

**WYDZIAŁ
MATEMATYKI
I FIZYKI STOSOWANEJ**
POLITECHNIKI RZESZOWSKIEJ

Wnioskowanie w warunkach niepewności

Projekt

Gabriel Lichacz

Rzeszów, 2022

Spis treści

1. Wstęp.....	3
2. Budowa sieci	3
2.1. Wstępna obróbka danych	3
2.2. Dane ciągłe.....	4
2.3. Dane dyskretyzowane	5
2.4. Score.....	7
2.5. Testy dyskretyzacji	7
3. Wnioskowanie	9
3.1. Prawdopodobieństwa warunkowe.....	9
3.1.1. Obliczone ręcznie.....	9
3.1.2. Obliczone w R	10
3.2. Rozkłady warunkowe.....	12
4. Podsumowanie.....	15
5. Spis ilustracji	16
6. Spis tabel	16
7. Kod źródłowy	16

1. Wstęp

Dane dotyczą zaburzeń wątroby i pochodzą z BUPA Medical Research Ltd. Pięć pierwszych zmiennych to badania krwi, które uważa się za wrażliwe na zaburzenia wątroby, wynikające z nadmiernego spożywania alkoholu. Każdy wiersz danych to pojedyncza osoba płci męskiej. Zbiór nie zawiera informacji o tym czy dana osoba posiada dolegliwości wątroby. Zestaw danych posiada 345 wierszy oraz sześć kolumn danych.

Zmienne w zbiorze:

- [1] mcv – wskaźnik średniej objętości krwinki czerwonej
- [2] alkphos – fosfataza alkaliczna
- [3] sgpt – aminotransferaza alaninowa
- [4] sgot – aminotransferaza asparaginianowa
- [5] gammagt – gamma-glutamylotranspeptydaza
- [6] drinks – ilość napojów alkoholowych objętości pół pinty wypijanych dziennie
- [7] class – pole z selektorem stworzonym przez badaczy BUPA do podziału danych na zbiory testowe

2. Budowa sieci

2.1. Wstępna obróbka danych

Dane po wczytaniu konwertuję dla pewności na typ numeric oraz usuwam kolumnę class. Nie wnosi ona nic do badanych danych a może wpłynąć negatywnie na model sieci.

	mcv	alkphos	sgpt	sgot	gammagt	drinks
1	85	92	45	27	31	0.0
2	85	64	59	32	23	0.0
3	86	54	33	16	54	0.0
4	91	78	34	24	36	0.0
5	87	70	12	28	10	0.0
6	98	55	13	17	17	0.0
7	88	62	20	17	9	0.5
8	88	67	21	11	11	0.5
9	92	54	22	20	7	0.5
10	90	60	25	19	5	0.5

rys. 2-1 Dane

Przeprowadzam test Shapiro-Wilka na normalność rozkładu.

```
shapiro_t <- c()
for(i in 1:(length(dane)-1)){
  shapiro_t[i] <- shapiro.test(dane[,i])$p
}
shapiro_t
```

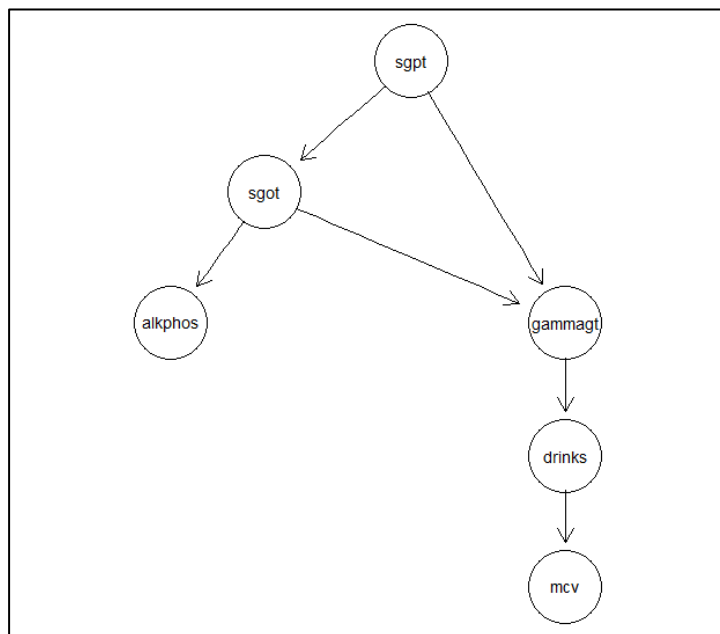
Kolumna	mcv	alkphos	sgpt	sgot	gammagt	drinks
Wartość p	3.340830e-06	3.604551e-07	2.579879e-23	1.402884e-19	6.480735e-25	1.686482e-18

tab. 1 Wartości p dla zmiennych

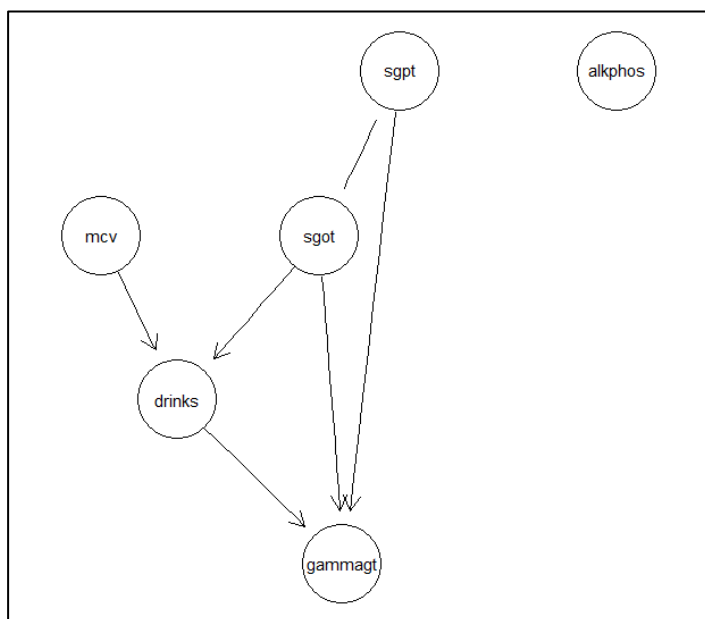
Obliczone wartości p są znacznie mniejsze niż $\alpha = 0,05$, co oznacza, że żadna ze zmiennych nie ma charakterystyki się rozkładem normalnym. Dane należy zdyskredytować.

2.2. Dane ciągłe

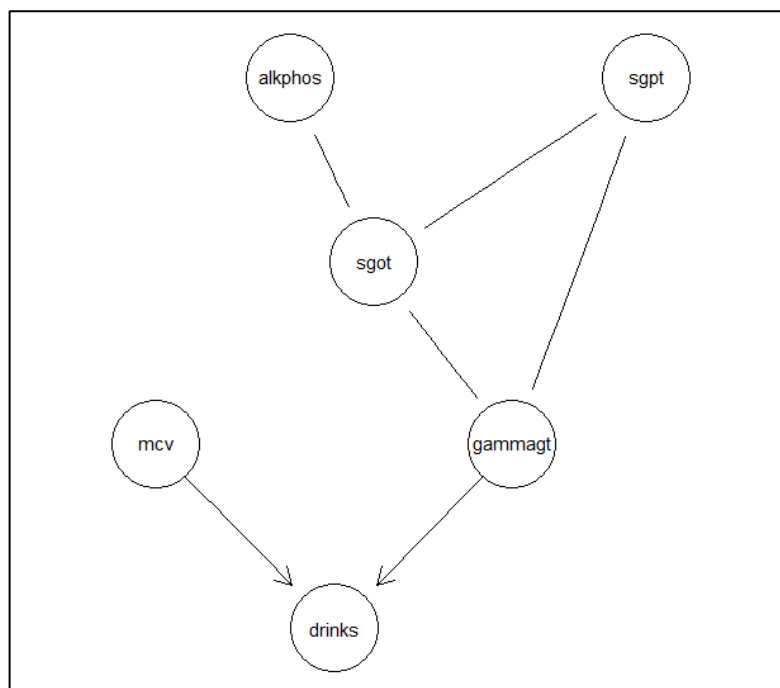
Modele zbudowane dla oryginalnych danych. Stworzone w celach poglądowych.



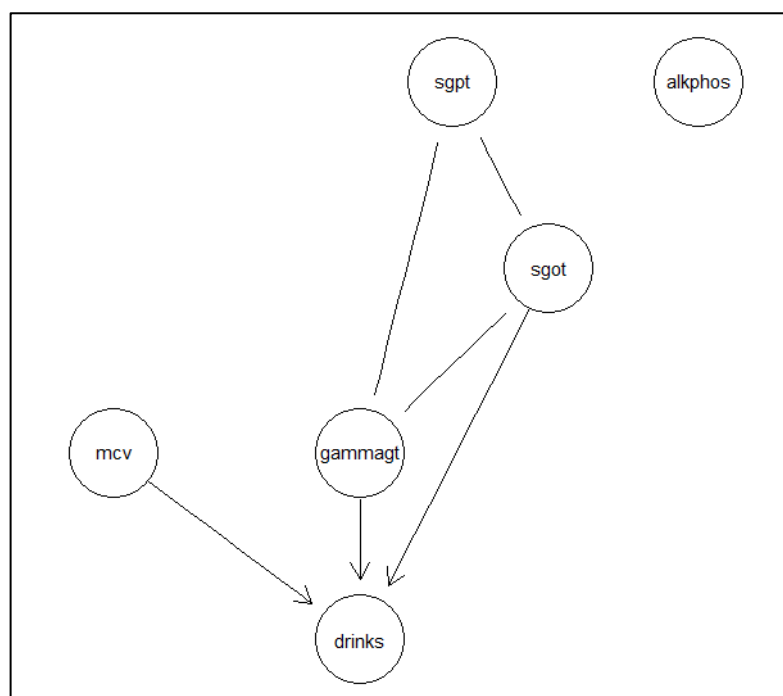
rys. 2-2 Sieć zbudowana algorytmem hc dla danych ciągłych



rys. 2-3 Sieć zbudowana algorytmem pc.stable dla danych ciągłych



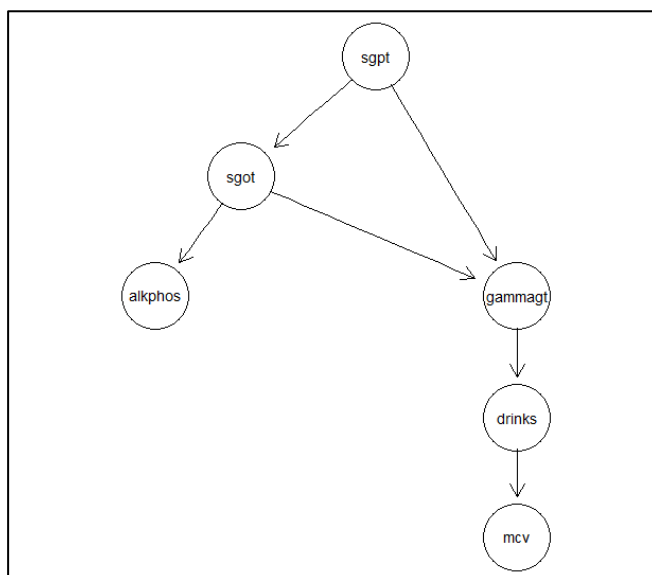
rys. 2-4 Sieć zbudowana algorytmem gs dla danych ciągłych



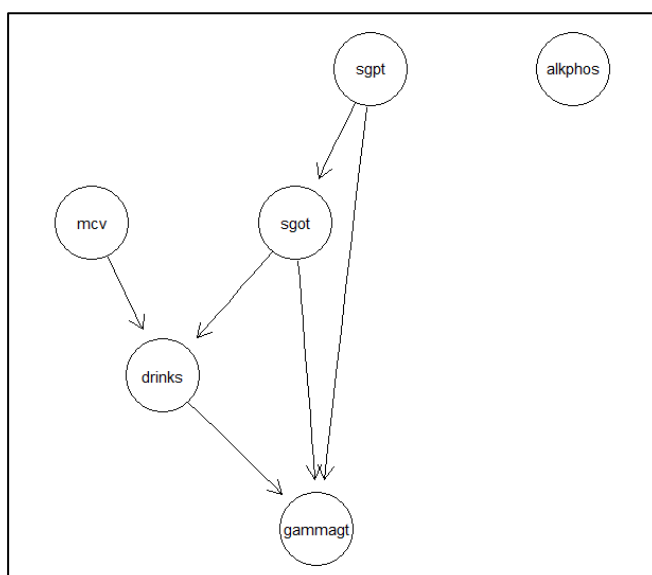
rys. 2-5 Sieć zbudowana algorytmem iamb dla danych ciągłych

2.3. Dane dyskretyzowane

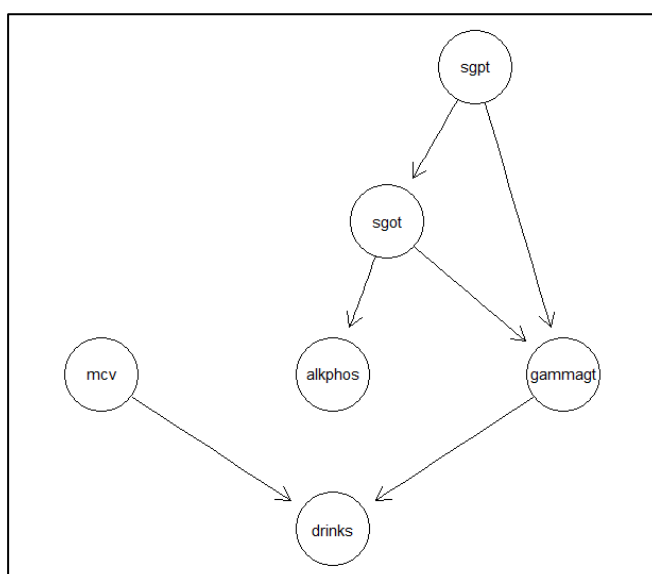
Sieci otrzymane po dyskretyzowaniu danych nie były zadowalające, przez co przebudowane zostały na wzór swoich odpowiedników dla danych ciągłych. W początkowej dyskretyzacji zmienne podzieliłem na odpowiednio 3, 3, 4, 5, 6, 7 przedziałów.



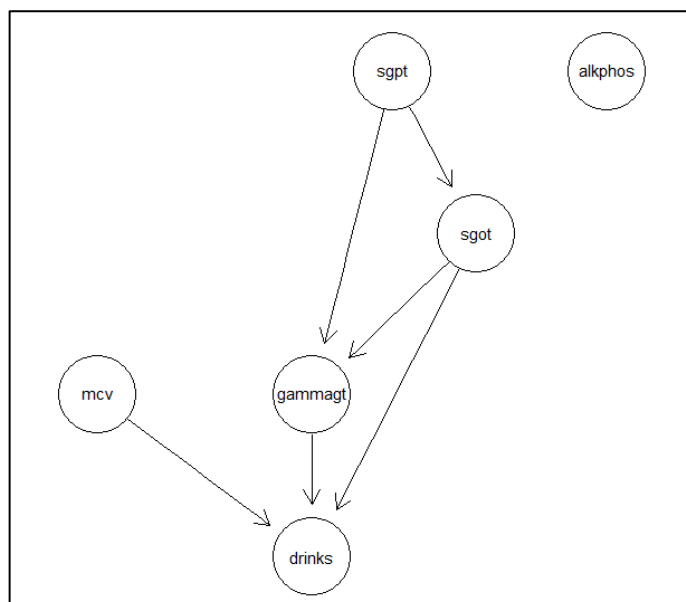
rys. 2-6 Sieć zbudowana algorytmem hc dla danych dyskretnych



rys. 2-7 Sieć zbudowana algorytmem pc.stable dla danych dyskretnych



rys. 2-8 Sieć zbudowana algorytmem gs dla danych dyskretnych



rys. 2-9 Sieć zbudowana algorytmem iamb dla danych dyskretnych

2.4. Score

Najlepszy wynik został otrzymany dla sieci zbudowanej algorytmem hc dla danych dyskretnych. Jest on najbliższej zera i wynosi -2071.689.

Nazwa algorytmu budującego sieć	Dane dyskretyzowane?	Score
hc	nie	-7766.483
	tak	-2071.689
pc.stable	nie	graf częściowo skierowany
	tak	-3898.508
gs	nie	graf częściowo skierowany
	tak	-2249.955
iamb	nie	graf częściowo skierowany
	tak	-3450.4

tab. 2 Score dla zbudowanych sieci

2.5. Testy dyskretyzacji

W celu osiągnięcia najlepszej oceny modelu przeprowadziłem testy na ile przedziałów należy podzielić każdą kolumnę danych. Do testów użyłem modelu sieci zbudowanej przy pomocy algorytmu hc.

Algorytm sumował ile wartości znajduje się w danym przedziale i liczył odległości między wartościami. Celem było znalezienie takiej liczby podziałów danych w kolumnie, by odległości były jak najmniejsze. Taki zabieg sprawiał, że dane były rozłożone równomiernie.

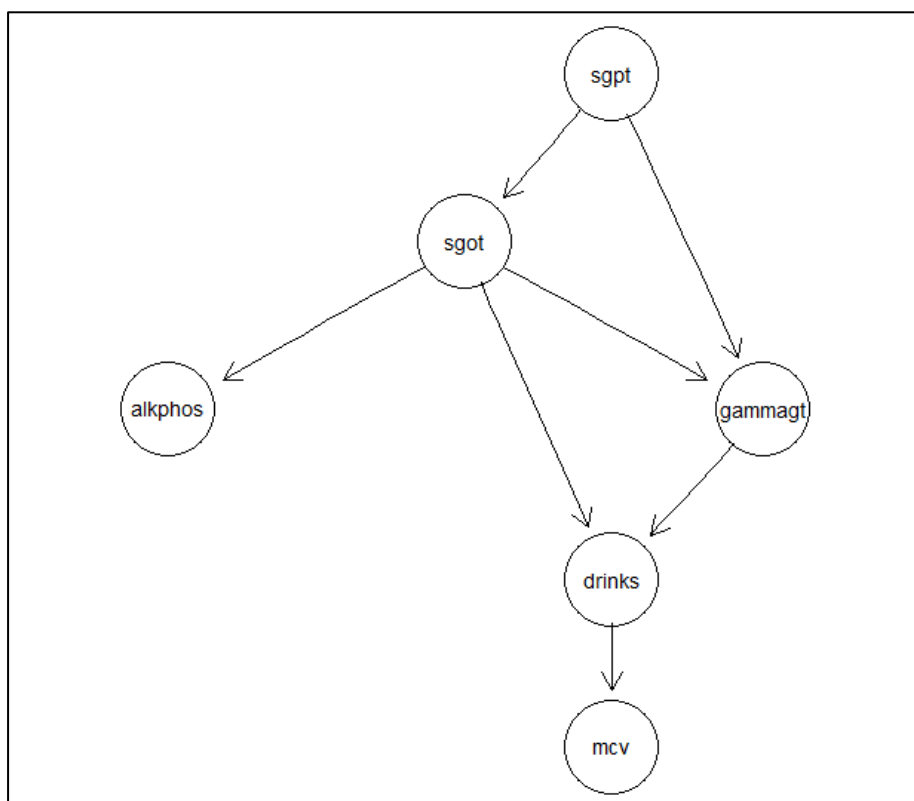
```

> test(max_dist = 1000, a_1 = T)
distance break_no
1      289        3
> test(max_dist = 1000, a_2 = T)
distance break_no
1      187        3
> test(max_dist = 1000, a_3 = T)
distance break_no
1      325        3
> test(max_dist = 1000, a_4 = T)
distance break_no
1      311        3
> test(max_dist = 1000, a_5 = T)
distance break_no
1      325        3
> test(max_dist = 1000, a_6 = T)
distance break_no
1      327        3

```

rys. 2-10 Wynik funkcji sprawdzającej optymalną liczbę przedziałów

Najlepszym sposobem okazało się podzielenie wszystkich kolumn na 3 przedziały. **Score** nowej sieci wynosił **-1154.221**, co jest znaczącym polepszeniem w porównaniu do poprzedniego -2071.689. Algorytm musiał przeprowadzić 30 iteracji testów aby nauczyć się tej sieci



rys. 2-11 Ulepszona sieć zbudowana algorytmem hc

3. Wnioskowanie

3.1. Prawdopodobieństwa warunkowe

3.1.1. Obliczone ręcznie

Prawdopodobieństwo wystąpienia *sgot* znajdującego się w przedziale [5, 30.6667] pod warunkiem *sgpt* będącego w przedziale [4, 54.3333] wynosi 90%.

$$\begin{aligned} & P(sgot = [5, 30.6667] \mid sgpt = [4, 54.3333]) \\ &= \frac{P(sgot = [5, 30.6667] \cap sgpt = [4, 54.3333])}{P(sgpt = [4, 54.3333])} = \\ &= \frac{\frac{283}{345}}{\frac{315}{345}} \approx \frac{0.82}{0.91} \approx 0.90 \end{aligned}$$

Prawdopodobieństwo wystąpienia *sgot* znajdującego się w przedziale [30.6667, 56.3333] pod warunkiem *sgpt* będącego w przedziale [104.6667, 155] wynosi 25%.

$$\begin{aligned} & P(sgot = [30.6667, 56.3333] \mid sgpt = [104.6667, 155]) = \\ &= \frac{P(sgot = [30.6667, 56.3333] \cap sgpt = [104.6667, 155])}{P(sgpt = [104.6667, 155])} = \\ &= \frac{\frac{1}{345}}{\frac{4}{345}} \approx \frac{0.003}{0.012} \approx 0.25 \end{aligned}$$

Prawdopodobieństwo wystąpienia *alkphos* znajdującego się w przedziale [61.3333, 99.6667] pod warunkiem *sgot* będącego w przedziale [5, 30.6667] i *sgpt* będącego w przedziale [104.6667, 155] wynosi 56%.

$$\begin{aligned} & P(alkphos = [61.3333, 99.6667] \mid sgot = [5, 30.6667], sgpt = [4, 54.3333]) \\ &= \frac{P(alkphos = [61.3333, 99.6667] \cap sgot = [5, 30.6667] \cap sgpt = [4, 54.3333])}{P(sgot = [5, 30.6667] \cap sgpt = [4, 54.3333])} = \\ &= \frac{\frac{158}{345}}{\frac{283}{345}} \approx \frac{0.46}{0.82} \approx 0.56 \end{aligned}$$

3.1.2. Obliczone w R

Jak widać funkcja w języku R zwróciła takie same wyniki jak przy obliczaniu prawdopodobieństw ręcznie.

```
sgot pod warunkiem sgpt = [4,54.3333]
sgot
      [5,30.6667] (30.6667,56.3333]      (56.3333,82]
      0.898412698      0.095238095      0.006349206

sgot pod warunkiem sgpt = (54.3333,104.667]
sgot
      [5,30.6667] (30.6667,56.3333]      (56.3333,82]
      0.11538462      0.84615385      0.03846154

sgot pod warunkiem sgpt = (104.667,155]
sgot
      [5,30.6667] (30.6667,56.3333]      (56.3333,82]
      0.00      0.25      0.75
```

rys. 3-1 Prawdopodobieństwo sgot pod warunkiem sgpt

```
alkphos pod warunkiem sgot = [4,54.3333]
alkphos
      [23,61.3333] (61.3333,99.6667]      (99.6667,138]
      0.36363636      0.56293706      0.07342657

alkphos pod warunkiem sgot = (54.3333,104.667]
alkphos
      [23,61.3333] (61.3333,99.6667]      (99.6667,138]
      0.2641509      0.6603774      0.0754717

alkphos pod warunkiem sgot = (104.667,155]
alkphos
      [23,61.3333] (61.3333,99.6667]      (99.6667,138]
      0.1666667      0.8333333      0.0000000
```

rys. 3-2 Prawdopodobieństwo alkphos pod warunkiem sgot

W przypadku prawdopodobieństwa wystąpienia gammagt pod warunkami sgpt i sgot wyniki prezentują się następująco:

```

gammagt pod warunkiem sgpt i sgot = [4,54.3333] [5,30.6667]
gammagt
  [5,102.333] (102.333,199.667] (199.667,297]
  0.971731449      0.021201413      0.007067138

gammagt pod warunkiem sgpt i sgot = [4,54.3333] (30.6667,56.3333]
gammagt
  [5,102.333] (102.333,199.667] (199.667,297]
  0.800000000      0.166666667      0.033333333

gammagt pod warunkiem sgpt i sgot = [4,54.3333] (56.3333,82]
gammagt
  [5,102.333] (102.333,199.667] (199.667,297]
  0.5          0.0          0.5

gammagt pod warunkiem sgpt i sgot = (54.3333,104.667] [5,30.6667]
gammagt
  [5,102.333] (102.333,199.667] (199.667,297]
  1          0          0

gammagt pod warunkiem sgpt i sgot = (54.3333,104.667] (30.6667,56.3333]
gammagt
  [5,102.333] (102.333,199.667] (199.667,297]
  0.7272727      0.1363636      0.1363636

gammagt pod warunkiem sgpt i sgot = (54.3333,104.667] (56.3333,82]
gammagt
  [5,102.333] (102.333,199.667] (199.667,297]
  0          1          0

gammagt pod warunkiem sgpt i sgot = (104.667,155] [5,30.6667]
gammagt
  [5,102.333] (102.333,199.667] (199.667,297]
  NaN          NaN          NaN

gammagt pod warunkiem sgpt i sgot = (104.667,155] (30.6667,56.3333]
gammagt
  [5,102.333] (102.333,199.667] (199.667,297]
  0          1          0

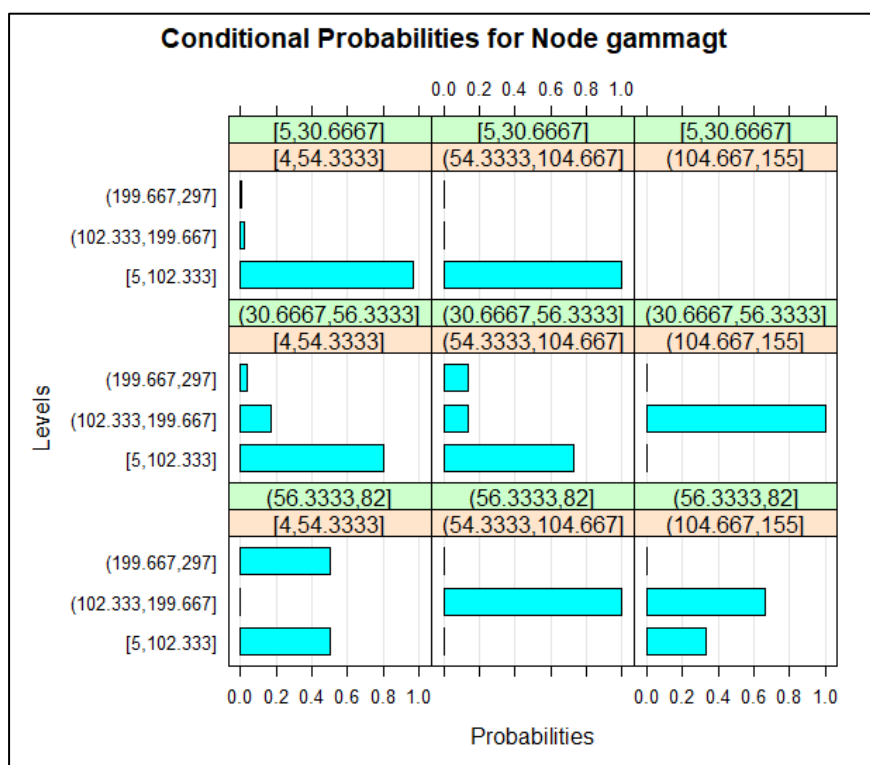
gammagt pod warunkiem sgpt i sgot = (104.667,155] (56.3333,82]
gammagt
  [5,102.333] (102.333,199.667] (199.667,297]
  0.3333333      0.6666667      0.0000000

```

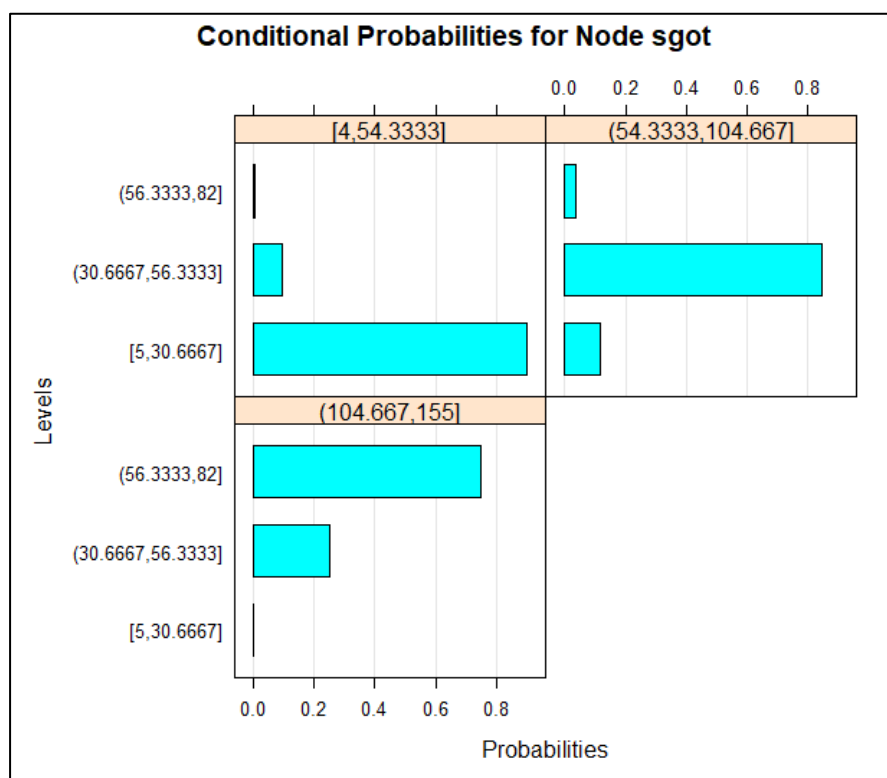
rys. 3-3 Prawdopodobieństwo gammagt pod warunkiem sgpt i sgot

3.2. Rozkłady warunkowe

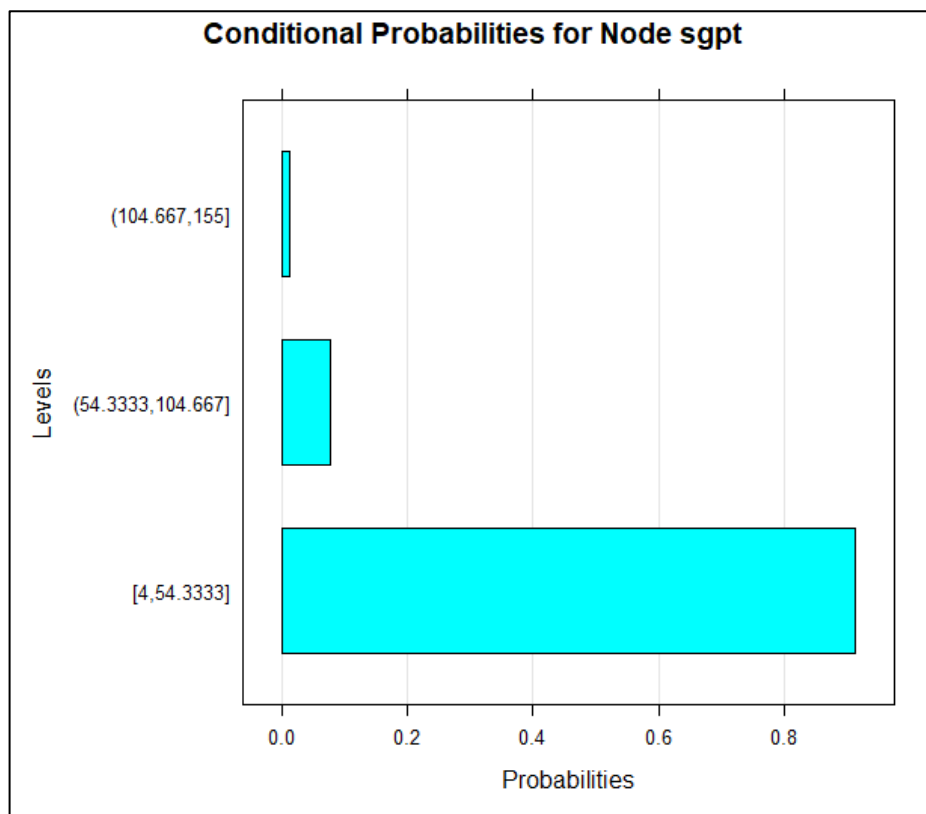
Prawdopodobieństwa można również zobrazować przy pomocy wykresów słupkowych.



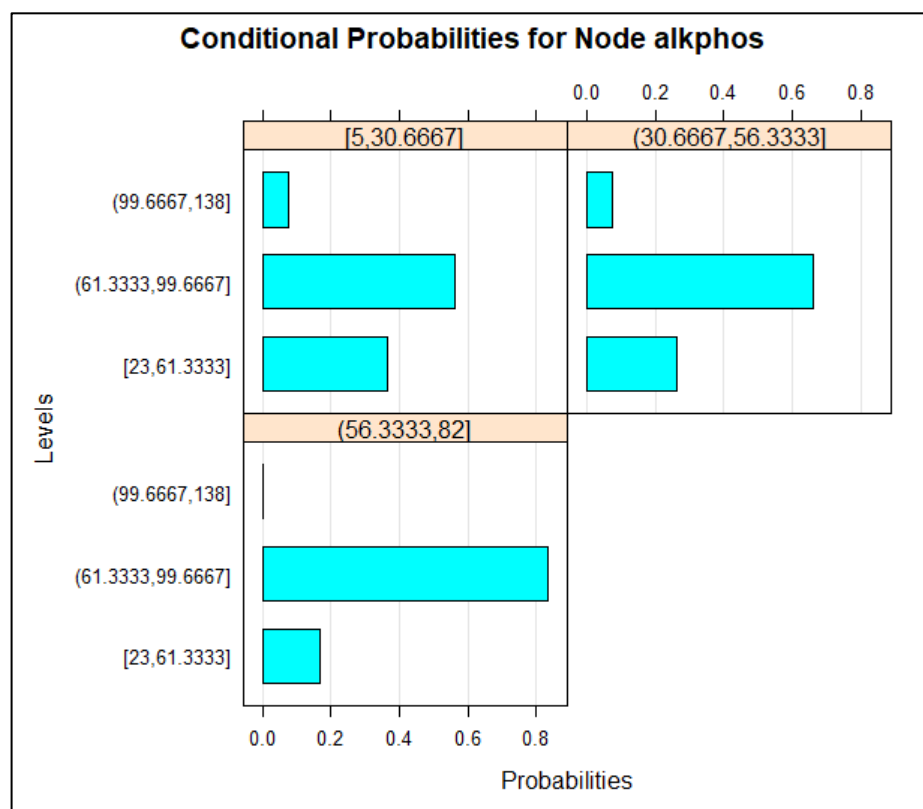
rys. 3-4 Rozkłady warunkowe dla węzła gammagt



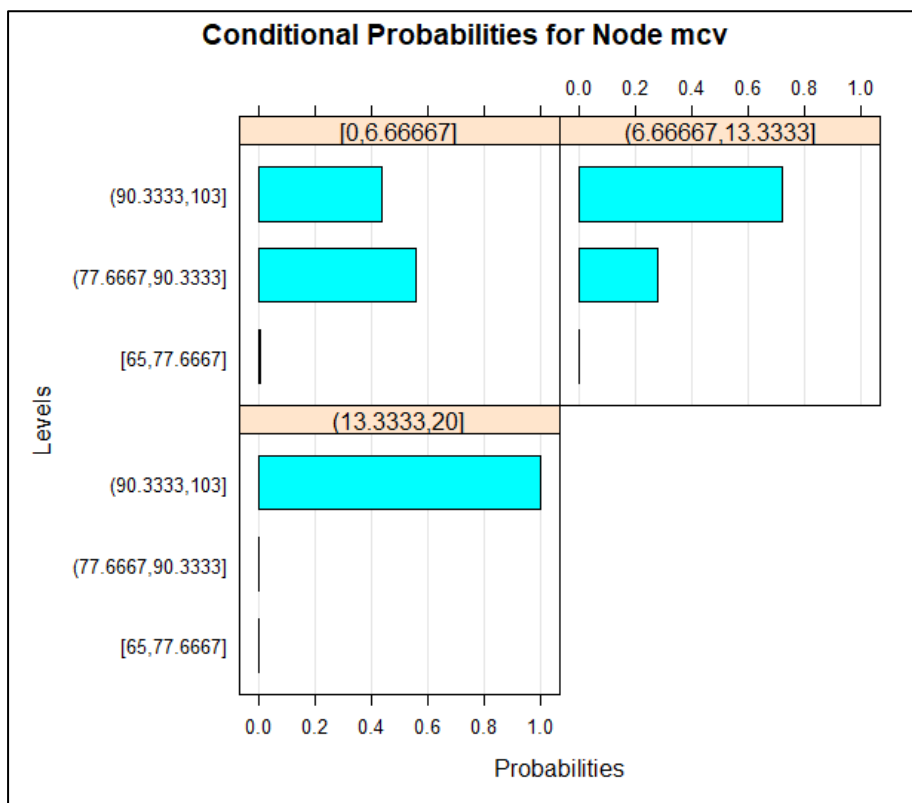
rys. 3-5 Rozkłady warunkowe dla węzła sgot



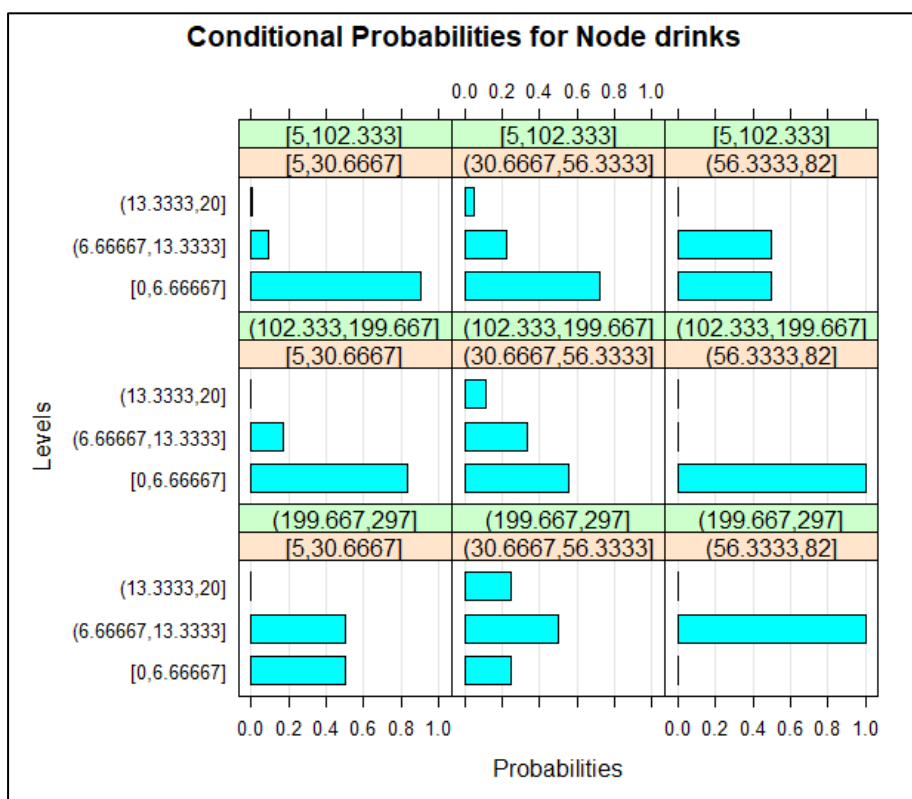
rys. 3-6 Rozkłady warunkowe dla węzła sgpt



rys. 3-7 Rozkłady warunkowe dla węzła alkphos



rys. 3-8 Rozkłady warunkowe dla węzła mcv



rys. 3-9 Rozkłady warunkowe dla węzła drinks

4. Podsumowanie

W przypadku wszystkich sprawdzanych modeli zmienna sgpt znajdowała się na samej górze i od niej zależały inne węzły. Zmienną najbardziej zależną od niej było sgot i gammagt. W większości przypadków alkphos znajdowało się poza modelem sieci lub uzależnione było od sgot. W tylko jednym przypadku gammagt wynikało z drinks – w każdym innym było odwrotnie. Podobna sytuacja dotyczy się zmiennych mcv i drinks. W co 3 modelu mcv było uzależnione od drinks – w reszcie odwrotnie.

5. Spis ilustracji

rys. 2-1 Dane	3
rys. 2-2 Sieć zbudowana algorytmem hc dla danych ciągłych	4
rys. 2-3 Sieć zbudowana algorytmem pc.stable dla danych ciągłych	4
rys. 2-4 Sieć zbudowana algorytmem gs dla danych ciągłych	5
rys. 2-5 Sieć zbudowana algorytmem iamb dla danych ciągłych	5
rys. 2-6 Sieć zbudowana algorytmem hc dla danych dyskretnych	6
rys. 2-7 Sieć zbudowana algorytmem pc.stable dla danych dyskretnych	6
rys. 2-8 Sieć zbudowana algorytmem gs dla danych dyskretnych	6
rys. 2-9 Sieć zbudowana algorytmem iamb dla danych dyskretnych	7
rys. 2-10 Wynik funkcji sprawdzającej optymalną liczbę przedziałów	8
rys. 2-11 Ulepszona sieć zbudowana algorytmem hc	8
rys. 3-7 Prawdopodobieństwo sgot pod warunkiem sgpt	10
rys. 3-8 Prawdopodobieństwo alkphos pod warunkiem sgot	10
rys. 3-9 Prawdopodobieństwo gammagt pod warunkiem sgpt i sgot	11
rys. 3-1 Rozkłady warunkowe dla węzła gammagt	12
rys. 3-2 Rozkłady warunkowe dla węzła sgot	12
rys. 3-3 Rozkłady warunkowe dla węzła sgpt	13
rys. 3-4 Rozkłady warunkowe dla węzła alkphos	13
rys. 3-5 Rozkłady warunkowe dla węzła mcv	14
rys. 3-6 Rozkłady warunkowe dla węzła drinks	14

6. Spis tabel

tab. 1 Wartości p dla zmiennych	4
tab. 2 Score dla zbudowanych sieci	7

7. Kod źródłowy

https://github.com/gabriellchacz/bayesian_network_liver