PRACE NAUKOWE UNIWERSYTETU EKONOMICZNEGO WE WROCŁAWIU nr 108
Współczesne tendencje rozwojowe badań operacyjnych 2010

Grzegorz Tarczyński
Uniwersytet Ekonomiczny we Wrocławiu

ROZWIĄZYWANIE ZAGADNIEŃ KLASYFIKACYJNYCH Z WYKORZYSTANIEM SIECI BAYESOWSKICH

Streszczenie: Sieci bayesowskie są narzędziem o szerokim zastosowaniu. Jednym z zagadnień, do których się je wykorzystuje, są problemy klasyfikacji danych. W artykule omówiono wybrane metody automatycznego generowania sieci bayesowskich. Przedstawiono również wyniki symulacji na różnych zbiorach danych. Porównano jakość otrzymanych klasyfikatorów i skomentowano problemy związane z zastosowaniem omawianych metod. Do obliczeń wykorzystano pakiet GeNie.

Słowa kluczowe: sieci bayesowskie, metody klasyfikacji, symulacje

1. Wstęp

Sieci bayesowskie są narzędziem o szerokim zastosowaniu. Jednym z zagadnień, do których się je wykorzystuje, jest problem wzorcowej klasyfikacji danych.

Celem artykułu jest przegląd popularnych metod generowania struktury (i parametrów) sieci bayesowskich oraz wykorzystanie ich do rozwiązywania zagadnień klasyfikacji danych i ocena jakości klasyfikacji. Podjęto również próbę odpowiedzi na pytanie, które algorytmy wymagają wspomagania ekonometrycznymi metodami doboru zmiennych do modelu. Do obliczeń wykorzystano pakiet GeNie.

2. Zasady działania sieci bayesowskich

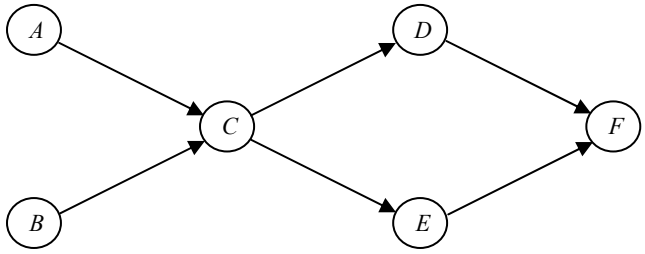
Sieć bayesowska to zorientowany acykliczny graf (*directed acyclic graph* – DAG)[1], w którym węzły reprezentują zmienne. Ze względów praktycznych ustala się, że zmienne przyjmują wartości dyskretne (często binarne). Występowanie łuku łączącego dwa węzły wiąże się z istnieniem zależności przyczynowo-skutkowej między zmiennymi reprezentowanymi przez te węzły. Brak łuku oznacza natomiast

[1] Niektórzy autorzy w teoretycznych rozważaniach przewidują również sieci bayesowskie reprezentowane przez grafy o łukach niezorientowanych.

brak takiej zależności. Innymi słowy, węzły odpowiadające zmiennym niezależnym nie są połączone łukami.

Z każdym węzłem powiązany jest rozkład prawdopodobieństwa. Węzłom początkowym (może być ich więcej niż jeden) odpowiadają rozkłady brzegowe, pozostałym węzłom zaś – rozkłady warunkowe.

Przykład sieci bayesowskiej przedstawiony jest na rys. 1. Z węzłami **A** i **B** powiązane są rozkłady brzegowe: $P(A)$ i $P(B)$, a z pozostałymi węzłami rozkłady warunkowe: $P(C|A,B)$, $P(D|C)$, $P(E|C)$, $P(F|D,E)$.



Rys. 1. Przykład sieci bayesowskiej

Źródło: opracowanie własne.

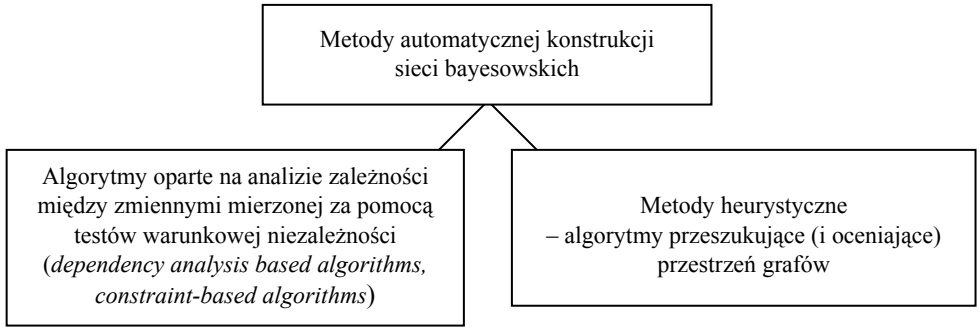
Do obliczeń pozostałych prawdopodobieństw wykorzystuje się wzór Bayesa:

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)} \tag{1}$$

Jeżeli znane są relacje występujące pomiędzy zmiennymi, sieci bayesowskie konstruuje się „ręcznie". Kiedy jednak nie dysponujemy taką wiedzą, możemy skorzystać z jednej z metod automatycznego generowania struktury sieci bayesowskiej. Wymaga to jednak odpowiedniego przygotowania danych: zamiany wartości ciągłych na dyskretne i, niekiedy, ograniczenia liczby zmiennych. Automatycznie wygenerowana sieć bayesowska może dostarczyć wiele informacji i pomóc w zrozumieniu relacji zachodzących pomiędzy zmiennymi.

3. Metody automatycznego generowania sieci bayesowskich

Istnieje wiele metod automatycznego generowania struktury sieci bayesowskiej. Podstawowa klasyfikacja obejmuje podział na metody oparte na testach warunkowej niezależności (*dependency analysis based algorithms, constraint-based algorithms*) i metody heurystyczne (rys. 2). W dalszej części omówione zostaną algorytmy **PC** i **EGS** z grupy metod opartych na testach warunkowej niezależności, heurystyczny algorytm **GTT** oraz metoda naiwna.



Rys. 2. Klasyfikacja metod automatycznej konstrukcji struktury sieci bayesowskich

Źródło: opracowanie własne.

Popularną metodą automatycznej konstrukcji struktury sieci bayesowskiej jest algorytm PC, oparty na testach warunkowej niezależności. Na etapie wstępnym algorytmu konstruowany jest spójny graf niezorientowany. Następnie przeprowadza się testy warunkowej niezależności, których efektem jest usunięcie łuków łączących węzły reprezentujące zmienne niezależne. Ostatnim etapem jest ustalenie zwrotów łuków.

Metoda PC jest bardzo wrażliwa na parametry: uporządkowanie węzłów (decyduje w jakiej kolejności przeprowadzać testy warunkowej niezależności) i poziom istotności $\alpha \in (0;1)$. Dla różnych wartości parametrów algorytm może wygenerować inne grafy. W prezentowanych w dalszej części wynikach eksperymentów przyjęto $\alpha = 0{,}05$.

Algorytm EGS (*essential graph search*) jest oparty na metodzie PC. Algorytm EGS wymaga zdefiniowania rozkładu $P(\alpha)$, następnie tworzone są grafy zgodnie z metodą PC dla różnych wartości α. Spośród wygenerowanych w ten sposób grafów wybiera się graf o maksymalnej wartości funkcji oceny.

Więcej informacji o algorytmach PC i EGS można znaleźć w pracy [Dash, Druzdzel 1999].

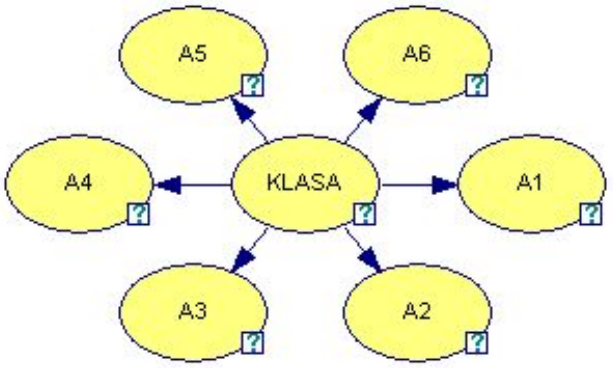
Algorytm GTT (*greedy thick thinning*) należy do grupy algorytmów heurystycznych. W metodzie tej generuje się zorientowany graf początkowy, a następnie sporządza listę możliwych modyfikacji grafu (dodanie łuku, usunięcie łuku, zmiana zwrotu). Jedynym ograniczeniem w zakresie modyfikacji grafu jest warunek niewystępowania cykli. W kolejnym etapie wybiera się taką zmianę, która zapewni największy wzrost wartości funkcji oceny. Modyfikacji dokonuje się do momentu, gdy jej przeprowadzenie nie przyniesie polepszenia funkcji oceny.

Wadą metod heurystycznych jest długi czas obliczeń i możliwość „utknięcia" w maksimach lokalnych funkcji oceny.

Metoda GTT szerzej opisana jest w pracy [Heckerman 1995].

Problemy klasyfikacyjne mogą być rozwiązywane również za pomocą sieci bayesowskich wygenerowanych przez metodę naiwną (*naive*). Metoda naiwna za-

kłada występowanie połączeń pomiędzy wszystkimi zmiennymi objaśniającymi a zmienną objaśnianą i brak połączeń zmiennych objaśniających między sobą (rys. 3). Jej zastosowanie wymaga więc przeprowadzenia wstępnej analizy ekonometrycznej doboru zmiennych do modelu. Łuki grafu wyrażają istnienie zależności przyczynowo-skutkowej pomiędzy parą zmiennych reprezentowanych przez węzły. Zmienne objaśniające powinny być więc silnie skorelowane ze zmienną objaśnianą i słabo między sobą (brak połączeń między węzłami reprezentującymi zmienne objaśniające).



Rys. 3. Przykład sieci wygenerowanej według podejścia naiwnego

Źródło: wydruk programu GeNie.

Zwroty łuków są tutaj dowolne, ale z praktycznych względów rysuje się łuki prowadzące od zmiennej objaśnianej do zmiennych objaśniających. Dzięki temu konieczna do wyznaczenia liczba prawdopodobieństw warunkowych jest znacznie mniejsza.

Dla grafu przedstawionego na rys. 3 niezbędne jest wyznaczenie rozkładu brzegowego zmiennej KLASA i rozkładów warunkowych zmiennych A1-A6.

Prawdopodobieństwa warunkowe zmiennej objaśnianej wyznacza się, korzystając ze wzoru Bayesa. Dla omawianego przykładu:

$$P(K \mid A1 \wedge A2 \wedge A3 \wedge A4 \wedge A5 \wedge A6) = \\ = \frac{P(A1 \wedge A2 \wedge A3 \wedge A4 \wedge A5 \wedge A6 \mid K)P(K)}{P(A1 \wedge A2 \wedge A3 \wedge A4 \wedge A5 \wedge A6)} \tag{2}$$

Stosując proste podstawienia, obliczamy wartość licznika i mianownika:

$$P(A1 \wedge A2 \wedge A3 \wedge A4 \wedge A5 \wedge A6 \mid K) = \\ = P(A1 \mid K)P(A2 \mid K)P(A3 \mid K)P(A4 \mid K)P(A5 \mid K)P(A6 \mid K),$$

$$P(A1 \wedge A2 \wedge A3 \wedge A4 \wedge A5 \wedge A6) =$$
$$= P(A1 \mid K)P(A2 \mid K)P(A3 \mid K)P(A4 \mid K)P(A5 \mid K)P(A6 \mid K)P(K) +$$
$$+P(A1 \mid \overline{K})P(A2 \mid \overline{K})P(A3 \mid \overline{K})P(A4 \mid \overline{K})P(A5 \mid \overline{K})P(A6 \mid \overline{K})P(\overline{K}).$$

Godny uwagi jest również algorytm TPDA (*three-phase dependency analysis*), zaproponowany przez J. Chenga i in. [Cheng, Bell, Liu 1997; Cheng *et al.* 2000]. Metoda oparta na teorii informacji obejmuje trzy fazy: szkicowanie (*drafting* – ustalanie wstępnej struktury grafu), „pogrubianie" (*thickening* – dodawanie nowych połączeń) i „odchudzanie" grafu (*thinning* – usuwanie zbędnych łuków). J. Cheng, stosując powyższą metodę, wygrał w 2001 r. KDD Cup (konkurs poświęcony pozyskiwaniu wiedzy z baz danych) [*KDD Cup*... 2002]. Metoda TPDA nie będzie omawiana w dalszej części tego artykułu.

4. Analiza jakości klasyfikacji dokonanej przez sieci bayesowskie

Analizie poddanych zostało 12 zbiorów danych, które są ogólnie dostępne na stronie http://www.datalab.uci.edu/. Są to dane z różnych dziedzin: ekonomii, psychologii, medycyny, techniki i politologii; trzy zbiory danych zostały sztucznie wygenerowane.

Zbiory danych dotyczą:

- ***credit***: decyzja o przyznaniu karty kredytowej, zbiór zawiera 654 obiekty opisane za pomocą 15 zmiennych (6 z nich przyjmuje wartości ciągłe, a 9 wartości dyskretne); obiekty podzielone są na dwie klasy (wnioski rozpatrzone pozytywnie i negatywnie);
- ***balance***: eksperyment psychologiczny, 625 obiektów, 4 zmienne (wartości dyskretne: {1,2,3,4,5}), 3 klasy;
- ***breast***: medycyna (zachorowalność na raka), 699 obiektów, 9 zmiennych (wartości dyskretne: 1-10), 2 klasy;
- ***diabetes***: medycyna, 768 obiektów, 8 zmiennych (wartości ciągłe) opisujących objawy cukrzycy, podział na 2 klasy (chorzy na cukrzycę/ zdrowi);
- ***heart***: medycyna (objawy choroby serca), 270 obiektów, 13 zmiennych (wartości ciągłe), dwie klasy;
- ***vote***: politologia (głosowania kongresmenów amerykańskich), 435 obiektów, 16 zmiennych przyjmujących wartości dyskretne: {był za, był przeciw, wstrzymał się od głosu}, podział na dwie klasy (demokraci i republikanie);
- ***sonar***: rozpoznawanie obiektów metalowych (min), 208 obiektów, 60 zmiennych przyjmujących wartości ciągłe z przedziału <0, 1>, podział na dwie klasy (obiekt jest skałą lub miną (metal));
- ***glass2***: analiza chemiczna szkła (szyb), 163 obiekty, 9 zmiennych przyjmujących wartości ciągłe, 2 klasy;
- ***cars***: motoryzacja, 392 obiekty, 8 zmiennych, z których 9 przyjmuje wartości ciągłe, a jedna: wartości dyskretne), 3 klasy (samochód europejski, amerykański, japoński).

Zbiory sztucznie wygenerowane: **monk1**, **monk2**, **monk3** opisane zostały przez 6 zmiennych które przyjmowały wartości dyskretne:
- *atr_1* : {1, 2, 3},
- *atr_2* : {1, 2, 3},
- *atr_3* : {1, 2},
- *atr_4* : {1, 2, 3},
- *atr_5* : {1, 2, 3, 4},
- *atr_6* : {1, 2}.

Obiekty ze zbiorów **monk1**, **monk2** i **monk3** podzielono na dwie klasy: „*yes*" i „*no*". Przydzielenie obiektu do klasy „*yes*" następowało zgodnie z następującymi regułami:
- **monk1**: (*atr_1* = *atr_2*) lub (*atr_5* = 1),
- **monk2**: (*atr_n* = 1) dla DOKŁADNIE DWÓCH *n*=1, 2,..., 6,
- **monk3**: (*atr_5* = 3 i *atr_4* = 1) lub (*atr_5* ≠ 4 i *atr_2* ≠ 3).

Dodatkowo do zbioru **monk3** wprowadzono szum: kilka obiektów ze zbioru uczącego zostało błędnie zaklasyfikowanych.

Dla zbioru **monk2** żadna z omówionych w poprzednim punkcie metod nie wygenerowała sieci bayesowskiej, dlatego nie będzie on podlegał dalszej analizie.

Wartości zmiennych ciągłych w każdym z omówionych zbiorów danych zostały przekształcone w jedną z trzech wartości dyskretnych (DUŻE, ŚREDNIE, MAŁE). Podobnie zrobiono dla przykładu ***breast***, gdzie wartości dyskretne 1, 2, 3 zastąpiono wartością MAŁE; 4, 5, 6, 7 – ŚREDNIE; 8, 9, 10 – DUŻE. Zamiana wartości ciągłych na dyskretne jest niezbędnym elementem, który należy przeprowadzić przed zastosowaniem metod generujących sieci bayesowskie. W ramach zbioru ***breast*** redukcja liczby kategorii pozwoliła na przyspieszenie działania tychże metod. Zabiegi te wiążą się oczywiście z utratą części informacji, przeprowadzenie ich jest jednak konieczne.

Dla przedstawionych zbiorów danych wygenerowano sieci bayesowskie za pomocą metod: naiwnej, PC, EGS i GTT. Otrzymane sieci bayesowskie wykorzystano do rozwiązania zagadnień klasyfikacji wzorcowej. Wyniki eksperymentów zawierają tab. 1-3. Liczby w kolumnach oznaczają, jaki procent danych został prawidłowo zaklasyfikowany. Pogrubioną czcionką wyróżniono najlepsze wyniki.

Metoda naiwna na danych uczących przyniosła najlepsze rezultaty dla zbiorów ***credit*** i **monk3** (podobnie jak pozostałe metody). Dla danych testowych i łącznych (uczące + testowe) z kolei są to zbiory ***balance***, ***glass2*** i **monk3**. Zaletą metody naiwnej jest jej prostota, wadą: konieczność wstępnej analizy danych (doboru zmiennych).

Metodą PC najlepsze wyniki uzyskano na zbiorach uczących, dużo słabsze zaś na testowych. Na przykład dla zbiorów uczących ***balance*** lub ***sonar*** uzyskano bardzo dobre (najlepsze) wyniki, dla zbiorów zaś testowych bardzo słabe (najgorsze). Nastąpiło więc „przeuczenie" sieci.

Tabela 1. Jakość wygenerowanych sieci bayesowskich (zbiór uczący)

Metoda generowania sieci bayesowskiej	Nazwa zbioru danych					
	credit	*balance*	*breast*	*diabetes*	*heart*	*vote*
naive	**86,13%**	89,66%	95,49%	74,61%	85,56%	94,33%
PC	85,16%	**100%**	**98,50%**	74,80%	85,00%	**95,33%**
EGS	85,16%	-	97,85%	71,48%	**91,11%**	**95,33%**
GTT	85,74%	89,66%	97,21%	**75,00%**	85,56%	**95,33%**
	sonar	*glass2*	*cars*	monk1	monk3	
naive	75,36%	74,07%	89,66%	79,84%	**93,44%**	
PC	**76,09%**	74,07%	88,51%	76,61%	**93,44%**	
EGS	64,69%	**75,00%**	91,95%	73,39%	**93,44%**	
GTT	70,29%	72,22%	**99,62%**	**100%**	**93,44%**	

Źródło: opracowanie własne.

Tabela 2. Jakość wygenerowanych sieci bayesowskich (zbiór testowy)

Metoda generowania sieci bayesowskiej	Nazwa zbioru danych					
	credit	*balance*	*breast*	*diabetes*	*heart*	*vote*
naive	84,44%	**88,04%**	95,71%	71,88%	85,56%	95,56%
PC	89,63%	27,75%	92,27%	72,27%	84,44%	95,56%
EGS	**92,59%**	-	93,56%	**77,34%**	71,11%	**96,30%**
GTT	84,44%	**88,04%**	**96,57%**	75,00%	**86,67%**	**96,30%**
	sonar	*glass2*	*cars*	monk1	monk3	
naive	54,29%	**74,55%**	87,02%	71,30%	**97,22%**	
PC	50,00%	72,73%	77,10%	69,44%	**97,22%**	
EGS	52,86%	70,91%	62,60%	75,00%	**97,22%**	
GTT	**67,14%**	72,73%	**97,71%**	**100%**	**97,22%**	

Źródło: opracowanie własne.

Tabela 3. Jakość wygenerowanych sieci bayesowskich (zbiór uczący i testowy)

Metoda generowania sieci bayesowskiej	Nazwa zbioru danych					
	credit	*balance*	*breast*	*diabetes*	*heart*	*vote*
naive	85,78%	**89,12%**	95,57%	73,70%	85,56%	94,71%
PC	86,09%	75,84%	96,42%	73,96%	84,81%	95,40%
EGS	**86,70%**	-	96,42%	73,44%	84,44%	**95,63%**
GTT	85,47%	**89,12%**	**97,00%**	**75,00%**	**85,93%**	**95,63%**
	sonar	*glass2*	*cars*	monk1	monk3	
naive	68,27%	**74,23%**	88,78%	73,20%	**96,39%**	
PC	67,31%	73,62%	84,69%	71,04%	**96,39%**	
EGS	60,58%	73,62%	82,14%	74,64%	**96,39%**	
GTT	**69,23%**	72,39%	**98,98%**	**100%**	**96,39%**	

Źródło: opracowanie własne.

Metoda EGS przyniosła wyniki porównywalne z metodą PC, co jest dość zaskakujące, gdyż bazuje ona na metodzie PC uruchamianej dla różnych wartości parametrów.

Najlepsze wyniki – aż dla 9 zbiorów danych – uzyskano metodą GTT. Jako jedyna w 100% prawidłowo odwzorowała regułę klasyfikacyjną zbioru **monk1**. Niestety, metoda GTT jest dość czasochłonna i dla zbiorów danych większych niż analizowane w niniejszej pracy jej zastosowanie może być mocno utrudnione. Już dla zbioru ***sonar***, zawierającego 60 zmiennych, wymusiła przeprowadzenie wstępnej eliminacji zmiennych, zgodnie z zasadami doboru zmiennych do modelu ekonometrycznego.

Dla zbioru **monk3** metody PC, EGS i GTT wygenerowały takie same struktury sieci bayesowskiej – stąd identyczne wyniki. Struktura sieci naiwnej była nieco inna, ponieważ dla tego przykładu nie przeprowadzono wstępnej analizy ekonometrycznej i nie usunięto zmiennych słabo skorelowanych ze zmienną objaśnianą. Metody PC, EGS i GTT wyeliminowały te zmienne, usuwając odpowiednie łuki grafu. Wyniki dla wszystkich czterech metod są jednak takie same, co więcej: wszystkie metody źle zaklasyfikowały te same obiekty. Wśród obiektów źle przydzielonych do grup są wszystkie zawierające szum. Można więc postawić tezę, że sieci bayesowskie prawidłowo odwzorowały regułę decyzyjną przydziału obiektów do klas. Nawet te obiekty, które zawierały błąd w danych, zostały prawidłowo zaklasyfikowane. Nie występuje więc tutaj zjawisko „przeuczenia", doskonale znane z zastosowań sieci neuronowych.

5. Wnioski

Przeprowadzone eksperymenty stanowią przesłankę do wyprowadzenia następujących wniosków:

- najlepsze wyniki uzyskano dla metody GTT, która jest jednak bardzo czasochłonna,
- szybka metoda PC przyniosła dobre wyniki dla zbiorów uczących,
- metoda naiwna daje w większości wyniki tylko nieznacznie gorsze od metody GTT,
- zastosowanie podejścia naiwnego wymaga wsparcia statystycznymi metodami doboru zmiennych do modelu.

Literatura

Cheng J., Bell D., Liu W., *Learning Bayesian Networks from Data: An efficient approach based on information theory,* Proceeding of the sixth ACM International Conference on Information and Knowledge Management, 1997.

Cheng J. *et al*., *Learning Bayesian networks from data: An information-theory based approach*, "Artificial Intelligence" 2002 vol. 137 no 1, s. 43-90.

Dash D., Druzdzel M., *A Hybrid Anytime Algorithm for the Construction of Casual Models from Sparse Data*, Morgan Kaufmann Publishers, Inc., San Francisco (CA) 1999, s. 142-149.

Heckerman D., *A Tutorial on Learning with Bayesian Networks*, Technical Report, Redmond, (WA) 1995.

KDD Cup 2001 Report, SIGKDD Explorations, 2002.

Pearl J., *Probabilistic Reasoning in Intelligent Systems*: *Networks of plausible inference*, Morgan Kaufmann Publishers, Inc., San Mateo (CA) 1988.

SOLVING CLUSTERING PROBLEMS WITH BAYESIAN NETS

Summary: Bayesian nets have very wide area of applications – they can be used for solving clustering problems too. The purpose of this paper is to present methods for learning both the parameters and structure of a Bayesian network. There are described such methods as naive, PC, EGS, GTT and TPDA. The algorithms were used to solve clustering problems based on real and artificial data. The paper contains the comparison of results and suggestion when one needs to use a particular method.