

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Методи ймовірного висновку в мережах Байєса

(тема)

Виконав:

студент II курсу, групи КСММ-19-1
Крікун А.О.
(прізвище, ініціали)

Спеціальність 123 – Комп'ютерна інженерія
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерні системи та мережі
(повна назва освітньої програми)

Керівник: доц. Ільїна І.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ

(підпис)

Коваленко А.А.

(прізвище, ініціали)

2020 р.

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління _____

Кафедра _____ електронних обчислювальних машин _____

Рівень вищої освіти _____ другий (магістерський) _____

Спеціальність _____ 123 – Комп'ютерна інженерія _____
(код і повна назва)

Тип програми _____ освітньо-професійна _____
(освітньо-професійна або освітньо-наукова)

Освітня програма _____ Комп'ютерні системи та мережі _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ РОБОТУ

студентові _____ Крікуну Артему Олександровичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Методи ймовірнісного висновку в мережах Байєса _____

затверджена наказом по університету від “ 30 ” жовтня 2020 р. № 1487Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 14 грудня 2020 р.

3. Вхідні дані до роботи _____

Мережа Байєса _____

Ймовірнісний висновок _____

Алгоритми кластеризації _____

LS-метод _____

4. Перелік питань, що потрібно опрацювати в роботі _____ Огляд мереж Байєса, алгоритмів та методів ймовірнісного висновку в них. Огляд на LS-метод, опис роботи даного методу, приведення прикладу роботи та порівняння результатів роботи методу з іншими програмними, що використовують мережі Байєса. _____

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Слайд презентація – 29 слайдів

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Огляд на мережі Байєса та методи ймовірнісного висновку	03.11.20-09.11.20	
2	Вибір методики дослідження	10.11.20-17.11.20	
3	Вибір інструментальних засобів	18.11.20-23.11.20	
4	Проведення експериментів та розрахунків	24.11.20-01.12.20	
5	Оформлення матеріалів атестаційної роботи	02.12.20-07.12.20	
6	Подання атестаційної роботи керівникові та її попередній захист	08.12.20-09.12.20	
7	Подання атестаційної роботи на рецензування	10.12.20-11.12.20	

Дата видачі завдання 02 листопада 2020 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис)

доц. Ільїна І.В.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка атестаційної роботи: 80 с., 31 рис., 39 табл., 1 дод., 22 джерел.

МЕРЕЖА БАЙЄСА, АРХІТЕКТУРА, ТЕОРІЯ ЙМОВІРНОСТІ, АЛГОРИТМ, LS-МЕТОД, ОБ'ЄДНАНЕ ДЕРЕВО, ЙМОВІРНІСНИЙ ВИСНОВОК.

Метою атестаційної роботи є дослідження мереж Байєса, методів ймовірнісного висновку в даних мережах, зокрема, огляд на LS-метод, алгоритм його роботи та його практичне застосування.

У ході виконання атестаційної роботи розглянуті дискретні мережі Байєса, їх класифікація та сфера застосування. Зроблен огляд на методи побудови ймовірнісного висновку в мережах Байєса, зроблен поетапний опис роботи LS-методу та змодельована ситуація в мережі Байєса, для якої були виконані розрахунки за допомогою LS-методу та іншого програмного продукту. Зроблен порівняльний аналіз результатів моделювання.

ABSTRACT

Master's thesis: 80 pages, 31 figures, 39 tables, 1 appendices, 22 sources.

BAYES NETWORK, ARCHITECTURE, PROBABILITY THEORY,
ALGORITHM, LS-METHOD, COMBINED TREE, PROBABILITY
CONCLUSION.

The major goal of this thesis is studying Bayesian networks, methods of probabilistic inference in these networks, in particular, an overview of the LS-method, the algorithm of its work and its practical application.

In the course of the master's thesis, discrete Bayesian networks, their classification and scope are considered. An overview of the methods of constructing a probabilistic conclusion in Bayesian networks, a step-by-step description of the LS-method and a simulated situation in the Bayesian network, for which calculations were performed using the LS-method and other software product. A comparative analysis of the simulation results is made.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ.....	8
ВСТУП	9
1 МЕРЕЖІ БАЙЄСА	10
1.1 Поняття мереж Байєса	10
1.2 Представлення мереж Байєса у вигляді графів.....	11
1.3 Типи мереж Байєса.....	14
1.3.1 Дискретні мережі Байєса.....	14
1.3.2 Неперервні мережі Байєса.....	16
1.3.3 Гібридні мережі Байєса	18
1.3.4 Динамічні мережі Байєса.....	19
1.4 Використання мереж Байєса	20
1.5 Формула Байєса	21
1.6 Приклад використання формули Байєса	23
1.7 Висновки по першому розділу.....	27
2 ЙМОВІРНІСНИЙ ВИСНОВОК	28
2.1 Алгоритми точного висновку	28
2.2 Апроксимаційні алгоритми	31
2.3 Види архітектур алгоритмів кластеризації.....	32
2.3.1 Архітектура Lauritzen-Spiegelhalter	32
2.3.2 Архітектура Hugin.....	33
2.3.3 Архітектура Shenoy-Shafer	34
2.4 Висновки по другому розділу	36
3 LS-МЕТОД ЙМОВІРНІСНОГО ВИСНОВКУ	37
3.1 Побудова об'єднаного дерева	37
3.2 Алгоритм пропагації	39
3.3 Приклад побудови ймовірнісного висновку	40

3.4 Порівняння архітектур за необхідними об'ємами пам'яті	59
3.5 Висновки до третього розділу	60
ВИСНОВКИ.....	61
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	62
ДОДАТОК А Графічний матеріал атестаційної роботи	65

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ
І ТЕРМІНІВ

МБ – мережа Байєса

ТУЙ – таблиця умовних ймовірностей

LS-метод – метод Lauritzen-Spiegelhalter

ВСТУП

Метою цієї роботи є вивчення байєсівських мереж, а також огляд методів імовірнісного виведення в байєсівських мережах. Детальна перевірка LS-методу імовірнісного виведення і принципу функціонування, порівняння результатів LS-методу з іншими, а також практичне застосування методів вирішення проблем в реальному житті.

Бурхливий розвиток комп'ютерних технологій за останні півстоліття і вдосконалення технологій зберігання і відтворення інформації призвели до накопичення великих баз даних. Однак отримати максимальну віддачу від наявних даних стало набагато складнішим завданням, ніж кілька десятиліть тому, і процес обробки інформації залишається одним з найважливіших, оскільки неінтерпретована інформація сама по собі не корисна.

Технологія інтелектуального аналізу даних з'явилася в 1978 році для вирішення проблеми виявлення прихованих відносин у великих базах даних. Більшість інструментів інтелектуального аналізу даних засновані на двох технологіях: машинному навчанні та візуалізації. Байєсівські мережі (МБ) об'єднують ці дві технології, цей напрямок наукового розвитку з'явилося на стику двох наук – теорії ймовірностей і теорії графів, а термін МБ з'явився в 1985 році. Ідея цієї реалізації полягає в тому, щоб представити причинно-наслідкові зв'язки проблеми у вигляді графіка.

Байєсівські мережі включають логічну інтерпретацію і модифікацію структури відносин між змінними проблеми, а також дозволяють явно враховувати попередній апріорний досвід експертів. Завдяки успішному представленню у вигляді графів МБ дуже зручні для вирішення прикладних проблем. Вони засновані на базових принципах і результатах теорії ймовірностей, розроблених протягом декількох сотень років, які забезпечують їх успіх у вирішенні практичних проблем.

1 МЕРЕЖІ БАЙЄСА

1.1 Поняття мереж Байєса

Мережа Байєса (МБ) представляє собою пару $\langle G, P \rangle$, де G – це направлений ациклічний граф, а $P = \{P(X_i | Pa(X_i)) \mid i = 1..n\}$ – множина таблиць умовних ймовірностей вершин. $P(X_i | Pa(X_i))$ – це відповідно умовний ймовірнісний розподіл вершини X_i залежно від інстанціювання її батьківських вершин – $Pa(X_i)$ [7].

Коли вершина приймає значення одного зі своїх станів, то вона називається інстанційованою.

Сукупний ймовірнісний розподіл в МБ обчислюється за формулою:

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | Pa(X_i)). \quad (1.1)$$

Математично МБ – це модель для представлення існуючих і відсутніх ймовірнісних залежностей. Зв'язок $A \rightarrow B$ є каузальним, якщо подія A є причиною події B , тобто якщо існує механізм, згідно з яким величина A приймає значення що впливає на B . МБ називається причинною, якщо всі її зв'язки є причинні. Існує ряд причин використання причинних моделей в штучному інтелекті:

- як правило, людина інтерпретує події з точки зору причини і слідства, що спрощує розуміння Користувачем причинно-наслідкових закономірностей;

- виявлення інваріантних причинно-наслідкових зв'язків в конкретній проблемі дозволяє передбачити наслідки випадкових подій і наслідки тих чи інших дій, тобто різних маніпуляцій або втручань;

- причинність і ймовірність тісно пов'язані між собою, так як причинність зазвичай передбачає існування ймовірнісних відносин, що

передають розуміння причинності; насправді необхідною умовою існування причинності є кореляція;

- аксіоматичні властивості МБ відповідають імовірнісним залежностям і випадкам їх відсутності, що виникають в причинній області;

- існують канонічні імовірнісні моделі, засновані на інтерпретації батьками вузла як причин або умов для цього вузла, а також на прийнятті незалежності причинно-наслідкових взаємодій; ці моделі зменшують кількість параметрів мережі, спрощують придбання знань і навіть сприяють зниженню обчислювальних витрат;

- причинні МБ підтримують деякі якісні аспекти рішень, які можуть бути визначені, щоб пояснити результати висновку; деякі з цих аспектів є у певних канонічних моделях.

1.2 Представлення мереж Байєса у вигляді графів

Теорія графів часто використовується для моделювання процесів різної природи, а МБ також бувають представлені у вигляді графу.

Граф – це набір вершин (вузлів), з'єднаних дугами. Дуга між двома вершинами вказує на наявність відносини між ними, а спрямована дуга вказує напрямком цього зв'язку – від причини до слідства. У байєсівській мережі всі дуги спрямовані, тобто це направлений граф.

Направлений граф – це пара $G = \langle V, E \rangle$, де $V = \{X_1, \dots, X_n\}$ – це множина вершин (змінні) і $E = \{(X_i, X_j) \mid X_i, X_j \in V\}$ – множина ребер графа.

Направлений граф є ациклічним, якщо він не містить спрямованих циклів, тобто для кожної вершини графа X_i немає спрямованого шляху, який починається і закінчується в X_i . Рисунок 1.1 а) показує приклад циклічного графу, рисунок 1.1 б) – ациклічного.

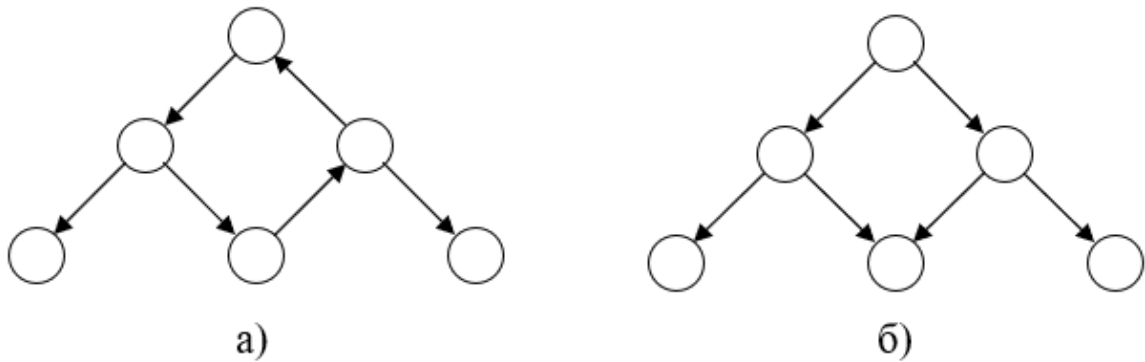


Рисунок 1.1 – Види графів: а) циклічний граф; б) ациклічний граф

На рисунку 1.2 в парі $(X_i, X_j) \in G$, X_i є батьком вершини X_j , а X_j відповідно є нащадком X_i .

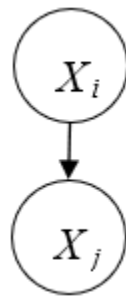


Рисунок 1.2 – Пара $(X_i, X_j) \in G$

Вершина X_i називається коренем в направленому графі, якщо в неї не входять ребра, а вершина X_j називається листком, якщо з неї не виходять ребра.

За типами структури графів МБ бувають деревами, однозв'язними мережами та багатозв'язними мережами [2].

Дерево – це направлений ациклічний граф, де кожна вершина може мати не більше одного батька (рисунок 1.3).

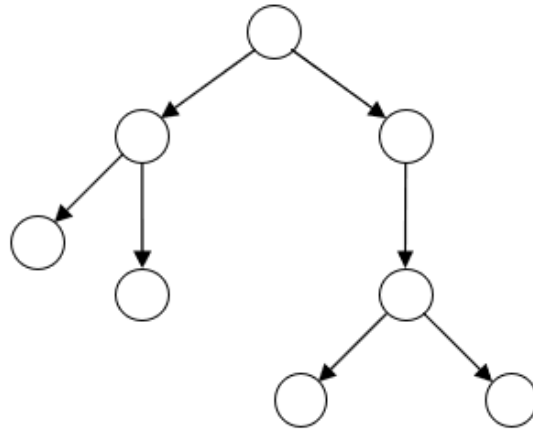


Рисунок 1.3 – Приклад дерева

Однозв'язна мережа або полідерево – це направлений ациклічний граф, в якому кожна вершина може мати більше одного батька, але існує тільки один (ненаправлений) шлях між будь-якими двома вершинами (рисунок 1.4).

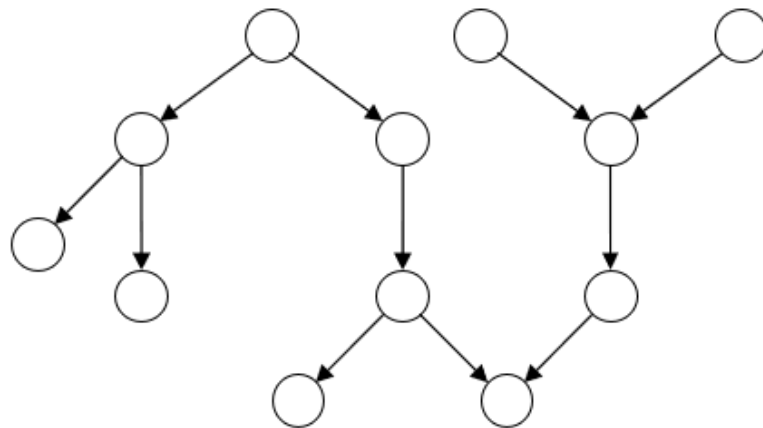


Рисунок 1.4 – Приклад полідерева

В багатозв'язних мережах між будь-якими двома вершинами може існувати декілька шляхів (рисунок 1.5). Саме такі мережі зустрічаються найчастіше при вирішенні різноманітних задач.

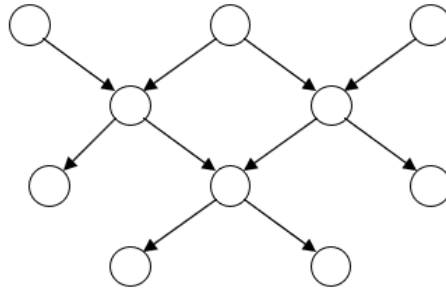


Рисунок 1.5 – Приклад багатозв’язної мережі

1.3 Типи мереж Байєса

В залежності від типів станів вершин та властивостей самих вершин виділяють наступні типи мереж Байєса:

- дискретні;
- неперервні;
- гібридні;
- динамічні.

1.3.1 Дискретні мережі Байєса

Дискретні МБ – це мережі, в яких змінні вершин дискретні, тобто вони мають кінцеву кількість станів.

Для опису ймовірнісного розподілу дискретної випадкової величини використовується таблиця – ряд розподілу. У байєсівській мережі він записується у вигляді таблиць умовних ймовірностей.

На рисунку 1.6 наведений приклад дискретної мережі діагностування причини відсутності зображення на екрані монітору, де:

- S – скачок напруги;
- M – пошкодження материнської плати комп’ютеру;
- V – пошкодження відеоадаптеру;
- I – немає зображення на екрані.

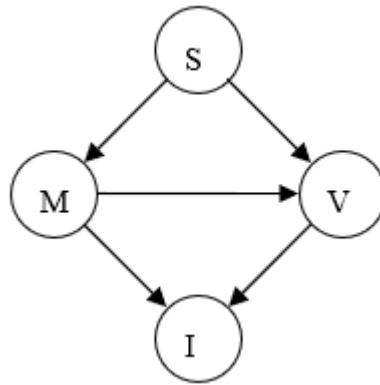


Рисунок 1.6 – Приклад дискретної МБ

Вказаній структурі мережі відповідає наступний набір таблиць умовних ймовірностей вершин (таблиця 1.1–1.4), де 1 означає позитивну відповідь, а 0 – негативну, тобто якщо $M=0$, це означає що материнська плата неушкоджена.

Таблиця 1.1 – Значення умовної ймовірності вершини S

Стан	Ймовірність
1	2
S=0	0,6
S=1	0,4

Таблиця 1.2 – Значення умовної ймовірності вершини M

Батьки	Стани батьків	
	2	3
S	S=0	S=1
Стан	Ймовірність	
M=0	0,6	0,4
M=1	0,4	0,6

Таблиця 1.3 – Значення умовної ймовірності вершини V

Батьки	Стани батьків			
1	2	3	4	5
S	S=0	S=0	S=1	S=1
M	M=0	M=1	M=0	M=1
Стан	Ймовірність			
V=0	0,67	0,5	1	0,33
V=1	0,33	0,5	0	0,67

Таблиця 1.4 – Значення умовної ймовірності вершини I

Батьки	Стани батьків			
1	2	3	4	5
M	M=0	M=0	M=1	M=1
V	V=0	V=1	V=0	V=1
Стан	Ймовірність			
I=0	1	0	0,5	0
I=1	0	1	0,5	1

1.3.2 Неперервні мережі Байєса

Неперервні МБ – це мережі, в яких змінні вершин є неперервними.

В багатьох випадках події можуть приймати будь-який стан з прийнятого діапазону. Тобто змінна X є неперервною випадковою величиною, а набір її можливих станів є весь діапазон значень, який вона може приймати: $X = \{x : a \leq x \leq b\}$ [1].

Функція розподілу або щільність розподілу використовується для опису розподілу ймовірностей безперервної випадкової величини.

Нехай вершина X має декілька батьків – $A = \{A_1, \dots, A_n\}$, тоді умовний розподіл випадкової величини задається такою формулою [4]:

$$f(X|A_i) = N(\mu_x + k_i * \mu_i; \sigma_x), \quad (1.2)$$

де μ_x – математичне сподівання величини X ;

μ_i – математичне сподівання величини A_i ;

k_i – ваговий коефіцієнт, який відображає зв'язок між вершиною X та її батьком A_i ;

σ_x – стандартне відхилення величини X :

$$N(\mu_x + k_i * \mu_i; \sigma_x) = \frac{1}{\sigma_x \sqrt{2\pi}} * \exp \left[-\frac{1}{2} \left(\frac{x - (\mu_x + k_i * \mu_i)}{\sigma_x} \right)^2 \right]. \quad (1.3)$$

Зв'язок між змінною X та її батьками $A = \{A_1, \dots, A_n\}$ можна представити за допомогою лінійної регресійної моделі:

$$X = k_1 * A_1 + \dots + k_n * A_n + Q_x, \quad (1.4)$$

де k_i – регресійний коефіцієнт, який відображає зв'язок між змінними X та A_i ;

Q_x – шумова компонента, що може бути записана у вигляді нормального розподілу з нульовим математичним сподіванням [4].

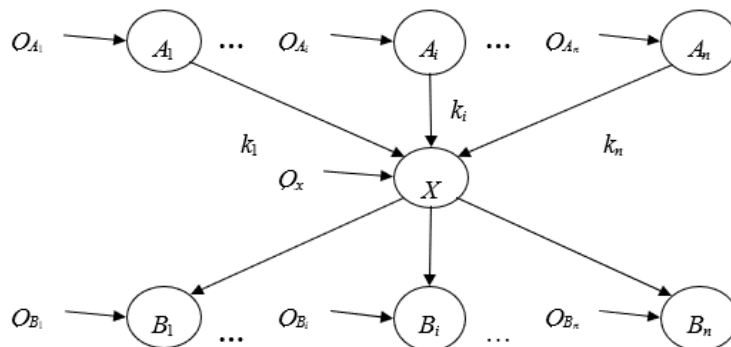


Рисунок. 1.7 – Приклад неперервної МБ

На рисунку 1.7 наведений приклад неперервної МБ з нормальним законом розподілу, де A_1, \dots, A_n – батьки X ; B_1, \dots, B_n – її нащадки; k_1, \dots, k_n – вагові коефіцієнти; $Q_{A_1}, \dots, Q_{A_n}, Q_X, Q_{B_1}, \dots, Q_{B_n}$ – шумові коефіцієнти.

1.3.3 Гібридні мережі Байєса

Гібридні МБ – це мережі, вершини яких містять як дискретні, так і неперервні змінні.

Використання дискретних і неперервних змінних в одній мережі вимагає деяких обмежень:

- дискретні вершини не можуть мати неперервних батьків;
- неперервні змінні мають нормальний закон розподілу, залежно від значень їх батьків;
- розподіл неперервної змінної X з дискретними батьками A і неперервними батьками B є нормальним.

Нехай дискретні випадкові величини A_1, \dots, A_p та неперервні випадкові величини B_1, \dots, B_m впливають на результуючу неперервну випадкову величину X , як показано на рисунку 1.8. Квадратні вершини мають дискретний розподіл, а круглі – неперервний.

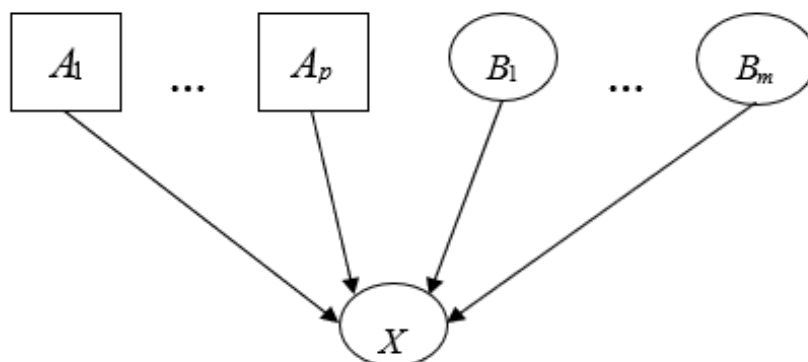


Рисунок 1.8 – Приклад гібридної МБ

Кожна з дискретних випадкових величин A_j ($j = 1, \dots, p$) може приймати значення A_{ij} ($i = 1, \dots, n_j$) з ймовірностями P_{ij} , для яких $\sum P_{ij} = 1$. Загальний вплив дискретних випадкових величин на X характеризується математичним сподіванням (μ_1, \dots, μ_p) та стандартним відхиленням ($\sigma_1, \dots, \sigma_p$). Загальний вплив неперервних величин та значень дискретних величин на кінцеву випадкову величину X позначається ваговими коефіцієнтами $k_{l,i,1,\dots,i_p}$ для $l = 1, \dots, m$.

Тоді характеристики величини X можна обчислити за формулами:

$$\mu_x = \sum_{i_1=1}^{n_1} \dots \sum_{i_p=1}^{n_p} P_{1,i_1} * \dots * P_{p,i_p} * (\mu_{i_1,\dots,i_p} + \sum_{l=1}^m k_{l,i_1,\dots,i_p} * \mu_l), \quad (1.5)$$

$$\sigma_x = \sqrt{\sum_{i_1=1}^{n_1} \dots \sum_{i_p=1}^{n_p} P_{1,i_1} * \dots * P_{p,i_p} * ((\mu_{i_1,\dots,i_p} + \sum_{l=1}^m k_{l,i_1,\dots,i_p} * \mu_l)^2 + \sigma_{i_1,\dots,i_p}^2 + \sum_{l=1}^m k_{l,i_1,\dots,i_p}^2 * \sigma_l^2) * \mu^2}. \quad (1.6)$$

1.3.4 Динамічні мережі Байєса

Динамічні МБ – це мережі, в яких значення вершин змінюється з плином часу, тому вони використовуються для моделювання тимчасових процесів. Динамічні байєсівські мережі ще також називають «часовими», оскільки структура моделі залишається незмінною, хоча приховані вузли можуть бути додані для опису поточного стану процесу [5].

Найпростіший тип динамічної МБ – це прихована модель Маркова, в якій кожен шар має приховану дискретну вершину і спостережувану дискретну або безперервну вершину (рисунок 1.9).

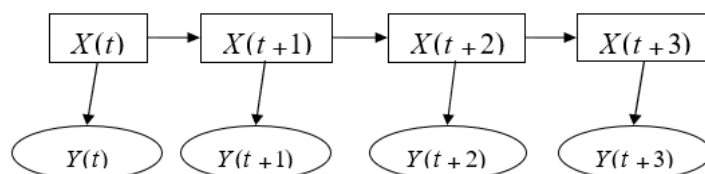


Рисунок 1.9 – Приклад динамічної мережі Байєса

Для визначення динамічної мережі необхідно вказати початковий розподіл $P(X(t))$, топологію всередині шару $P(X(t + i) | X(t + i - 1))$ та між двома шарами - $P(Y(t) | X(t))$ [6].

Як приклад використання динамічних МБ можна привести розпізнавання мови. В ньому спостережуваними змінними Y будуть фонемі, а прихованими X – літери вимовлених слів. Така мережа складається з набору блоків, які повторюються в різний час.

1.4 Використання мереж Байєса

Медицина стала однією з перших областей, де байєсівські мережі знайшли своє застосування. Системи, засновані на них, зосереджені на діагностиці, прогнозуванні результатів різних методів лікування і виборі оптимального лікування захворювання одного пацієнта.

Військові завжди активно використовували нові технології, тому в цьому напрямку використовуються і байєсівські мережі. Системи навчання використовують байєсівські мережі для моделювання захисту територіальної зони з моря. До них відносяться різні тактичні сценарії конфліктуючих сторін, інформація про рух кораблів, розвідувальні та інші дані. Послідовне отримання інформації про дії супротивників дозволяє передбачати ймовірності подій протягом усього конфлікту.

Існує багато розробок для космосу з використанням байєсівських мереж. Одним з основних завдань NASA є оцінка стану елементів різних систем, в яких використовуються датчики, тому діагностика помилок датчиків і елементів системи є важливою проблемою. Використання байєсівських мереж дозволяє вирішити цю проблему на високому рівні – мережа з більш ніж 400 вершин може бути об'єднана в арифметичну схему, що здатна правильно реагувати на запити.

Також байєсівські мережі знайшли себе у робототехніці, аналізі ризиків виникнення злочинів, в менеджменті та інших сферах.

1.5 Формула Байєса

Поняття ймовірності відіграє значну роль у повсякденному житті, але математична теорія ймовірностей була сформульована порівняно недавно (у XVII столітті). Класично ймовірність події означає відношення випадків, в яких воно відбулося, до загальної кількості спостережень. Однак існує кілька інтерпретацій теорії ймовірностей, де до найбільш домінуючих відносяться об'єктивістська, суб'єктивістська і логічна [7].

Об'єктивістський підхід полягає в тому, що частота виникнення події при великій кількості спостережень призводить до об'єктивної ймовірності цієї події.

Суб'єктивістський підхід розглядає ймовірнісну міру як міру довіри до ставлення людини до істини висловлювання. Люди, які беруть участь у події, можуть мати різні ступені довіри до одного і того ж твердження. Термін «байєсівський» часто використовується як синонім суб'єктивної ймовірності.

Логічний підхід розглядає ймовірність як ступінь довідності логічно вивіреного висновку. Цю точку зору можна розглядати як розширення звичайної логіки.

Ці підходи використовують різні схеми виводу, тобто дві різні школи ймовірнісних розрахунків – школу Паскаля (загальноприйняту) і школу Бекона (індуктивну). Розрахунки Бекона використовують логічні правила для доведення або спростування гіпотез, і ця школа обчислень використовує логічний погляд на ймовірність. Розрахунки Паскаля використовують байєсівські правила для перевірки та обробки заходів довіри, а об'єктивістські та суб'єктивістські підходи використовують цю школу обчислень.

Припустимо, що у нас є дві події – $A \in \Omega$ і $B \in \Omega$.

Ймовірність того, що подія A відбудеться за умови, що відбулась подія B записується у вигляді $P(A | B)$ і називається умовною ймовірністю події A при заданій події B . Ймовірність, що обидві події відбудуться $P(A \cap B)$ називається сукупною ймовірністю.

Умовна ймовірність $P(A|B)$ дорівнює відношенню сукупної ймовірності до ймовірності події B , за умови що вона не дорівнює нулю:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}. \quad (1.7)$$

Аналогічно формула має місце для події B :

$$P(B|A) = \frac{P(B \cap A)}{P(A)}. \quad (1.8)$$

Завдяки властивості комутативності сукупної ймовірності $P(A \cap B) = P(B \cap A)$ ми отримуємо правило Байєса [4]:

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}. \quad (1.9)$$

Якщо множини A і B не перетинаються між собою, то $P(A \cup B) = P(A) + P(B)$. За теорією множин B можна виразити так:

$$B = (B \cap A) \cup (B \cap \bar{A}). \quad (1.10)$$

Так як ці об'єднання не перетинаються між собою, то:

$$P(B) = P((B \cap A) \cup (B \cap \bar{A})) = P(B \cap A) + P(B \cap \bar{A}) = P(B|A) * P(A) + P(B|\bar{A}) * P(\bar{A}). \quad (1.11)$$

Повертаючись від множин до подій, ми можемо підставити отриманий вираз в формулу Байєса:

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B|A) * P(A) + P(B|\bar{A}) * P(\bar{A})}. \quad (1.12)$$

Перепишемо цю формулу в термінах гіпотез H і спостережень E :

$$P(H|E) = \frac{P(E|H)*P(H)}{P(E|H)*P(H)+P(E|\bar{H})*P(\bar{H})}. \quad (1.13)$$

Цей вираз показує зв'язок між гіпотезою і спостереженням і спостереженням з гіпотезою, яка ще не підтверджена. Вважається, що апріорна ймовірність гіпотези $P(H)$ і ймовірність $P(E|H)$ визначаються до отримання спостережень. Ймовірність $P(H|E)$ називається апостеріорною ймовірністю гіпотези h за спостереженням E .

Підсумовуючи цей вираз для випадку n гіпотез, отримаємо наступну формулу:

$$P(H_k|E) = \frac{P(E|H_k)*P(H_k)}{\sum_{i=1}^n P(E|H_i)*P(H_i)}. \quad (1.14)$$

1.6 Приклад використання формули Байєса

На рисунку 1.10 зображена мережа діагностування причини відсутності зображення на екрані монітору, де:

- S – скачок напруги;
- M – ушкодження материнської плати комп'ютеру;
- V – ушкодження відеоадаптеру;
- I – відсутність зображення на екрані.

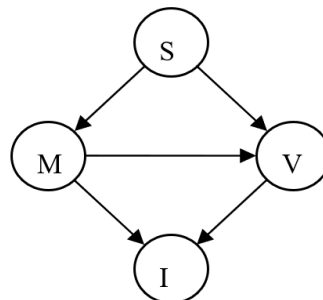


Рисунок 1.10 – Діагностична МБ

В таблиці 1.5 вказаний набір даних по даній мережі на основі 10 спостережень:

Таблиця 1.5 – Дані спостережень

№	Спостереження			
	S	M	V	I
1	2	3	4	5
1	1	0	0	0
2	0	1	0	1
3	0	1	1	1
4	1	1	1	1
5	0	1	1	1
6	0	0	0	0
7	1	1	0	0
8	0	0	1	1
9	1	0	0	0
10	0	0	0	0

В таблиці 1.6 вказані значення сукупного розподілу всієї мережі, що отримані на основі даних спостережень:

Таблиця 1.6 – Сукупний розподіл

№	Спостереження				Ймовірність всієї мережі
	S	M	V	I	
1	2	3	4	5	6
1	0	0	0	0	0,2
2	0	0	1	1	0,1
3	0	1	0	1	0,1
4	0	1	1	1	0,2

Продовження таблиці 1.6

1	2	3	4	5	6
5	1	0	0	0	0,2
6	1	1	0	0	0,1
7	1	1	1	1	0,1

З отриманих значень можна обчислити значення сукупного розподілу ймовірностей вершин, що відображені на рисунку 1.11.

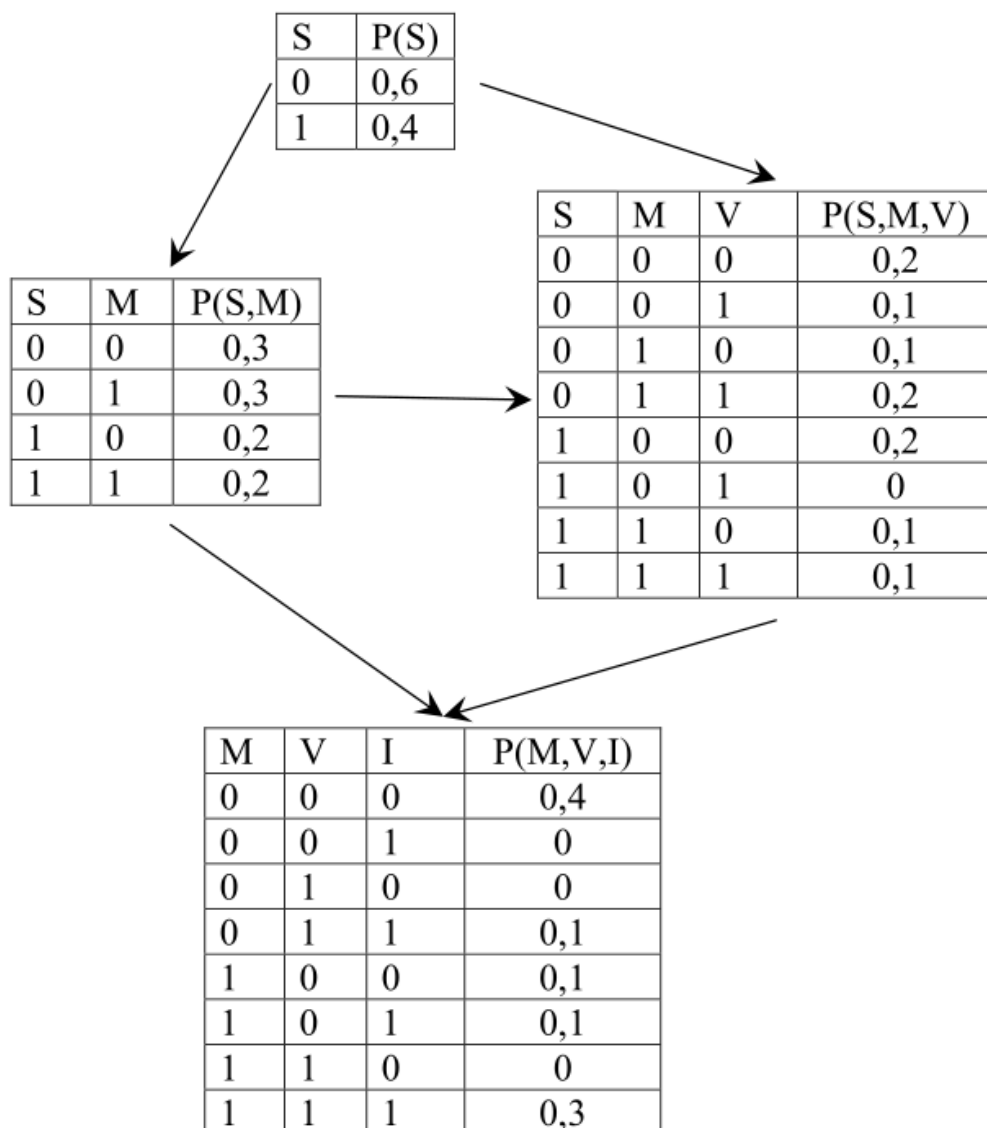


Рисунок 1.11 – Мережа Байєса у вигляді сукупного розподілу ймовірностей вершин

Використовуючи формулу Байєса за значеннями сукупного розподілу ймовірностей вершин і значеннями ймовірностей вершин, можна обчислити значення умовних ймовірностей, які будуть формувати таблицю умовних ймовірностей (рисунок 1.12).

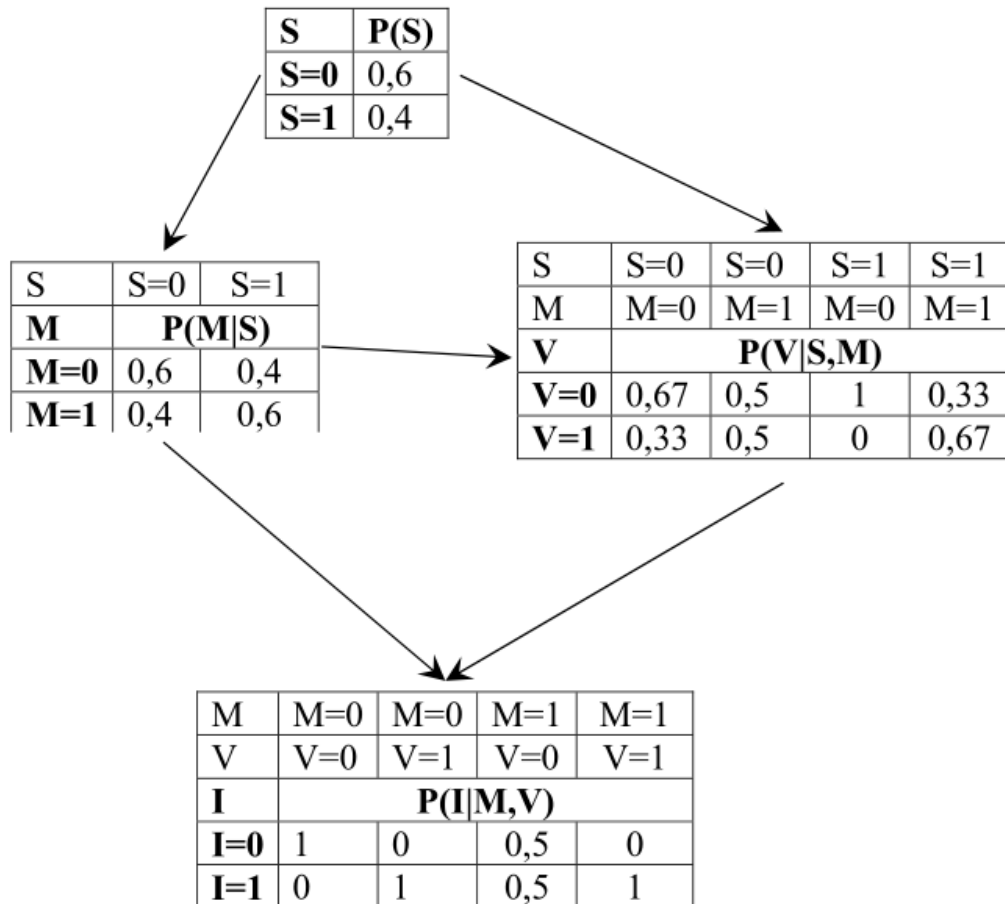


Рисунок 1.12 – Мережа Байєса у вигляді таблиць умовних ймовірностей

Повна сукупна ймовірність даної МБ розраховується за формулою 1.15:

$$P(S, M, V, I) = P(S) * P(M|S) * P(V|S, M) * P(I|M, V). \quad (1.15)$$

Обчислені значення наведені в таблиці 1.7, де з'явилися нові події та їх ймовірності.

Таблиця 1.7 – повний сукупний розподіл

№	Спостереження				Ймовірність всієї мережі
	S	M	V	I	
1	2	3	4	5	6
1	0	0	0	0	0,201
2	0	0	0	1	0
3	0	0	1	0	0
4	0	0	1	1	0,099
5	0	1	0	0	0,0495
6	0	1	0	1	0,0495
7	0	1	1	0	0
8	0	1	1	1	0,201
9	1	0	0	0	0,2
10	1	0	0	1	0
11	1	0	1	0	0
12	1	0	1	1	0
13	1	1	0	0	0,05
14	1	1	0	1	0,05
15	1	1	1	0	0
16	1	1	1	1	0,1

1.7 Висновки по першому розділу

У першому розділі розглядалося таке поняття, як байєсівська мережа, наведено її запис і структуру. Були також описані основні типи існуючих мереж і розглянуті системи, що використовують байєсівські мережі для вирішення проблем і задач; їх використання в різних областях людського життя.

2 ЙМОВІРНІСНИЙ ВИСНОВОК

Байєсівську мережу можна розглядати як ймовірнісну експертну систему, в якій база знань представлена топологією мережі і таблицею умовних ймовірностей кожної вершини. Головна мета побудови бази знань – це її використання для побудови висновків, тобто знаходження відповідей на запити.

Мета імовірнісного висновку полягає в тому, щоб знайти $P(X|E)$ – апостеріорні ймовірності необхідних вершин X при деякому значенні спостережуваних вершин E . Найпростішим випадком ймовірнісного висновку можна вважати той випадок, коли шукана вершина тільки одна [22].

Залежно від розміру розв’язуваних проблем можна виділити два класи ймовірнісного висновку: точний та апроксимаційний. При вирішенні реальних великих завдань застосування точного ймовірнісного висновку стає неможливим через велику обчислювальну складність, і саме тоді використовуються методи апроксимації, які приблизно виконують обчислення.

2.1 Алгоритми точного висновку

Алгоритми точного ймовірнісного висновку включають наступні алгоритми:

- алгоритм Перла розповсюдження повідомлення для однозв’язних мереж – ідея алгоритму полягає в тому, що нова порція спостережень розглядається як збурення, що поширюється мережею шляхом відправки повідомлень між сусідніми вершинами. Інформація щодо стану батьківської вершини передається через пі-повідомлення, а про стан нащадка – лямбда-повідомленням (рисунок 2.1).

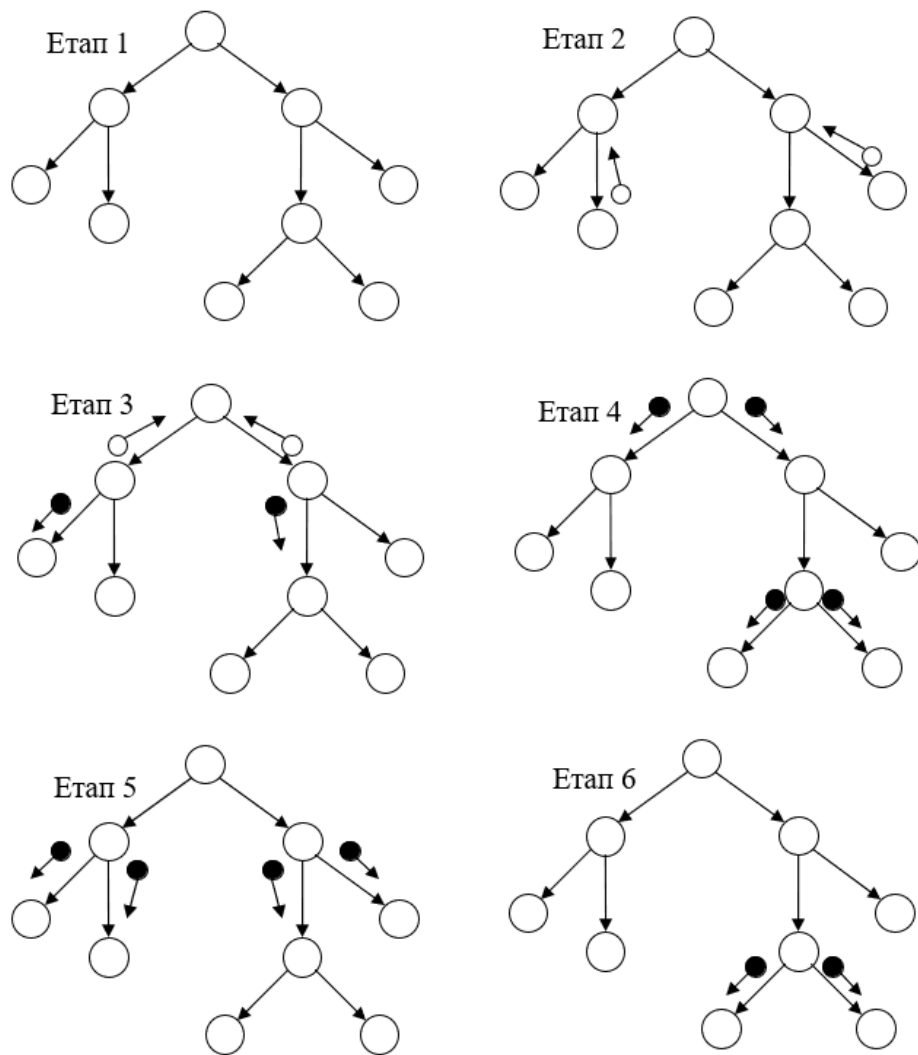


Рисунок 2.1 – Етапи розсилання повідомлень

До недоліків цього алгоритму відноситься той факт, що чим більше вершин в дереві, тим більше складність алгоритму і, отже, вимагає більше часу і ресурсів для його виконання;

- алгоритм визначеного перетину – ідея цього алгоритму полягає в тому, щоб змінити структуру багатозв'язної мережі на декілька однозв'язних, шляхом інстанціювання вершин, які входять в перетин [18]. Складність цього алгоритму залежить від кількості інстанціювань, що необхідно розглянути і росте експоненційно. Ось чому звести до мінімуму розмір певного перетину є важливим завданням, але ця задача є NP-складною і цей момент можна віднести до недоліків даного алгоритму. До плюсів даного алгоритму можна

віднести те, що завдяки зміні структури мережі спрощується формула обчислення сукупного ймовірнісного розподілу та вона буде мати наступний вигляд:

$$P(A, B, C, D) = P(A = A_1, B, C, D) * P(A = A_1) + P(A = A_2, B, C, D) * P(A = A_2). \quad (2.1)$$

- алгоритм виключення змінних – основна ідея цього алгоритму полягає в тому, щоб обчислити ймовірність вершини за формулою, заснованою на формулі розкладання загального розподілу ймовірностей мережі [20]. Складність алгоритму залежить від кількості операцій додавання і множення, що, в свою чергу, залежить від порядку виключення змінних. Однак задача знаходження оптимального порядку виключення є NP-складною. Рисунок 2.2 показує поетапне виключення вершин, щоб визначити ймовірність останньої вершини;

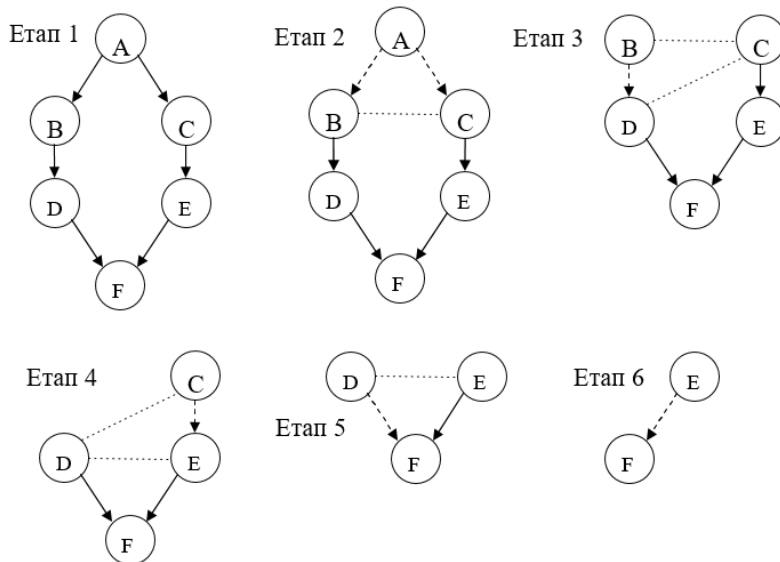


Рисунок 2.2 – Робота алгоритму виключення змінних

- алгоритми кластеризації – в 1988 Lauritzen та Spiegelhalter запропонували так званий LS метод ймовірнісного висновку [13]. Згодом в 1990 році Дженсен запропонував його модифікацію, яка отримала назву Hugin,

за назвою програми, де вона була впроваджен. У 1990 році Shenoy і Shafer запропонували власну архітектуру, а в 1997 році модифікували її для поліпшення обчислювальної ефективності [5]. Використання так званих об'єднаних дерев в цих алгоритмах дозволяє використати ідею обміну повідомленнями ймовірнісного висновку Перла;

- інші алгоритми – до цих алгоритмів можна віднести алгоритм символічного висновку, алгоритм реверсування та диференційний підхід.

2.2 Апроксимаційні алгоритми

Відомі випадки, коли формування ймовірнісного висновку з використанням точних методів стає неможливим через високу обчислювальну складність. З цієї причини були розроблені алгоритми апроксимації, які дозволяють вирішити цю проблему, жертвуючи точністю обчислень. [12].

Апроксимаційні алгоритми поділяють на:

- алгоритми стохастичної вибірки – вони створюють множину випадково обраних подій або інстанціювань в мережі відповідно до таблиці умовних ймовірностей моделі і потім апроксимують ймовірності шуканих змінних на основі частоти виникнення подій у вибірці. Точність методів залежить від розміру вибірки, а не від структури мережі;

- алгоритми неповного висновку – їх ще називають методами спрощення моделі. Їх ідея полягає в тому, щоб спростити мережу до тієї, де можна застосовувати методи точного висновку [17];

- варіаційні алгоритми – в основі варіаційних алгоритмів лежить ідея усереднення значень ймовірностей вершин, тобто в розрахунку враховуються тільки значущі вершини. Мережа перетворюється в підграф початкового графу, в якому деякі вершини позбавляються зв'язків до тих пір, поки не стане можливим застосувати точний ймовірнісний алгоритм;

- пошукові алгоритми – основою пошукових алгоритмів є ідея переходу від задачі ймовірнісного висновку до оптимізаційної задачі пошуку найбільш

ймовірного значення. Вважається, що найбільш вірогідною частиною мережі є відносно невелика кількість вершин, і якщо шукаються найбільш ймовірні екземпляри, то можна отримати межі апостеріорних ймовірностей значень вершин. [14].

2.3 Види архітектур алгоритмів кластеризації

2.3.1 Архітектура Lauritzen-Spiegelhalter

В архітектурі LS спочатку створюється зв'язне дерево, а потім в ньому поширюється повідомлення. Об'єднане дерево ми отримується з ациклічного ненаправленого графу.

Для перетворення мережі Байєса в моралізований граф необхідно ввести додаткове ребро між кожною батьківською парою, а потім замінити направлений зв'язок між вершинами ненаправлений зв'язком. Потім отриманий граф повинен бути триангульований, тобто необхідно зробити його таким, щоб для будь-якого його простого циклу довжиною більше трьох, існувало ребро, яке з'єднує дві незв'язані вершини цього циклу.

Після триангуляції графу можна впорядкувати його кліки.

Зв'язне дерево – це таке дерево, де вершини якого є підмножинами змінних і змінна, яка міститься у двох різних вершинах, повинна бути в кожній вершині на шляху між цим двома вершинами. Об'єднаним деревом називається таке зв'язне дерево, вершинами якого є кліки раніше триангульованого графу. Потім ми вводимо між кліками ребра, що містять сепаратори – перетини множин вершин цих клік, видаляємо зайві і вибираємо корінь дерева.

З кожною клікою ми будемо пов'язувати таблицю умовної ймовірності, яка містить змінні кліки.

Поширення повідомлення по дереву відбувається в два етапи – сходження догори та донизу [7].

Правила сходження догори:

- кожен нащадок чекає, щоб відправити повідомлення своїм батькам, поки не отримає повідомлення від всіх своїх нащадків. Якщо їх у нього немає, то він може негайно відправляти повідомлення;

- коли нащадок готовий відправити повідомлення, він формує його маргіналізуючи свою поточну таблицю по сепаратору – підсумовуючи змінні, яких нема в сепараторі [15]. Після того, як він надіслав повідомлення батькам, він ділить свою таблицю на це повідомлення;

- коли батько отримує повідомлення, він множить свою таблицю на нього.

Сходження догори закінчується, коли корінь дерева отримав повідомлення від усіх своїх нащадків.

Правила сходження донизу:

- кожен батько чекає, щоб відправити повідомлення своїм нащадкам, поки сам не отримає повідомлення від своїх батьків. Корінь дерева не має батьків, тому повідомлення відправляється негайно;

- коли батько готовий надсилати повідомлення, він формує його шляхом маргіналізації своєї поточної таблиці;

- коли нащадок отримує повідомлення, він множить свою таблицю на нього.

Сходження донизу закінчується, коли всі листя дерева отримали повідомлення від своїх батьків. Після цього можна отримати значення кожної змінної з таблиці кожної кліки.

2.3.2 Архітектура Nugin

Нехай мається об'єднане дерево і відповідні таблиці для кожної кліки. Сепаратор кожного ребра дерева також містить відповідні таблиці. Процес пропагації також відбувається в два етапи – сходження догори та донизу. В архітектурі Nugin на етапі сходження догори відправник не ділить свою

таблицю на повідомлення, а замість цього записує його в сепаратор. Це економить підрахунки, але й потребує більшого об'єму пам'яті. На етапі сходження донизу сепаратор ділить нове повідомлення на те, яке він раніше зберігав і саме на це відношення множить свою таблицю отримувач повідомлення. Економія обчислень відбувається завдяки діленню таблиць сепараторів, що мають менший розмір, ніж таблиці клік.

Правила пропagaції:

- кожна некоренева вершина чекає відправки повідомлення, поки сама не отримає повідомлення;

- корінь дерева чекає відправки повідомлення, поки не отримає повідомлення від усіх нащадків;

- коли вершина готова до відправки повідомлення, вона формує його маргіналізуючи свою поточну таблицю по сепаратору і тоді відправляє це повідомлення сепаратору між нею та отримувачем;

- коли сепаратор отримує нове повідомлення від одного зі своїх сусідів, він ділить його на свою поточну, стару таблицю, відправляє результат іншому сусідові і потім замінює старе значення на нове;

- коли вершина отримує повідомлення, вона множить свою таблицю на нього.

Після завершення процесу пропagaції значення окремої змінної можна отримати з таблиці сепаратора, який її містить, раціональніше – найменшого.

Якщо ж жоден сепаратор не містить необхідної змінної, то її значення обчислюється з таблиці кліки.

2.3.3 Архітектура Shenoy-Shafer

В архітектурі Shenoy-Shafer існує набір таблиць умовних ймовірностей, які визначають сукупний розподіл. Змінні таблиць формують гіперграф H , до якого додаються підмножини, значення яких треба знайти.

На першому кроці формується з підмножин гіперграфу H бінарне зв'язне дерево. Бінарне зв'язне дерево – це зв'язне дерево, кожна вершина якого має не більше трьох сусідів (рисунок 2.3).

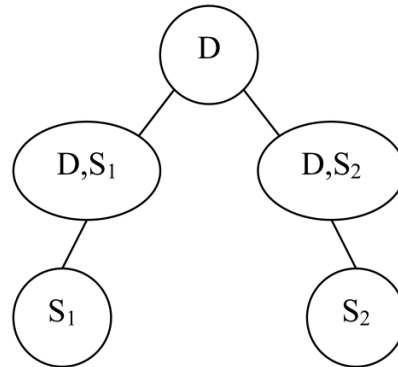


Рисунок 2.3 – Бінарне зв'язне дерево

Кожній вершині дерева присвоюється відповідна таблиця. Вершина, значення якої потрібно знайти, запрошує повідомлення від своїх сусідів.

Правила пропagaції мають наступний вигляд [8]:

- розрахунок повідомлення – припустимо, що r і s сусіди і s надсилає запит на повідомлення до r ; r , в свою чергу, надсилає запити на повідомлення від своїх інших сусідів і після отримання цих повідомлень, від розраховує повідомлення s наступним чином. Фактично, повідомлення, яке r надсилає своєму сусіду s являє собою комбінацію усіх повідомлень від інших сусідів, маргіналізованих по $r \cap s$. Лист у дереві має лише одного сусіда, тому коли він отримує запит на повідомлення, від може відправити його негайно;

- розрахунок шуканого значення вершини – коли вершина s отримала повідомлення від усіх сусідів і об'єднала їх власною таблицею, можна знайти шукане значення.

Кожна вершина в двійковому зв'язному дереві може зберігати нуль, одну, дві або більше таблиць – одну для кожної вхідної таблиці і одну для вихідної. Кожен сепаратор може зберігати не більше двох таблиць для двох повідомлень в обох напрямках.

2.4 Висновки по другому розділу

В даному розділі розглянута задача побудови ймовірнісного висновку, що є складною з обчислювальної точки зору і неоднозначною. Було зроблено огляд на існуючі алгоритми ймовірнісного висновку, які можна розділити на дві великі групи в залежності від розміру розв'язуваних задач: точного та апроксимаційного висновку. Необхідність розробки та застосування алгоритмів апроксимації пов'язана з проблемою побудови ймовірнісного висновку в великих мережах, де точністю обчислень можна поступитись для меншого об'єму обчислень.

Була розглянута група алгоритмів кластеризації і показано, що при реалізації ймовірнісного висновку необхідно спочатку привести структуру МБ до вигляду зв'язного дерева, зв'язати умовні таблиці ймовірностей первинної мережевої структури з вершинами цього дерева, а потім використовувати в ньому алгоритми пропагації, що ґрунтуються на проходженні повідомлень по дереву в обох напрямках та послідовному перерахуванню таблиць умовних ймовірностей вершин дерева.

3 LS-МЕТОД ЙМОВІРНІСНОГО ВИСНОВКУ

Ідея Lauritzen-Spiegelhalter методу ймовірнісного висновку є основоположною ідеєю методів кластеризації, де для реалізації ймовірнісного висновку необхідно спочатку привести структуру мережі Байєса до вигляду об'єднаного дерева, а потім використовувати алгоритм розповсюдження повідомлень по дереву догори та донизу і послідовно перераховувати таблиці умовних ймовірностей вершин дерева [19].

У загальному вигляді LS-метод передбачає виконання двох етапів. На першому етапі виконується побудова об'єднаного дерева клік з первинної структури мережі та заповнення вершин цього дерева таблицями умовних ймовірностей мережі. На другому етапі обчислюються значення ймовірностей станів вершин на основі алгоритмів розповсюдження значень ймовірності по об'єднаному дереву [6].

Загалом, метою дослідження є доведення ефективності даного методу серед інших архітектур ймовірнісного висновку.

3.1 Побудова об'єднаного дерева

Перший етап складається з декількох підетапів, які необхідно виконати, щоб сформулювати ймовірнісний висновок:

- моралізація графа – всі вершини мережі, у яких є батьки послідовно перебираються [15]. Якщо батьки вершини не пов'язані один з одним, то між ними вводиться зв'язок, так званий сусід;
- приведення графу до ненаправленого виду;
- триангуляція графа [15];
- побудова дерева суміжності – послідовний перебір всіх клік, отриманих на стадії триангуляції;
- побудова об'єднаного дерева – всі кліки послідовно зв'язуються одна

з одною, обираючи для зв'язку ребра з найбільшими сепараторами, видаляючи інші ребра;

- заповнення об'єднаного дерева таблицями – починається з листя дерева і послідовно проходить через всі кліки. Проводиться розгляд невідмічених вершин в ще необробленій кліці.

Для зручності коренем в об'єднаному дереві обирається кліка з найбільшим числом вершин. Якщо така вершина не одна, то обирається кліка, в якій є найбільша кількість ребер.

Процес триангуляції графа включає в себе:

- перевірку, чи знаходяться сусіди аналізованої вершини поруч один з одним. Якщо так, то така вершина вважається простою, або симпліціальною. Така вершина разом зі своїми сусідами утворює кліку та виключається з розгляду разом з її ребрами;

- якщо після перебору всіх вершин, мережа, що залишилася до розгляду, не порожня, то шукається вершина з найбільшим числом сусідів і така вершина стає симпліціальною шляхом введення додаткових ребер між її несуміжними сусідами, а потім ця вершина виключається з розгляду. Якщо вершин більше немає, то граф триангульований;

- після розгляду усієї мережі до первісного моралізованого ненаправленого графа додаються додаткові ребра і такий граф вважається триангульованим.

Побудова дерева суміжності складається з:

- перевірки, чи є кліка підмножиною інших неперебраних клік. Якщо являється, то така кліка знищується;

- якщо в кліці і інших неперебраних кліках збігається хоча б одна вершина, то між відповідними кліками вводиться ребро, що містить сепаратор-перетин множин вершин цих клік.

Процес заповнення об'єднаного дерева таблицями:

- якщо серед невідмічених вершин є така, яка не зустрічається в інших необроблених кліках, то таблиця повинна мати вигляд P (поточна

вершина | інші вершини в кліці) – у такому разі ця вершина відмічається і кліка вважається обробленою;

- якщо в первинній МБ такої таблиці немає або така вершина не одна, то використовується таблиця сукупного розподілу ймовірності – у такому разі усі вершини кліки відмічені і кліка оброблена;

- якщо в кліці є вже відмічені вершини і всього одна невідмічена вершина, то використовується таблиця виду $P(\text{невідмічена вершина})$ з первинної МБ – у такому разі ця вершина відмічена і кліка вважається обробленою;

- якщо кліка містить вже відмічені вершини, та кілька невідмічених вершин, то використовується таблиця сукупного розподілу ймовірності невідмічених вершин, ці вершини відмічаються, а кліка вважається обробленою;

- якщо в необробленій кліці усі вершини вже відмічені, то кліка заповнюється таблицею виду $P(\text{будь-яка вершина})$ і вважається обробленою.

3.2 Алгоритм пропагації

Для кожного спостереження змінної вибирається таблиця, що містить цю змінну. Задаються нульовими усі входження, які суперечать спостереженню [10].

Процес сходження догори – кожен лист дерева посилає повідомлення своєму батьку. Повідомлення є результатом маргіналізації підсумовування змінних таблиці, які не включені до сепаратора. Після того, як віправник відіслав повідомлення, він ділить свою поточну таблицю умовних ймовірностей на це повідомлення. Коли одержувач отримує повідомлення, він множить його на свою таблицю умовних ймовірностей і отримує нову таблицю. В свою чергу він також відправляє повідомлення своєму батьку після отримання всіх повідомлень від своїх нащадків. Процес триває до тих пір, поки корінь не отримає повідомлення від усіх своїх нащадків [16].

Процес сходження донизу – корінь посилає повідомлення кожному зі своїх нащадків. Він ділить свою таблицю умовних ймовірностей на повідомлення, що отримав від нащадка, проводить маргіналізацію таблиці за сепаратором і посилає результат. Коли нащадок отримує повідомлення від свого батька, він множить його на свою поточну таблицю умовних ймовірностей, утворюючи таким чином свою нову таблицю. Далі він маргіналізує її по сепаратору і відправляє своєму нащадку. Процес триває доти, поки усі листя не отримають повідомлення. Таким чином, таблиці результатів будуть повторно перераховані для кожної змінної, за умови, що є спостереження, які потім повинні бути нормалізовані.

Процес розрахунку ймовірностей станів вершин – для кожної вершини виконується пошук клік, в яких вона міститься. Якщо серед них є кліка, яка не є листом, то вибирається ця кліка. В іншому випадку – будь-яка з них. Для того, щоб визначити ймовірність кожного стану вершини необхідно підсумувати усі значення ймовірностей для цього конкретного стану в таблиці кліці.

3.3 Приклад побудови ймовірнісного висновку

Як приклад можна розглянути оцінювання кредитоспроможності осіб для отримання кредиту у банку. В якості змінних процесу будуть використовуватись такі атрибути:

- T – вид контракту (англ., type of contract);
- A – вік (англ., age);
- G – стать (англ., gender);
- M – сімейний стан (англ., marital status);
- S – сума кредиту (англ., sum of credit);
- P – наявність поручителя (англ., personal guarantor);
- R – результат (англ., result).

На рисунку 3.1 приведена отримана структура мережі Байєса, а в таблицях 3.1-3.7 – значення умовних ймовірностей вершин даної мережі.

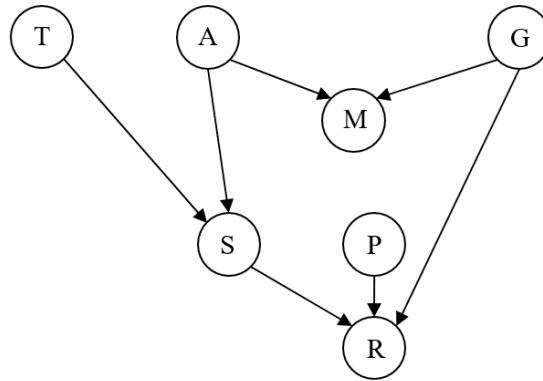


Рисунок 3.1 – Структура мережі Байєса

Таблиця 3.1 – Значення умовної ймовірності вершини Т

Стан	Ймовірність
1	2
T1 – для робочого	0,875
T2 – для пенсіонера	0,061
T3 – для підприємця	0,064

Таблиця 3.2 – Значення умовної ймовірності вершини А

Стан	Ймовірність
1	2
A1 – старше 40 років	0,45
A2 – молодше 40 років	0,55

Таблиця 3.3 – Значення умовної ймовірності вершини G

Стан	Ймовірність
1	2
G1 – жінка	0,47
G2 – чоловік	0,53

Таблиця 3.4 – Значення умовної ймовірності вершини Р

Стан	Ймовірність
1	2
P1 – нема поручителя	0,44
P2 – є поручитель	0,56

Таблиця 3.5 – Значення умовної ймовірності вершини М

Батьки	Стани батьків			
1	2	3	4	5
A	A1	A2	A1	A2
G	G1	G1	G2	G2
Стан	Ймовірність			
M1 – цивільний шлюб	0,0361	0,0841	0,0321	0,0903
M2 – розлучений	0,1996	0,0771	0,0681	0,0819
M3 – одружений	0,5848	0,5455	0,8517	0,1218
M4 – неодружений	0,0792	0,2853	0,0341	0,706
M5 – вдова	0,1003	0,008	0,014	0

Таблиця 3.6 – Значення умовної ймовірності вершини S

Батьки	Стани батьків					
1	2	3	4	5	6	7
T	T1	T1	T2	T2	T3	T3
A	A1	A2	A1	A2	A1	A2
Стан	Ймовірність					
S1 – більше 10000 грн	0,55	0,6	0,63	0,8	0,4	0,46
S2 – менше 10000 грн	0,45	0,4	0,37	0,2	0,6	0,54

Таблиця 3.7 – Значення умовної ймовірності вершини R

Батьки	Стани батьків							
	2	3	4	5	6	7	8	9
G	G1	G1	G1	G1	G2	G2	G2	G2
P	P1	P1	P2	P2	P1	P1	P2	P2
S	S1	S2	S1	S2	S1	S2	S1	S2
Стан	Ймовірність							
R1 – кредит схвалено	0,96	0,91	0,99	0,99	0,86	0,76	0,985	0,9659
R2 –кредит не схвалено	0,04	0,09	0,01	0,01	0,14	0,24	0,015	0,0341

Далі проводимо процедури моралізації (рисунок 3.2), приведення до ненаправленості (рисунок 3.3) та триангуляції графу.

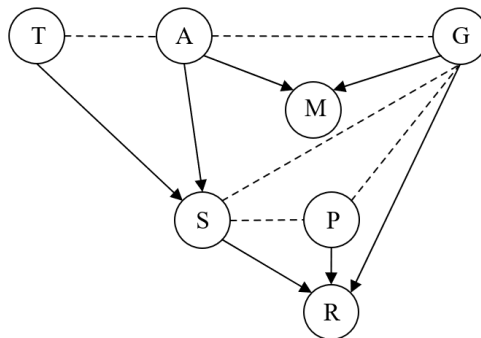


Рисунок 3.2 – Процес моралізації графу

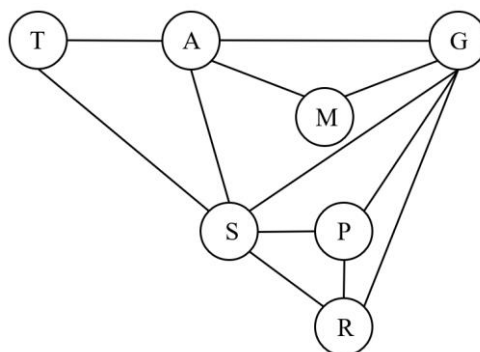


Рисунок 3.3 – Готовий ненаправлений граф

На етапі триангуляції виділяються симпліціальні вершини, потім вони видаляються з графу разом зі своїми ребрами. В результаті отримуємо триангульований граф (рисунок 3.4 – 3.6).

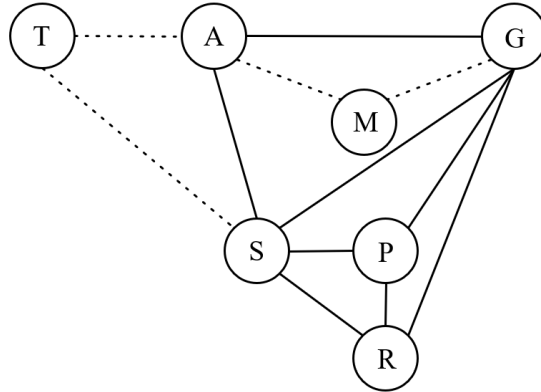


Рисунок 3.4 – Триангуляція графа. Виключення вершин T, M

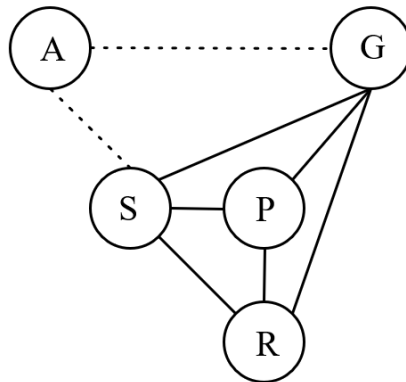


Рисунок 3.5 – Триангуляція графа. Виключення вершини A

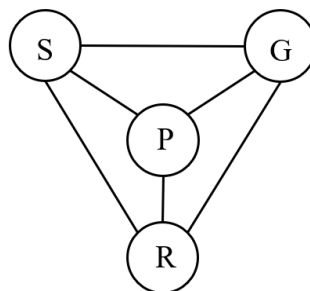


Рисунок 3.6 – Триангульований граф

Наступний етап: побудова дерева клік та об'єднаного дерева (рисунок 3.7 – 3.8).

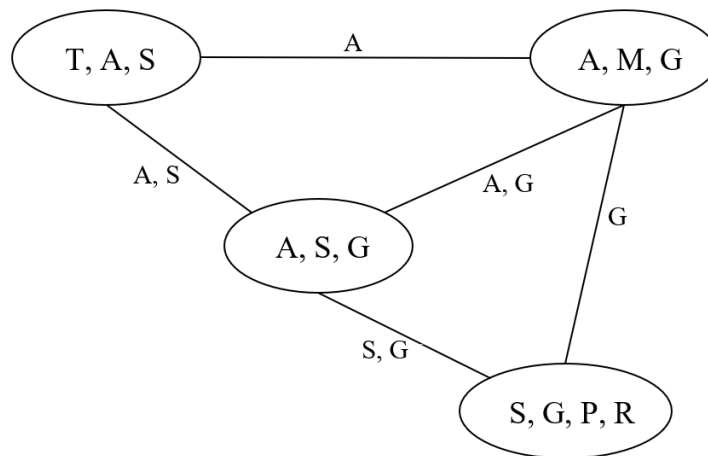


Рисунок 3.7 – Дерево клік

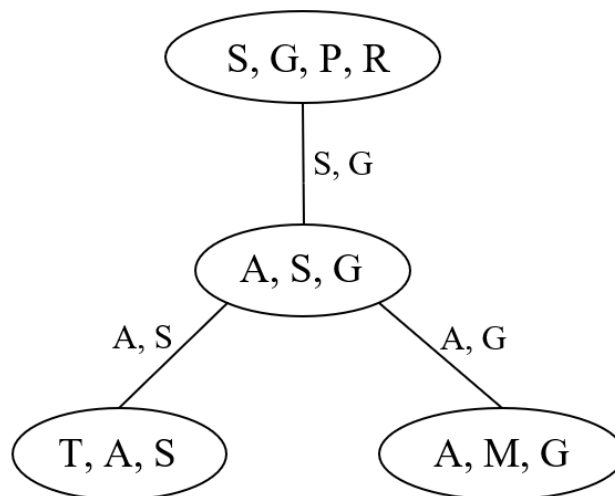


Рисунок 3.8 – Об'єднане дерево

У даній кліці маємо одну невідмічену вершину, яка більше не зустрічається в інших кліках – це вершина Т. Так як таблиці $P(T|S,A)$ не існує в структурі мережі, тому необхідно використовувати таблицю сукупного розподілу ймовірностей (таблиця 3.8).

Таблиця 3.8 – Лист «Т, А, S»

S1	S2	T	A
1	2	3	4
0,2165	0,1771	T1	A1
0,2887	0,1925	T1	A2
0,0172	0,0101	T2	A1
0,0268	0,0067	T2	A2
0,0115	0,0172	T3	A1
0,0161	0,019	T3	A2

У даній кліці маємо також одну вершину, яка більше не зустрічається в інших кліках – вершину M. Таблиця вигляду $P(M|A,G)$ існує і тому ми використовуємо її (таблиця 3.9).

Таблиця 3.9 – Лист «А, М, G»

M1	M2	M3	M4	M5	A	G
1	2	3	4	5	6	7
0,0361	0,1996	0,5848	0,0792	0,1003	A1	G1
0,0321	0,0681	0,8517	0,0341	0,014	A1	G2
0,0841	0,0771	0,5455	0,2853	0,008	A2	G1
0,0903	0,0819	0,1218	0,706	0	A2	G2

У даній кліці знаходяться дві відмічені вершини – вершина A та вершина S і одна невідмічена – вершина G, тому використовується таблиця вигляду $P(G)$ (таблиця 3.10).

Таблиця 3.10 – Лист «A, S, G»

S1	S2	A	G
1	2	3	4
0,47	0,47	A1	G1
0,53	0,53	A1	G2
0,47	0,47	A2	G1
0,53	0,53	A2	G2

У даній кліці є дві невідмічені вершини, яких немає в необроблених кліках – це вершини P та R, тому використовується таблиця сукупного розподілу ймовірностей (таблиця 3.11).

Таблиця 3.11 – Корінь «R, S, P, G»

R1	R2	S	P	G
1	2	3	4	5
0,1145	0,0047	S1	P1	G1
0,1157	0,0188	S1	P1	G2
0,1503	0,0015	S1	P2	G1
0,1686	0,0025	S1	P2	G2
0,0796	0,0078	S2	P1	G1
0,0749	0,0236	S2	P1	G2
0,1102	0,0011	S2	P2	G1
0,1212	0,0042	S2	P2	G2

Другий етап побудови ймовірнісного висновку: нехай інстанційованими вершинами будуть $P=P1$, $G=G2$. Для спостереження $P=P1$ заповнюються нулями входження $P=P2$ в корені об'єднаного дерева (таблиця 3.12).

Таблиця 3.12 – Значення кореня «R, S, P, G»

R1	R2	S	P	G
1	2	3	4	5
0,1145	0,0047	S1	P1	G1
0,1157	0,0188	S1	P1	G2
0	0	S1	P2	G1
0	0	S1	P2	G2
0,0796	0,0078	S2	P1	G1
0,0749	0,0236	S2	P1	G2
0	0	S2	P2	G1
0	0	S2	P2	G2

Для спостереження $G=G2$, нулями заповнюються входження $G=G1$ в кліці «A, S, G» (таблиця 3.13).

Таблиця 3.13 – Значення листа «A, S, G»

S1	S2	A	G
1	2	3	4
0	0	A1	G1
0,53	0,53	A1	G2
0	0	A2	G1
0,53	0,53	A2	G2

Процес сходження догори – передаємо повідомлення від кліки «T, A, S» до кліки «A, S, G», тобто, проводимо маргіналізацію по A та S (таблиця 3.14) та отримуємо нову таблицю кліки «T, A, S» (таблиця 3.15).

Таблиця 3.14 – Маргіналізація по А та S

S1	S2	A
1	2	3
0,2453	0,2046	A1
0,3317	0,2182	A2

Таблиця 3.15 – Нова таблиця кліки «Т, А, S»

S1	S2	T	A
1	2	3	4
0,8825	0,8659	T1	A1
0,8703	0,8821	T1	A2
0,0704	0,0496	T2	A1
0,0808	0,0307	T2	A2
0,0469	0,0844	T3	A1
0,0488	0,0871	T3	A2

Далі проводиться передача повідомлення від кліки «А, М, G» до кліки «А, S, G» (таблиця 3.16), отримуються нові таблиці кліки «А, М, G» та кліки «А, S, G» (таблиця 3.17 – 3.18). Проводиться передача повідомлення від кліки «А, S, G» до кореня «S, G, R, P» (таблиця 3.19) та отримуються нові таблиці кліки «А, S, G» та кореня «S, G, R, P» (таблиця 3.20 – 3.21).

Таблиця 3.16 – Маргіналізація по А та G

	A	G
1	2	3
1	A1	G1
1	A1	G2
1	A2	G1
1	A2	G2

Таблиця 3.17 – Нова таблиця кліки «А, М, G»

M1	M2	M3	M4	M5	A	G
1	2	3	4	5	6	7
0,0361	0,1996	0,5848	0,0792	0,1003	A1	G1
0,0321	0,0681	0,8517	0,0341	0,014	A1	G2
0,0841	0,0771	0,5455	0,2853	0,008	A2	G1
0,0903	0,0819	0,1218	0,706	0	A2	G2

Таблиця 3.18 – Нова таблиця кліки «А, S, G»

S1	S2	A	G
1	2	3	4
0	0	A1	G1
0,13	0,1084	A1	G2
0	0	A2	G1
0,1758	0,1156	A2	G2

Таблиця 3.19 – Маргіналізація по S та G

S1	S2	G
1	2	3
0	0	G1
0,3058	0,2241	G2

Таблиця 3.20 – Нова таблиця кліки «А, S, G»

S1	S2	A	G
1	2	3	4
0	0	A1	G1
0,4251	0,4839	A1	G2
0	0	A2	G1
0,5748	0,516	A2	G2

Таблиця 3.21 – Нова таблиця кореня «S, G, R, P»

R1	R2	S	P	G
1	2	3	4	5
0	0	S1	P1	G1
0,0353	0,0057	S1	P1	G2
0	0	S1	P2	G1
0	0	S1	P2	G2
0	0	S2	P1	G1
0,0168	0,0053	S2	P1	G2
0	0	S2	P2	G1
0	0	S2	P2	G2

Процес сходження донизу – являє собою зворотній процес для процесу сходження догори. Процес сходження донизу представлений у таблицях 3.22 – 3.27.

Таблиця 3.22 – Маргіналізація по S та G

S1	S2	G
1	2	3
0,1193	0,0874	G1
0,1345	0,0986	G2

Таблиця 3.23 – Нова таблиця кліки «A, S, G»

S1	S2	A	G
1	2	3	4
0	0	A1	G1
0,0572	0,40477	A1	G2
0	0	A2	G1
0,0773	0,0509	A2	G2

Таблиця 3.24 – Маргіналізація по А та G

	A	G
1	2	3
0	A1	G1
0,1049	A1	G2
0	A2	G1
0,1282	A2	G2

Таблиця 3.25 – Нова таблиця кліки «А, М, G»

M1	M2	M3	M4	M5	A	G
1	2	3	4	5	6	7
0	0	0	0	0	A1	G1
0,0033	0,0071	0,0893	0,0035	0,0014	A1	G2
0	0	0	0	0	A2	G1
0,0115	0,0105	0,0156	0,0905	0	A2	G2

Таблиця 3.26 – Маргіналізація по А та S

S1	S2	A
1	2	3
0,0572	0,0477	A1
0,0773	0,0509	A2

Таблиця 3.27 – Нова таблиця кліки «Т, А, S»

S1	S2	T	A
1	2	3	4
0,0504	0,0413	T1	A1
0,0673	0,0449	T1	A2
0,004	0,0023	T2	A1

Продовження таблиці 3.27

1	2	3	4
0,0062	0,0015	T2	A2
0,0026	0,004	T3	A1
0,0037	0,0044	T3	A2

Після виконання процедур сходження до гори та донизу отримано декілька таблиць для кожної вершини, які далі треба нормалізувати (таблиці 3.28 – 3.31)

Таблиця 3.28 – Нормалізована таблиця кліки «Т, А, S»

S1	S2	T	A
1	2	3	4
0,2252	0,1126	T1	A1
0,4061	0,1305	T1	A2
0,0179	0,0064	T2	A1
0,0377	0,0045	T2	A2
0,0119	0,0109	T3	A1
0,0227	0,0128	T3	A2

Таблиця 3.29 – Нормалізована таблиця кліки «А, М, G»

M1	M2	M3	M4	M5	A	G
1	2	3	4	5	6	7
0	0	0	0	0	A1	G1
0,0143	0,0303	0,3795	0,0151	0,0062	A1	G2
0	0	0	0	0	A2	G1
0,05	0,0453	0,0675	0,3913	0	A2	G2

Таблиця 3.30 – Нормалізована таблиця кліки «A, S, G»

S1	S2	A	G
1	2	3	4
0	0	A1	G1
0,2765	0,169	A1	G2
0	0	A2	G1
0,3739	0,1803	A2	G2

Таблиця 3.31 – Нова таблиця кореня «S, G, R, P»

R1	R2	S	P	G
1	2	3	4	5
0	0	S1	P1	G1
0,5594	0,091	S1	P1	G2
0	0	S1	P2	G1
0	0	S1	P2	G2
0	0	S2	P1	G1
0,2655	0,0838	S2	P1	G2
0	0	S2	P2	G1
0	0	S2	P2	G2

Далі проводиться розрахунок ймовірностей станів мережи Байєса: для вершини T беремо значення з кліки «T, A, S», для вершини A – з кліки «A, S, G», для вершини M – з кліки «A, M, G», для вершини S – з кліки «A, S, G» та для вершини R – з кореня дерева.

$$P(T1) = 0,2252 + 0,1126 + 0,4061 + 0,1305 = 0.8744;$$

$$P(T2) = 0,0179 + 0,0064 + 0,0377 + 0,0045 = 0,0665;$$

$$P(T3) = 0,0119 + 0,0109 + 0,0227 + 0,0128 = 0,0583;$$

$$P(A1) = 0,2765 + 0,169 = 0,4455;$$

$$P(A2) = 0,3739 + 0,1803 = 0,5542;$$

$$P(M1) = 0,0143 + 0,05 = 0,0643;$$

$$P(M2) = 0,0303 + 0,0453 = 0,0756;$$

$$P(M3) = 0,3795 + 0,0675 = 0,447;$$

$$P(M4) = 0,0151 + 0,3913 = 0,4064;$$

$$P(M5) = 0,0062;$$

$$P(S1) = 0,2765 + 0,3739 = 0,6504;$$

$$P(S2) = 0,169 + 0,1803 = 0,3493;$$

$$P(R1) = 0,5594 + 0,2655 = 0,8249;$$

$$P(R2) = 0,091 + 0,0838 = 0,1748$$

Для порівняння результатів обчислень, отриманих за допомогою LS-методу було прийняте рішення використати програму Netica. Для побудови ймовірнісного висновку в програмі використовується метод виключення зв'язних дерев.

Результати обчислень наведено у таблиці 3.32, результат роботи у програмі Netica – на рисунках 3.9 – 3.16.

Таблиця 3.32 – Порівняння результатів обчислень за допомогою LS-метода та програми Netica

Стан	Ймовірність	
	LS-метод	Програма Netica
1	2	3
T1	0,8744	0,875
T2	0,0665	0,061
T3	0,0583	0,064
A1	0,4455	0,45
A2	0,5542	0,55
G1	0	0
G2	1	1

Продовження таблиці 3.32

1	2	3
M1	0,0643	0,0641
M2	0,0756	0,0757
M3	0,447	0,45
M4	0,4064	0,404
M5	0,0062	0,0063
S1	0,6504	0,5770
S2	0,3493	0,423
P1	1	1
P2	0	0
R1	0,8249	0,818
R2	0,1748	0,182

Netica - [S Table (in Bayes net test1)]

File Edit Table Window Help

Node: R

Chance Probability

Apply OK

Reset Close

P	S	G	R1	R2
P1	S1	G1	.96	.04
P1	S1	G2	.86	.14
P1	S2	G1	.91	.09
P1	S2	G2	.76	.24
P2	S1	G1	.99	.01
P2	S1	G2	.985	.015
P2	S2	G1	.99	.01
P2	S2	G2	.9659	.0341

Рисунок 3.9 – Таблиця умовних ймовірностей вершини R

Netica - [T Table (in Bayes net test1)]

File Edit Table Window Help

Node: T

Chance Probability

T1	T2	T3
.875	.061	.064

Apply OK

Reset Close

Рисунок 3.10 – Таблица умовних ймовірностей вершини Т

Netica - [T Table (in Bayes net test1)]

File Edit Table Window Help

Node: S

Chance Probability

T A	S1	S2
T1 A1	.55	.45
T1 A2	.6	.4
T2 A1	.63	.37
T2 A2	.8	.2
T3 A1	.4	.6
T3 A2	.46	.54

Apply OK

Reset Close

Рисунок 3.11 – Таблица умовних ймовірностей вершини S

Netica - [R Table (in Bayes net test1)]

File Edit Table Window Help

Node: P

Chance Probability

P1	P2
.44	.56

Apply OK

Reset Close

Рисунок 3.12 – Таблица умовних ймовірностей вершини P

A G	M1	M2	M3	M4	M5
A1 G1	.0361	.1996	.5848	.0792	.1003
A1 G2	.0321	.0681	.8517	.0341	.014
A2 G1	.0841	.0771	.5455	.2853	.008
A2 G2	.0903	.0819	.1218	.706	0

Рисунок 3.13 – Таблица умовних ймовірностей вершини М

G1	G2
.47	.53

Рисунок 3.14 – Таблица умовних ймовірностей вершини G

A1	A2
.45	.55

Рисунок 3.15 – Таблица умовних ймовірностей вершини А

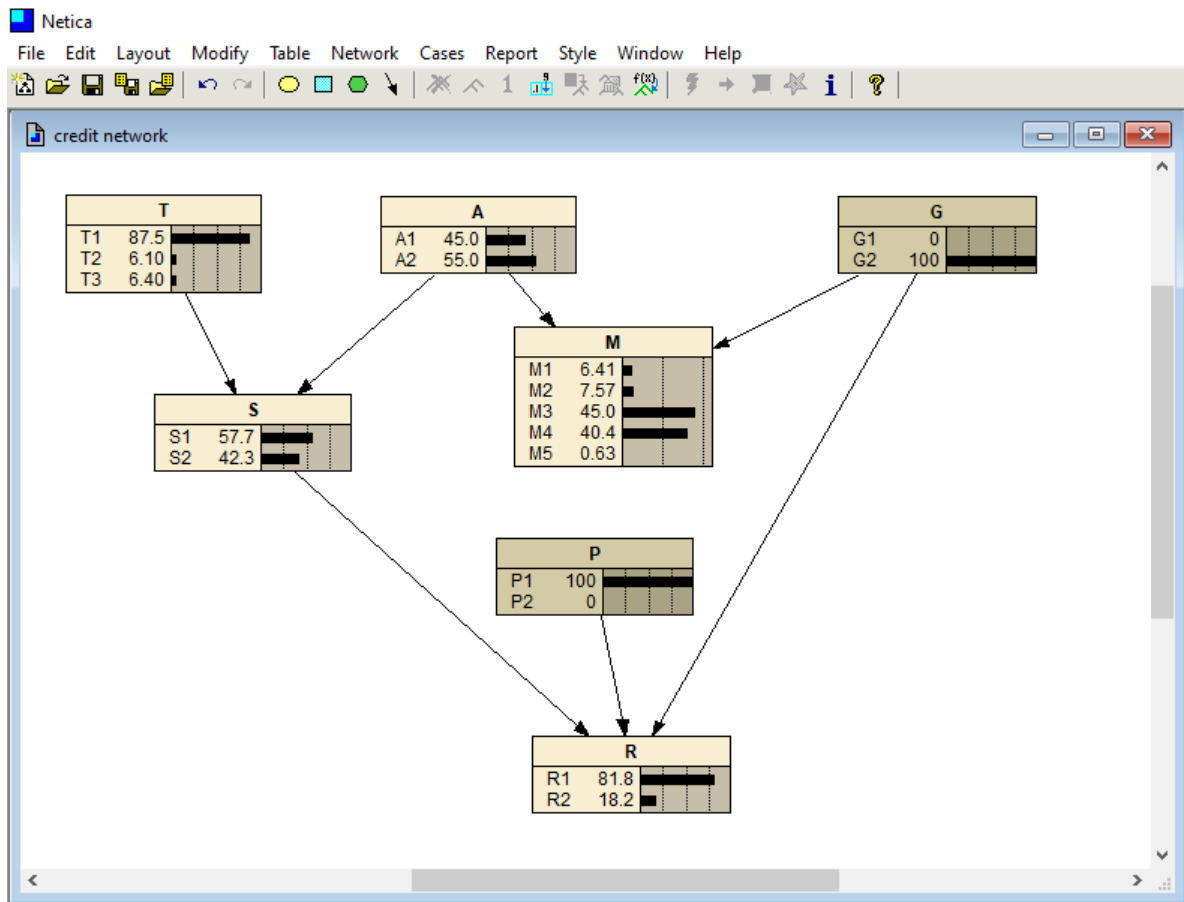


Рисунок 3.16 – Результат роботи в програмі Netica

3.4 Порівняння архітектур за необхідними об'ємами пам'яті

В архітектурі Lauritzen-Spiegelhalter, кожна кліка в об'єднаному дереві зберігає одну таблицю. Таким чином, загальний обсяг пам'яті залежить від кількості кліків в об'єднаному дереві і розмірності станів клік [13].

В архітектурі Hugin, кожна кліка в об'єднаному дереві і кожен сепаратор між двома сусідніми кліками зберігають таблицю.

В архітектурі Shenoy-Shafer кожна вершина може не зберігати ніякої таблиці, зберігати одну або дві і більше. Кожен сепаратор між двома сусідніми вершинами має одну або дві таблиці, залежно від того, які повідомлення потребуються. Якщо обидві сусідні вершини запрошують повідомлення одна від одної, то сепаратор має дві таблиці, а якщо повідомлення потрібне тільки одному з сусідів, то відповідно одну.

Спробуємо розрахувати необхідний об'єм пам'яті для виконання задачі розрахунку видачі кредиту для кожної з архітектур.

Для всіх трьох архітектур нам необхідний однаковий об'єм пам'яті для вхідних таблиць ($3+2+2+2+20+16 = 57$), для спостереження ($2+2 = 4$) та для вихідних таблиць ($3+2+2+5+2+2+2 = 18$).

Для архітектури Lauritzen-Spiegelhalter потрібно зберігати кліки, для яких потрібно ще виділити пам'ять ($16+8+12+20 = 56$).

Для архітектури Hugin також необхідно виділяти пам'ять на кліки, а ще на сепаратори ($16+8+12+20+4+4+4 = 68$).

В архітектурі Shenoу-Shafer використовується дуже багато сепараторів, що значно підвищує кількість виділяємої пам'яті, тому розрахунок її значення не є доцільним.

Отже, підсумовуючи, отримуємо наступні значення кількості пам'яті для архітектур:

- архітектура Lauritzen-Spiegelhalter: $57+4+18+56 = 135$;
- архітектура Hugin: $57+4+18+68 = 147$;
- архітектура Shenoу-Shafer – не розраховується.

Тому можна зробити висновок, що архітектура Lauritzen-Spiegelhalter є більш ефективною за необхідними об'ємами пам'яті, ніж інші представлені архітектури.

3.5 Висновки до третього розділу

В даному розділі було детально розглянуто LS-метод ймовірнісного висновку, приведений його поетапний алгоритм, та розглянутий приклад його реалізації в реальній задачі отримання кредиту.

Наведені результати роботи методу на даному прикладі та порівняння з результатами з програми Netica, свідчать про успішну роботу даного алгоритму.

ВИСНОВКИ

У роботі проводиться дослідження дискретних мереж Байєса, що можуть розглядатись як ймовірнісна експертна система, в яких база знань представляється топологією мережі і таблицею умовних ймовірностей кожної вершини; методів побудови ймовірнісного висновку в них і їх практичного застосування при вирішенні задач у реальному житті.

В першому розділі розглядалися основні поняття байєсівських мереж, було наведено їх формальний запис та структуру. Також були описані основні типи існуючих мереж і були розглянуті конкретні системи, що використовують мережі Байєса для вирішення проблем та задач.

У другому розділі була розглянута задача побудови ймовірнісного висновку, яка є складною з обчислювальної точки зору і неоднозначною. Було зроблено огляд існуючих алгоритмів ймовірнісного висновку, які за розміром вирішуваних задач можна поділити на дві великі групи: точного та апроксимаційного висновку. Групу точних алгоритмів кластеризації було розглянуто більш детально і було показано, що при реалізації ймовірнісного висновку необхідно спочатку привести структуру МБ до вигляду зв'язного дерева, зв'язати з вершинами цього дерева таблиці умовних ймовірностей первинної структури мережі, а потім використовувати в ньому алгоритми пропагації, що ґрунтуються на проходженні повідомлень по дереву в обох напрямках та послідовному перерахуванню таблиць умовних ймовірностей вершин дерева.

У третьому розділі був розписаний LS-метод ймовірнісного висновку, приведений його поетапний алгоритм та розглянуто приклад його реалізації в задачі отримання кредиту.

Також виконане порівняння з іншими методами кластеризації і на базі конкретних числових даних показано, що приведений LS-метод є найефективніший з точки зору використання об'ємів пам'яті.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Тулупьев А. Л. Байесовские сети: Логико-вероятностный подход [Текст] / Александр Львович Тулупьев, Сергей Игоревич Николенко, Александр Владимирович Сироткин. – СПб.: Наука, 2006. – 607с.
2. Бидюк П. И. Метод вероятностного вывода в байесовских сетях по обучающим данным [Текст] / Петр Иванович Бидюк, Александр Николаевич Терентьев. - Кибернетика и системный анализ. – 2007. – № 3. – С. 93-99.
3. Бидюк П. И. Построение и методы обучения байесовских сетей [Текст] / Петр Иванович Бидюк, Александр Николаевич Терентьев. - Таврический вестник информатики и математики. – Симферополь: КНЦ НАНУ, 2004. – № 2. – С. 139–153.
4. Згуровский М. З. Методы построения байесовских сетей на основе оценочных функций [Текст] / Михаил Захарович Згуровский, Петр Иванович Бидюк, Александр Николаевич Терентьев. - Кибернетика и системный анализ. – 2008. – № 2. – С. 81-88.
5. Jensen F. V., Olesen K. G. and Andersen S. K. An algebra of Bayesian belief universes for knowledge-based systems [Text] / Networks. – Wilmington: Wiley Periodicals Inc., 1990. – Vol. 20, № 5. – P. 637-659.
6. Lauritzen S. L. and Jensen F. V. Local computation with valuations from a commutative semigroup [Text] / Annals of Mathematics and Artificial Intelligence, – New York: Springer, 1997. – Vol. 21, № 1. – P. 51-69.
7. Pearl J. Fusion, propagation and structuring in belief networks [Text] / Artificial Intelligence, imprint: Elsevier, September 1986. – Vol. 29, №3. – P. 241–288.
8. Shenoy P. P. Binary joint trees for computing marginals in the ShenoyShafer architecture [Text] / International Journal of Approximate Reasoning, publisher: Elsevier, August 1997. – Vol. 17, № 2. – P. 239-263.

9. Shachter R. D. Evidence absorption and propagation through evidence reversals [Text] / Fifth Workshop on Uncertainty in Artificial Intelligence (UAI'89), University of Windsor, Ontario, Canada, 8–10 August, 1989. – Amsterdam: North-Holland, 1990. – P. 303-310.

10. Guo H. Probabilistic inference in Bayes networks [Text] / Report on research activities, Kansas State University, department of computing and information sciences, November 2000. – 27 p.

11. Lokeswarappa K. G. Junction trees: motivation [Text] / Seminar CSE 714 on advanced topics in machine learning, March 2005. – 57 p.

12. Terent'yev A. N., Bidyuk P. I., Korshevnyuk L. A. Bayesian network as instrument of intelligent data analysis [Text] / Journal of Automation and Information Sciences. – New York: Begell House Inc., 2007. – 39. – P. 28-38.

13. Lepar V. and Shenoy P. P. A comparison of Lauritzen-Spiegelhalter, Hugin, and Shenoy-Shafer architectures for computing marginals of probability distributions [Text] / Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI'98), Madison, Wisconsin, USA, 24–26 July, 1998. – SF.: Morgan Kaufmann, 1998. – P. 328-337.

14. Guo H. Real time Bayesian networks inference [Text] / Report on research activities, Kansas State University, department of computing and information sciences, March – November 2001. – 42 p.

15. Spiegelhalter D., Dawid P., Lauritzen S. and Cowell R. Bayesian analysis in expert systems [Text] / Statistical Science, 8. – 1993. – P.219–282.

16. Li Z. and D'Ambrosio B. Efficient Inference in Bayes Nets as a combinatorial optimization problem. [Text] / International Journal of Approximate Reasoning, publisher: Elsevier science. –1994. – Vol 11, № 1. – P. 55-81.

17. Castillo E., Gutiérrez J. M. and Hadi A. S. A new method for efficient symbolic propagation in discrete Bayesian networks. [Text] / Networks. – Wilmington: Wiley Periodicals Inc, 1996. – Vol. 28. – P. 31-43.

18. Shachter R. D. Evidence absorption and propagation through evidence reversals. [Text] / Fifth workshop on Uncertainty in Artificial Intelligence (UAI'89),

University of Windsor, Ontario, Canada, 8–10 August, 1989. – Amsterdam: North-Holland, 1990. – P. 303-310.

19. Darwiche A.. A differential approach to inference in Bayesian networks. [Text] / Journal of the association for computing machinery (JACM). – 2003 – Vol. 50, №3. – P. 280-305.

20. Sarkar S. Using tree-decomposable structures to approximate belief networks. [Text] / Proceedings of the ninth annual conference on Uncertainty in Artificial Intelligence (UAI'93), Washington, Columbia, USA, 9–11 July, 1993. – SF.: Morgan Kaufmann, 1993. – P. 376-382.

21. Wellman M. P. and Liu C. L. State-space abstraction for anytime evaluation of probabilistic networks. [Text] / Proceedings of the tenth annual conference on Uncertainty in Artificial Intelligence (UAI'94), Seattle, Washington, USA, 29–31 July, 1994. – SF.: Morgan Kaufmann, 1994. – P.567-574.

22. Ільїна І.В., Крікун А.О. Методи ймовірнісного висновку в мережах Байєса / Проблеми інформатизації: тези доповідей восьмої міжнародної конференції – Черкаси – Харків – Баку – Бельсько-Бяла – 2020; 26-27 листопада 2020 року; Том 2:секція 5, с.26