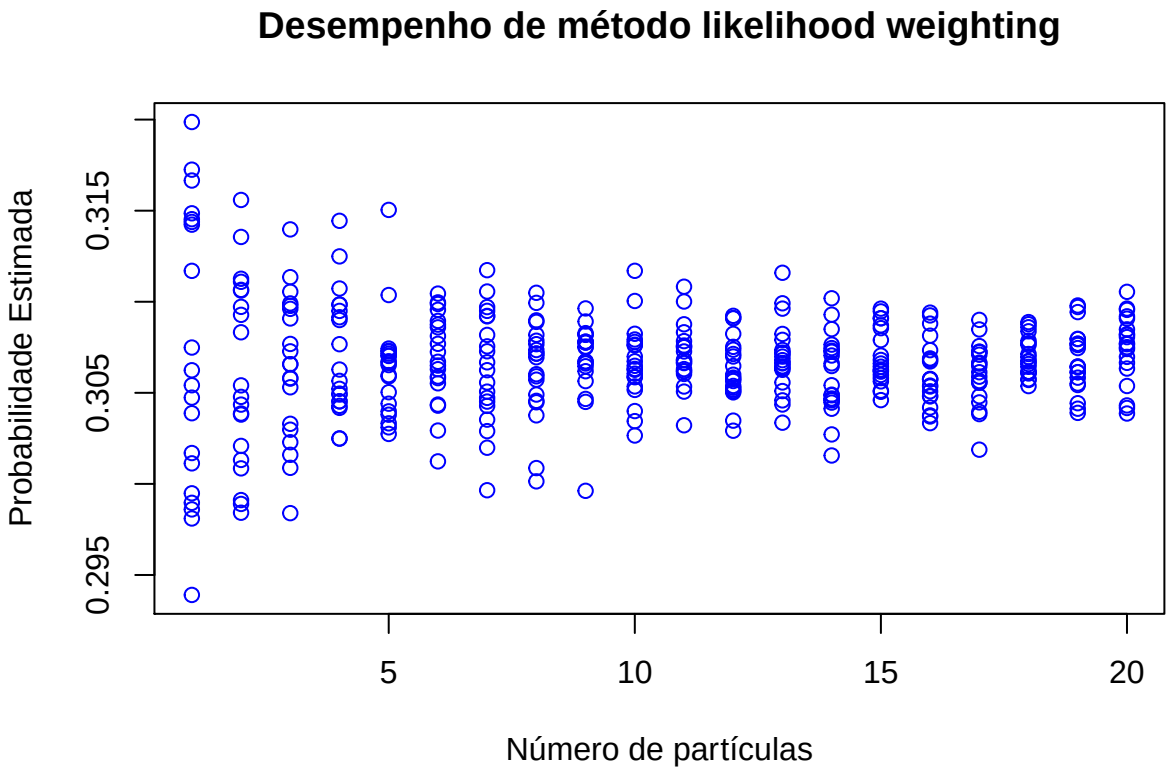
Rede simpificada



Performing likelihood weighting

TODO

With cpquery by setting method = “lw” and specifying the evidence as a named list with one element for each node we are conditioning on



Inferência

In practice, probabilistic reasoning on Bayesian networks has its roots embedded in Bayesian statistics and focuses on the computation of posterior probabilities or densities. For example, suppose we have learned a Bayesian network B with. Bayesian inference on the other hand is often a follow-up to Bayesian net- work learning and deals with inferring the state of a set of variables given the state of others as evidence.

Bayesian networks, like other statistical models, can be used to answer questions about the nature of the data that go beyond the mere description of the observed sample. Techniques used to obtain those answers based on new evidence are known in general as inference. For Bayesian networks, the process of answering these questions is also known as probabilistic reasoning or belief updating, while the questions themselves are called queries.

In practice, probabilistic reasoning on Bayesian networks has its roots embedded in Bayesian statistics and focuses on the computation of posterior probabilities or densities. For example, suppose we have learned a Bayesian network B with

structure G and parameters Θ , under one of the distributional assumptions detailed in Sect. 2.2.4. Subsequently, we want to investigate the effects of a new piece of evidence E on the distribution of X using the knowledge encoded in B, that is, to investigate the posterior distribution $P(X|E, B) = P(X|E, G, \Theta)$. The approaches used for this kind of analysis vary depending on the nature of E and on the nature of information we are interested in. The two most common kinds of evidence are as follows:

- Hard evidence, an instantiation of one or more variables in the network. In other words,
- Soft evidence, a new distribution for one or more variables in the network. Since both the network structure and the distributional assumptions are treated as fixed, soft evidence is usually specified as a new set of parameters,

As far as queries are concerned, we will focus on conditional probability queries (CPQ) and maximum a

posteriori (MAP) queries, also known as most probable explanation (MPE) queries. Both apply mainly to hard evidence, even though they can be used in combination with soft evidence.

Predição

Sabe-se que paciente **está** no CTI, estima-se qual a distribuição marginal de probabilidade das variáveis

- IDADE
- RENAL
- EVOLUCAO
- ANTIVIRAL

Oberve que as variáveis RENAL é independentes das demais na rede

---

Legenda para a interpretação das probabilidade das variáveis

EVOLUCAO

1 - Cura, 2 - Obito por COVID-19, 3 - Óbito por outras causas, 9 - Ignorado

ANTIVIRAL

1 - Oseltamivir, 2 - Zanamivir, 3 - Outro

RENAL

1 - sim, 2 - não, 3 - ignorado

Cenário 1

- UTI: não
- SUPORT_VEN: não

```
## $IDADE
## IDADE
##   [1,37]  (37,73] (73,109]
##    0.072    0.673    0.255
##
## $EVOLUCAO
## EVOLUCAO
##      1      2      3      9
## 0.7973 0.1686 0.0018 0.0323
##
## $RENAL
## RENAL
##     1     2     9
## 0.051 0.929 0.020
##
## $ANTIVIRAL
## ANTIVIRAL
##     1     2     9
## 0.043 0.821 0.136
```

Cenário 2

- UTI: sim
- SUPORT_VEN: invasivo

```
## $IDADE
## IDADE
##   [1,37]  (37,73] (73,109]
##    0.052    0.628    0.320
##
## $RENAL
## RENAL
##     1     2     9
## 0.051 0.929 0.020
##
## $EVOLUCAO
## EVOLUCAO
##      1      2      3      9
## 0.0992 0.8842 0.0032 0.0133
##
## $ANTIVIRAL
## ANTIVIRAL
##     1     2     9
## 0.038 0.798 0.164
```

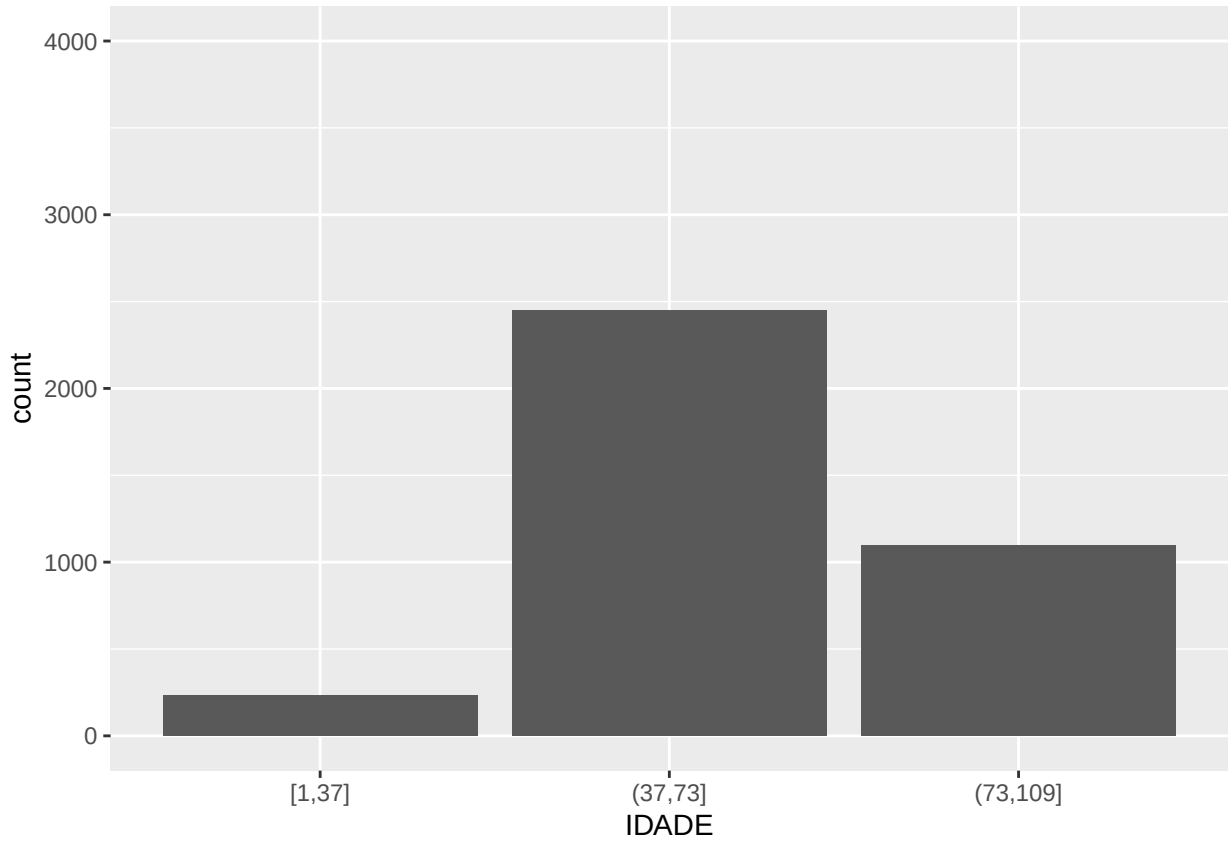
Cenário 3

- UTI: sim
- SUPORT_VEN: nao invasivo

```
## $IDADE
## IDADE
##   [1,37]  (37,73] (73,109]
##    0.077    0.666    0.257
##
## $RENAL
## RENAL
##     1     2     9
## 0.051 0.929 0.020
##
## $EVOLUCAO
## EVOLUCAO
##      1      2      3      9
## 0.4678 0.5077 0.0021 0.0223
##
## $ANTIVIRAL
## ANTIVIRAL
##     1     2     9
## 0.043 0.818 0.139
```

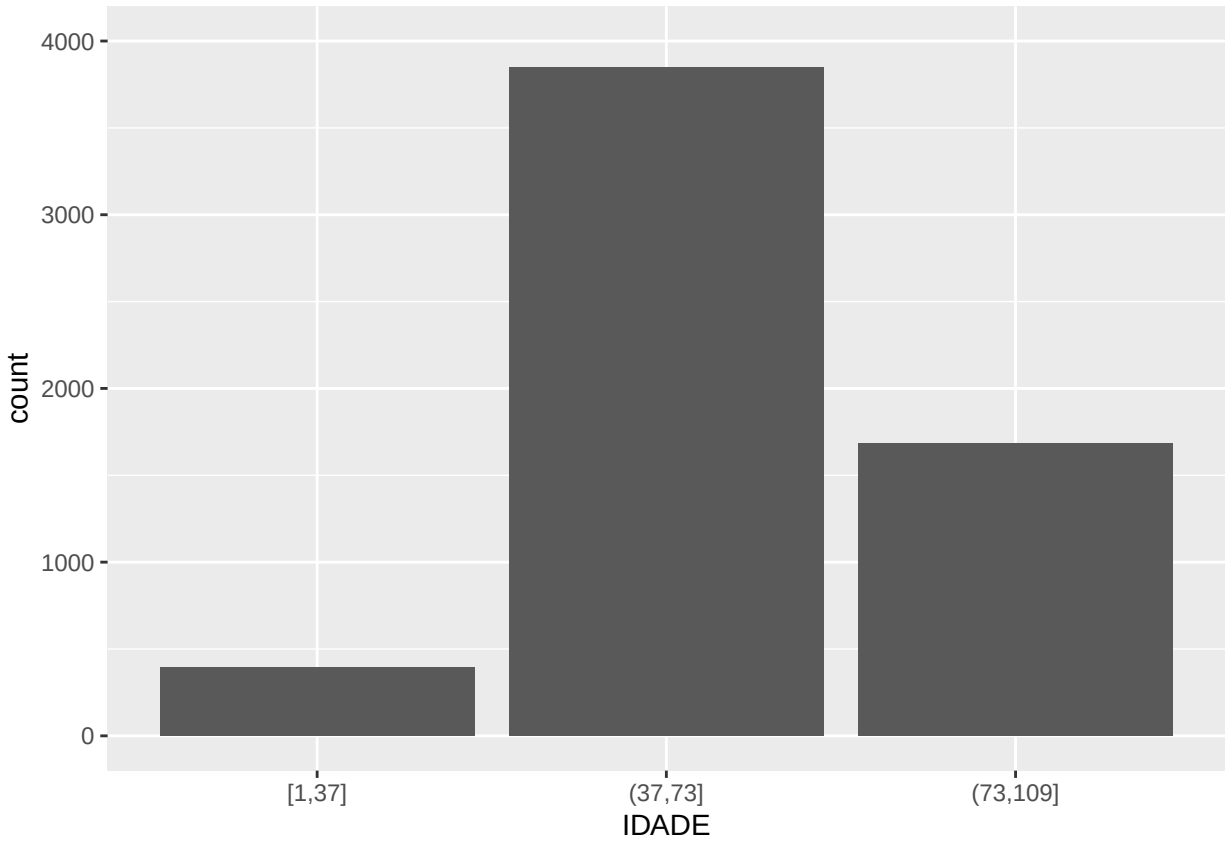
```
SxT = cpdist(fitt3, nodes = c("IDADE", "RENAL", "EVOLUCAO", "ANTIVIRAL"), evidence = UTI == "1")
ggplot(SxT, aes(IDADE)) + geom_histogram(stat = "count") + ylim(0, 4000)
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```




```
SxT = cpdist(fitt3, nodes = c("IDADE", "RENAL", "EVOLUCAO", "ANTIVIRAL"), evidence = UTI == "2")
ggplot(SxT, aes(IDADE)) + geom_histogram(stat = "count") + ylim(0, 4000)
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```


4000
3000
2000
1000
0
count
[1,37]
(37,73]
(73,109]
IDADE