УДК 62-50

М.З. ЗГУРОВСКИЙ, П.И. БИДЮК, А.Н. ТЕРЕНТЬЕВ

МЕТОДЫ ПОСТРОЕНИЯ БАЙЕСОВСКИХ СЕТЕЙ НА ОСНОВЕ ОЦЕНОЧНЫХ ФУНКЦИЙ

Ключевые слова: *байесовская сеть, оценочные методы поиска, вычислительные характеристики, минимальная длина описания.*

ВВЕДЕНИЕ

Во всех странах мира накапливаются большие объемы информации, требующие надлежащей обработки и принятия решений на основе результатов этой обработки. Методы интеллектуального анализа данных (ИАД), к которым принадлежат и байесовские сети (БС), предоставляют возможность автоматического поиска закономерностей, характерных для многомерных данных. В основе большинства инструментов интеллектуального анализа данных лежат две технологии: машинное обучение и визуализация информации. Байесовские сети объединяют в себе эти две технологии.

Широкое применение БС нашли в медицинской и технической диагностике в условиях неполной и неточной информации, в системах классификации данных различной природы, системах автоматического распознавания речевых сигналов, маркетинге, бизнесе и во многих других сферах деятельности. В общем случае БС дает возможность воспроизвести причинно-следственные связи между событиями и определить вероятность наступления той или иной ситуации при получении новой информации относительно изменения состояния любого узла (переменной) сети. Степень целесообразности применения данного метода моделирования и формирования вероятностного вывода зависит от умения корректно осуществить постановку задачи, выбрать переменные процесса, которые в достаточной мере характеризуют его динамику или статику, найти необходимые данные и использовать их для обучения сети, а также корректно сформулировать результат-вывод с помощью построенной сети.

В англоязычной литературе термин «построение БС» означает реализацию следующих процессов: 1) поиск оптимальной структуры БС, т.е. направленного ациклического графа, наиболее адекватно соответствующего обучающим данным или исследуемому процессу; 2) вычисление значений таблиц условных вероятностей БС для соответствующих узлов этого графа.

Цель настоящей статьи — анализ существующих методов решения задачи выбора оптимальной структуры байесовской сети, описание заложенных в них принципов и практического использования.

ФОРМАЛЬНАЯ МАТЕМАТИЧЕСКАЯ ЗАПИСЬ БС

Байесовская сеть — это графическая модель процесса или объекта произвольной природы, представленная парой $< G, B >$. Первой компонентой G является направленный ациклический граф, соответствующий случайным переменным объекта или процесса. Он записывается как набор условий независимости: каждая переменная не зависит от ее родителей в G. Вторая компонента пары B представляет множество параметров, определяющих сеть. Эта компонента содержит параметры $\Theta_{x^{(i)}\,|pa\,(X^i)} = P\,(X^{(i)}|\,pa\,(X^{(i)}))$ для каждого возможного

значения $x^{(i)} \in X^{(i)}$ и $pa(X^{(i)}) \in Pa(X^{(i)})$, где $Pa(X^{(i)})$ означает набор родителей переменной $X^{(i)} \in G$. Каждая переменная $X^{(i)} \in G$ представляется в виде вершины. Если рассматривают более одного графа, то для определения родителей переменной $X^{(i)}$ в графе G используют обозначение $Pa^G(X^{(i)})$. Полная совместная вероятность БС вычисляется по формуле $P_B(X^{(1)}, \ldots, X^{(N)}) = \prod_{i=1}^{N} P_B(X^{(i)} \mid Pa(X^{(i)}))$.

Множество обучающих данных записывается следующим образом: $D = \{d_1, \ldots, d_n\}$, $d_i = \{x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(N)}\}$. Здесь нижний индекс — это номер наблюдения, а верхний — номер переменной; n — количество наблюдений, каждое из которых состоит из N $(N \geq 2)$ переменных $X^{(1)}, X^{(2)}, \ldots, X^{(N)}$. Каждая j-я переменная $(j = 1, \ldots, N)$ имеет $A^{(j)} = \{0, 1, \ldots, \alpha^{(j)} - 1\}$ $(\alpha^{(j)} \geq 2)$ состояний, а каждая структура $g \in G$ БС представляется N множествами предков $(\Pi^{(1)}, \ldots, \Pi^{(N)})$. Иными словами, для каждой вершины $j = 1, \ldots, N$ множество $\Pi^{(j)}$ — набор родительских вершин таких, что $\Pi^{(j)} \subseteq \{X^{(1)}, \ldots, X^{(N)}\} \setminus \{X^j\}$ (вершина не может быть предком самой себе, т. е. петли в графе отсутствуют).

ГРАФИЧЕСКОЕ ОТОБРАЖЕНИЕ СТРУКТУРЫ БС

Дерево — такая структура БС, в которой любая вершина может иметь не более одной вершины-предка (рис. 1).

Поли-дерево — такая структура БС, в которой любая вершина может иметь более одной вершины-предка, но при этом между любыми двумя вершинами должно быть не более одного связывающего их пути (рис. 2).

Сети — сетевая структура, в которой любая вершина может иметь более чем одну вершину-предка; при этом между любыми двумя вершинами может быть более одного связывающего их пути (рис. 3).

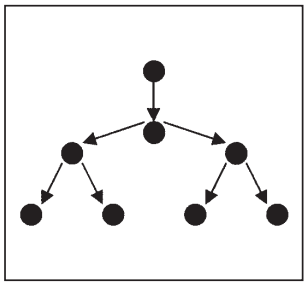

Рис. 1. Структура БС в виде дерева

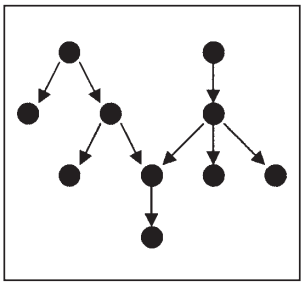

Рис. 2. Структура БС в виде поли-дерева

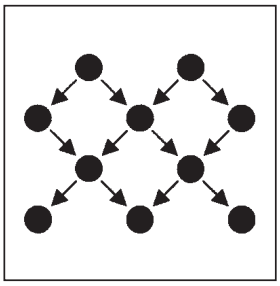

Рис. 3. Структура БС в виде сети

Деревья и поли-деревья называют также односвязными сетями, а сети — многосвязными сетями.

АЛГОРИТМЫ ПОСТРОЕНИЯ СТРУКТУРЫ СЕТЕЙ БАЙЕСА

Чу и Лиу (Chow and Liu) в 1968 г. предложили алгоритм для построения БС в виде дерева [1], основанный на использовании значений обоюдной информации между вершинами. В качестве решения метод выдает структуру БС со значением совместного распределения вероятности, наиболее соответствующе-

го обучающим данным. Построение структуры БС осуществляется за $O(N^2)$ шагов, где N — число вершин сети, однако этот алгоритм не применим для многосвязных БС.

В 1988 г. Рибан и Перл (Rebane and Pearl) предложили усовершенствованный измененный алгоритм Чу и Лиу для построения БС в виде поли-дерева [2]. Купер и Гершкович (Cooper and Herskovits) в 1990 г. разработали алгоритм Кутато (Kutato) [3]. На этапе инициализации алгоритма с учетом, что все вершины БС независимы, вычисляется энтропия этой сети. Затем происходит добавление дуг между вершинами в сети таким образом, чтобы минимизировать энтропию БС. Для работы алгоритма требуется наличие упорядоченого множества вершин.

Алгоритм SGS [4], предложенный в 1991 г., при построении структуры обходится без упорядоченого множества вершин, но вместо этого ему приходится выполнять экспоненциальное количество тестов на условную независимость между вершинами. Купер и Гершкович в 1992 г. предложили широко известный алгоритм К2 [5], который выполняет поиск структуры с максимальным значением функции Купера–Гершковича (КГ). Для работы алгоритма требуется наличие упорядоченного множества вершин.

Алгоритм Лема–Бахуса (Lam–Bacchus) [6], предложенный в 1996 г., выполняет эвристическое построение структуры сети, используя обоюдную информацию между вершинами, а в качестве оценочной функции применяется описание минимальной длины (ОМД).

Алгоритм Бенедикта (Benedict) [7], предложенный в 1996 г., выполняет эвристический поиск на основе упорядоченого множества вершин, анализируя условные независимости в структуре сети на основе d-разделения, а в качестве оценочной функции используется энтропия.

Алгоритм СВ [8] предложен в 1995 г. Он использует тест на условную независимость между вершинами сети для построения упорядоченного множества вершин. При построении структуры сети используется функция КГ.

Алгоритм Фридмана–Голдшмидта (Friedman–Goldszmidt) [9] предложен в 1996 г. При построении сети применяется анализ ее локальных подструктур, а в качестве оценочных функций используются ОМД и оценка Байеса.

В алгоритме WKD [10], предложенном в 1996 г., в качестве оценочной функции при построении сети используется функция сообщения минимальной длины, которая имеет сходство с ОМД.

Алгоритм Сузуки (Suzuki) [11], предложенный в 1999 г., основан на методе ветвей и границ для задания последовательности построения структуры сети, а в качестве оценочной функции используется ОМД.

ФУНКЦИЯ КУПЕР–ГЕРШКОВИЧА

В работе [5] Купер и Гершкович предложили метод КГ для обучения БС, который основан на поиске структуры БС с максимальным значением функции КГ. Функция КГ структуры $g \in G$ при заданной последовательности из n наблюдений $x^n = d_1 d_2 \ldots d_n$ записывается уравнением

$$P(g, x^n) = P(g) \cdot \prod_{j \in J} \left(\prod_{s \in S(j,g)} \frac{(\alpha^{(j)} - 1)! \cdot \prod\limits_{q \in A^{(j)}} (n[q, s, j, g]!)}{(n[s, j, g] + \alpha^{(j)} - 1)!} \right).$$

Здесь $P(g)$ — априорная вероятность структуры $g \in G$, которую часто опуска-

ют при вычислениях; запись $j \in J = \{1, \ldots, N\}$ означает перебор всех вершин структуры сети g, а $s \in S(j, g)$ — перебор множества всех наборов значений, принимаемых предками j-й вершины;

$$n(s, j, g) = \sum_{i=1}^{n} I(\pi_i^{(j)} = s);\ n[q, s, j, g] = \sum_{i=1}^{n} I(x_i = q, \pi_i^{(j)} = s),$$

где $\pi^{(j)} = \Pi^{(j)}$, функция $I(E) = 1$, когда предикат $E = \text{true}$, в противном случае $I(E) = 0$.

Алгоритм обучения БС с использованием функции КГ основан на цикличесском переборе всех возможных ациклических сетевых структур. В g^* сохраняется оптимальная сетевая структура. Оптимальной структурой будет та, которая имеет наибольшее значение функции $P(g, x^n)$:

1) $g^* \leftarrow g_0 (\in G)$;

2) $\forall\, g \in G - \{g_0\}$: если $P(g, x^n) > P(g^*, x^n)$, то $g^* \leftarrow g$;

3) на выходе имеем g^* в качестве решения.

Однако при использовании функции КГ необходимо учитывать вычислительные ограничения моделирующих систем, связанные с конечной длиной разрядной сетки. Приведем тривиальный пример, когда в структуре имеются две вершины: $X^{(1)}$ и $X^{(2)}$, а множество обучающих примеров включает миллион записей $D = \{d^{(1)}, \ldots, d^{(1.000.000)}\}$. В таком случае при нахождении $P(g, x^n)$ потребуется вычислить факториал вида $(n[s, j, g] + \alpha^{(j)} - 1)! = (1.000.000 + \alpha^{(j)} - 1)!$, в то время как такие 32-разрядные программы как MatLab и MathCAD способны вычислить факториалы не более $170!$.

МОДИФИЦИРОВАННАЯ ЛОГАРИФМИЧЕСКАЯ ФУНКЦИЯ КУПЕРА И ГЕРШКОВИЧА

Для более широкого использования функции КГ следует избавиться от факториала. Для этого прологарифмируем уравнение, описывающее функцию КГ:

$$\log(P(g, x^n)) = \log\left(P(g) \cdot \prod_{j \in J} \left(\prod_{s \in S(j,g)} \frac{(\alpha^{(j)} - 1)! \prod\limits_{q \in A^{(j)}} (n[q, s, j, g]!)}{(n[s, j, g] + \alpha^{(j)} - 1)!} \right) \right) =$$

$$= \log(P(g)) + \sum_{j \in J} \left(\sum_{s \in S(j,g)} \left(\sum_{i=1}^{\alpha^{(j)} - 1} i + \sum_{q \in A^{(j)}} \left(\sum_{i=1}^{n[q,s,j,g]} \right) - \sum_{i=1}^{n[s,j,g] + \alpha^{(j)} - 1} i \right) \right) =$$

$$= \log(P(g)) + \sum_{j \in J} \left(\sum_{s \in S(j,g)} \left(\sum_{q \in A^{(j)}} \left(\sum_{i=1}^{n[q,s,j,g]} \right) - \sum_{i=\alpha^{(j)}}^{n[s,j,g] + \alpha^{(j)} - 1} i \right) \right).$$

Полученное выражение умножим на -1 и для экономии вычислительных ресурсов исключим из него $\log(P(g))$. Как и в [5] предполагаем, что априорные вероятности $P(g)$ всех структур равны. Теперь вместо поиска структуры с максимальным значением функции КГ следует осуществлять поиск структуры с минимальным значением модифицированной логарифмической функции Купера и

Гершковича (МЛКГ):

$$F(g, x^n) = \sum_{j \in J} \left(\sum_{s \in S(j,g)} \left(\sum_{i=\alpha^{(j)}}^{n[s,j,g]+\alpha^{(j)}-1} i \right) \right) -$$

$$- \sum_{j \in J} \left(\sum_{s \in S(j,g)} \left(\sum_{q \in A^{(j)}} \left(\sum_{i=1}^{n[q,s,j,g]} i \right) \right) \right).$$

Как показали вычислительные эксперименты, функции КГ и МЛКГ выдают на одних и тех же обучающих данных абсолютно одинаковые решения. Однако на маленьких сетях (до 10 вершин) алгоритм с использованием функции КГ работает быстрее, чем МЛКГ, а на сетях с большим количеством вершин ситуация противоположна.

ФУНКЦИЯ ОПИСАНИЯ МИНИМАЛЬНОЙ ДЛИНЫ

При построении БС в качестве оценочной часто используют функцию ОМД [6, 9, 11, 12] или ее модификации. Для заданной последовательности $x^n = d_1, d_2, \ldots, d_n$ из n наблюдений ОМД структуры $g \in G$ вычисляется по формуле

$$L(g, x^n) = H(g, x^n) + \frac{k(g)}{2} \cdot \log(n),$$

где $k(g)$ — количество независимых условных вероятностей в сетевой структуре g; $H(g, x^n)$ — эмпирическая энтропия:

$$H(g, x^n) = \sum_{j \in J} H(j, g, x^n), \; k(g) = \sum_{j \in J} k(j, g).$$

ОМД j-й вершины вычисляется по формуле

$$L(j, g, x^n) = H(j, g, x^n) + \frac{k(j, g)}{2} \cdot \log(n),$$

где $k(j, g)$ — количество независимых условных вероятностей j-й вершины:

$$k(j, g) = (a^{(j)} - 1) \cdot \prod_{k \in \varphi(j)} \alpha^k,$$

$\varphi(j) \subseteq \{1, \ldots, j-1, j+1, \ldots, N\}$ — это такое множество, при котором $\Pi^{(j)} = \{X^{(k)} : k \in \varphi^{(j)}\}$.

Эмпирическая энтропия j-й вершины вычисляется согласно выражению

$$H(j, g, x^n) = \sum_{s \in S(j,g)} \sum_{q \in A^{(j)}} - n[q, s, j, g] \cdot \log \frac{n[q, s, j, g]}{n[s, j, g]};$$

$$n(s, j, g) = \sum_{i=1}^{n} I(\pi_i^{(j)} = s); \; n[q, s, j, g] = \sum_{i=1}^{n} I(x_i = q, \pi_i^{(j)} = s),$$

где $\pi^{(j)} = \Pi^{(j)}$ означает, что $X^{(k)} = x^{(k)} \; \forall \, k \in \varphi^{(j)}$; функция $I(E) = 1$, когда предикат $E = \text{true}$, в противном случае $I(E) = 0$.

ЭВРИСТИЧЕСКИЙ ПОИСК С ИСПОЛЬЗОВАНИЕМ УПОРЯДОЧЕННОГО МНОЖЕСТВА ВЕРШИН

Купер и Гершкович [5], а также Дехтер [13] и многие другие исследователи [8] для уменьшения пространства структур сети предлагают считать множество вершин упорядоченным. Иными словами, имеется упорядоченое множество вершин вида $\{X^{(1)}, X^{(2)}, \ldots, X^{(N)}\}$, где вершина $X^{(1)}$ — главная корневая вершина, у которой нет предков; $X^{(2)}$ — дочерняя вершина по отношению к $X^{(1)}$; $X^{(3)}$ — дочерняя вершина по отношению к какой-то предыдущей вершине или ко всем предыдущим вершинам одновременно и т.д.

В статье [5] предлагается эвристический метод, известный в литературе как алгоритм К2. К вершине $X^{(N)}$ по очереди добавляются предки от $X^{(1)}$ до $X^{(N-1)}$. С помощью функции КГ вычисляются $P(g, x^{(n)})$ для каждой комбинации. В качестве предка для вершины $X^{(N)}$ оставляют вершину $X^{(i)}$, при которой $P(g, x^{(n)})$ принимает максимальное значение. После этого к вершинам $X^{(N)}$ и $X^{(N-1)}$ по очереди добавляются предки от $X^{(1)}$ до $X^{(N-2)}$ и вычисляются $P(g, x^{(n)})$. На выходе метод выдает структуру сети g, для которой $P(g, x^{(n)})$ принимает максимальное значение.

Наличие упорядоченого множества вершин существенно сокращает пространство всех возможных нециклических структур. Но в этом случае появляется новая нетривиальная задача — как по множеству обучающих данных получить упорядоченное множество вершин сети. Наиболее очевидный способ — привлечь экспертов. Однако может возникнуть потребность моделирования данных в такой предметной области, где квалифицированных экспертов нет.

ЗНАЧЕНИЕ ОБОЮДНОЙ ИНФОРМАЦИИ МЕЖДУ ВЕРШИНАМИ

Для оценки степени зависимости двух произвольных переменных x^i и x^j в работе [1] Шоу и Лиу предложили в 1968 г. использовать значение обоюдной информации (mutual information) $MI(x^i, x^j)$. Для расчета предложено следующее выражение:

$$MI(x^i, x^j) = \sum_{x^i, x^j} P(x^i, x^j) \cdot \log\left(\frac{P(x^i, x^j)}{P(x^i) \cdot P(x^j)}\right).$$

По своей сути значение обоюдной информации является аналогом корреляции, но по своему содержанию — это оценка количества информации о переменной x^j, содержащейся в переменной x^i. Значение обоюдной информации принимает неотрицательные значения $MI(x^i, x^j) \geq 0$, а в случае, если вершины x^i и x^j полностью независимы одна от другой, то $MI(x^i, x^j) = 0$, так как $P(x^i, x^j) = P(x^i) \cdot P(x^j)$; следовательно,

$$\log\left(\frac{P(x^i, x^j)}{P(x^i) \cdot P(x^j)}\right) = \log\left(\frac{P(x^i) \cdot P(x^j)}{P(x^i) \cdot P(x^j)}\right) = \log(1) = 0.$$

Значение обоюдной информации используется вместо упорядоченого множества вершин при построений БС [1, 2, 6, 12].

РЕЗУЛЬТАТЫ ВЫЧИСЛИТЕЛЬНЫХ ЭКСПЕРИМЕНТОВ

Наиболее распространенными оценочными функциями при построении структур БС являются функции КГ и ОМД, а также их модификации. Поэтому в настоящей статье выполнено сравнение алгоритмов, использующих МЛКГ и ОМД относительного времени, затраченого на построение БС. Для определения порядка добавления дуг в БС использовалось значение обоюдной информации [12], а для построения — выборка генетических данных, состоящая из 600 обучающих записей. На рис. 4 и 5 показаны графики временных затрат вычисления алгоритмов. Как видно, для построения БС, состоящих из более чем 30 вершин, алгоритм с использованием ОМД работает быстрее, чем с МЛКГ.

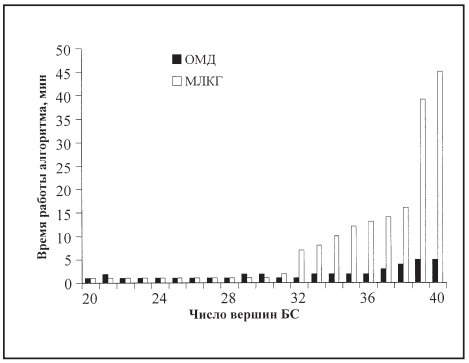

Рис. 4. Диаграмма времени вычисления алгоритмов с использованием МЛКГ и ОМД

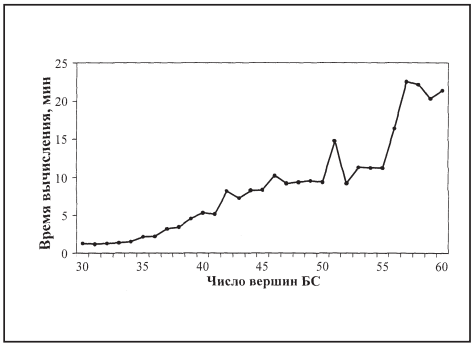

Рис. 5. График временных затрат алгоритма с использованием ОМД

ЗАКЛЮЧЕНИЕ

Использование БС является востребованным современным подходом в области обработки информации при моделировании процессов различной природы и сложности. В статье выполнен анализ десяти методов построения БС, использующих оценочные функции. Наиболее распространенными оценочными функциями являются функции КГ и ОМД, а также их модификации. Результаты вычислительных экспериментов показали, что на коротких обучающих выборках (до 170 записей) и сетях, состоящих не более чем из 10 вершин, алгоритм с использованием функции КГ работает быстрее по сравнению с МЛКГ и ОМД. Однако алгоритмы, использующие МЛКГ и ОМД, в отличие от КГ, работают с обучающими выборками любого размера. Вычислительные эксперименты показали, что методы построения БС с применением функций КГ и ее модификаций, выполняют переобучение БС, т. е. такие сети содержат лишние дуги.

При сравнении функций МЛКГ и ОМД на выборках генетических данных, состоящих из 600 обучающих записей, функция МЛКГ показала лучшее время вычисления на сетях, состоящих не более чем из 30 вершин. Но алгоритм с использованием ОМД по сравнению с МЛКГ работает в несколько раз быстрее на больших сетях, состоящих из более чем 30 вершин.

СПИСОК ЛИТЕРАТУРЫ

1. Chow C.K., Liu C.N. Approximating discrete probability distributions with dependence trees // IEEE Transactions on information theory. — 1968. — **4**, N 3. — P. 462–467.

2. Rebane G., Pearl J. The recovery of causal poly-trees from statistical data // Intern. Jour. of Approx. Reas. — 1988. — **2**, N 3. — P. 175–182.

3. Herskovits E., Cooper G. Kutato: an entropy-driven system for construction of probabilistic expert systems from databases // Proc. of the Sixth Intern. Conf. on Uncertainty in Arti-

ficial Intelligence (UAI'90), Cambridge, MA, USA. — New York: Elsevier science, 1991. — P. 54–62.

4. Spirtes P., Glymour C., Scheines R. From probability to causality // Philos. Studies. — Amsterdam: Springer Netherlands. — 1991. — **64**, N 1. — P. 1–36.
5. Cooper G., Herskovits E. A Bayesian method for the induction of probabilistic networks from data // Machine Learning. — 1992. — **9**. — P. 309–347.
6. Lam W., Bacchus F. Learning Bayesian belief networks: an approach based on the MDL principle // Computational Intelligence. — 1994. — **10**, N 4. — P. 269–293.
7. Acid S., Campos L. Benedict : an algorithm for learning probabilistic belief networks // Proc. of the Sixth International Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'96), Granada, Spain. — New York: Springer, 1997. — P. 979–984.
8. Singh M., Valtorta M. Construction of Bayesian network structures from data: a brief survey and an efficient algorithm // Intern. Jour. of Approx. Rea. — 1995. — **12**. — P. 111–131.
9. Friedman N., Goldszmidt M. Learning Bayesian networks with local structure // Proc. of the Twelfth International Conf. on Uncertainty in Artificial Intelligence (UAI'96), Portland, Oregon, USA. — SF.: Morgan Kaufmann, 1996. — P. 252–262.
10. Wallace C., Korb K., Dai H. Causal discovery via MML / Proc. of the Thirteenth Intern. Conf. on Machine Learning (ICML'96), Bari, Italy. — SF.: Morgan Kaufmann, 1996. — P. 516–524.
11. Suzuki J. Learning Bayesian belief networks based on the MDL principle: an efficient algorithm using the branch and bound technique // IEICE Trans. on Inform. and Systems. — 1999. — P. 356–367.
12. Терентьев А.Н., Бидюк П.И. Эвристический метод построения байесовских сетей // Мат. машины и системы. — 2006. — **3**. — С. 12–23.
13. Dechter R. Bucket elimination: a unifying framework for reasoning // ACM Press. — 1996. — **28,** N 61. — P. 1–51.

Поступила 27.06.2007