Міністерство освіти і науки України

Дніпровський національний університет імені Олеся Гончара

Я. С. Бондаренко, Д. О. Рачко, А. О. Розливан

ПОСІБНИК ДО ВИВЧЕННЯ ДИСЦИПЛІНИ “ІМОВІРНІСНІ ГРАФІЧНІ МОДЕЛІ” ЧАСТИНА 1. БАЙЄСІВСЬКА МЕРЕЖА

Дніпро

2019

УДК 519.21:004.032.26
Б81

Рецензенти: канд. фіз.-мат. наук, доц. М.Є. Ткаченко,
канд. фіз.-мат. наук, доц. Н.І. Послайко

Б81 Бондаренко Я. С. Посібник до вивчення дисципліни “Імовірнісні графічні моделі”. Частина 1. Байєсівська мережа [Текст] / Я.С. Бондаренко, Д.О. Рачко, А.О. Розливан. – Дніпро: Ліра, 2019. – 64 с.

Викладено теоретичні положення щодо пода́ння байєсівської мережі, формування ймовірнісного висновку та побудови моделей міркувань в байєсівській мережі.

Для студентів механіко-математичного факультету ДНУ спеціальності “Статистика”.

Рекомендовано до друку вченою радою
механіко-математичного факультету
Дніпровського національного університету імені Олеся Гончара
протокол №11 від 21.06.2019 року

Навчальне видання

Яна Сергіївна Бондаренко
Деніс Олексійович Рачко
Анастасія Олександрівна Розливан

Посібник до вивчення дисципліни
“Імовірнісні графічні моделі”
Частина 1. Байєсівська мережа

Друкується за авторською редакцією

Підписано до друку 05.07.2019. Формат 60×84/16. Папір друкарський. Друк плоский. Ум. друк. арк. 3,72. Тираж 20 пр. Зам. № 210.

Друкарня «Ліра», вул. Наукова, 5, м. Дніпро, 49107. Свідоцтво про внесення до Державного реєстру серія ДК №6042 від 26.02.2018 р.

ВСТУП

Для розуміння процесу прийняття рішень в умовах невизначеності розглянемо задачу, названу на честь ведучого американської телегри "Let′s Make a Deal" Монті Холла (Monty Hall, 1921–2017).

Уявіть, що ви стали учасником гри, в якій вам необхідно вибрати одну з трьох дверей. За однією з дверей знаходиться автомобіль, за двома іншими дверима – кози. Ви вибираєте одну з дверей, наприклад, номер 1, після цього ведучий, який знає, де знаходиться автомобіль, а де – кози, відкриває одну з решти дверей, наприклад, номер 3, за якою знаходиться коза. Після цього він запитує вас – чи не бажаєте ви змінити свій вибір і вибрати двері номер 2? Чи збільшаться ваші шанси виграти автомобіль, якщо ви приймете пропозицію ведучого і змініте свій вибір?

Учаснику гри заздалегідь відомі наступні правила: автомобіль рівноймовірно розміщений за будь-якою з трьох дверей; ведучий в будь-якому випадку зобов'язаний відкрити двері з козою (але не ту, яку вибрав гравець) і запропонувати гравцеві змінити свій вибір; якщо у ведучого є вибір, яку з двох дверей відкрити, він обирає будь-яку з них з однаковою ймовірністю.

Введемо позначення: H_1 – автомобіль знаходиться за дверима номер 1, H_2 – автомобіль знаходиться за дверима номер 2, H_3 – автомобіль знаходиться за дверима номер 3. Події H_1, H_2, H_3 утворюють повну групу подій. Апріорні ймовірності подій H_i дорівнюють

$$P(H_1) = P(H_2) = P(H_3) = 1/3.$$

Подія A – учасник обирає двері номер 1, ведучий відкриває двері номер 3. Обчислимо апостеріорні ймовірності подій $H_i, i = 1,2,3$, за умов, що подія A відбулася: $P(H_1|A), P(H_2|A), P(H_3|A)$.

Ведучий відкриває двері номер 2 або двері номер 3 з однаковою ймовірністю, якщо автомобіль знаходиться за дверима номер 1, тому $P(A|H_1) = 0,5$. Ведучий відкриває двері номер 3 з імовірністю 1, якщо

автомобіль знаходиться за дверима номер 2, тому $P(A|H_2) = 1$. Ведучий не відкриває двері номер 3, якщо автомобіль знаходиться за дверима номер 3, тому $P(A|H_3) = 0$.

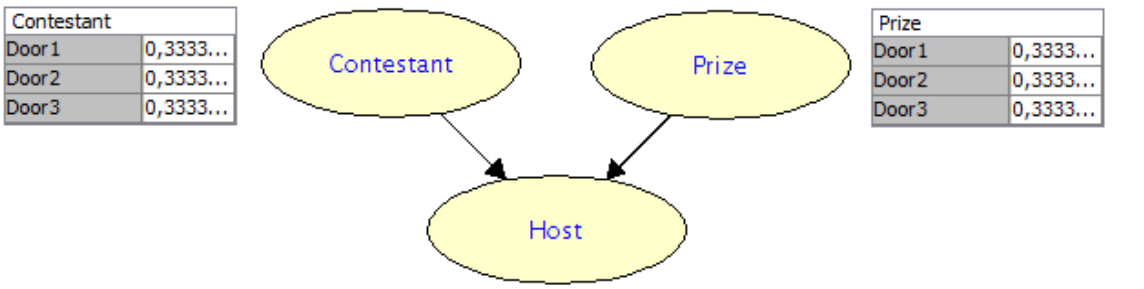
Апостеріорні ймовірності подій $H_i, i = 1,2,3$, дорівнюють

$$P(H_i|A) = \frac{P(A|H_i)P(H_i)}{P(A|H_1)P(H_1) + P(A|H_2)P(H_2) + P(A|H_3)P(H_3)},$$

$$P(H_1|A) = \frac{1}{3}, \qquad P(H_2|A) = \frac{2}{3}, \qquad P(H_3|A) = 0.$$

Шанси виграти автомобіль збільшаться, якщо прийняти пропозицію ведучого й зміните вибір, оскільки $P(H_2|A) > P(H_1|A)$.

Імовірнісною графічною моделлю задачі Монті Холла є байєсівська мережа подана орієнтованим ациклічним графом з вершинами $Contestant$, $Prize$, $Host$, ребра якого характеризують причинно-наслідкові зв'язки між вершинами. Імовірнісні розподіли вершин $Contestant$, $Prize$ та умовний імовірнісний розподіл вершини $Host$ наведено на рисунку 1.



Host

Contestant	Door1			Door2			Door3		
Prize	Door1	Door2	Door3	Door1	Door2	Door3	Door1	Door2	Door3
Door1	0,0	0,0	0,0	0,0	0,5	1,0	0,0	1,0	0,5
Door2	0,5	0,0	1,0	0,0	0,0	0,0	1,0	0,0	0,5
Door3	0,5	1,0	0,0	1,0	0,5	0,0	0,0	0,0	0,0

Рис. 1. Байєсівська мережа Monty Hall

Використовуючи байєсівську мережу можна моделювати різноманітні ситуації, тобто задавати вершинам мережі деякі значення станів і отримувати найбільш імовірні значення станів для інших вершин мережі.

Учасник обирає одну з дверей, наприклад, номер 1, після цього ведучий, який знає, де знаходиться автомобіль, а де – кози, відкриває одну з решти дверей, наприклад, номер 3, за якою знаходиться коза. Після цього він запитує – чи не бажаєте ви змінити свій вибір і вибрати двері номер 2? Шанси виграти автомобіль збільшаться, якщо прийняти пропозицію ведучого й зміните вибір (див. рис. 2).

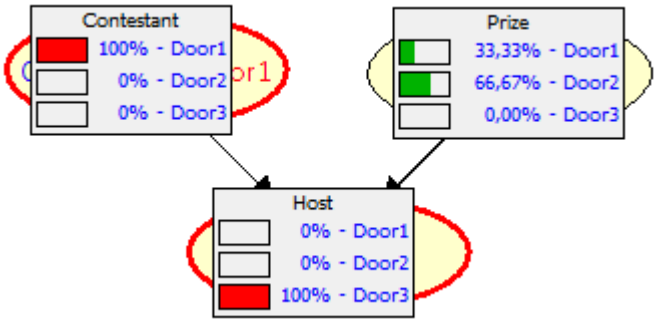


Рис. 2. Апостеріорний розподіл вершини $Prize$

Учасник обирає одну з дверей, наприклад, номер 1, після цього ведучий, який знає, де знаходиться автомобіль, а де – кози, відкриває одну з решти дверей, наприклад, номер 2, за якою знаходиться коза. Після цього він запитує – чи не бажаєте ви змінити свій вибір і вибрати двері номер 3? Шанси виграти автомобіль збільшаться, якщо прийняти пропозицію ведучого й зміните вибір (див. рис. 3).

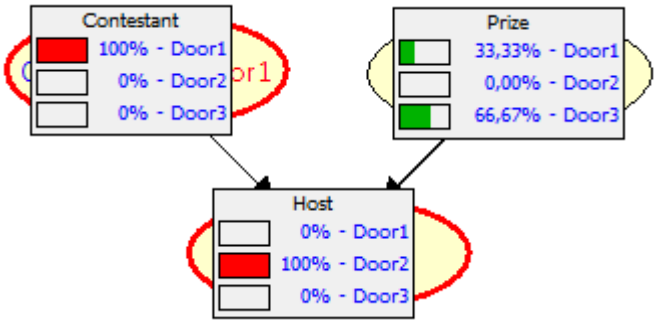


Рис. 3. Апостеріорний розподіл вершини $Prize$

Пода́ння байєсівської мережі, формування ймовірнісного висновку та побудова моделей міркувань в байєсівській мережі розглянуто в частині 1 посібника. Точні та наближені алгоритми формування ймовірнісного висновку в байєсівській мережі, вибір структури мережі розглядатимуться в наступних частинах навчального посібника “Імовірнісні графічні моделі”.

1. Основні поняття теорії ймовірностей

1.1. Умовна ймовірність

Означення. Трійку $\{\Omega, \mathcal{F}, P\}$, де Ω – простір елементарних подій; $\mathcal{F}$ – σ-алгебра підмножин простору Ω; P – імовірність на $\mathcal{F}$, називатимемо *імовірнісним простором*.

Імовірнісний простір $\{\Omega, \mathcal{F}, P\}$ є математичною моделлю стохастичного експерименту, при цьому Ω – простір елементарних подій – математична модель наслідків стохастичного експерименту; $\mathcal{F}$ – σ-алгебра підмножин простору Ω – математична модель алгебри подій стохастичного експерименту (множини з $\mathcal{F}$ ще називають подіями); P – імовірність на $\mathcal{F}$ – математична модель частоти події в послідовності експериментів.

Означення. Нехай $\{\Omega, \mathcal{F}, P\}$ – імовірнісний простір, $B \in \mathcal{F}$, $P(B) > 0$. *Умовною ймовірністю* $P(A|B)$ події A відносно події B називатимемо відношення

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

Теорема (формула множення). Для будь-яких подій A та $B \in \mathcal{F}$ таких, що $P(A) > 0, P(B) > 0$,

$$P(A \cap B) = P(A|B)P(B) = P(B|A)P(A).$$

Теорема допускає узагальнення.

Нехай $A_1, A_2, \dots, A_n$ – такі події, що $P(A_1 \cap A_2 \cap \dots \cap A_{n-1}) > 0$, тоді

$$P(A_n \cap A_{n-1} \cap \dots \cap A_2 \cap A_1) =$$

$$= P(A_n|A_{n-1} \cap \dots \cap A_2 \cap A_1)P(A_{n-1}|A_{n-2} \cap \dots \cap A_2 \cap A_1) \dots P(A_2|A_1)P(A_1)$$

Означення. Події $H_1, H_2, \dots, H_n$ утворюють *повну групу* подій, якщо

1) $\bigcup_{i=1}^{n} H_i = \Omega$ (хоча б одна подія в експерименті відбулася),
2) $H_i \cap H_j = \emptyset, i \neq j$ (події попарно несумісні).

Якщо подія A може відбутися в експерименті з однією з подій $H_1, H_2, \dots, H_n$, які утворюють повну групу подій, то ймовірність події A обчислюється за формулою повної ймовірності.

Теорема (формула повної ймовірності). Нехай $H_1, \dots, H_n$ – повна група подій, при цьому $P(H_i) > 0, i = 1,2, \dots, n$. Тоді для будь-якої події A

$$P(A) = \sum_{i=1}^{n} P(A|\, H_i)P(H_i).$$

Доведення. Скористаємося тим, що $\bigcup_{i=1}^{n} H_i = \Omega$, представимо подію A у вигляді

$$A = A \cap \Omega = A \bigcap \left(\bigcup_{i=1}^{n} H_i \right) = \bigcup_{i=1}^{n} (A \cap H_i).$$

Оскільки події H_i попарно несумісні $\left(H_i \cap H_j = \emptyset, i \neq j\right)$, то події $A \cap H_i$, $i = 1,2, \dots, n$, також попарно несумісні:

$$(A \cap H_i) \cap \left(A \cap H_j\right) = \emptyset, i \neq j.$$

Тому

$$P(A) = P\left(\bigcup_{i=1}^{n} (A \cap H_i) \right) = \sum_{i=1}^{n} P(A \cap H_i) = \sum_{i=1}^{n} P(A|\, H_i)P(H_i).$$

Якщо в результаті експерименту подія A відбулася й необхідно переоцінити ймовірності події $H_1, H_2, \dots, H_n$, які утворюють повну групу подій, з урахуванням результатів експерименту, то ймовірності подій $H_i, i = 1,2, \dots, n$, обчислюються за формулами Байєса.

Теорема (формули Байєса). Нехай $H_1, \dots, H_n$ – повна група подій, при цьому $P(H_i) > 0, i = 1,2, \dots, n$. Тоді для будь-якої події A $(P(A) > 0)$

$$P(H_i|A) = \frac{P(A|H_i)P(H_i)}{\sum_{k=1}^{n} P(A|\, H_k)P(H_k)}, i = 1,2, \dots, n.$$

Доведення.

$$P(H_i|A) = \frac{P(H_i \cap A)}{P(A)} = \frac{P(A|H_i)P(H_i)}{\sum_{k=1}^{n} P(A|\, H_k)P(H_k)}, i = 1,2, \dots, n.$$

Апріорні ймовірності $P(H_i)$ подій $H_i, i = 1,2, \dots, n$ відомі до проведення експерименту. Апостеріорні ймовірності $P(H_i|A)$ подій $H_i, i = 1,2, \dots, n$ обчислюються після проведення експерименту.

1.2. Незалежні події

Означення. Нехай $\{\Omega, \mathcal{F}, P\}$ – імовірнісний простір. Події $A, B \in \mathcal{F}$ називатимемо *незалежними*, якщо

$$P(A \cap B) = P(A)P(B).$$

Теорема. Події A та B незалежні ($P(B) > 0$) тоді й тільки тоді, коли

$$P(A|B) = P(A).$$

Доведення. Якщо події A та B незалежні, то

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

Якщо $P(A|B) = P(A)$, то

$$P(A \cap B) = P(A|B)P(B) = P(A)P(B).$$

Останнє означає, що події A та B незалежні.

Означення. Нехай $\{\Omega, \mathcal{F}, P\}$ – імовірнісний простір. Події $A, B \in \mathcal{F}$ називатимемо *умовно незалежними, якщо відомо, що подія $C \in \mathcal{F}$ відбулася*, якщо

$$P(A \cap B|C) = P(A|C)P(B|C).$$

Теорема. Події A та B умовно незалежні, якщо відомо, що подія C відбулася ($P(C) > 0, P(B|C) > 0$) тоді й тільки тоді, коли

$$P(A|B \cap C) = P(A|C).$$

Доведення. Розглянемо умовну ймовірність

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

Якщо події A, B умовно незалежні, якщо відомо, що подія C відбулася, то

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

Якщо $P(A|B \cap C) = P(A|C)$, то

$$P(A \cap B|C) = P(A|B \cap C)P(B|C) = P(A|C)P(B|C).$$

Останнє означає, що події A та B незалежні, якщо подія C відбулася.

Зазначимо, що з незалежності подій не випливає умовна незалежність подій, і навпаки.

1.3. Дискретна випадкова величина та її розподіл

Випадкова величина – це величина, яка набуває тих чи інших значень залежно від наслідку стохастичного експерименту. Наслідки стохастичного експерименту описуються точками ω множини наслідків Ω дискретного ймовірнісного простору $\{\Omega, P\}$, тому випадкову величину означаємо як функцію $X = X(\omega)$ на множині Ω.

Випадковою величиною на дискретному імовірнісному просторі $\{\Omega, P\}$ називатимемо функцію $X = X(\omega) = \big(X_1(\omega), X_2(\omega), \dots, X_n(\omega)\big)$ зі значеннями в R^n, задану на просторі елементарних подій Ω.

Якщо $n > 1$, то випадкову величину X називають *багатовимірною або випадковою зі значеннями в R^n (випадковим вектором),* зокрема, якщо $n = 2$ – *двовимірною,* якщо $n = 1$ – *одновимірною* або просто випадковою величиною.

Означення. Випадкова величина, яка набуває не більш ніж зліченне число значень, називається *дискретною* випадковою величиною.

Випадкова величина, задана на дискретному ймовірнісному просторі, завжди дискретна.

Означення. Нехай X – випадкова величина зі значеннями в R^1, задана на просторі елементарних подій Ω. Точку $x \in R^1$ називатимемо *можливим значенням* випадкової величини X, якщо $P(X = x) > 0$; множину можливих значень X будемо позначати через $Val(X)$.

Означення. *Розподілом* дискретної випадкової величини X зі значенням в R^1 називатимемо функцію

$$P_X: x \to P(X = x), x \in Val(X),$$

означену на множині $Val(X)$ різних можливих значень випадкової величини X, що ставить у відповідність кожному можливому значенню $x \in Val(X)$ імовірність $P(X = x)$, з якою X набуває це значення.

Розподіл дискретної випадкової величини X зі значенням в R^1 часто записують у вигляді таблиці, у верхньому рядку якої вказують різні

можливі значення випадкової величини, а в нижньому – ймовірності, з якими ці значення набуваються.

Означення. *Спільним розподілом* випадкових величин X, Y називають розподіл векторної випадкової величини (X, Y)

$$P\colon (x, y) \to P(X = x, Y = y), x \epsilon Val(X), y \epsilon Val(Y).$$

Теорема. Нехай $P(x, y)$ спільний розподіл випадкових величин X, Y. Тоді

$$P(X = x) = \sum_y P(x, y),\ P(Y = y) = \sum_x P(x, y),$$

Означення. *Умовним розподілом* випадкової величини X, якщо відомо, що випадкова величина Y набуває значення y називатимемо

$$P(X = x|Y = y) = \frac{P(X = x, Y = y)}{P(Y = y)}, x \epsilon Val(X), y \epsilon Val(Y).$$

Звідси спільний розподіл випадкових величин X, Y дорівнює

$$P(X = x, Y = y) = \ P(Y = y)P(X = x|Y = y).$$

Теорема Байєса для випадкових величин X, Y набуває вигляду:

$$P(X = x|Y = y) = \frac{P(Y = y|X = x)P(X = x)}{P(Y = y)} =$$

$$= \frac{P(Y = y|X = x)P(X = x)}{\sum_{z \in Val(X)} P(Y = y|X = z)P(X = z)}.$$

Теорема (формула множення).

$$P(X_1, \dots, X_n) = P(X_1 = x_1)P(X_2 = x_2|X_1 = x_1)P(X_3 = x_3|X_1 = x_1, X_2 = x_2)$$

$$\dots P(X_k = x_k|X_1 = x_1, \dots, X_{k-1} = x_{k-1}),$$

$x_1 \epsilon Val(X_1), x_2 \epsilon Val(X_2), \dots, x_k \epsilon Val(X_k)$.

1.4. Незалежні випадкові величини

Випадкові величини X, Y *незалежні* тоді й тільки тоді, коли для всіх можливих значень x_i, y_j випадкових величин X, Y

$$P\big(X = x_i, Y = y_j\big) = P(X = x_i)P\big(Y = y_j\big),\ x_i \in Val(X), y_j \in Val(Y).$$

Випадкові величини X, Y *умовно незалежні за умови заданої випадкової величини* Z тоді й тільки тоді, коли для всіх можливих значень x_i, y_j, z_k випадкових величин X, Y, Z

$$P\big(X = x_i, Y = y_j \big| Z = z_k\big) = P(X = x_i|Z = z_k)P(Y = y_j|Z = z_k),$$
$$x_i \in Val(X), y_j \in Val(Y), z_k \in Val(Z).$$

Умовну незалежність випадкових величин скорочено позначатимемо так

$$P(X, Y|Z) = P(X|Z)P(Y|Z),$$

або,

$$(X \perp Y|Z).$$

Здобудемо еквівалентне означення умовної незалежності. Розділимо праву і ліву частини рівності $P(X, Y|Z) = P(X|Z)P(Y|Z)$ на $P(Y|Z)$:

$$\frac{P(X, Y|Z)}{P(Y|Z)} = P(X|Z).$$

Домножимо чисельник і знаменник лівої частини останньої рівності на $P(Z)$:

$$\frac{P(X, Y|Z)P(Z)}{P(Y|Z)P(Z)} = P(X|Z).$$

Застосуємо формулу множення до чисельника та знаменника в лівій частині рівності:

$$\frac{P(X, Y, Z)}{P(Y, Z)} = P(X|Z).$$

Скористаємося означенням умовної ймовірності й здобудемо еквівалентне означення умовної незалежності випадкових величин X, Y за умови заданої випадкової величини Z:

$$P(X|Y, Z) = P(X|Z).$$

2. Пода́ння байєсівської мережі

2.1. Формула множення для байєсівської мережі

Вивчатимемо спільний розподіл випадкових величин $X_1, \ldots, X_n$. Навіть у найпростішому випадку, коли кожна випадкова величина набуває два значення, спільний розподіл $P(X_1, \ldots, X_n)$ вимагає визначення $2^n - 1$ імовірностей для різних комбінацій можливих значень випадкових величин $X_1, \ldots, X_n$. Досить складно проводити обчислення з такою великою кількістю значень, а також зберігати їх в пам'яті комп'ютера. Практично неможливо здобути експертні оцінки ймовірностей всіх комбінацій можливих значень випадкових величин. Для оцінювання невідомих параметрів спільного розподілу необхідні великі обсяги реальних даних. Саме ці проблеми виявилися основними труднощами при застосуванні ймовірнісних методів до побудови експертних систем до тих пір, поки не був запропонований підхід, викладений в роботі [1]. Компактне подання спільного розподілу випадкових величин $X_1, \ldots, X_n$ базується на двох ключових ідеях – використанні *умовних імовірнісних розподілів* випадкових величин та *умовної незалежності* випадкових величин.

Означення. *Байєсівською мережею* називатимемо пару $\mathcal{B} = (G, P)$, де G – орієнтований ациклічний граф, P – спільний розподіл, що факторизується на графі G.

Означення. Граф G *задає множину умовних незалежностей, пов'язаних зі спільним розподілом* P, якщо кожна вершина X_i умовно не залежить від вершин, які не є її нащадками $\left(NonDescendants_{X_i}\right)$ при заданих батьківських вершинах для вершини X_i $\left(Parents_{X_i}\right)$:

$$P(X_i | X_1, X_2, \ldots, X_{i-1}) = P\left(X_i \middle| Parents_{X_i}\right), i = 1, \ldots, n,$$

де вершини $X_1, \ldots, X_n$ розташовані в топологічному порядку відносно графа G.

Формула множення для байєсівської мережі. Нехай G – байєсівська мережа, побудована на $X_1,\dots,X_n$. Спільний розподіл $P(X_1,\dots,X_n)$ факторизується на графі G, якщо його можна подати як добуток умовних імовірнісних розподілів $P\left(X_i|Parents_{X_i}\right)$:

$$P(X_1,\dots,X_n)=\prod_{i=1}^{n}P\left(X_i|Parents_{X_i}\right).$$

Теорема. Нехай G – байєсівська мережа, побудована на випадкових величинах $X_1,\dots,X_n$, P – спільний розподіл випадкових величин $X_1,\dots,X_n$. Якщо граф G задає множину умовних незалежностей, пов'язаних зі спільним розподілом P, то P факторизується на графі G.

Доведення. Згідно з формулою множення спільний розподіл $X_1,\dots,X_n$ дорівнює

$$P(X_1,X_2,\dots,X_n)=P(X_1)P(X_2|X_1)P(X_3|X_1,X_2)\dots$$
$$\dots P(X_i|X_1,\dots,X_{i-1})\dots\ P(X_n|X_1,\dots,X_{n-1}).$$

Розглянемо один множник $P(X_i|X_1,\dots,X_{i-1})$. Оскільки граф G задає множину умовних незалежностей, пов'язаних зі спільним розподілом P, то для кожної вершини X_i:

$$\left(X_i\perp NonDescendants_{X_i}|Parents_{X_i}\right),i=1,\dots,n.$$

Всі батьківські вершини для X_i містяться в множині $\{X_1,\dots,X_{i-1}\}$, а жоден з нащадків вершини X_i не належить множині $\{X_1,\dots,X_{i-1}\}$. Отже,

$$\{X_1,\dots,X_{i-1}\}=Parents_{X_i}\cup\boldsymbol{Z},$$

де $\boldsymbol{Z}\subseteq NonDescendants_{X_i}$. Тоді

$$\left(X_i\perp\boldsymbol{Z}|Parents_{X_i}\right).$$

Отже,

$$P(X_i|X_1,\dots,X_{i-1})=P\left(X_i\middle|Parents_{X_i}\cup\boldsymbol{Z}\right)=P\left(X_i\middle|Parents_{X_i}\right).$$

Тоді спільний розподіл дорівнює

$$P(X_1,X_2,\dots,X_n)=P\left(X_1\middle|Parents_{X_1}\right)P\left(X_2\middle|Parents_{X_2}\right)\dots P\left(X_n\middle|Parents_{X_n}\right),$$

тобто P факторизується на графі G. Теорема доведена.

Теорема. Нехай G – байєсівська мережа, побудована на випадкових величинах $X_1, \dots, X_n$, P – спільний розподіл випадкових величин $X_1, \dots, X_n$. Якщо P факторизується на графі G, то граф G задає множину умовних незалежностей, пов'язаних зі спільним розподілом P.

Доведення. Спільний розподіл P факторизується на графі G, тобто справедлива формула множення для байєсівської мережі

$$P(X_1, X_2, \dots, X_n) = P\left(X_1 \middle| Parents_{X_1}\right) P\left(X_2 \middle| Parents_{X_2}\right) \dots P\left(X_n \middle| Parents_{X_n}\right).$$

Покажемо, що $(X_i \perp \boldsymbol{Z} \middle| Parents_{X_i}), \boldsymbol{Z} \subseteq NonDescendants_{X_i}$, зокрема,

$$P(X_i | X_1, X_2, \dots, X_{i-1}) = P\left(X_i \middle| Parents_{X_i}\right).$$

Дійсно,

$$P(X_i | X_1, X_2, \dots, X_{i-1}) = \frac{P(X_1, X_2, \dots, X_i)}{P(X_1, X_2, \dots, X_{i-1})} =$$

$$= \frac{P\left(X_1 \middle| Parents_{X_1}\right) \dots P\left(X_i \middle| Parents_{X_i}\right)}{P\left(X_1 \middle| Parents_{X_1}\right) \dots P\left(X_{i-1} \middle| Parents_{X_{i-1}}\right)} = P\left(X_i \middle| Parents_{X_i}\right).$$

Теорема доведена.

2.2. Байєсівська мережа Insurance

Байєсівська мережа Insurance представлена орієнтованим ациклічним графом, в вершинах якого знаходяться характеристики клієнтів та їх автомобілів, а орієнтовані ребра представляють вплив однієї характеристики на іншу. Структура мережі та умовні ймовірнісні розподіли вершин подані в [2]. Так концентрація уваги клієнта $Focused$ та його вік Age впливають на успішність клієнта в автошколі $Good\ Student$; концентрація уваги клієнта $Focused$, його вік Age та проходження курсу екстремального водіння $Extra\ Training$ впливають на кваліфікованість водія $Driver\ Quality$; кваліфікованість водія $Driver\ Quality$ впливає на кількість аварій $Driving\ History$; концентрація уваги клієнта $Focused$ впливає на розмір автомобіля $Vehicle\ Size$; кваліфікованість водія $Driver\ Quality$, розмір автомобіля $Vehicle\ Size$, рік випуску

автомобіля $Vehicle\ Year$, наявність подушки безпеки в автомобілі $Airbag$ та наявність антиблокувальної системи в автомобілі $Antilock\ Brakes$ впливає на ступінь тяжкості аварії $Accident$; рік випуску автомобіля $Vehicle\ Year$ та ступінь тяжкості аварії $Accident$ впливають на витрати страхової компанії $Cost$.

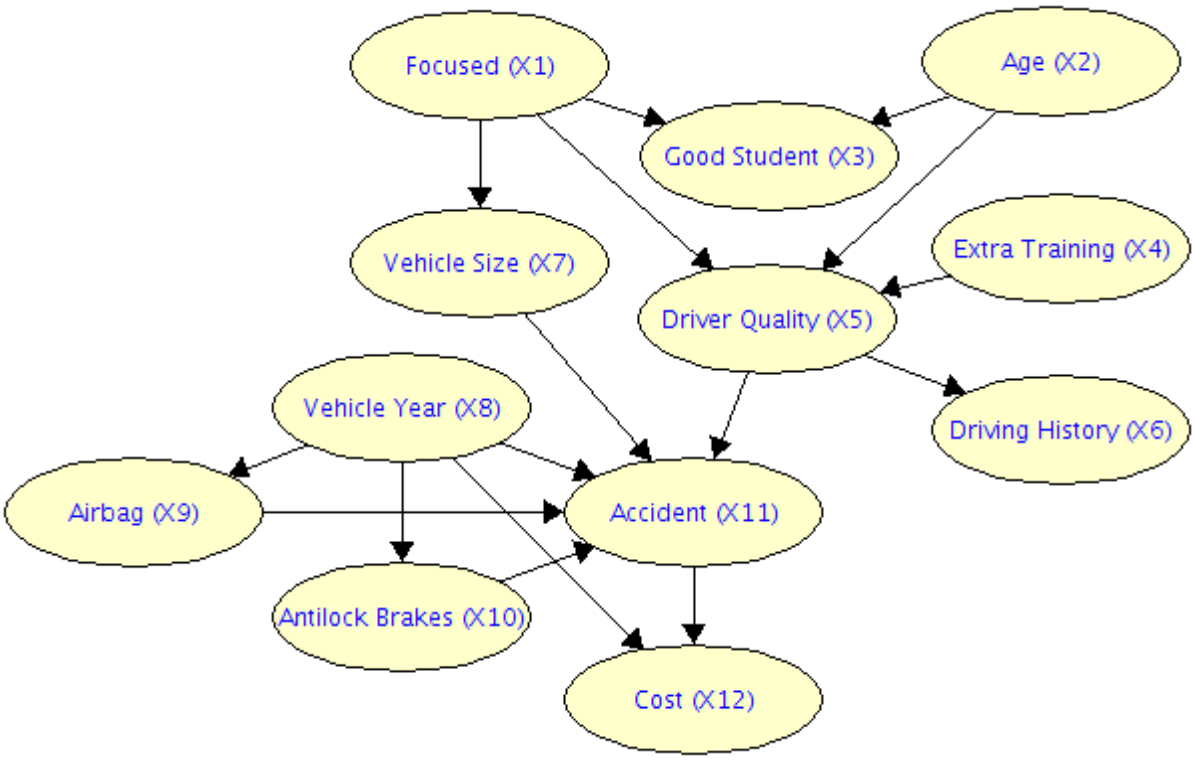


Рис. 2.2.1. Байєсівська мережа Insurance [2]

Розглянемо умовні незалежності в байєсівській мережі Insurance. Маємо:

$$(X_2 \perp X_1),$$
$$(X_4 \perp X_3, X_2, X_1),$$
$$(X_5 \perp X_3 | X_4, X_2, X_1),$$
$$(X_6 \perp X_4, X_3, X_2, X_1 | X_5),$$
$$(X_7 \perp X_6, X_5, X_4, X_3, X_2 | X_1),$$
$$(X_8 \perp X_7, X_6, X_5, X_4, X_3, X_2, X_1),$$
$$(X_9 \perp X_7, X_6, X_5, X_4, X_3, X_2, X_1 | X_8),$$
$$(X_{10} \perp X_9, X_7, X_6, X_5, X_4, X_3, X_2, X_1 | X_8),$$
$$(X_{11} \perp X_6, X_4, X_3, X_2, X_1 | X_{10}, X_9, X_8, X_7, X_5),$$
$$(X_{12} \perp X_{10}, X_9, X_7, X_6, X_5, X_4, X_3, X_2, X_1 | X_{11}, X_8).$$

Спільний розподіл вершин $X_1, X_2, \dots, X_{12}$ задається за допомогою формули множення:

$$P(X_1, X_2, \dots, X_{12}) = P(X_{12}|X_1, X_2, \dots, X_{11})P(X_{11}|X_1, X_2, \dots, X_{10}) \cdot$$
$$\cdot P(X_{10}|X_1, X_2, \dots, X_9)P(X_9|X_1, X_2, \dots, X_8)P(X_8|X_1, X_2, \dots, X_7) \cdot$$
$$\cdot P(X_7|X_1, X_2, \dots, X_6)P(X_6|X_1, X_2, \dots, X_5)P(X_5|X_1, X_2, X_3, X_4) \cdot$$
$$\cdot P(X_4|X_1, X_2, X_3)P(X_3|X_1, X_2)P(X_2|X_1)P(X_1)$$

і з урахуванням множини умовних незалежностей перепишеться у вигляді:

$$P(X_1, X_2, \dots, X_{12}) = P(X_{12}|X_{11}, X_8)P(X_{11}|X_{10}, X_9, X_8, X_7, X_5)P(X_{10}|X_8) \cdot$$
$$\cdot P(X_9|X_8)P(X_8)P(X_7|X_1)P(X_6|X_5)P(X_5|X_1, X_2, X_4)P(X_4) \cdot$$
$$\cdot P(X_3|X_2, X_1)P(X_2)P(X_1).$$

Перевага подання спільного розподілу $P(X_1, X_2, \dots, X_{12})$ як добутку умовних імовірнісних розподілів $P\left(X_i|Parents_{X_i}\right)$ полягає в тому, що нам необхідно знати в 95 разів менше оцінок імовірностей можливих значень станів вершин, ніж при безпосередньому заданні спільного розподілу.

2.3. Байєсівська мережа Credit

Байєсівська мережа *Credit* представлена орієнтованим ациклічним графом, в вершинах якого знаходяться характеристики клієнтів, а орієнтовані ребра представляють вплив однієї характеристики на іншу. Структура мережі та умовні ймовірнісні розподіли вершин подані в [2]. Так вік клієнта *Age* та відношення боргу до доходу *Ratio of Debts to Income* впливають на історію платежів *Payment History*, вік клієнта *Age* та історія платежів *Payment History* впливають на надійність клієнта *Reliability*, дохід *Income* та активи *Assets* впливають на майбутній дохід *Future Income*, майбутній дохід *Future Income*, відношення боргу до доходу *Ratio of Debts to Income* та надійність клієнта *Reliability* впливають на кредитоспроможність клієнта *Credit Worthiness*.

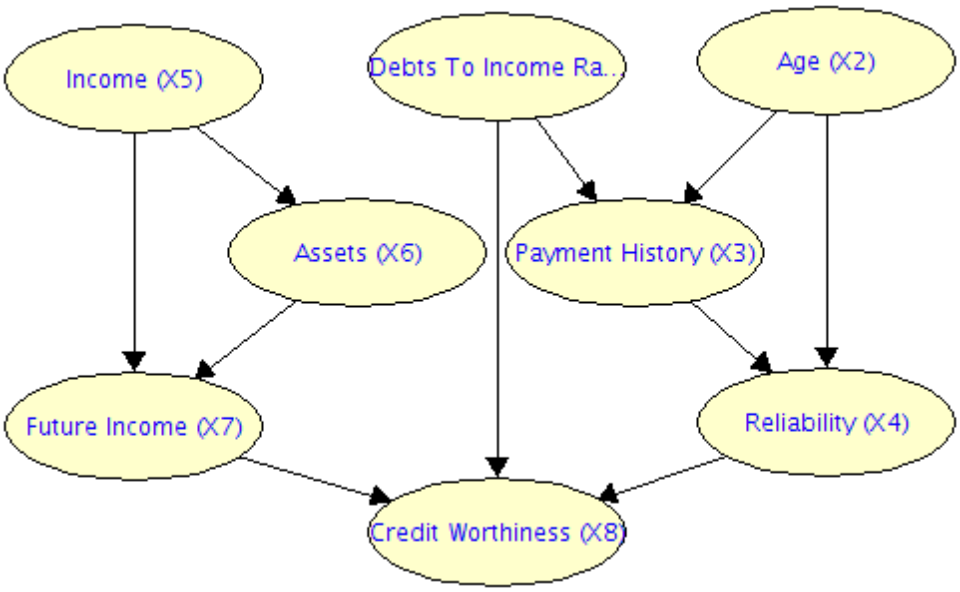


Рис. 2.3.1. Байєсівська мережа Credit [2]

Розглянемо умовні незалежності в байєсівській мережі Credit. Маємо:

$$(X_2 \perp X_1),$$

$$(X_4 \perp X_1 | X_2, X_3),$$

$$(X_5 \perp X_1, X_2, X_3, X_4),$$

$$(X_6 \perp X_1, X_2, X_3, X_4 | X_5),$$

$$(X_7 \perp X_1, X_2, X_3, X_4 | X_5, X_6),$$

$$(X_8 \perp X_2, X_3, X_5, X_6 | X_1, X_4, X_7).$$

Спільний розподіл вершин $X_1, X_2, \dots, X_8$ задається за допомогою формули множення:

$$P(X_1, \dots, X_8) = P(X_8|X_1, X_2, \dots, X_7)P(X_7|X_1, X_2, \dots, X_6)P(X_6|X_1, X_2, \dots, X_5) \cdot$$
$$\cdot P(X_5|X_1, X_2, X_3, X_4)P(X_4|X_1, X_2, X_3)P(X_3|X_1, X_2)P(X_2|X_1)P(X_1)$$

і з урахуванням множини умовних незалежностей перепишеться у вигляді:

$$P(X_1, \dots, X_8) = P(X_1)P(X_2)P(X_3|X_1, X_2)P(X_4|X_2, X_3)P(X_5) \cdot$$
$$\cdot P(X_6|X_5)P(X_7|X_5, X_6)P(X_8|X_1, X_4, X_7).$$

Перевага подання спільного розподілу $P(X_1, X_2, \dots, X_8)$ як добутку умовних імовірнісних розподілів $P\left(X_i|Parents_{X_i}\right)$ полягає в тому, що нам необхідно знати в 25 разів менше оцінок імовірностей можливих значень станів вершин, ніж при безпосередньому заданні спільного розподілу.

3. Імовірнісний висновок

3.1. Апріорний та апостеріорний розподіли

Побудувавши структуру мережі самостійно або залучивши експертів для її побудови, дослідник отримує мережу певної топології. Використовуючи її, можна моделювати різноманітні ситуації, тобто задавати вершинам мережі деякі значення станів і отримувати найбільш імовірні значення станів для інших вершин мережі. Процес обчислення апостеріорних імовірностей станів вершини X_i при заданих станах спостережуваних вершин називатимемо *ймовірнісним висновком*.

Апріорний розподіл будь-якої вершини X_i обчислюється за спільним розподілом $P(X_1, X_2, \dots, X_n)$ вершин $X_1, X_2, \dots, X_n$:

$$P(X_i) = \sum_{X_1,\dots,X_{i-1},X_{i+1},\dots,X_n} P(X_1, \dots, X_n),$$

де спільний розподіл $P(X_1, \dots, X_n)$ обчислюється за формулою множення для байєсівської мережі:

$$P(X_1, \dots, X_n) = \prod_{i=1}^{n} P\left(X_i | Parents_{X_i}\right).$$

Апостеріорний розподіл вершини X_i за умови заданого стану вершини X_j обчислюється за формулою умовної ймовірності:

$$P\left(X_i | X_j\right) = \frac{P\left(X_i, X_j\right)}{P\left(X_j\right)},$$

$$P\left(X_i, X_j\right) = \sum_{X_1,\dots,X_{i-1},X_{i+1},\dots,X_{j-1},X_{j+1},\dots,X_n} P(X_1, \dots, X_n),$$

$$P\left(X_j\right) = \sum_{X_1,\dots,X_{j-1},X_{j+1},\dots,X_n} P(X_1, \dots, X_n).$$

де спільний розподіл $P(X_1, \dots, X_n)$ обчислюється за формулою множення для байєсівської мережі:

$$P(X_1, \dots, X_n) = \prod_{i=1}^{n} P\left(X_i | Parents_{X_i}\right).$$

3.2. Алгоритм Variable Elimination

Розглянемо найпростіший алгоритм точного імовірнісного висновку Variable Elimination для обчислення розподілів станів вершини. Алгоритм виключення змінної полягає в послідовному обчисленні сум добутків множників за всіма станами вершин, які виключаються.

Algorithm 1. Sum-product variable elimination algorithm [1, p. 298]

Procedure Sum-Product-VE (
Φ, // набір множників
$\boldsymbol{Z}$, // множина вершин, які виключаються
)

1 Нехай $Z_1, \dots, Z_k$ порядок виключення вершин знаходиться згідно з алгоритмом Greedy-Ordering [1, p. 314]
2 **for** $i = 1, \dots, k$
3 $\Phi \leftarrow \text{Sum} - \text{Product} - \text{Eliminate} - \text{Var}(\Phi, Z_i)$
4 $\phi^* \leftarrow \prod_{\phi \in \Phi} \phi$
5 **return** ϕ^*

Procedure Sum-Product-Eliminate-Var (
Φ, // набір множників
Z, // вершина, яка виключається
)

1 $\Phi' \leftarrow \{\phi \in \Phi : Z \in Scope[\phi]\}$
2 $\Phi'' \leftarrow \Phi - \Phi'$
3 $\psi \leftarrow \prod_{\phi \in \Phi'} \phi$
4 $\tau \leftarrow \sum_Z \psi$
5 **return** $\Phi'' \cup \{\tau\}$

3.3. Імовірнісний висновок в байєсівській мережі Insurance

Розподіл витрат страхової компанії X_{12} має вигляд

$$P(X_{12}) = \sum_{X_1,\dots,X_{11}} P(X_1,\dots,X_{12}) =$$

$$\sum_{X_1,\dots,X_{11}} P(X_{12}|X_{11},X_8)P(X_{11}|X_{10},X_9,X_8,X_7,X_5)P(X_{10}|X_8)P(X_9|X_8)P(X_8)\cdot$$

$$\cdot P(X_7|X_1)P(X_6|X_5)P(X_5|X_4,X_2,X_1)P(X_4)P(X_3|X_2,X_1)P(X_2)P(X_1).$$

При обчисленні розподілу витрат страхової компанії безпосереднє застосування формули множення для байєсівської мережі вимагає 110 590 арифметичних операцій (101 376 множень, 9214 додавань). Застосуємо алгоритм Variable Elimination. Так ми значно зменшимо число операцій та час обчислення розподілу $P(X_{12})$.

$$P(X_{12}) = \sum_{X_{11}}\sum_{X_8} P(X_{12}|X_{11},X_8)P(X_8)\sum_{X_{10}} P(X_{10}|X_8)\sum_{X_9} P(X_9|X_8)$$

$$\sum_{X_7}\sum_{X_5} P(X_{11}|X_{10},X_9,X_8,X_7,X_5)\sum_{X_1} P(X_1)P(X_7|X_1)$$

$$\sum_{X_3}\sum_{X_2} P(X_3|X_2,X_1)P(X_2)\sum_{X_6} P(X_6|X_5)\sum_{X_4} P(X_5|X_4,X_2,X_1)P(X_4).$$

Порядок виключення вершин набуває вигляду:

$$Z = \{X_4, X_6, X_2, X_3, X_1, X_5, X_7, X_9, X_{10}, X_8, X_{11}\}.$$

Виключаємо вершину X_4. Здобуваємо

$$\psi_1(X_1,X_2,X_4,X_5) = P(X_5|X_1,X_2,X_4)P(X_4),$$

$$\tau_1(X_1,X_2,X_5) = \sum_{X_4} \psi_1(X_1,X_2,X_4,X_5),$$

$$P(X_{12}) = \sum_{X_{11}}\sum_{X_8} P(X_{12}|X_{11},X_8)P(X_8)\sum_{X_{10}} P(X_{10}|X_8)\sum_{X_9} P(X_9|X_8)$$

$$\sum_{X_7}\sum_{X_5} P(X_{11}|X_{10},X_9,X_8,X_7,X_5)\sum_{X_1} P(X_1)P(X_7|X_1)$$

$$\sum_{X_3}\sum_{X_2} P(X_3|X_2,X_1)P(X_2)\tau_1(X_1,X_2,X_5)\sum_{X_6} P(X_6|X_5).$$

Виключаємо вершину X_6. Отримаємо

$$\psi_2(X_6, X_5) = P(X_6|X_5),$$

$$\tau_2(X_5) = \sum_{X_6} \psi_2(X_6, X_5),$$

$$P(X_{12}) = \sum_{X_{11}} \sum_{X_8} P(X_{12}|X_{11}, X_8) P(X_8) \sum_{X_{10}} P(X_{10}|X_8) \sum_{X_9} P(X_9|X_8)$$

$$\sum_{X_7} \sum_{X_5} P(X_{11}|X_{10}, X_9, X_8, X_7, X_5) \tau_2(X_5) \sum_{X_1} P(X_1) P(X_7|X_1)$$

$$\sum_{X_3} \sum_{X_2} P(X_3|X_2, X_1) P(X_2) \tau_1(X_1, X_2, X_5).$$

Виключаємо вершину X_2. Отримаємо

$$\psi_3(X_5, X_3, X_2, X_1) = P(X_3|X_1, X_2) P(X_2) \tau_1(X_1, X_2, X_5),$$

$$\tau_3(X_5, X_3, X_1) = \sum_{X_2} \psi_3(X_5, X_3, X_2, X_1),$$

$$P(X_{12}) = \sum_{X_{11}} \sum_{X_8} P(X_{12}|X_{11}, X_8) P(X_8) \sum_{X_{10}} P(X_{10}|X_8) \sum_{X_9} P(X_9|X_8)$$

$$\sum_{X_7} \sum_{X_5} P(X_{11}|X_{10}, X_9, X_8, X_7, X_5) \tau_2(X_5) \sum_{X_1} P(X_1) P(X_7|X_1) \sum_{X_3} \tau_3(X_5, X_3, X_1).$$

Виключаємо вершину X_3. Отримаємо

$$\psi_4(X_5, X_3, X_1) = \tau_3(X_5, X_3, X_1),$$

$$\tau_4(X_5, X_1) = \sum_{X_3} \psi_4(X_5, X_3, X_1),$$

$$P(X_{12}) = \sum_{X_{11}} \sum_{X_8} P(X_{12}|X_{11}, X_8) P(X_8) \sum_{X_{10}} P(X_{10}|X_8) \sum_{X_9} P(X_9|X_8)$$

$$\sum_{X_7} \sum_{X_5} P(X_{11}|X_{10}, X_9, X_8, X_7, X_5) \tau_2(X_5) \sum_{X_1} P(X_1) P(X_7|X_1)\, \tau_4(X_5, X_1).$$

Виключаємо вершину X_1. Отримаємо

$$\psi_5(X_7, X_5, X_1) = P(X_1) P(X_7|X_1) \tau_4(X_5, X_1),$$

$$\tau_5(X_7, X_5) = \sum_{X_1} \psi_5(X_7, X_5, X_1),$$

$$P(X_{12}) = \sum_{X_{11}} \sum_{X_8} P(X_{12}|X_{11}, X_8) P(X_8) \sum_{X_{10}} P(X_{10}|X_8) \sum_{X_9} P(X_9|X_8)$$

$$\sum_{X_7} \sum_{X_5} P(X_{11}|X_{10}, X_9, X_8, X_7, X_5) \tau_2(X_5)\, \tau_5(X_7, X_5).$$

Виключаємо вершину X_5. Отримаємо

$$\psi_6(X_{11}, X_{10}, X_9, X_8, X_7, X_5) = P(X_{11}|X_{10}, X_9, X_8, X_7, X_5) \tau_2(X_5) \tau_5(X_7, X_5),$$

$$\tau_6(X_{11}, X_{10}, X_9, X_8, X_7) = \sum_{X_5} \psi_6(X_{11}, X_{10}, X_9, X_8, X_7, X_5),$$

$$P(X_{12}) = \sum_{X_{11}} \sum_{X_8} P(X_{12}|X_{11}, X_8) P(X_8) \sum_{X_{10}} P(X_{10}|X_8) \sum_{X_9} P(X_9|X_8)$$

$$\sum_{X_7} \tau_6(X_{11}, X_{10}, X_9, X_8, X_7).$$

Виключаємо вершину X_7. Отримаємо

$$\psi_7(X_{11}, X_{10}, X_9, X_8, X_7) = \tau_6(X_{11}, X_{10}, X_9, X_8, X_7),$$

$$\tau_7(X_{11}, X_{10}, X_9, X_8) = \sum_{X_7} \psi_7(X_{11}, X_{10}, X_9, X_8),$$

$$P(X_{12}) =$$

$$= \sum_{X_{11}} \sum_{X_8} P(X_{12}|X_{11}, X_8) P(X_8) \sum_{X_{10}} P(X_{10}|X_8) \sum_{X_9} \tau_7(X_{11}, X_{10}, X_9, X_8) P(X_9|X_8).$$

Виключаємо вершину X_9. Отримаємо

$$\psi_8(X_{11}, X_{10}, X_9, X_8) = \tau_7(X_{11}, X_{10}, X_9, X_8) P(X_9|X_8),$$

$$\tau_8(X_{11}, X_{10}, X_8) = \sum_{X_9} \psi_8(X_{11}, X_{10}, X_9, X_8),$$

$$P(X_{12}) = \sum_{X_{11}} \sum_{X_8} P(X_{12}|X_{11}, X_8) P(X_8) \sum_{X_{10}} \tau_8(X_{11}, X_{10}, X_8) P(X_{10}|X_8).$$

Виключаємо вершину X_{10}. Отримаємо

$$\psi_9(X_{11}, X_{10}, X_8) = \tau_8(X_{11}, X_{10}, X_8) P(X_{10}|X_8),$$

$$\tau_9(X_{11}, X_8) = \sum_{X_{10}} \psi_9(X_{11}, X_{10}, X_8),$$

$$P(X_{12}) = \sum_{X_{11}} \sum_{X_8} \tau_9(X_{11}, X_8) P(X_{12}|X_{11}, X_8) P(X_8).$$

Виключаємо вершину X_8. Отримаємо

$$\psi_{10}(X_{12}, X_{11}, X_8) = \tau_9(X_{11}, X_8)P(X_{12}|X_{11}, X_8)P(X_8),$$

$$\tau_{10}(X_{12}, X_{11}) = \sum_{X_8} \psi_{10}(X_{12}, X_{11}, X_8),$$

$$P(X_{12}) = \sum_{X_{11}} \tau_{10}(X_{12}, X_{11}).$$

Виключаємо вершину X_{11}. Отримаємо

$$\psi_{11}(X_{12}, X_{11}) = \tau_{10}(X_{12}, X_{11}),$$

$$\tau_{11}(X_{12}) = \sum_{X_{11}} \psi_{11}(X_{12}, X_{11}),$$

$$P(X_{12}) = \tau_{11}(X_{12}).$$

Таблиця 3.3.1. Алгоритм Variable Elimination

Крок	Виключена змінна	Використані множники	Використані змінні	Новий множник
1	X_4	$P(X_5\|X_1, X_2, X_4), P(X_4)$	X_1, X_2, X_4, X_5	$\tau_1(X_1, X_2, X_5)$
2	X_6	$P(X_6\|X_5)$	X_5, X_6	$\tau_2(X_5)$
3	X_2	$\tau_1(X_1, X_2, X_5), P(X_3\|X_1, X_2), P(X_2)$	X_1, X_2, X_3, X_5	$\tau_3(X_5, X_3, X_1)$
4	X_3	$\tau_3(X_5, X_3, X_1)$	X_1, X_3, X_5	$\tau_4(X_5, X_1)$
5	X_1	$P(X_7\|X_1), \tau_4(X_5, X_1), P(X_1)$	X_1, X_5, X_7	$\tau_5(X_7, X_5)$
6	X_5	$P(X_{11}\|X_{10}, X_9, X_8, X_7, X_5), \tau_2(X_5), \tau_5(X_7, X_5)$	$X_{11}, X_{10}, X_9, X_8 X_7, X_5$	$\tau_6\binom{X_{11}, X_{10}, X_9,}{X_8, X_7}$
7	X_7	$\tau_6(X_{11}, X_{10}, X_9, X_8, X_7)$	$X_{11}, X_{10}, X_9, X_8, X_7$	$\tau_7\binom{X_{11}, X_{10},}{X_9, X_8}$
8	X_9	$\tau_7(X_{11}, X_{10}, X_9, X_8), P(X_9\|X_8)$	X_8, X_9, X_{10}, X_{11}	$\tau_8(X_{11}, X_{10}, X_8)$
9	X_{10}	$\tau_8(X_{11}, X_{10}, X_8), P(X_{10}\|X_8)$	X_8, X_{10}, X_{11}	$\tau_9(X_{11}, X_8)$
10	X_8	$P(X_{12}\|X_{11}, X_8), \tau_9(X_{11}, X_8), P(X_8)$	X_8, X_{11}, X_{12}	$\tau_{10}(X_{12}, X_{11})$
11	X_{11}	$\tau_{10}(X_{12}, X_{11})$	X_{11}, X_{12}	$\tau_{11}(X_{12})$

Реалізація алгоритму Variable Elimination для обчислення розподілу витрат компанії наведена в додатку 2 (виконано 444 арифметичні операції: 316 множень, 128 додавань, час обчислення склав 0,0625 сек).

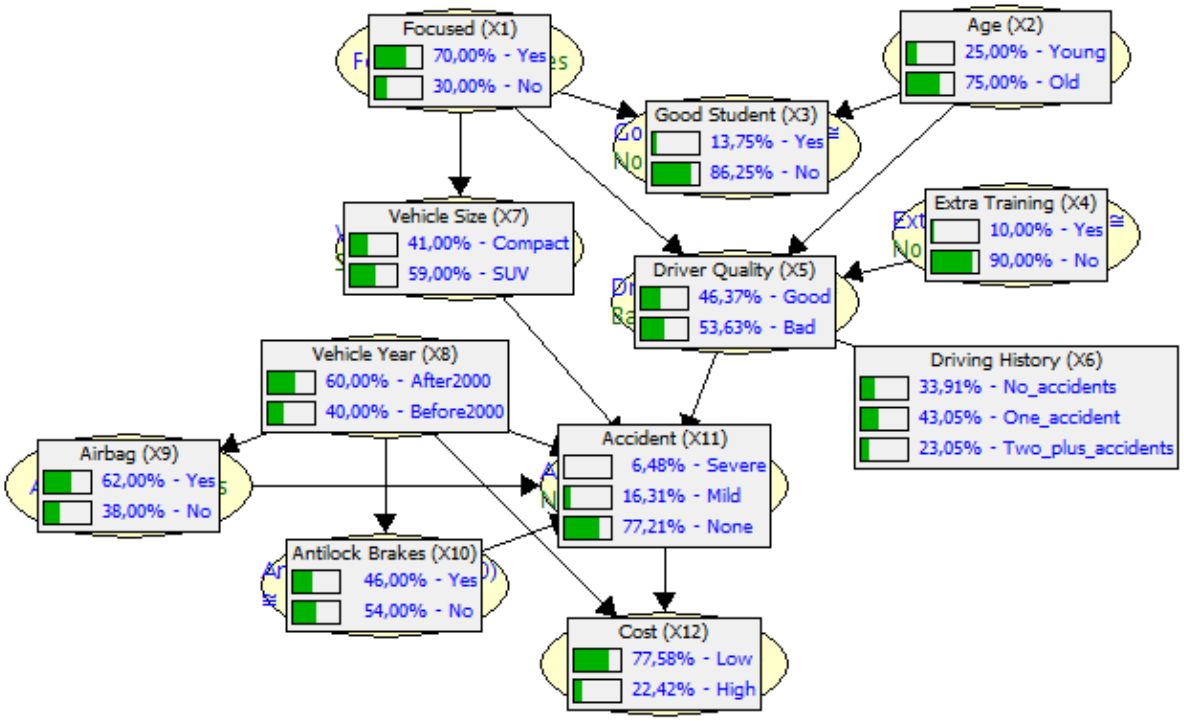


Рис. 3.3.1. Розподіл витрат страхової компанії

$$P(X_{12} = Low) = 0{,}7758,\ P(X_{12} = High) = 0{,}2242.$$

Апостеріорний розподіл витрат страхової компанії X_{12} має вигляд

$$P(X_{12}|X_{10} = Yes) = \frac{P(X_{12}, X_{10} = Yes)}{P(X_{10} = Yes)} =$$

$$= \frac{1}{P(X_{10} = Yes)} \sum_{X_1,\dots,X_9,X_{11}} P(X_{12}|X_{11}, X_8) \cdot$$

$$\cdot P(X_{11}|X_{10} = Yes, X_9, X_8, X_7, X_5) P(X_{10} = Yes|X_8) P(X_9|X_8) P(X_8) P(X_7|X_1) \cdot$$

$$\cdot P(X_6|X_5) P(X_5|X_4, X_2, X_1) P(X_4) P(X_3|X_2, X_1) P(X_2) P(X_1).$$

При обчисленні апостеріорного розподілу витрат страхової компанії безпосереднє застосування формули множення для байєсівської мережі вимагає 55 295 операцій (50 688 множень, 4606 додавань, одне ділення). Застосуємо алгоритм Variable Elimination. Так ми значно зменшимо число арифметичних операцій та час обчислення розподілу $P(X_{12}|X_{10} = Yes)$:

$$\frac{1}{P(X_{10} = Yes)} \sum_{X_{11}} \sum_{X_8} P(X_{12}|X_{11}, X_8) P(X_8) P(X_{10} = Yes|X_8)$$

$$\sum_{X_9} P(X_9|X_8) \sum_{X_7} \sum_{X_5} P(X_{11}|X_{10} = Yes, X_9, X_8, X_7, X_5) \sum_{X_1} P(X_1) P(X_7|X_1)$$

$$\sum_{X_3} \sum_{X_2} P(X_3|X_2, X_1) P(X_2) \sum_{X_6} P(X_6|X_5) \sum_{X_4} P(X_5|X_4, X_2, X_1) P(X_4).$$

Таблиця 3.3.2. Алгоритм Variable Elimination

Крок	Виключена змінна	Використані множники	Використані змінні	Новий множник
$1'$	X_4	$P(X_5\|X_1,X_2,X_4), P(X_4)$	X_1,X_2,X_4,X_5	$\tau_1'(X_1,X_2,X_5)$
$2'$	X_6	$P(X_6\|X_5)$	X_5,X_6	$\tau_2'(X_5)$
$3'$	X_2	$\tau_1'(X_1,X_2,X_5), P(X_3\|X_1,X_2), P(X_2)$	X_1,X_2,X_3,X_5	$\tau_3'(X_5,X_3,X_1)$
$4'$	X_3	$\tau_3'(X_5,X_3,X_1)$	X_1,X_3,X_5	$\tau_4'(X_5,X_1)$
$5'$	X_1	$P(X_7\|X_1), \tau_4'(X_5,X_1), P(X_1)$	X_1,X_5,X_7	$\tau_5'(X_7,X_5)$
$6'$	X_5	$P\left(X_{11}\middle\|\begin{matrix}X_{10}=Yes,X_9,\\X_8,X_7,X_5\end{matrix}\right), \tau_2'(X_5), \tau_5'(X_7,X_5)$	$X_{11},X_9,$ X_8,X_7,X_5	$\tau_6'\begin{pmatrix}X_{11},X_9,\\X_8,X_7,\\X_{10}=Yes\end{pmatrix}$
$7'$	X_7	$\tau_6'\begin{pmatrix}X_{11},X_{10}=Yes,\\X_9,X_8,X_7\end{pmatrix}$	$X_{11},X_9,$ X_8,X_7	$\tau_7'\begin{pmatrix}X_{11},X_9,X_8,\\X_{10}=Yes\end{pmatrix}$
$8'$	X_9	$\tau_7'(X_{11},X_{10}=Yes,X_9,X_8), P(X_9\|X_8)$	X_8,X_9,X_{11}	$\tau_8'\begin{pmatrix}X_8,X_{11},\\X_{10}=Yes\end{pmatrix}$
$10'$	X_8	$P(X_{12}\|X_{11},X_8), P(X_8),$ $\tau_8'(X_{11},X_{10}=Yes,X_8),$ $P(X_{10}=Yes\|X_8)$	X_8,X_{11},X_{12}	$\tau_{10}'\begin{pmatrix}X_{12},X_{11},\\X_{10}=Yes\end{pmatrix}$
$11'$	X_{11}	$\tau_{10}'(X_{12},X_{11},X_{10}=Yes)$	X_{11},X_{12}	$\tau_{11}'\begin{pmatrix}X_{12},\\X_{10}=Yes\end{pmatrix}$

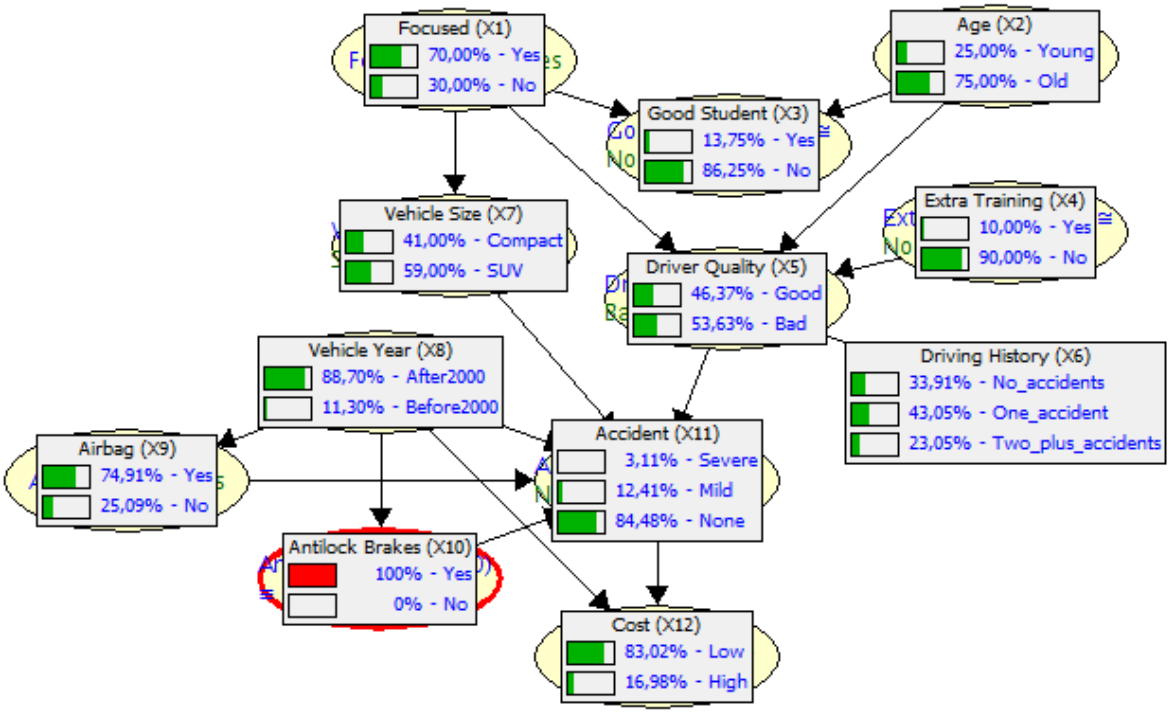


Рис. 3.3.2. Апостеріорний розподіл витрат страхової компанії

$P(X_{12}=High|X_{10}=Yes)=0{,}1698$, $P(X_{12}=Low|X_{10}=Yes)=0{,}8302$

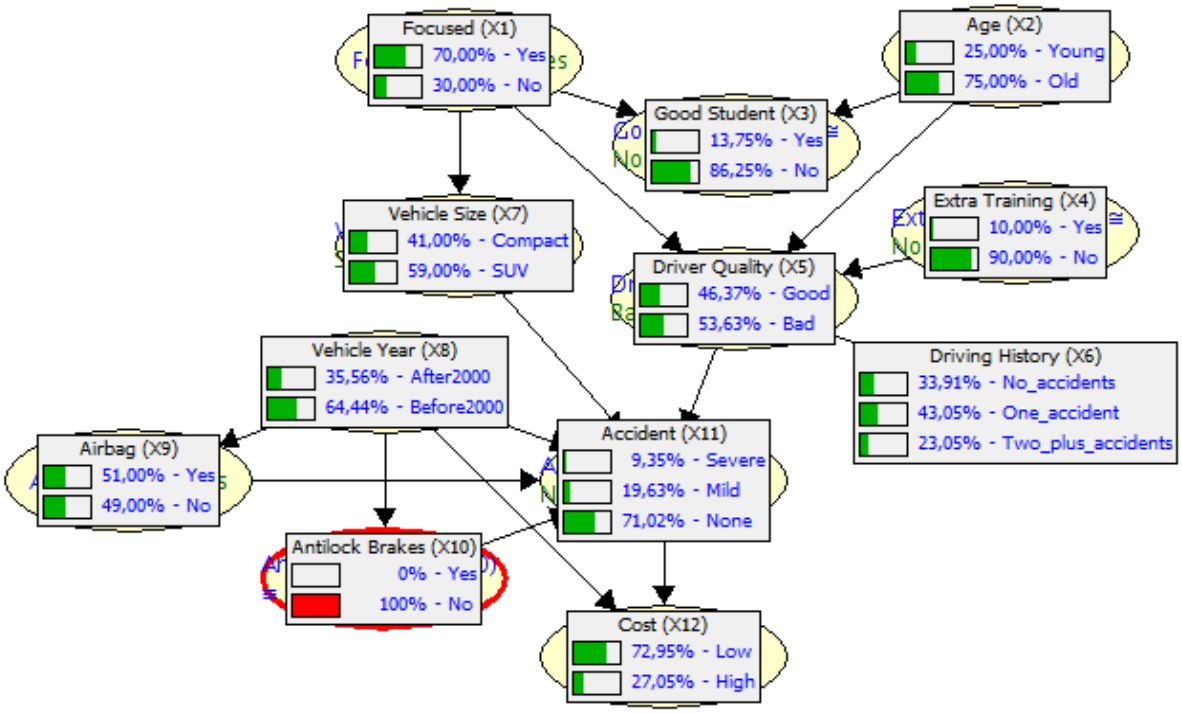


Рис. 3.3.3. Апостеріорний розподіл витрат страхової компанії

$$P(X_{12} = High|X_{10} = No) = 0{,}2705,\ P(X_{12} = Low|X_{10} = No) = 0{,}7295$$

Реалізація алгоритму Variable Elimination для обчислення розподілу витрат компанії наведена в додатку 2 (виконано 241 арифметична операція: 160 множень, 80 додавань, одне ділення, час обчислення склав 0,0938 сек).

Апостеріорний розподіл витрат страхової компанії X_{12} має вигляд

$$P(X_{12}|X_{10} = Yes, X_9 = Yes) = \frac{P(X_{12}, X_{10} = Yes, X_9 = Yes)}{P(X_{10} = Yes, X_9 = Yes)} =$$

$$= \frac{1}{P(X_{10} = Yes, X_9 = Yes)} \sum_{X_1,\dots,X_8,X_{11}} P(X_{12}|X_{11}, X_8) \cdot$$

$$\cdot P(X_{11}|X_{10} = Yes, X_9 = Yes, X_8, X_7, X_5) P(X_{10} = Yes|X_8) P(X_9 = Yes|X_8)$$

$$\cdot P(X_8) P(X_7|X_1) P(X_6|X_5) P(X_5|X_4, X_2, X_1) P(X_4) P(X_3|X_2, X_1) P(X_2) P(X_1).$$

При обчисленні апостеріорного розподілу витрат страхової компанії безпосереднє застосування формули множення для байєсівської мережі вимагає 27 647 операцій (25 344 множення, 2 302 додавання, одне ділення). Застосуємо алгоритм Variable Elimination. Так ми значно зменшимо число операцій та час обчислення розподілу $P(X_{12}|X_{10} = Yes, X_9 = Yes)$.

$$P(X_{12}|X_{10}=Yes,X_9=Yes)=$$

$$=\frac{1}{P(X_{10}=Yes,X_9=Yes)}\sum_{X_{11}}\sum_{X_8}P(X_{12}|X_{11},X_8)P(X_8)P(X_{10}=Yes|X_8)$$

$$P(X_9=Yes|X_8)\sum_{X_7}\sum_{X_5}P(X_{11}|X_{10}=Yes,X_9=Yes,X_8,X_7,X_5)\sum_{X_1}P(X_1)P(X_7|X_1)$$

$$\sum_{X_3}\sum_{X_2}P(X_3|X_2,X_1)P(X_2)\sum_{X_6}P(X_6|X_5)\sum_{X_4}P(X_5|X_4,X_2,X_1)P(X_4).$$

Таблиця 3.3.3. Алгоритм Variable Elimination

Крок	Виключена змінна	Використані множники	Використані змінні	Новий множник
$1'$	X_4	$P(X_5\|X_1,X_2,X_4),P(X_4)$	X_1,X_2,X_4,X_5	$\tau_1'(X_1,X_2,X_5)$
$2'$	X_6	$P(X_6\|X_5)$	X_5,X_6	$\tau_2'(X_5)$
$3'$	X_2	$\tau_1'(X_5,X_2,X_1),$ $P(X_3\|X_1,X_2),$ $P(X_2)$	X_1,X_2,X_3,X_5	$\tau_3'(X_5,X_3,X_1)$
$4'$	X_3	$\tau_3'(X_5,X_3,X_1)$	X_1,X_3,X_5	$\tau_4'(X_5,X_1)$
$5'$	X_1	$P(X_7\|X_1),\tau_4'(X_5,X_1),$ $P(X_1)$	X_1,X_5,X_7	$\tau_5'(X_7,X_5)$
$6'$	X_5	$P\left(X_{11}\middle\|\begin{matrix}X_{10}=Yes,X_9=Yes,\\X_8,X_7,X_5\end{matrix}\right)$ $\tau_2'(X_5),$ $\tau_5'(X_7,X_5)$	X_{11},X_8,X_7,X_5	$\tau_6'\left(\begin{matrix}X_{11},\\X_{10}=Yes,\\X_9=Yes,\\X_8,X_7\end{matrix}\right)$
$7'$	X_7	$\tau_6'\left(\begin{matrix}X_{11},X_{10}=Yes,\\X_9=Yes,X_8,X_7\end{matrix}\right)$	X_{11},X_8,X_7	$\tau_7'\left(\begin{matrix}X_{11},\\X_{10}=Yes,\\X_9=Yes,\\X_8\end{matrix}\right)$
$10'$	X_8	$P(X_{12}\|X_{11},X_8),P(X_8),$ $\tau_7'\left(\begin{matrix}X_{11},X_{10}=Yes,\\X_9=Yes,X_8\end{matrix}\right),$ $P(X_9=Yes\|X_8),$ $P(X_{10}=Yes\|X_8)$	X_8,X_{11},X_{12}	$\tau_{10}'\left(\begin{matrix}X_{12},X_{11},\\X_{10}=Yes,\\X_9=Yes\end{matrix}\right)$
$11'$	X_{11}	$\tau_{10}'\left(\begin{matrix}X_{12},X_{11},X_{10}=Yes,\\X_9=Yes\end{matrix}\right)$	X_{11},X_{12}	$\tau_{11}'\left(\begin{matrix}X_{12},\\X_9=Yes,\\X_{10}=Yes\end{matrix}\right)$

Реалізація алгоритму Variable Elimination для обчислення розподілу витрат компанії наведена в додатку 2 (виконано 217 операцій: 160 множень, 56 додавань, 1 ділення, час обчислення склав 0,078 сек).

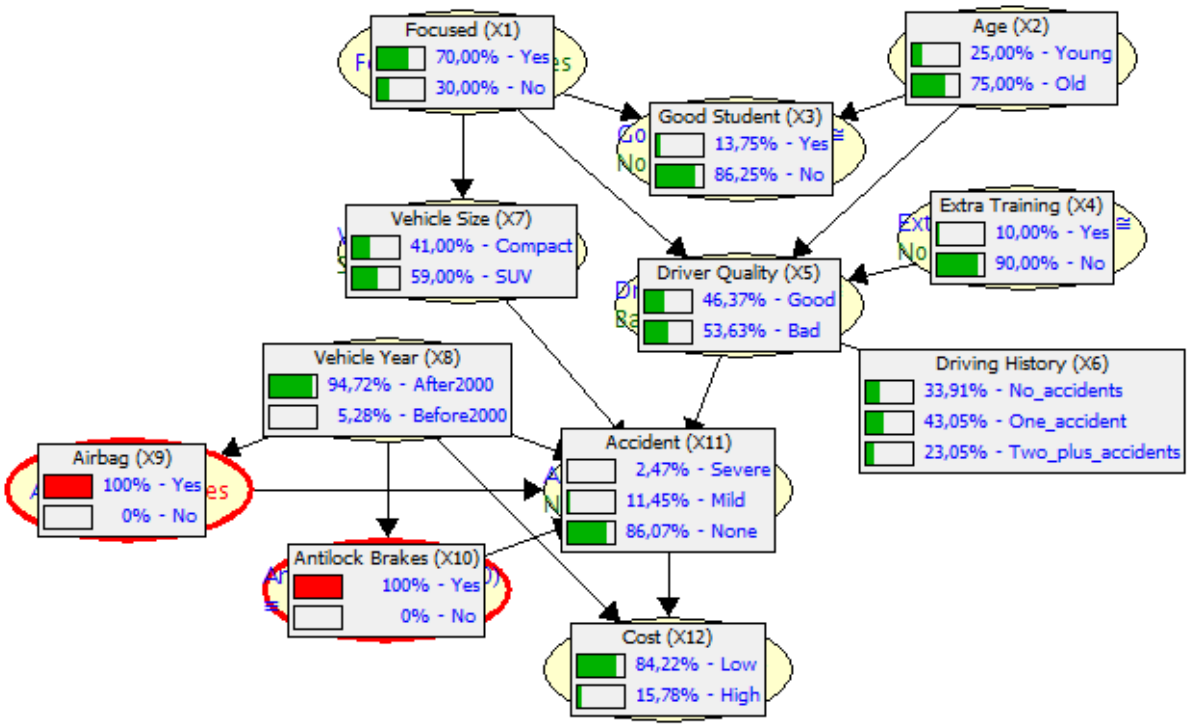


Рис. 3.3.4. Апостеріорний розподіл витрат страхової компанії

$$P(X_{12} = High|X_{10} = Yes, X_9 = Yes) = 0{,}1578$$

$$P(X_{12} = Low|X_{10} = Yes, X_9 = Yes) = 0{,}8422$$

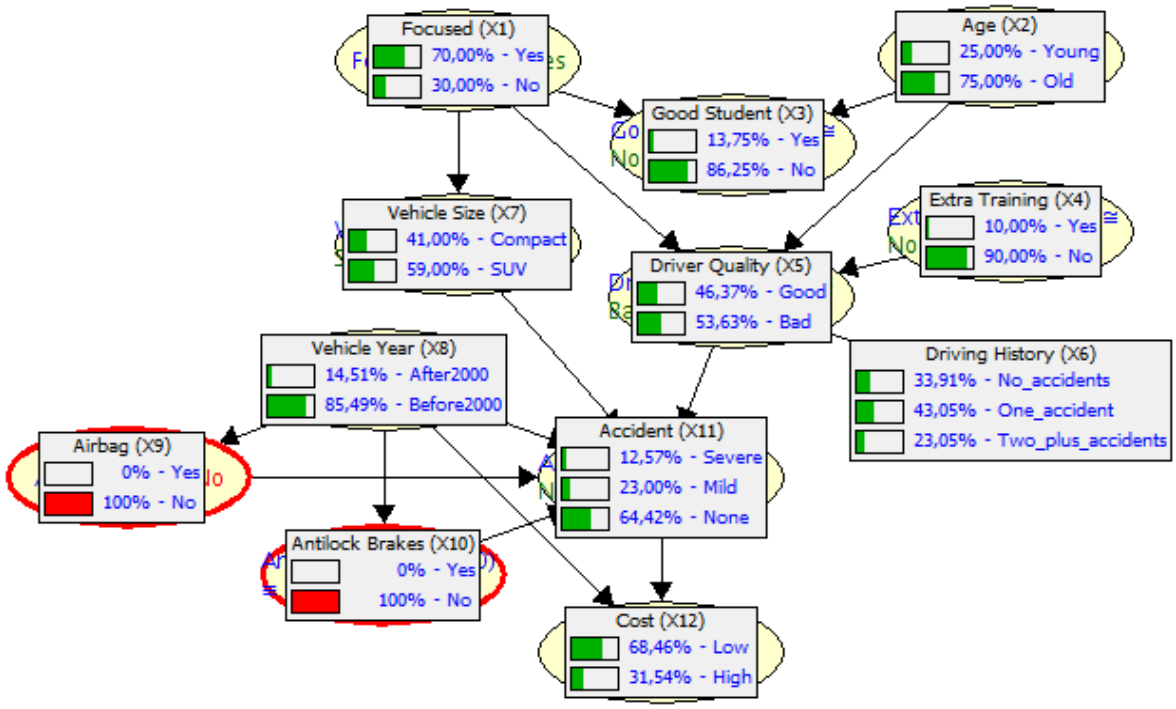


Рис. 3.3.5. Апостеріорний розподіл витрат страхової компанії

$$P(X_{12} = High|X_{10} = No, X_9 = No) = 0{,}3154$$

$$P(X_{12} = Low|X_{10} = No, X_9 = No) = 0{,}6846$$

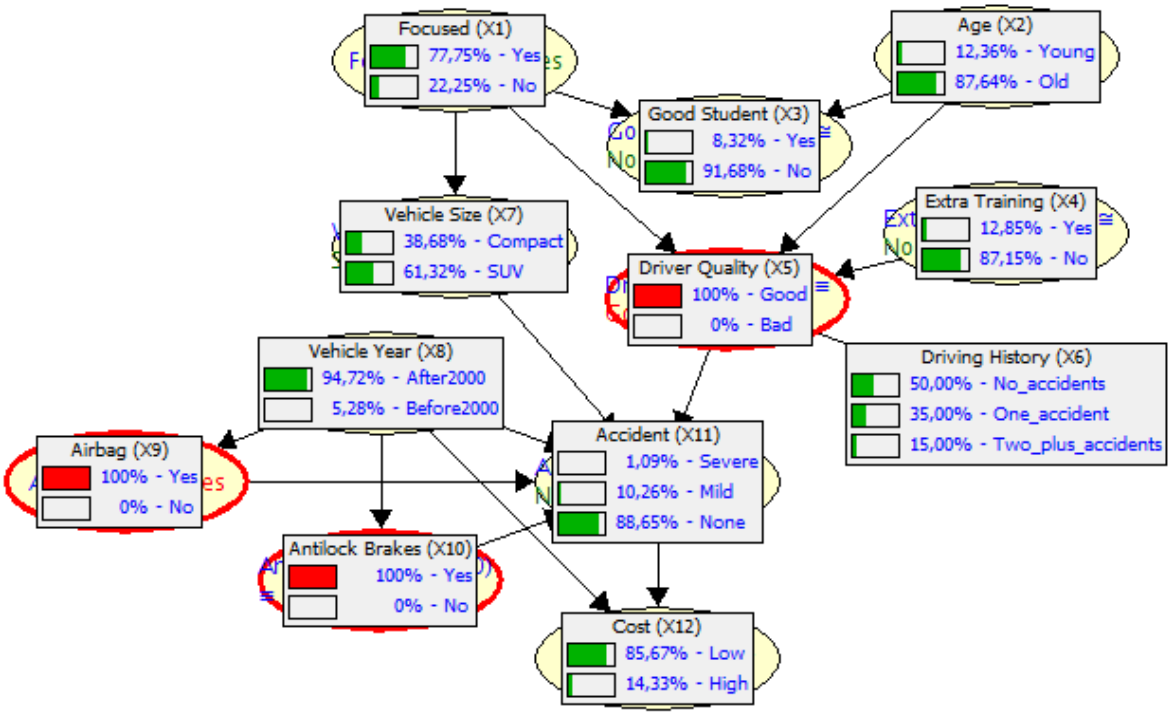


Рис. 3.3.6. Апостеріорний розподіл витрат страхової компанії

$$P(X_{12} = High|X_{10} = Yes, X_9 = Yes, X_5 = Good) = 0{,}1433,$$

$$P(X_{12} = Low|X_{10} = Yes, X_9 = Yes, X_5 = Good) = 0{,}8567$$

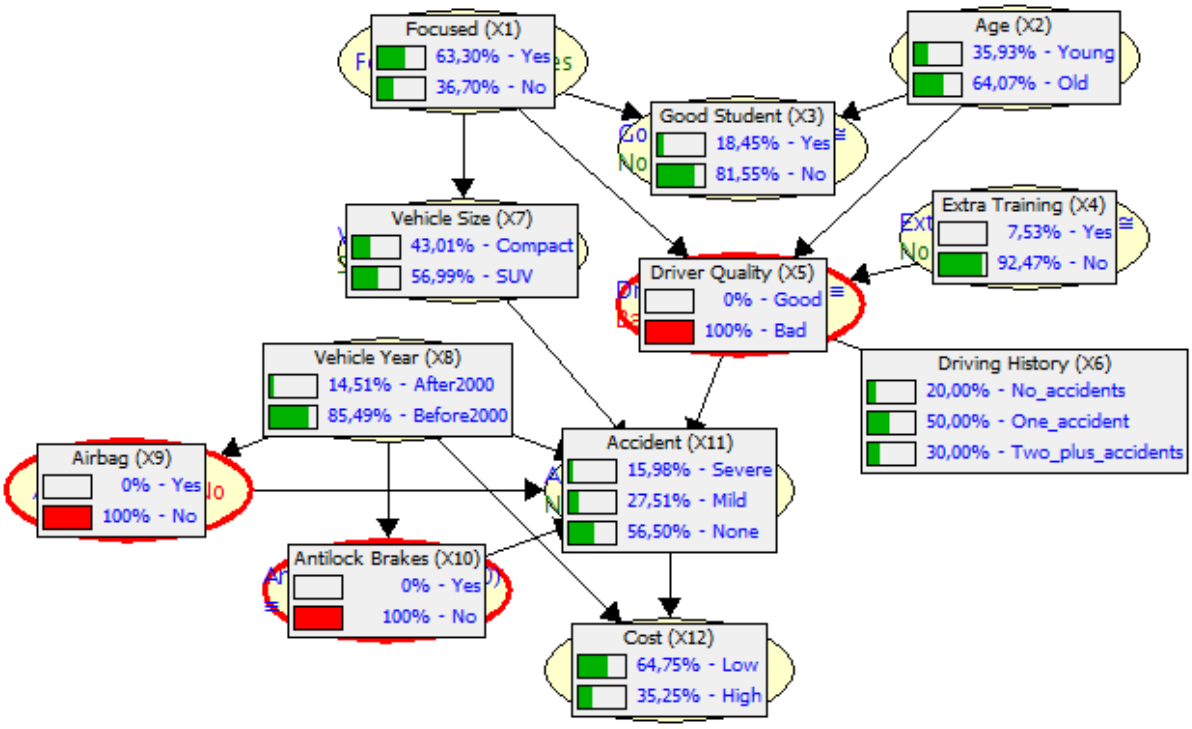


Рис. 3.3.7. Апостеріорний розподіл витрат страхової компанії

$$P(X_{12} = High|X_{10} = No, X_9 = No, X_5 = Bad) = 0{,}3525,$$

$$P(X_{12} = Low|X_{10} = No, X_9 = No, X_5 = Bad) = 0{,}6475$$

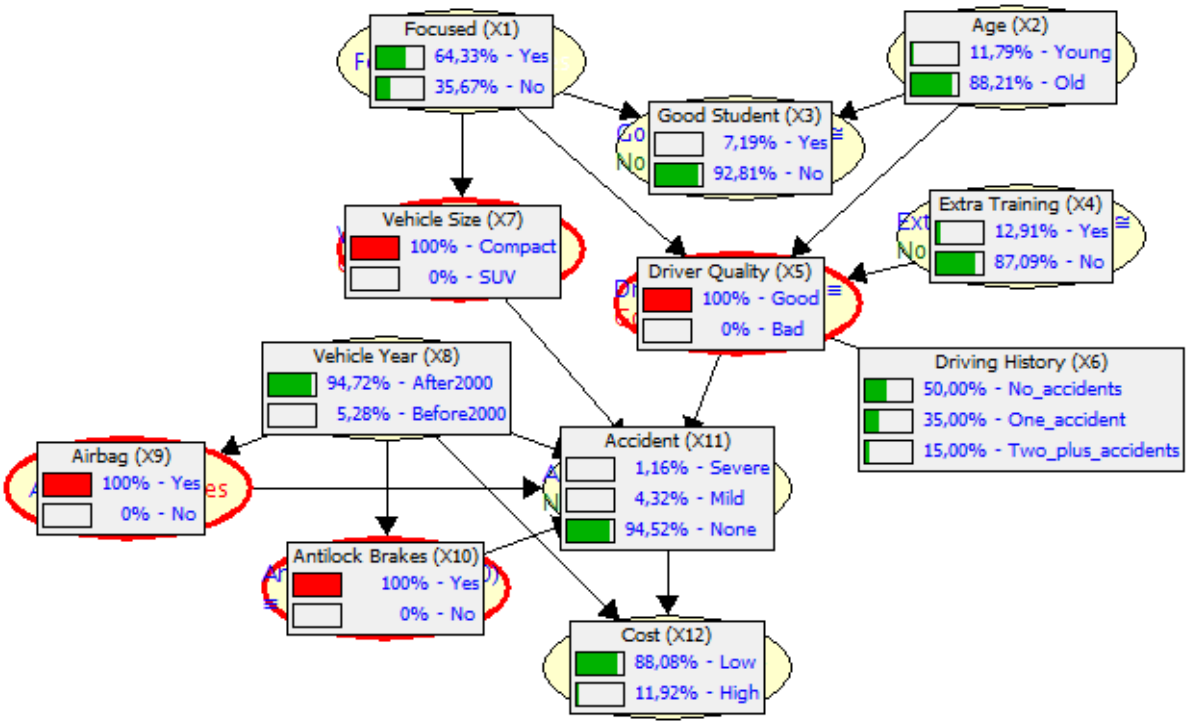


Рис. 3.3.8. Апостеріорний розподіл витрат страхової компанії

$$P(X_{12} = High|X_{10} = Yes, X_9 = Yes, X_5 = Good, X_7 = Compact) = 0{,}1192$$

$$P(X_{12} = Low|X_{10} = Yes, X_9 = Yes, X_5 = Good, X_7 = Compact) = 0{,}8808$$

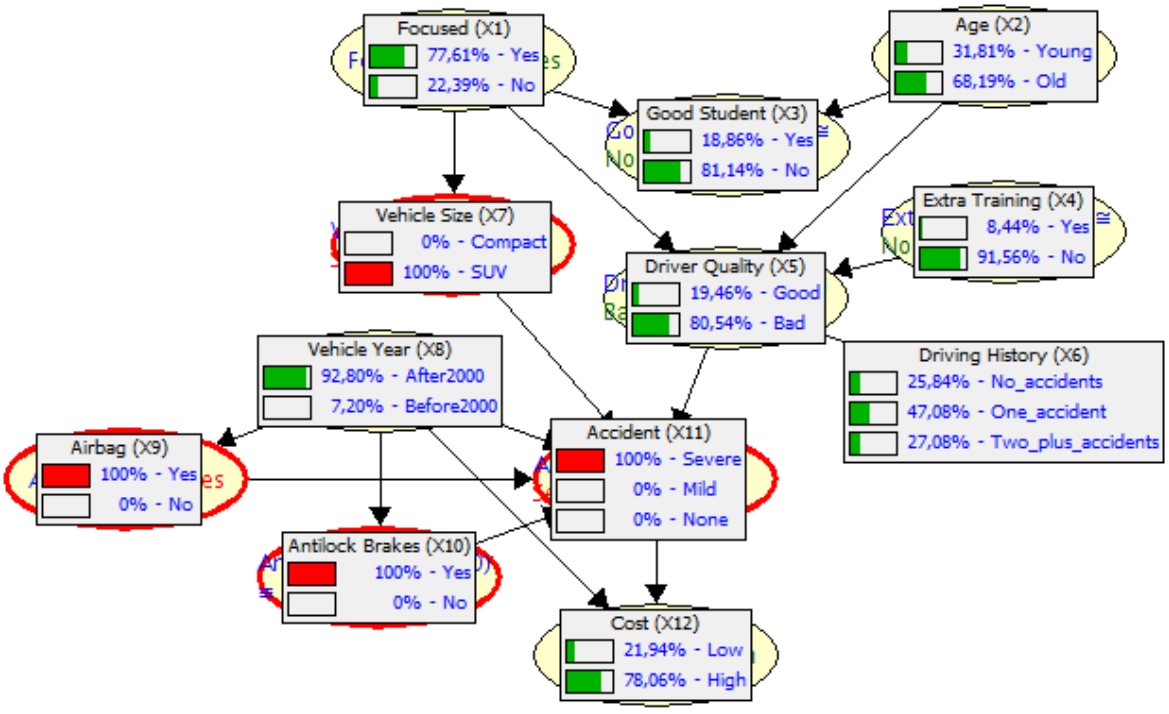


Рис. 3.3.9. Апостеріорний розподіл витрат страхової компанії

$$P(X_{12} = High|X_{10} = Yes, X_9 = Yes, X_7 = SUV, X_{11} = Severe) = 0{,}7806,$$

$$P(X_{12} = Low|X_{10} = Yes, X_9 = Yes, X_7 = SUV, X_{11} = Severe) = 0{,}2194$$

3.4. Імовірнісний висновок в байєсівській мережі Credit

Розподіл кредитоспроможності клієнта X_8 має вигляд

$$P(X_8) = \sum_{X_1,\dots,X_8} P(X_1,\dots,X_8) =$$

$$= \sum_{X_1,\dots,X_7} P(X_1)P(X_2)P(X_3|X_1,X_2)P(X_4|X_2,X_3)P(X_5)\,P(X_6|X_5)\cdot$$

$$\cdot\, P(X_7|X_5,X_6)P(X_8|X_1,X_4,X_7).$$

Для обчислення розподілу кредитоспроможності клієнта безпосереднє застосування формули множення для байєсівської мережі вимагає 10366 арифметичних операцій (9072 множень, 1294 додавань). Застосуємо алгоритм Variable Elimination. Так ми значно зменшимо число операцій та час обчислення розподілу $P(X_8)$.

$$P(X_8) = \sum_{X_7}\sum_{X_6}\sum_{X_5} P(X_5)P(X_6|X_5)P(X_7|X_5,X_6)$$

$$\sum_{X_4}\sum_{X_3}\sum_{X_2} P(X_2)P(X_4|X_2,X_3)\sum_{X_1} P(X_1)P(X_3|X_1,X_2)P(X_8|X_1,X_4,X_7).$$

Порядок виключення вершин набуває вигляду:

$$Z = \{X_1, X_2, X_3, X_4, X_5, X_6, X_7\}.$$

Виключаємо вершину X_1. Отримаємо

$$\psi_1(X_1,X_2,X_3,X_4,X_7,X_8) = P(X_1)P(X_3|X_1,X_2)P(X_8|X_1,X_4,X_7),$$

$$\tau_1(X_2,X_3,X_4,X_7,X_8) = \sum_{X_1} \psi_1(X_1,X_2,X_3,X_4,X_7,X_8),$$

$$P(X_8) = \sum_{X_7}\sum_{X_6}\sum_{X_5} P(X_5)P(X_6|X_5)P(X_7|X_5,X_6)$$

$$\sum_{X_4}\sum_{X_3}\sum_{X_2} P(X_2)P(X_4|X_2,X_3)\tau_1(X_2,X_3,X_4,X_7,X_8).$$

Виключаємо вершину X_2. Отримаємо

$$\psi_2(X_2,X_3,X_4,X_7,X_8) = P(X_2)P(X_4|X_2,X_3)\tau_1(X_2,X_3,X_4,X_7,X_8),$$

$$\tau_2(X_3, X_4, X_7, X_8) = \sum_{X_2} \psi_2(X_2, X_3, X_4, X_7, X_8),$$

$$P(X_8) = \sum_{X_7} \sum_{X_6} \sum_{X_5} P(X_5)P(X_6|X_5)P(X_7|X_5, X_6) \sum_{X_4} \sum_{X_3} \tau_2(X_3, X_4, X_7, X_8).$$

Виключаємо вершину X_3. Отримаємо

$$\psi_3(X_3, X_4, X_7, X_8) = \tau_2(X_3, X_4, X_7, X_8),$$

$$\tau_3(X_4, X_7, X_8) = \sum_{X_3} \psi_3(X_3, X_4, X_7, X_8),$$

$$P(X_8) = \sum_{X_7} \sum_{X_6} \sum_{X_5} P(X_5)P(X_6|X_5)P(X_7|X_5, X_6) \sum_{X_4} \tau_3(X_4, X_7, X_8).$$

Виключаємо вершину X_4. Отримаємо

$$\psi_4(X_4, X_7, X_8) = \tau_3(X_4, X_7, X_8),$$

$$\tau_4(X_7, X_8) = \sum_{X_4} \psi_4(X_4, X_7, X_8),$$

$$P(X_8) = \sum_{X_7} \tau_4(X_7, X_8) \sum_{X_6} \sum_{X_5} P(X_5)P(X_6|X_5)P(X_7|X_5, X_6).$$

Виключаємо вершину X_5. Отримаємо

$$\psi_5(X_5, X_6, X_7) = P(X_5)P(X_6|X_5)P(X_7|X_5, X_6),$$

$$\tau_5(X_6, X_7) = \sum_{X_5} \psi_5(X_5, X_6, X_7),$$

$$P(X_8) = \sum_{X_7} \tau_4(X_7, X_8) \sum_{X_6} \tau_5(X_6, X_7).$$

Виключаємо вершину X_6. Отримаємо

$$\psi_6(X_6, X_7) = \tau_5(X_6, X_7),$$

$$\tau_6(X_7) = \sum_{X_6} \psi_6(X_6, X_7),$$

$$P(X_8) = \sum_{X_7} \tau_4(X_7, X_8)\tau_6(X_7).$$

Виключаємо вершину X_7. Отримаємо

$$\psi_7(X_7) = \tau_4(X_7, X_8)\tau_6(X_7),$$

$$\tau_7(X_8) = \sum_{X_7} \psi_7(X_7),\ P(X_8) = \tau_7(X_8).$$

Таблиця 3.4.1. Алгоритм Variable Elimination

Крок	Виключена змінна	Використані множники	Використані змінні	Новий множник
1	X_1	$P(X_1), P(X_3\|X_1, X_2),$ $P(X_8\|X_1, X_4, X_7)$	$X_1, X_2, X_3, X_4,$ X_7, X_8	$\tau_1(X_2, X_3, X_4, X_7, X_8)$
2	X_2	$\tau_1(X_2, X_3, X_4, X_7, X_8)$ $P(X_2), P(X_4\|X_2, X_3)$	$X_2, X_3, X_4,$ X_7, X_8	$\tau_2(X_3, X_4, X_7, X_8)$
3	X_3	$\tau_2(X_3, X_4, X_7, X_8)$	X_3, X_4, X_7, X_8	$\tau_3(X_4, X_7, X_8)$
4	X_4	$\tau_3(X_4, X_7, X_8)$	X_4, X_7, X_8	$\tau_4(X_7, X_8)$
5	X_5	$P(X_5), P(X_6\|X_5),$ $P(X_7\|X_5, X_6)$	X_5, X_6, X_7	$\tau_5(X_6, X_7)$
6	X_6	$\tau_5(X_6, X_7)$	X_6, X_7	$\tau_6(X_7)$
7	X_7	$\tau_4(X_7, X_8), \tau_6(X_7)$	X_7	$\tau_7(X_8)$

Реалізація алгоритму Variable Elimination для обчислення розподілів вершин байєсівської мережі Credit наведена в додатку 3.

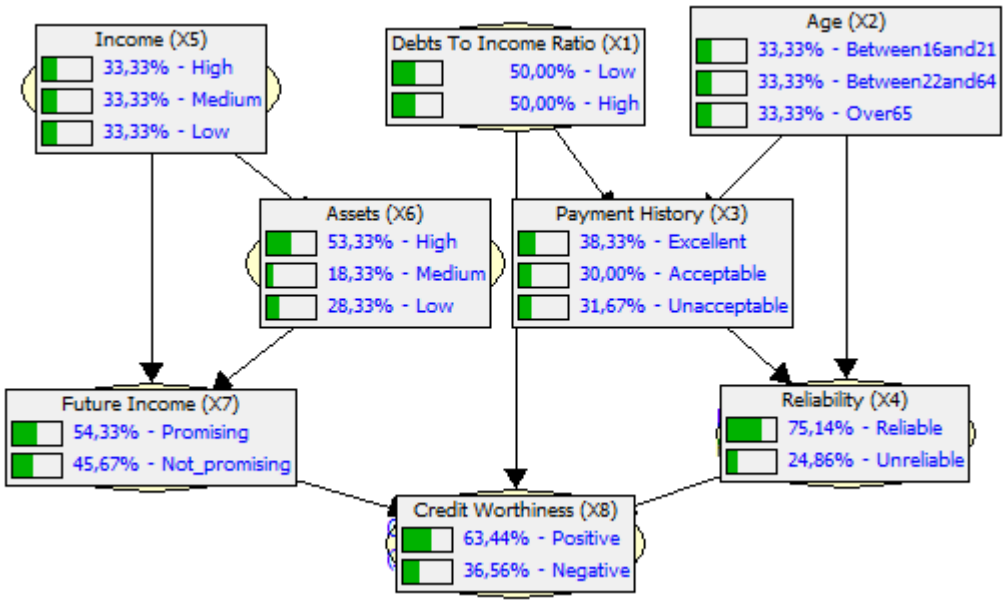


Рис. 3.4.1. Розподіл кредитоспроможності клієнта

4. Моделі міркувань

4.1. Моделі міркувань в байєсівській мережі Insurance

Розглянемо *причинно-наслідкові моделі міркувань* в мережі Insurance. Всі апріорні та апостеріорні розподіли вершин обчислюються за допомогою алгоритму Variable Elimination, при цьому порядок виключення вершин знаходиться згідно з алгоритмом Greedy-Ordering.

Апостеріорний розподіл вершини $Driver\ Quality\ (X_5)$ за умов заданого стану вершини $Extra\ Training\ (X_4)$ дорівнює:

$$P(X_5|X_4) = \frac{P(X_5, X_4)}{P(X_4)},$$

де спільний розподіл вершин X_5, X_4 обчислюється за формулою

$$P(X_5, X_4) = \sum_{X_1,X_2,X_3,X_6,X_7,X_8,X_9,X_{10},X_{11},X_{12}} P(X_1, \dots, X_{12}).$$

1. Імовірність високої кваліфікації водія *за умови*, що водій пройшов курс екстремального водіння дорівнює

$$P(X_5 = Good|X_4 = Yes) = \frac{P(X_5 = Good, X_4 = Yes)}{P(X_4 = Yes)} = 0{,}5959.$$

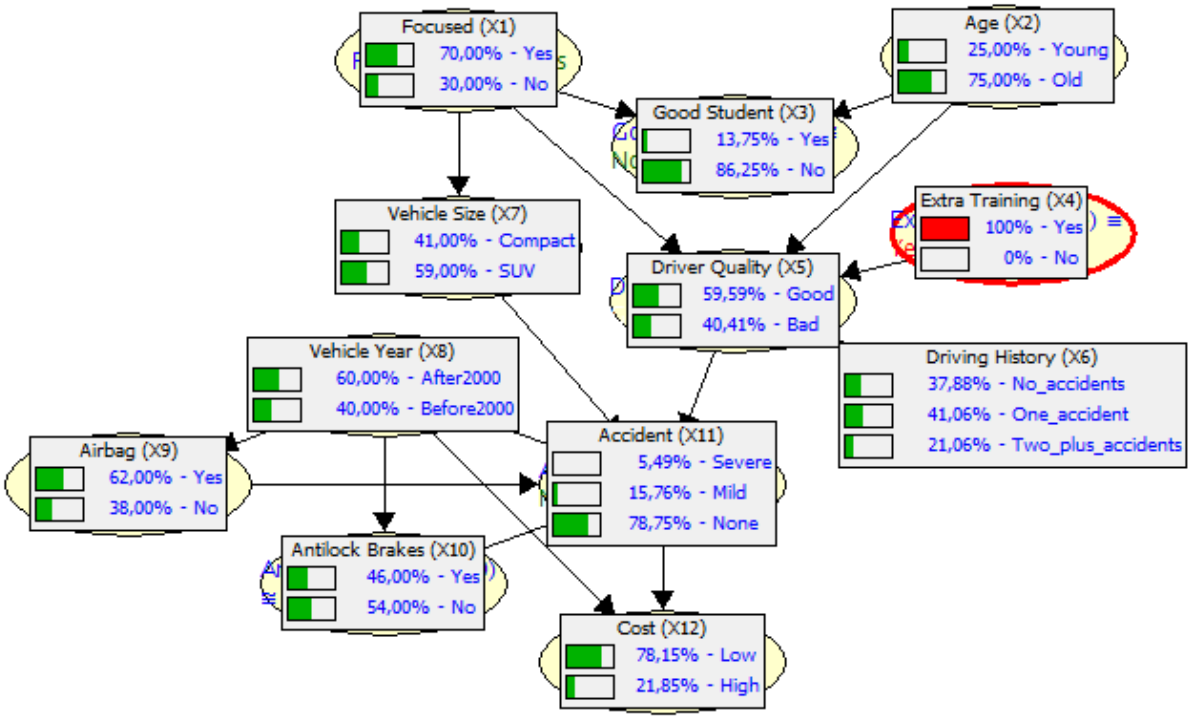


Рис. 4.1.1. До визначення $P(X_5 = Good|X_4 = Yes)$

Імовірність високої кваліфікації водія *за умови*, що водій пройшов курс екстремального водіння збільшилась з $P(X_5 = Good) = 0{,}4637$ (див. рис. 3.3.1) до 0,5959.

2. Імовірність високої кваліфікації водія *за умови*, що водій не пройшов курс екстремального водіння дорівнює

$$P(X_5 = Good | X_4 = No) = \frac{P(X_5 = Good, X_4 = No)}{P(X_4 = No)} = 0{,}449.$$

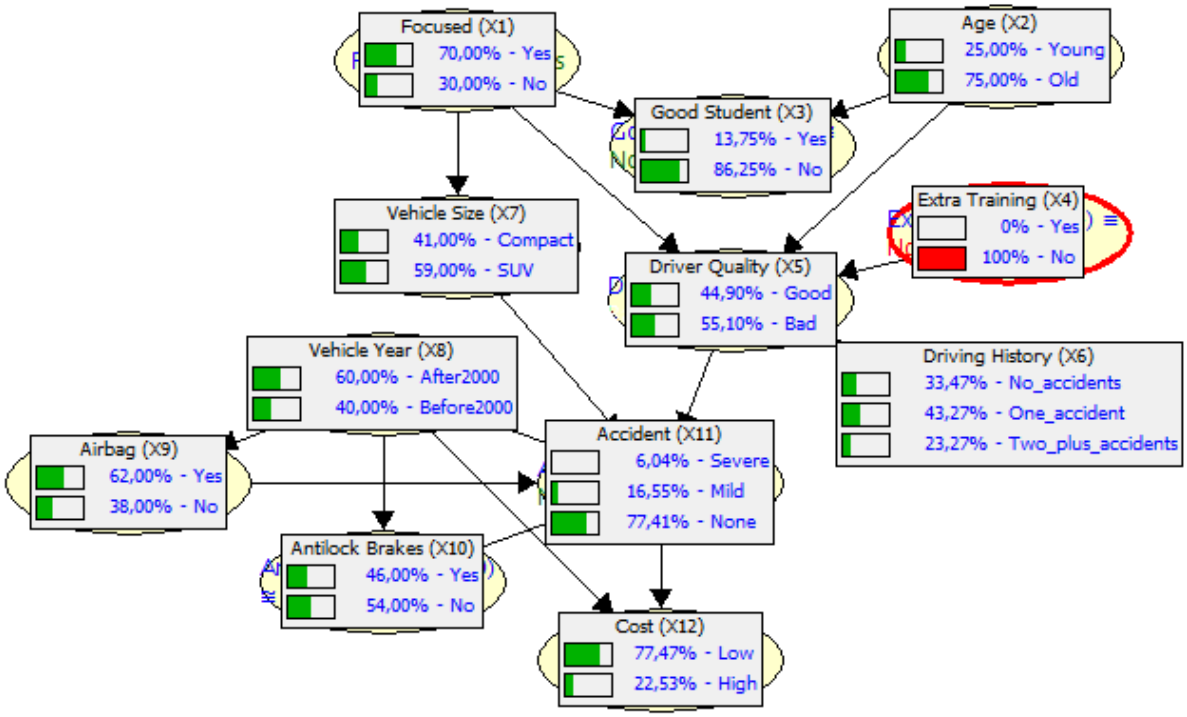


Рис. 4.1.2. До визначення $P(X_5 = Good | X_4 = No)$

Імовірність високої кваліфікації водія *за умови*, що водій не пройшов курс екстремального водіння зменшилась з $P(X_5 = Good) = 0{,}4637$ до 0,449.

Апостеріорний розподіл вершини $Driver\ Quality(X_5)$ за умов заданих станів вершин $Age\ (X_2)$ та $Extra\ Trainig\ (X_4)$ дорівнює

$$P(X_5 | X_4, X_2) = \frac{P(X_5, X_4, X_2)}{P(X_4, X_2)},$$

де спільні розподіли вершин обчислюються за формулами

$$P(X_5, X_4, X_2) = \sum_{X_1, X_3, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}} P(X_1, \dots, X_{12}),$$

$$P(X_4, X_2) = P(X_4) P(X_2).$$

1. Імовірність високої кваліфікації водія *за умов*, що водій похилого віку та пройшов курс екстремального водіння дорівнює

$$P(X_5 = Good|X_4 = Yes, X_2 = Old) =$$

$$= \frac{P(X_5 = Good, X_4 = Yes, X_2 = Old)}{P(X_4 = Yes, X_2 = Old)} = 0{,}696.$$

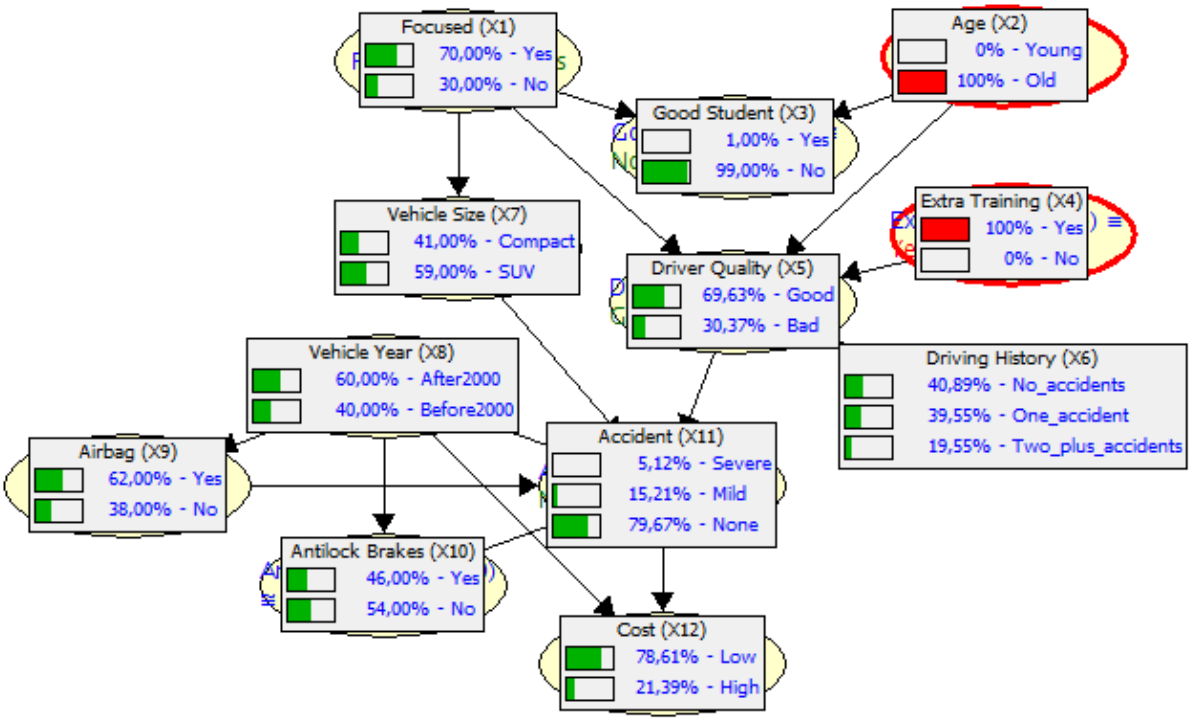


Рис. 4.1.3. До визначення $P(X_5 = Good|X_4 = Yes, X_2 = Old)$

Імовірність високої кваліфікації водія *за умов*, що водій похилого віку та пройшов курс екстремального водіння зросла з $P(X_5 = Good) = 0{,}4637$ до $0{,}6963$.

2. Імовірність високої кваліфікації водія *за умов*, що водій молодий та не пройшов курс екстремального водіння

$$P(X_5 = Good|X_4 = No, X_2 = Young) =$$

$$= \frac{P(X_5 = Good, X_4 = No, X_2 = Young)}{P(X_4 = No, X_2 = Young)} = 0{,}2219.$$

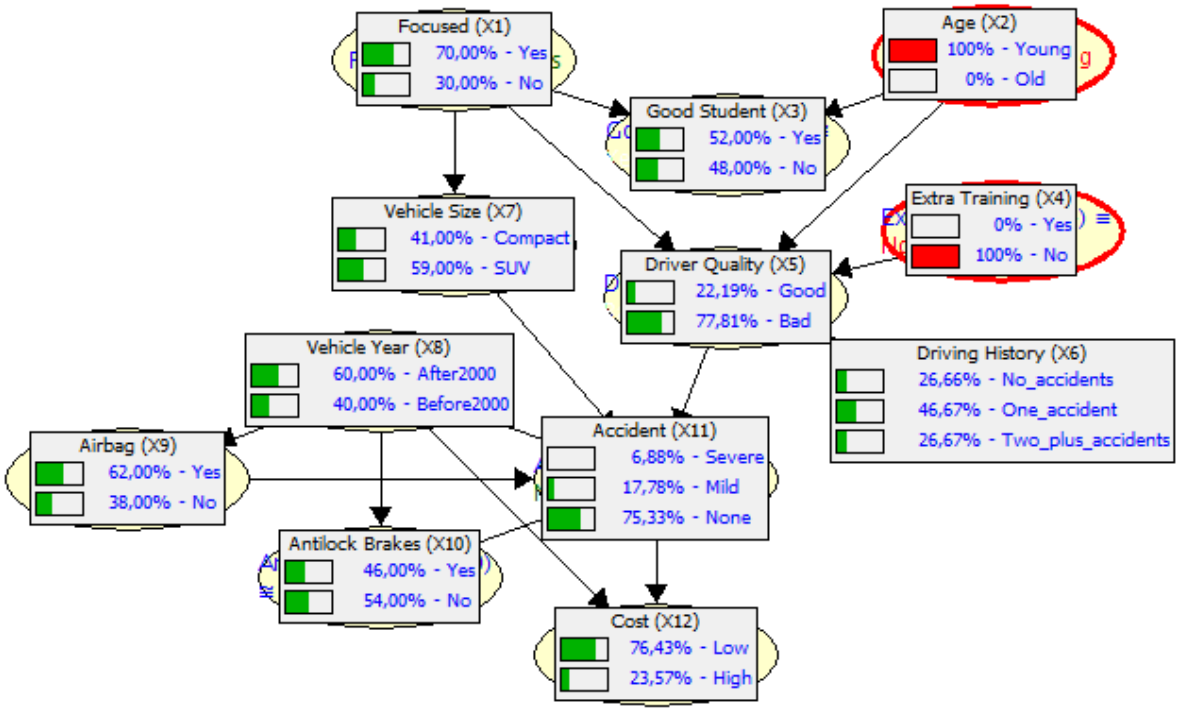


Рис. 4.1.4. До визначення $P(X_5 = Good|X_4 = Yes, X_2 = Young)$

Імовірність високої кваліфікації водія *за умов*, що водій молодий та не пройшов курс екстремального водіння зменшилася з 0,4637 до 0,2219.

Розглянемо *наслідково-причинні моделі міркувань* в мережі Insurance. Всі апріорні та апостеріорні розподіли вершин обчислюються за допомогою алгоритму Variable Elimination, при цьому порядок виключення вершин знаходиться згідно з алгоритмом Greedy-Ordering.

Апостеріорний розподіл вершини $Extra\ Trainig\ (X_4)$ за умов заданого стану вершини $Driver\ Quality\ (X_5)$ дорівнює

$$P(X_4|X_5) = \frac{P(X_5, X_4)}{P(X_5)},$$

де спільний розподіл вершин X_5, X_4 обчислюється за формулою

$$P(X_5, X_4) = \sum_{X_1, X_2, X_3, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}} P(X_1, \dots, X_{12}).$$

1. Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій є висококваліфікованим фахівцем дорівнює

$$P(X_4 = Yes|X_5 = Good) = \frac{P(X_5 = Good, X_4 = Yes)}{P(X_5 = Good)} = 0{,}1285.$$

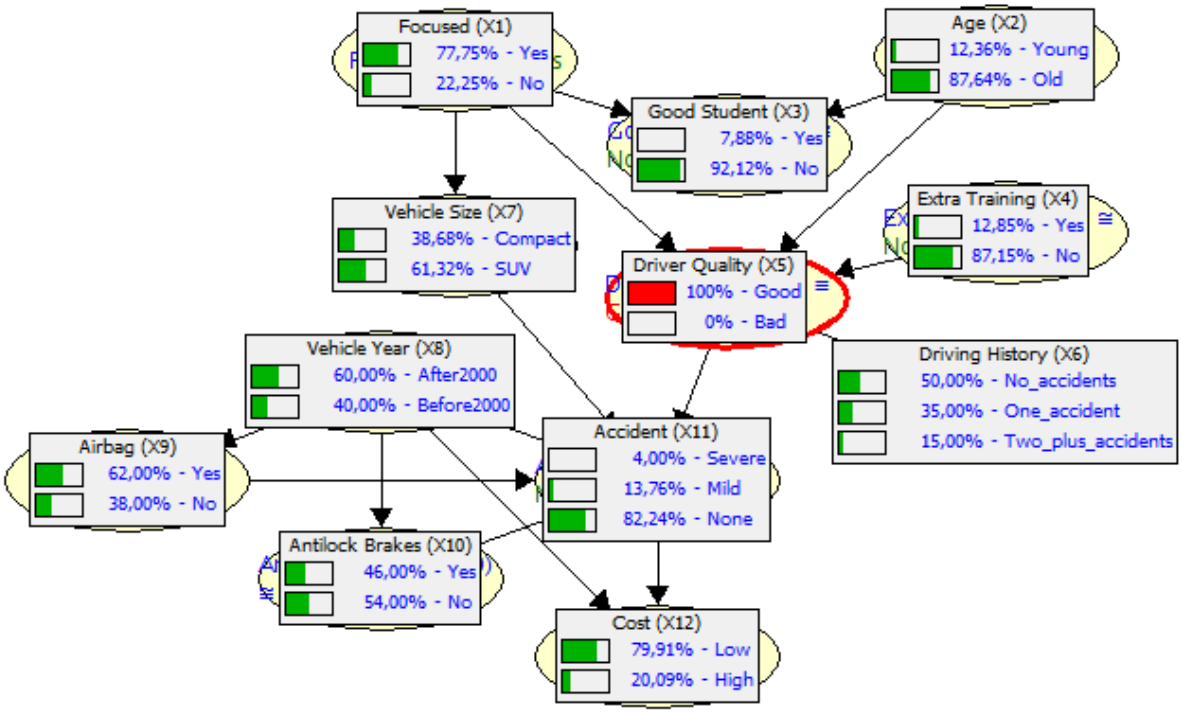


Рис. 4.1.5. До визначення $P(X_4 = Yes|X_5 = Good)$

Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій є висококваліфікованим фахівцем збільшилась з $P(X_4 = Yes) = 0{,}1$ (див. рис. 3.3.1) до $P(X_4 = Yes|X_5 = Good) = 0{,}1285$.

2. Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій не є висококваліфікованим фахівцем дорівнює

$$(X_4 = Yes|X_5 = Bad) = \frac{P(X_5 = Bad, X_4 = Yes)}{P(X_5 = Bad)} = 0{,}0753.$$

Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій не є висококваліфікованим фахівцем зменшилась з $P(X_4 = Yes) = 0{,}1$ до $P(X_4 = Yes|X_5 = Bad) = 0{,}0753$.

Розглянемо *змішані моделі міркувань* в мережі Insurance.

Апостеріорний розподіл вершини $Extra\ Trainig\ (X_4)$ за умов заданих станів вершин $Driver\ Quality\ (X_5)$, $Age(X_2)$ дорівнює

$$P(X_4|X_5, X_2) = \frac{P(X_5, X_4, X_2)}{P(X_5, X_2)},$$

де спільні розподіли вершин обчислюються за формулами

$$P(X_5, X_4, X_2) = \sum_{X_1, X_3, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}} P(X_1, \dots, X_{12}).$$

$$P(X_5, X_2) = \sum_{X_1, X_3, X_4, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}} P(X_1, \dots, X_{12}).$$

1. Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій молодий та висококваліфікований фахівець дорівнює

$$P(X_4 = Yes | X_5 = Good, X_2 = Young) =$$

$$= \frac{P(X_4 = Yes, X_5 = Good, X_2 = Young)}{P(X_5 = Good, X_2 = Young)} = 0{,}1285.$$

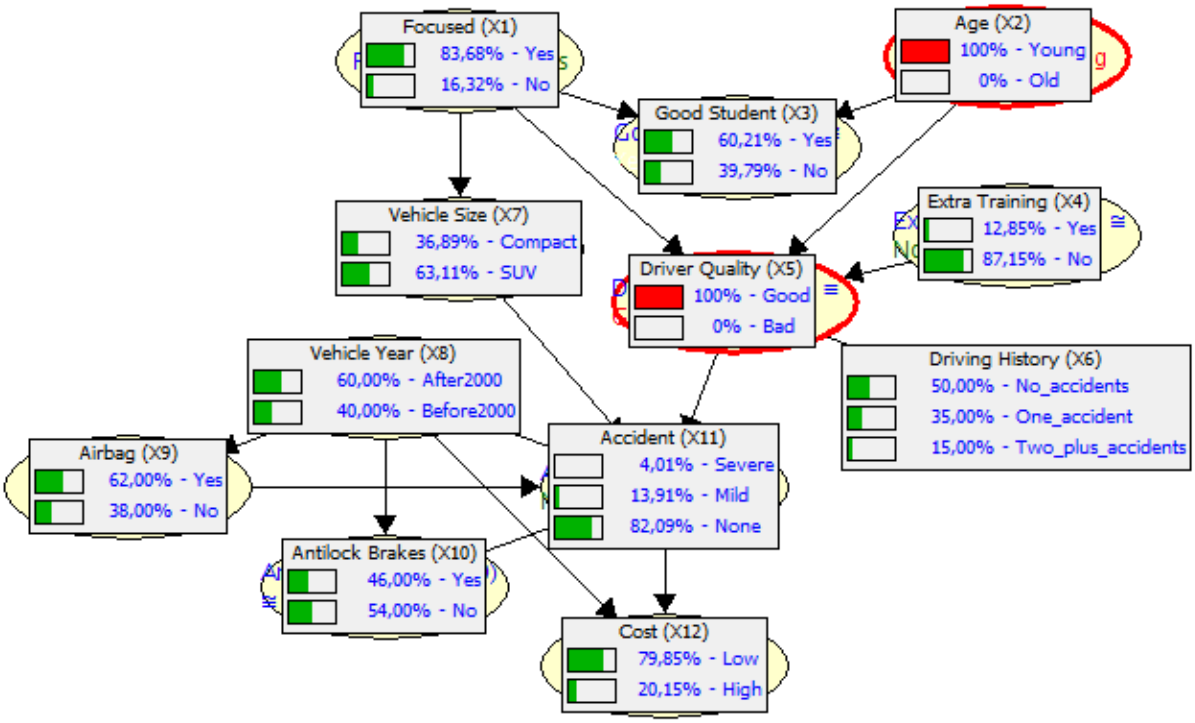


Рис. 4.1.6. До визначення $P(X_4 = Yes | X_5 = Good, X_2 = Young)$

Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій молодий та висококваліфікований фахівець збільшилася з $P(X_4 = Yes) = 0{,}1$ до $P(X_4 = Yes | X_5 = Good, X_2 = Young) = 0{,}1285$.

2. Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій молодий та не є висококваліфікованим фахівцем:

$$P(X_4 = Yes | X_5 = Bad, X_2 = Young) =$$

$$= \frac{P(X_4 = Yes, X_5 = Bad, X_2 = Young)}{P(X_5 = Bad, X_2 = Young)} = 0{,}0915.$$

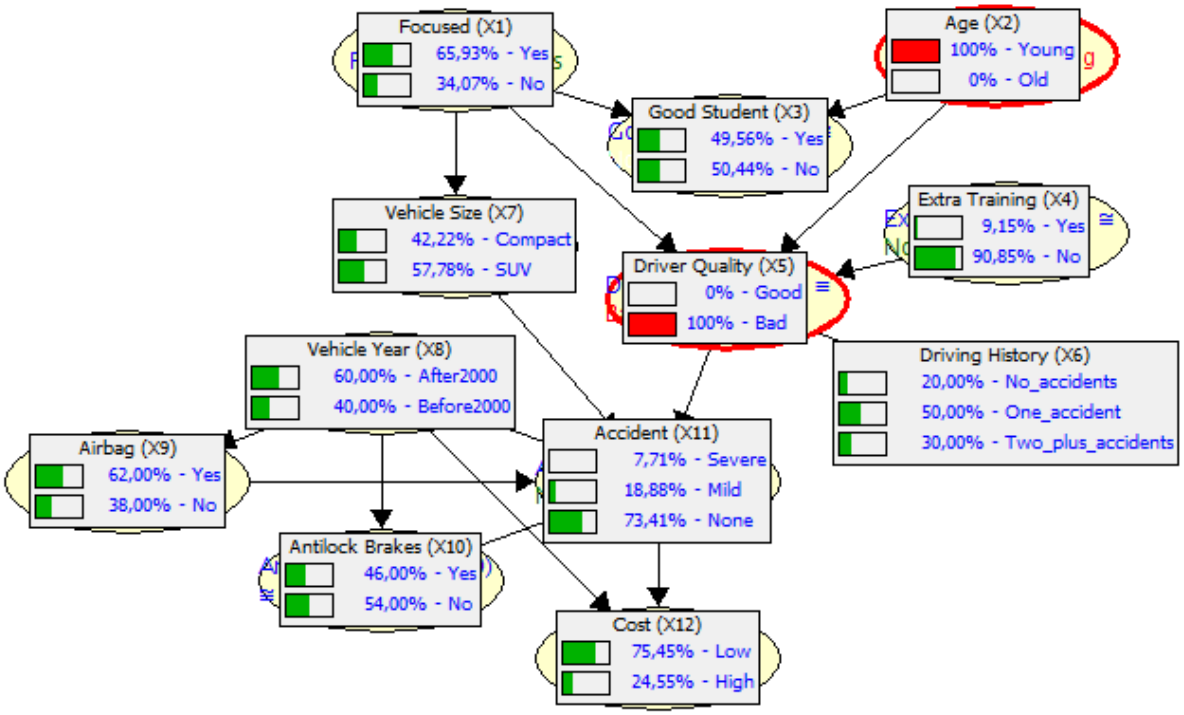


Рис. 4.1.7. До визначення $P(X_4 = Yes|X_5 = Bad, X_2 = Young)$

Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій молодий та не є висококваліфікованим фахівцем зменшилася з $P(X_4 = Yes) = 0{,}1$ до $P(X_4 = Yes|X_5 = Bad, X_2 = Young) = 0{,}0915$.

Апостеріорний розподіл вершини $Extra\ Trainig\ (X_4)$ за умов заданих станів вершин $Driver\ Quality\ (X_5)$, $Good\ Student\ (X_3)$ дорівнює

$$P(X_4|X_5, X_3) = \frac{P(X_5, X_4, X_3)}{P(X_5, X_3)},$$

де спільні розподіли вершин обчислюються за формулами

$$P(X_5, X_4, X_3) = \sum_{X_1, X_2, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}} P(X_1, \dots, X_{12}),$$

$$P(X_5, X_3) = \sum_{X_1, X_2, X_4, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}} P(X_1, \dots, X_{12}).$$

1. Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій висококваліфікований фахівець та добре навчався в автошколі збільшилась з $P(X_4 = Yes) = 0{,}1$ до $P(X_4 = Yes|X_5 = Good, X_3 = Yes) = 0{,}1247$.

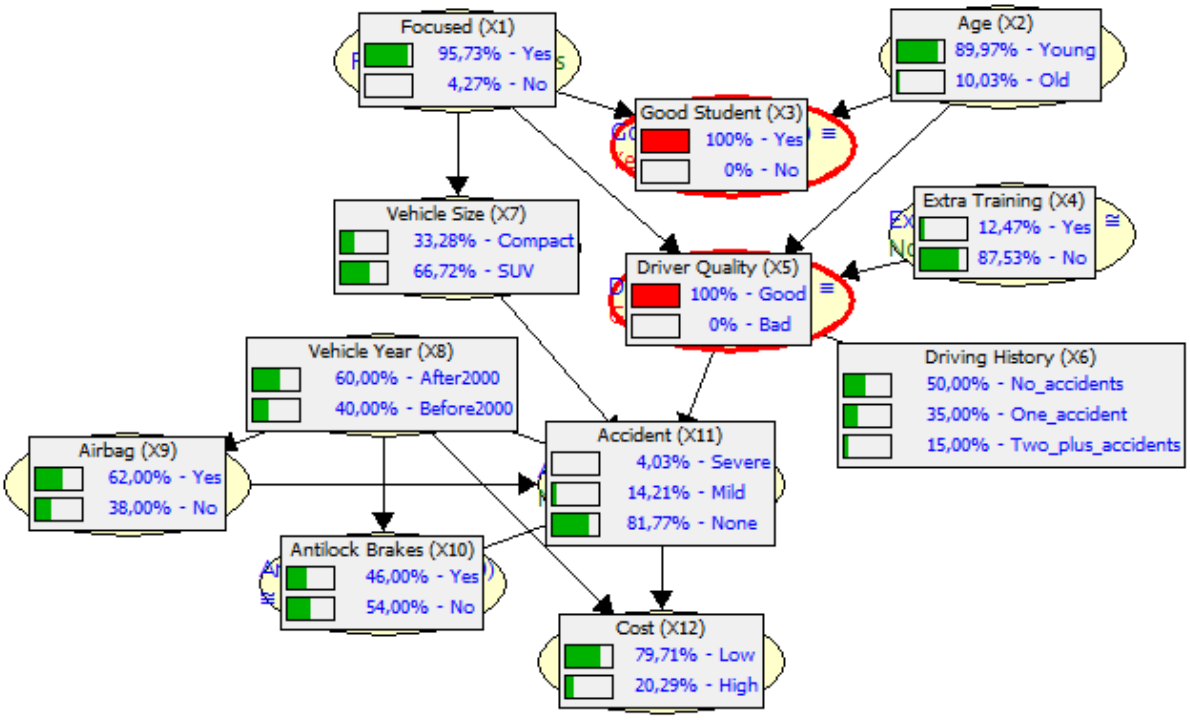


Рис. 4.1.8. До визначення $P(X_4 = Yes|X_5 = Good, X_3 = Yes)$

2. Імовірність проходження водієм курсу екстремального водіння *за умови*, що водій не є висококваліфікованим фахівцем та погано навчався в автошколі зменшилась з $P(X_4 = Yes) = 0{,}1$ до $0{,}073$.

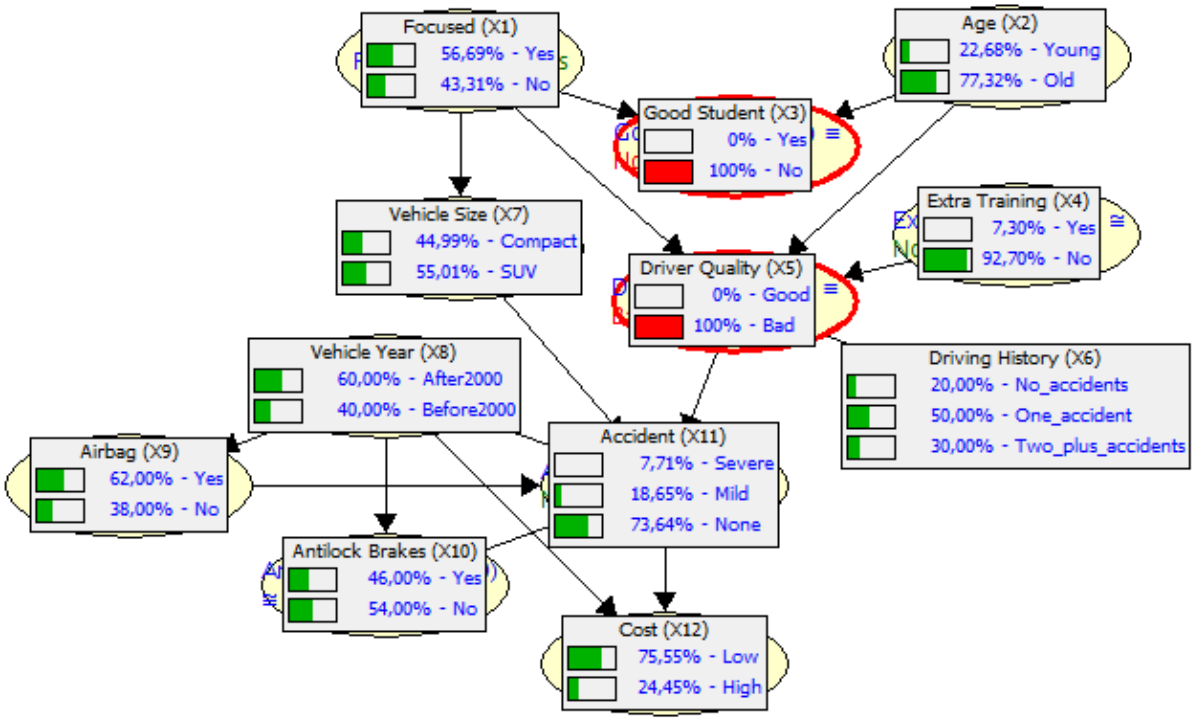


Рис. 4.1.9. До визначення $P(X_4 = Yes|X_5 = Bad, X_3 = No)$

4.2. Моделі міркувань в байєсівській мережі Credit

Розглянемо *причинно-наслідкові моделі міркувань* в мережі Credit. Всі апріорні та апостеріорні розподіли вершин обчислюються за допомогою алгоритму Variable Elimination, при цьому порядок виключення вершин знаходиться згідно з алгоритмом Greedy-Ordering.

Апостеріорний розподіл вершини $Payment\ History\ (X_3)$ за умов заданих станів вершин $Ratio\ of\ Debts\ To\ Income\ (X_1)$ та $Age\ (X_2)$ дорівнює:

$$P(X_3|X_1, X_2) = \frac{P(X_3, X_1, X_2)}{P(X_1, X_2)},$$

де спільні розподіли вершин X_3, X_1, X_2 та X_1, X_2 обчислюються за формулами

$$P(X_3, X_1, X_2) = \sum_{X_4, X_5, X_6, X_7, X_8} P(X_1, \dots, X_8),$$

$$P(X_1, X_2) = \sum_{X_3, X_4, X_5, X_6, X_7, X_8} P(X_1, \dots, X_8).$$

1. Імовірність того, що клієнт має незадовільну історію платежів *за умови,* що відношення заборгованості до доходу високе та вік клієнта між 16 та 21, зросла з $P\{X_3 = Unacceptable\} = 0{,}3167$ (див. рис. 3.4.1) до 0,7.

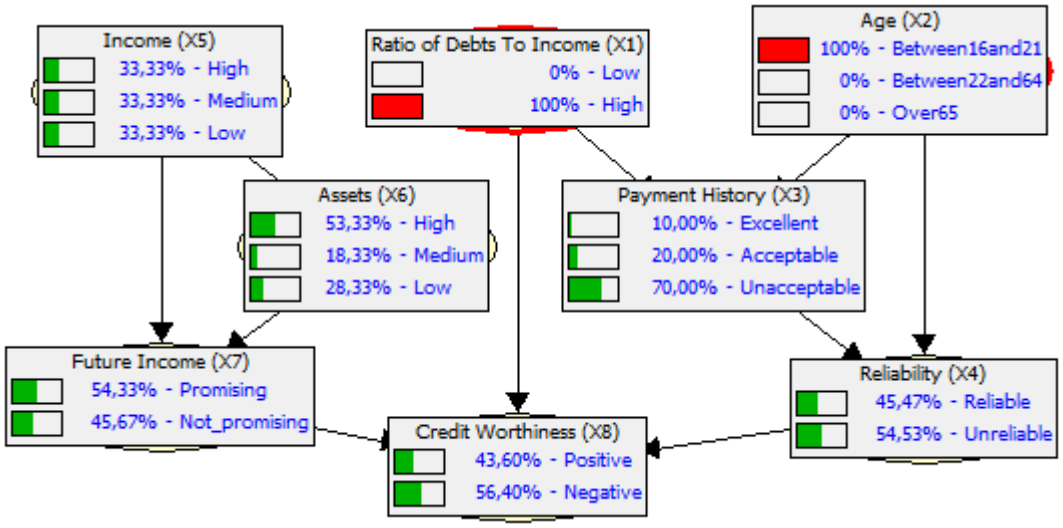


Рис. 4.2.1. До визначення

$$P(X_3 = Unacceptable|X_1 = High, X_2 = Between16and21)$$

2. Імовірність того, що клієнт має незадовільну історію платежів *за умови,* що відношення заборгованості до доходу низьке та вік клієнта між 16 та 21, зросла з $P\{X_3 = Unacceptable\} = 0{,}3167$ до 0,5.

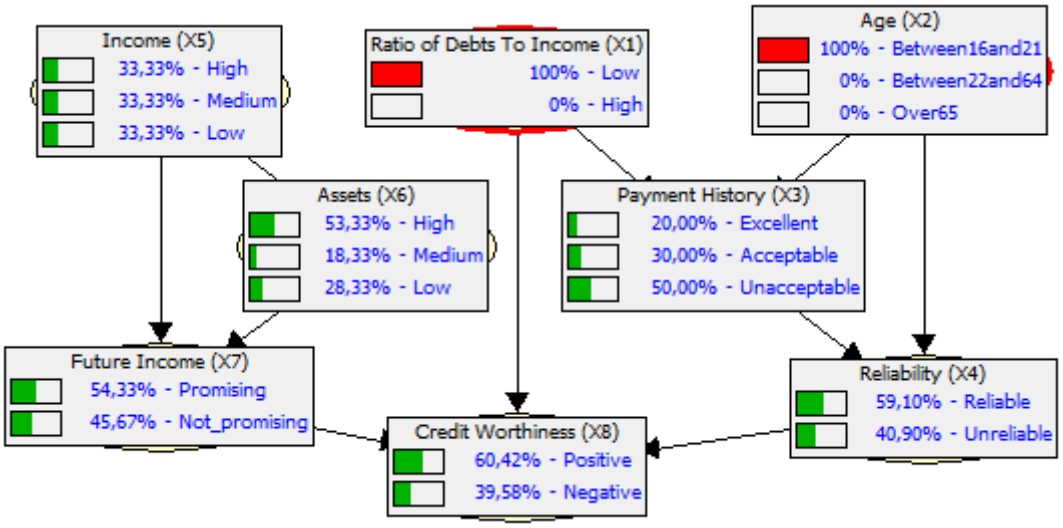


Рис. 4.2.2. До визначення

$$P(X_3 = Unacceptable | X_1 = Low, X_2 = Between16and21)$$

3. Імовірність того, що клієнт має незадовільну історію платежів *за умови,* що відношення заборгованості до доходу низьке та вік клієнта більше 65, зменшилася з $P\{X_3 = Unacceptable\} = 0{,}3167$ до 0,1.

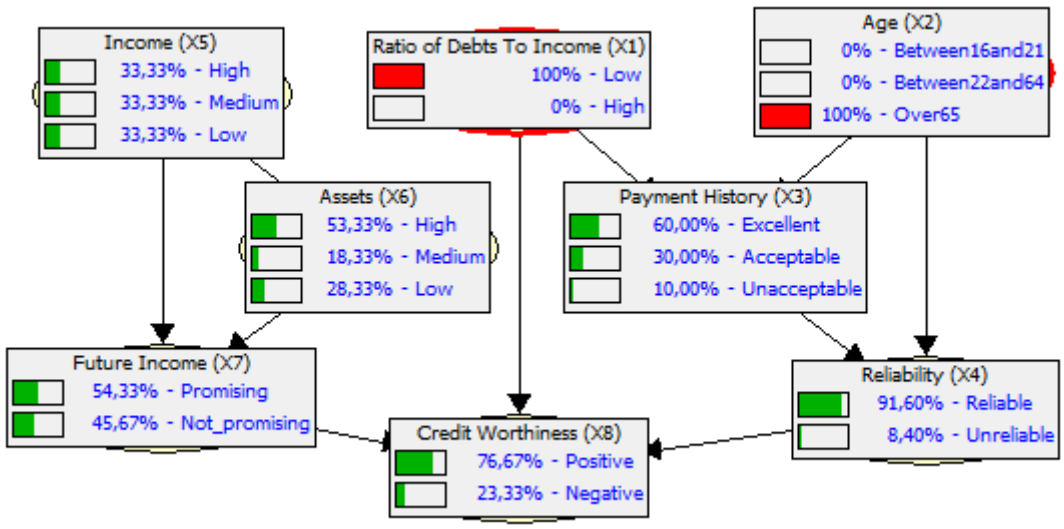


Рис. 4.2.3. До визначення

$$P(X_3 = Unacceptable\ | X_1 = Low, X_2 = Over65)$$

Розглянемо *наслідково-причинні моделі міркувань* в мережі Credit. Всі апріорні та апостеріорні розподіли вершин обчислюються за допомогою алгоритму Variable Elimination, при цьому порядок виключення вершин знаходиться згідно з алгоритмом Greedy-Ordering.

Апостеріорний розподіл вершини $Payment\ History\ (X_3)$ за умов заданого стану вершини $Reliability\ (X_4)$ дорівнює:

$$P(X_3|X_4) = \frac{P(X_3, X_4)}{P(X_4)},$$

де спільний розподіл вершин X_3, X_4 обчислюється за формулою

$$P(X_3, X_4) = \sum_{X_1,X_2,X_5,X_6,X_7,X_8} P(X_1, \dots, X_8),$$

а розподіл вершини X_4 обчислюється за формулою

$$P(X_4) = \sum_{X_1,X_2,X_3,X_5,X_6,X_7,X_8} P(X_1, \dots, X_8).$$

1. Імовірність того, що клієнт має незадовільну історію платежів *за умови,* що клієнт є надійним, зменшилася з $P\{X_3 = Unacceptable\} = 0{,}3167$ до $P(X_3 = Unacceptable\ |X_4 = Reliable) = 0{,}1484$.

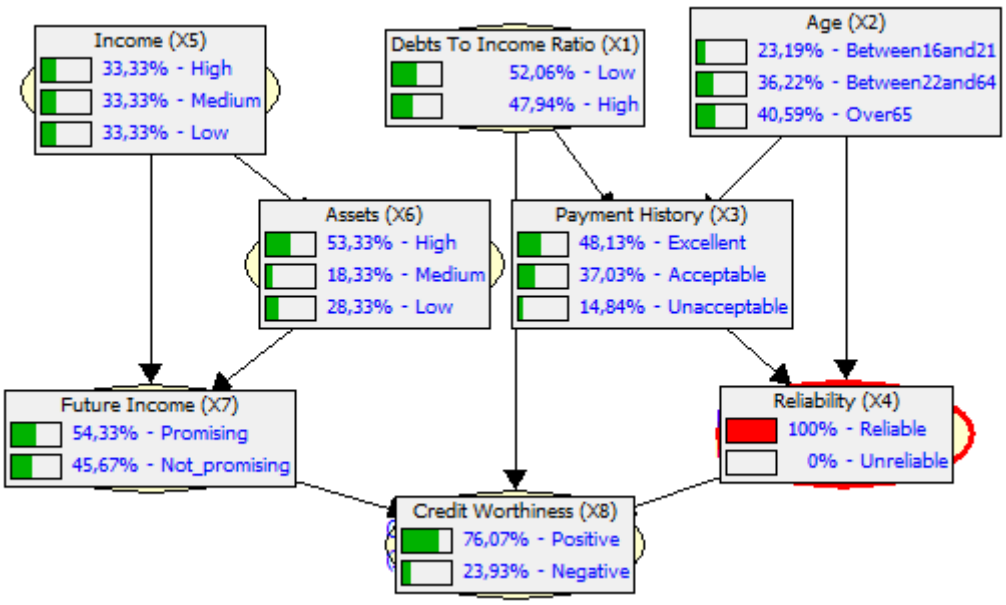


Рис. 4.2.4. До визначення

$P(X_3 = Unacceptable\ |X_4 = Reliable)$

2. Імовірність того, що клієнт має незадовільну історію платежів *за умови,* що клієнт не є надійним, збільшилась з $P\{X_3 = Unacceptable\} = 0{,}3167$ до $P(X_3 = Unacceptable\,|X_4 = Unreliable) = 0{,}8254$.

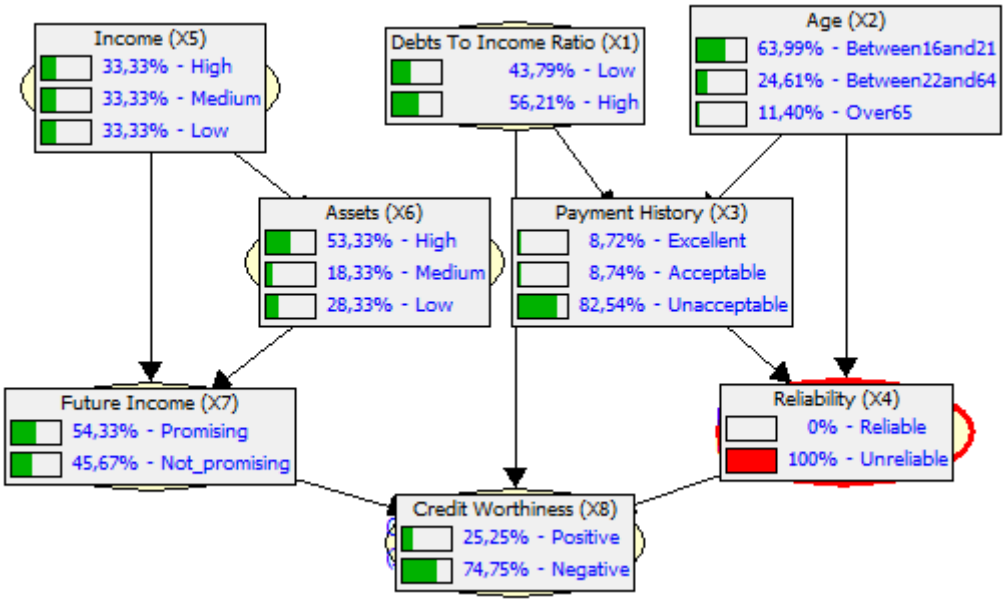


Рис. 4.2.5. До визначення $P(X_3 = Unacceptable\,|X_4 = Unreliable)$

Розглянемо *змішані моделі міркувань* в мережі Credit.

Апостеріорний розподіл вершини $Debts\ To\ Income\ Ratio\ (X_1)$ за умов заданих станів вершин $Age\ (X_2)$ та $Payment\ History\ (X_3)$ дорівнює:

$$P(X_1|X_2, X_3) = \frac{P(X_1, X_2, X_3)}{P(X_2, X_3)}$$

де спільний розподіл вершин X_1, X_2, X_3 обчислюється за формулою

$$P(X_1, X_2, X_3) = \sum_{X_4, X_5, X_6, X_7, X_8} P(X_1, \dots, X_8),$$

а спільний розподіл вершин X_2, X_3 обчислюється за формулою

$$P(X_2, X_3) = \sum_{X_1, X_4, X_5, X_6, X_7, X_8} P(X_1, \dots, X_8).$$

1. Імовірність того, що клієнт має високе відношення заборгованості до доходу *за умови*, що його історія платежів незадовільна та вік клієнта між 16 та 21 збільшилась з $P(X_1 = High) = 0{,}5$ (див. рис. 3.4.1) до 0,5833.

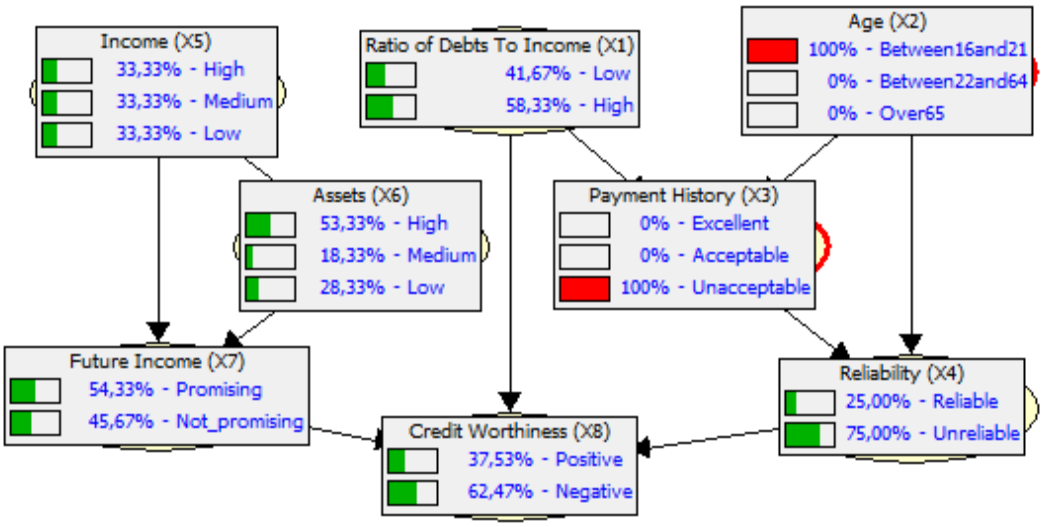


Рис. 4.2.6. До визначення

$$P(X_1 = High | X_2 = Between16and21, X_3 = Unacceptable\ )$$

2. Імовірність того, що клієнт має високе відношення заборгованості до доходу *за умови*, що його історія платежів відмінна та вік клієнта між 16 та 21 зменшилась з 0,5 (див. рис. 3.4.1) до 0,3333.

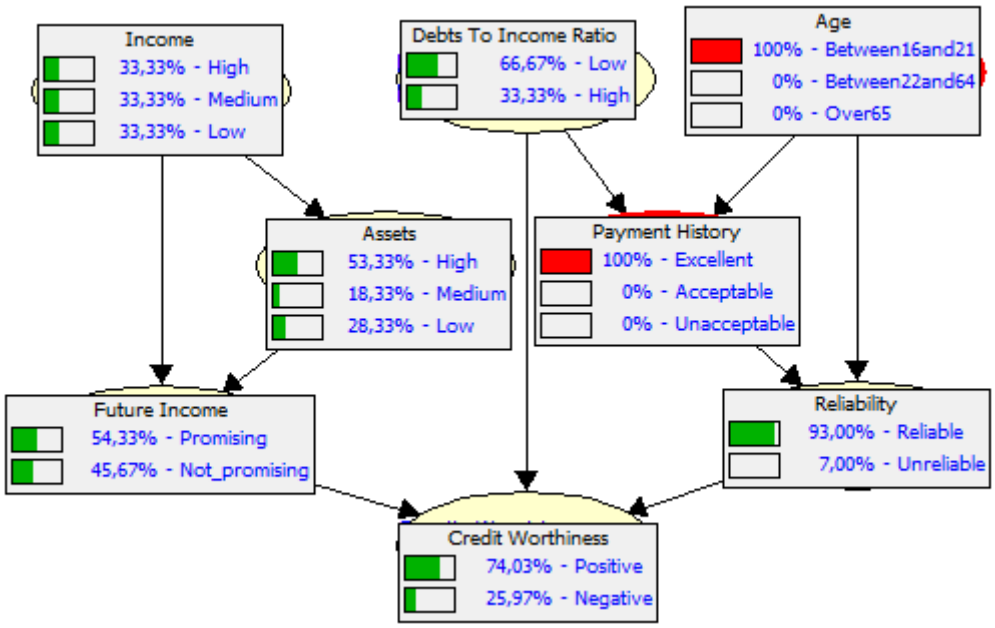


Рис. 4.2.7. До визначення

$$P(X_1 = High | X_2 = Between16and21, X_3 = Excellent)$$

5. d-відокремлення

5.1. Означення d-відокремлення

Розглянемо вплив стану вершини X на стани вершини Y за умови: стан вершини Z заданий, стан вершини Z не заданий.

1 випадок. Непрямий причинний зв'язок (causal trail).

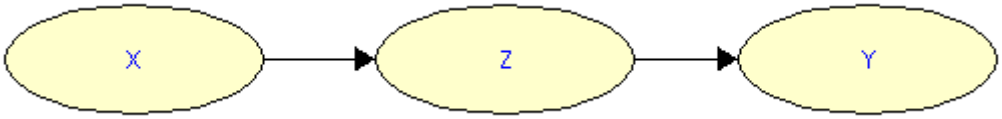


Рис. 5.1.1. $X \to Z \to Y$

Вершини X та Y *умовно незалежні* при заданому стані вершини Z. Дійсно,

$$P(Y|X,Z) = \frac{P(X,Y,Z)}{P(X,Z)} = \frac{P(X)P(Z|X)P(Y|Z)}{P(X)P(Z|X)} = P(Y|Z).$$

Вплив стану вершини X на стани вершини Y блокується при заданому стані вершини Z.

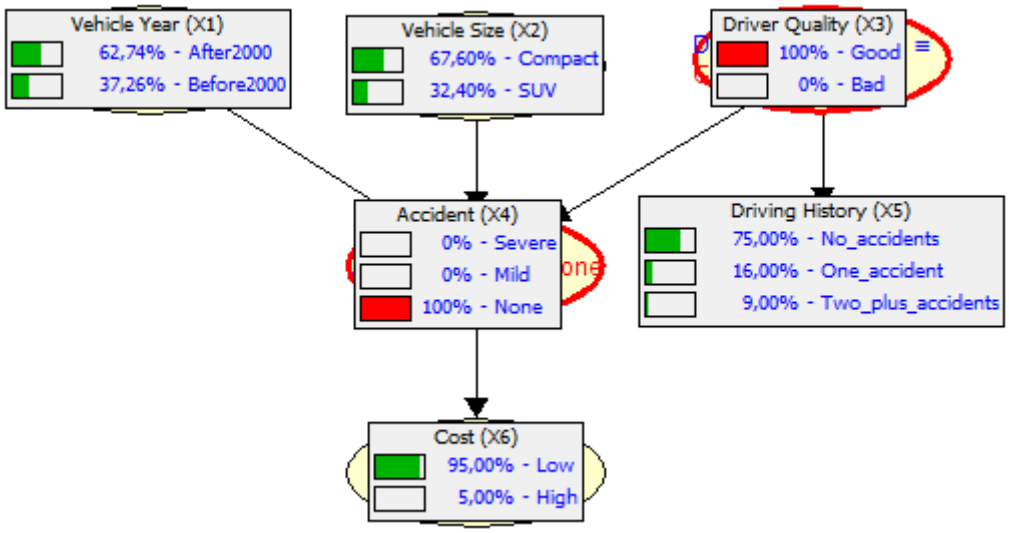


Рис. 5.1.2а. $Driver\ Quality \to Accident \to Cost$

Розглянемо шлях $Driver\ Quality \to Accident \to Cost$, при цьому ступінь тяжкості аварії $Accident$ відома. Якість водіння $Driver\ Quality$ не впливає на витрати страхової компанії $Cost$, якщо відома ступінь тяжкості аварії $Accident$:

$$P(X_6 = Low|X_4 = None, X_3 = Good) = P(X_6 = Low|X_4 = None) = 0{,}95.$$

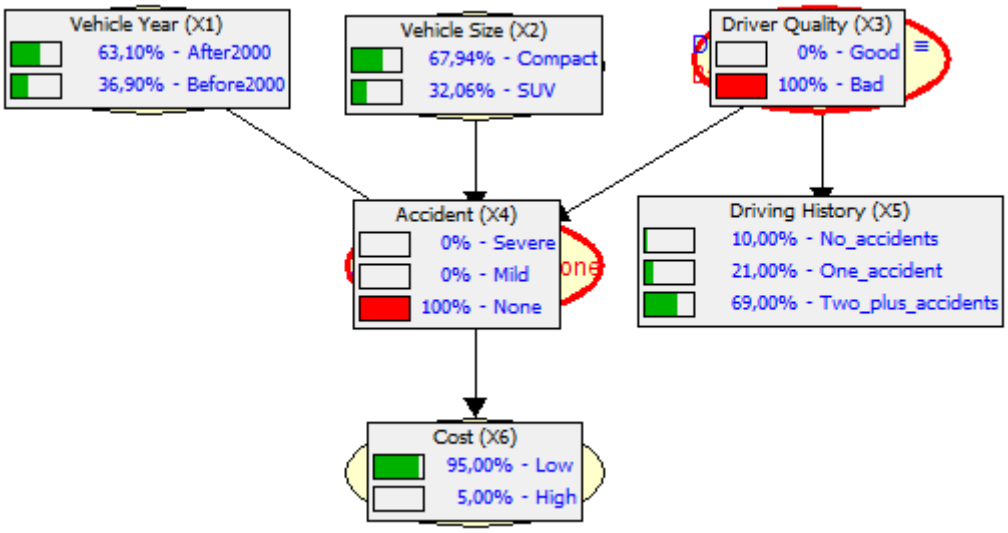


Рис. 5.1.2б. $Driver\ Quality \rightarrow Accident \rightarrow Cost$

$P(X_6 = Low|X_4 = None, X_3 = Bad) = P(X_6 = Low|X_4 = None) = 0{,}95.$
Вплив стану вершини $Driver\ Quality$ на стани вершину $Cost$ блокується при заданому стані вершини $Accident$.

Вершини X та Y *залежні*, якщо стан вершини Z не заданий. Вплив стану вершини X на стани вершину Y не блокується при незаданому стані вершини Z.

2 випадок. Непрямий наслідковий зв'язок (evidential trail).

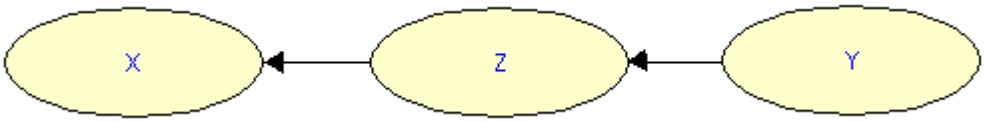


Рис. 5.1.3. $Y \rightarrow Z \rightarrow X$

Міркування аналогічні міркуванням, викладеним у випадку 1.

3 випадок. Common cause.

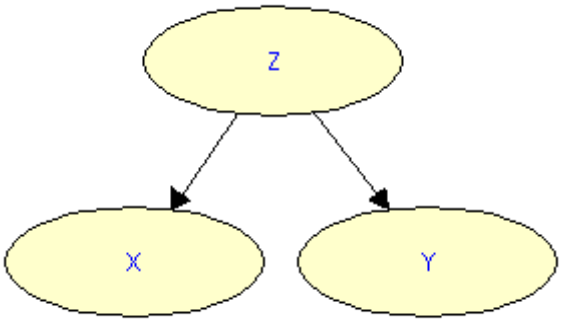


Рис. 5.1.4. $X \leftarrow Z \rightarrow Y$

Вершини X та Y *умовно незалежні* при заданому стані вершини Z. Дійсно,

$$P(Y|X,Z) = \frac{P(X,Y,Z)}{P(X,Z)} = \frac{P(X)P(Z|X)P(Y|Z)}{P(X)P(Z|X)} = P(Y|Z).$$

Вплив стану вершини X на стани вершини Y блокується при заданому стані вершини Z.

Розглянемо шлях

$$Accident \leftarrow Driver\ Quality \rightarrow Driving\ History,$$

при цьому якість водіння $Driver\ Quality$ відома. Ступінь тяжкості аварії $Accident$ не впливає на кількість аварій $Driving\ History$, в які потрапляв водій, якщо відома його якість водіння $Driver\ Quality$:

$$P(X_5 = No|X_4 = None, X_3 = Good) = P(X_5 = No|X_3 = Good) = 0{,}75.$$

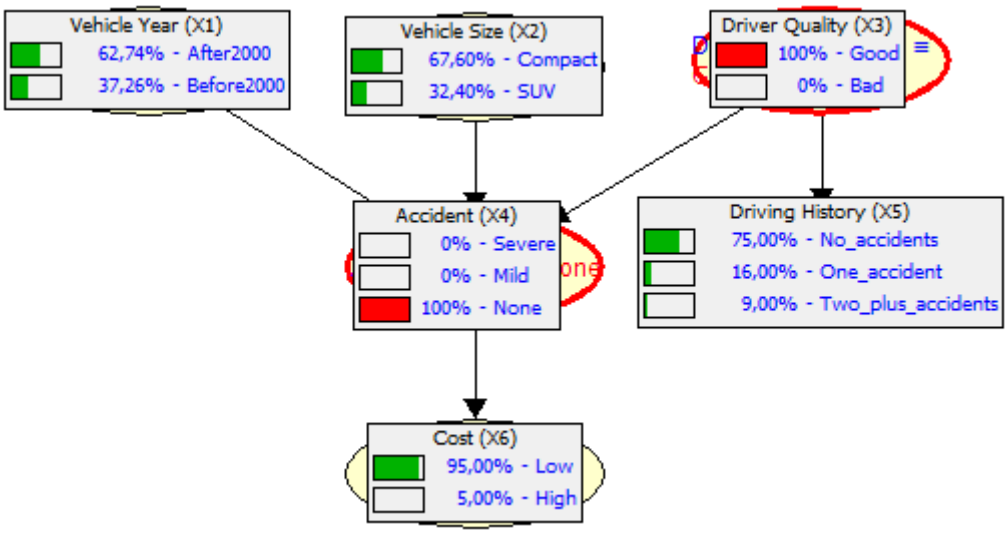


Рис. 5.1.5а. $Accident \leftarrow Driver\ Quality \rightarrow Driving\ History$

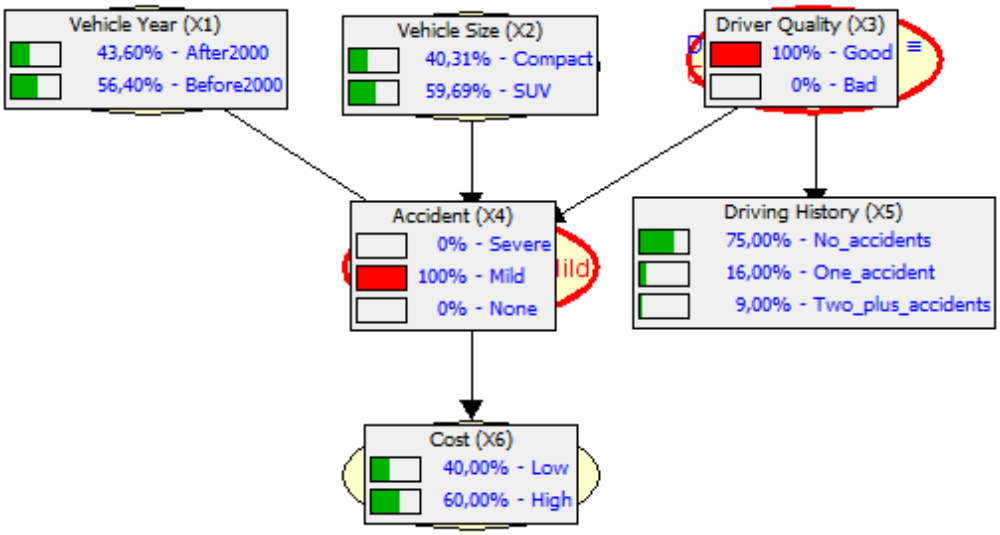


Рис. 5.1.5б. $Accident \leftarrow Driver\ Quality \rightarrow Driving\ History$

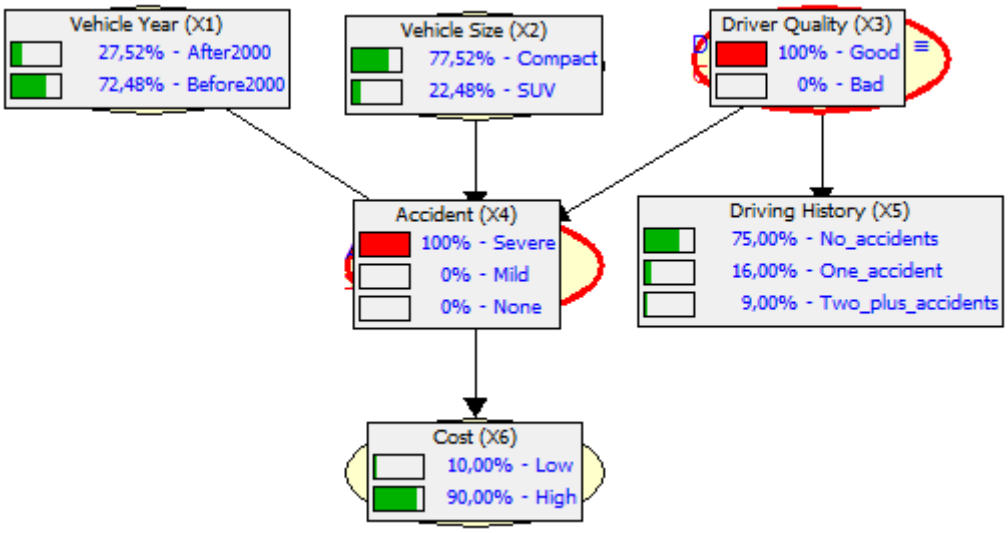


Рис. 5.1.5в. $Accident \leftarrow Driver\ Quality \rightarrow Driving\ History$

$$P(X_5 = No|X_4 = Mild, X_3 = Good) = P(X_5 = No|X_3 = Good) = 0{,}75.$$

$$P(X_5 = No|X_4 = Severe, X_3 = Good) = P(X_5 = No|X_3 = Good) = 0{,}75.$$

Вплив стану вершини $Accident$ на стани вершини $Driving\ History$ блокується при заданому стані вершини $Driver\ Quality$.

Вершини X та Y *залежні*, якщо стан вершини Z не заданий. Вплив стану вершини X на стани вершини Y не блокується при незаданому стані вершини Z.

4 випадок. Common effect (v-structure).

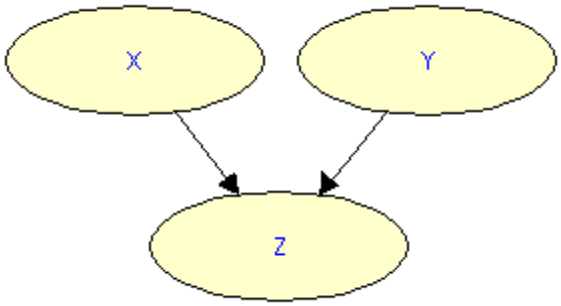


Рис. 5.1.6. $X \rightarrow Z \leftarrow Y$

Вершини X та Y *залежні*, якщо стан вершини Z заданий. Вплив стану вершини X на стани вершину Y не блокується при заданому стані вершини Z. Вершини X та Y *незалежні*, якщо стан вершини Z не заданий:

$$P(Y|X) = P(Y).$$

Вплив стану вершини X на стани вершину Y блокується при незаданому стані вершини Z.

Розглянемо шлях $Vehicle\ Size \rightarrow Accident \leftarrow Driver\ Quality$, при цьому ступінь тяжкості аварії $Accident$ відома.

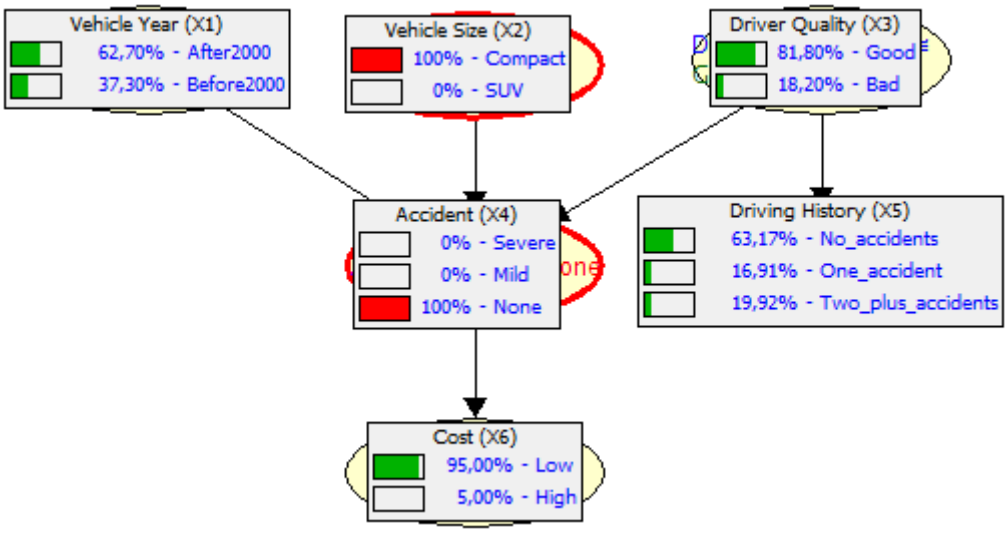


Рис. 5.1.7а. $Vehicle\ Size \rightarrow Accident \leftarrow Driver\ Quality$

$P(X_3 = Good|X_4 = None, X_2 = Compact) = 0{,}8108.$

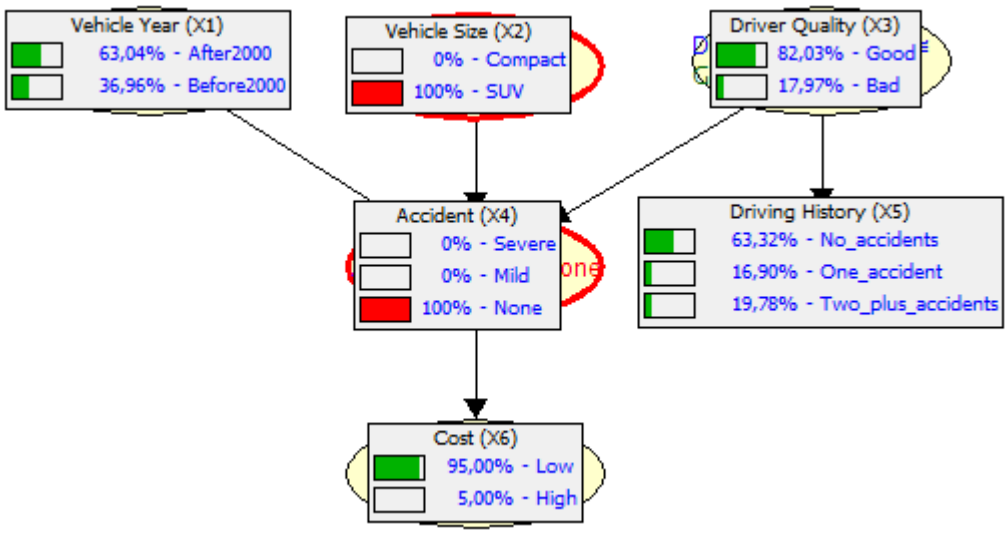


Рис. 5.1.7б. $Vehicle\ Size \rightarrow Accident \leftarrow Driver\ Quality$

$P(X_3 = Good|X_4 = None, X_2 = SUV) = 0{,}8203.$

Вплив стану вершини $Vehicle\ Size$ на стани вершини $Driver\ Quality$ не блокується при заданому стані вершини $Accident$.

Шлях $X \rightleftharpoons Z \rightleftharpoons Y$ вважатимемо активним, якщо існує вплив стану вершини X на стани вершини Y. Зокрема,

Causal trail $X \to Z \to Y$: шлях активний тоді й тільки тоді, коли стан вершини Z не заданий.

Evidential trail $X \leftarrow Z \leftarrow Y$: шлях активний тоді й тільки тоді, коли стан вершини Z не заданий.

Common cause $X \leftarrow Z \to Y$: шлях активний тоді й тільки тоді, коли стан вершини Z не заданий.

Common effect $X \to Z \leftarrow Y$: шлях активний тоді й тільки тоді, коли стан вершини Z або стан одного з нащадків вершини Z заданий.

Означення. Нехай G – байєсівська мережа, $\boldsymbol{Z}$ – підмножина вершин, стани яких задані. Шлях $X_1 \rightleftharpoons \cdots \rightleftharpoons X_n$ називатимемо **активним** при заданих станах вершин $\boldsymbol{Z}$, якщо для будь-якої v-структури $X_{i-1} \to X_i \leftarrow X_{i+1}$ вершина X_i або один з її нащадків належать множині $\boldsymbol{Z}$ й інших вершин в множині $\boldsymbol{Z}$ немає.

Означення d-відокремлення (directed separation). Нехай $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ – підмножини вершин ациклічного орієнтованого графа ($\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ не перетинаються). Вершини $\boldsymbol{X}$ та $\boldsymbol{Y}$ **d -відокремлені** вершинами $\boldsymbol{Z}$ тоді й тільки тоді, коли вершини $\boldsymbol{Z}$ блокують всі активні шляхи з будь-якої вершини, що належить $\boldsymbol{X}$ в будь-яку вершину, що належить $\boldsymbol{Y}$.

Наприклад, в байєсівській мережі шлях $X_2 \to X_4 \leftarrow X_3 \to X_5$ неактивний для $\boldsymbol{Z} = \emptyset$, тому що v-структура $X_2 \to X_4 \leftarrow X_3$ не активована. Шлях $X_2 \to X_4 \leftarrow X_3 \to X_5$ активний для $\boldsymbol{Z} = \{X_4\}$, тому що v-структура $X_2 \to X_4 \leftarrow X_3$ активована. З іншої сторони, шлях $X_2 \to X_4 \leftarrow X_3 \to X_5$ неактивний для $\boldsymbol{Z} = \{X_4, X_3\}$, оскільки X_3 блокує шлях $X_4 \leftarrow X_3 \to X_5$.

5.2. Алгоритм d-відокремлення

Розглянемо алгоритм знаходження досяжних вершин із вершини X при заданих вершинах $\boldsymbol{Z}$ через активні шляхи.

Algorithm 3.1 Algorithm for finding nodes reachable from X given $\boldsymbol{Z}$ via active trails [1, p.75]

Procedure Reachable (
G, // *граф байєсівської мережі*
X, // *вхідна вершина*
$\boldsymbol{Z}$ // *вершини* $\boldsymbol{Z}$*, що спостерігаються*
Pa_Y // *батьківські вершини для вершини* Y
Ch_Y // *дочірні вершини для вершини* Y
)

Фаза I: Вибір (виділення) в множину $\boldsymbol{A}$ всіх вершин, які знаходяться в множині $\boldsymbol{Z}$ або мають нащадків в множині $\boldsymbol{Z}$

$\boldsymbol{L} = \boldsymbol{Z}$ // *множина вершин, які ми відвідаємо*
$\boldsymbol{A} = \emptyset$
до тих пір, поки $\boldsymbol{L} \neq \emptyset$
обираємо вершину Y з множини $\boldsymbol{L}$
прибираємо вершину Y з множини $\boldsymbol{L}$
якщо $Y \notin \boldsymbol{A}$, то
$\boldsymbol{L} = \boldsymbol{L} \cup Pa_Y$ // *додаємо батьків вершини* Y *в множину* $\boldsymbol{L}$
$\boldsymbol{A} = \boldsymbol{A} \cup \{Y\}$ // *додаємо вершину* Y *в множину* $\boldsymbol{A}$

Фаза II: переміщення активними шляхами, починаючи з вершини X
$\boldsymbol{L} = \{(X, \uparrow)\}$ // *множина (вершина, напрямок), які ми відвідаємо*
$\boldsymbol{V} = \emptyset$ // *множина (вершина, напрямок), які ми відвідали*
$\boldsymbol{R} = \emptyset$ // *множина вершин, досяжних активними шляхами*
до тих пір, поки $\boldsymbol{L} \neq \emptyset$
обираємо (вершину, напрямок) (Y, d) з множини $\boldsymbol{L}$
прибираємо (вершину, напрямок) (Y, d) з множини $\boldsymbol{L}$
якщо $(Y, d) \notin \boldsymbol{V}$, то // *якщо* (Y, d) *ми ще не відвідали, то*
$\boldsymbol{R} = \boldsymbol{R} \cup \{Y\}$ // *додаємо вершину* Y *в множину* $\boldsymbol{R}$
$\boldsymbol{V} = \boldsymbol{V} \cup \{(Y, d)\}$ // *додаємо* (Y, d) *в множину* $\boldsymbol{V}$
якщо $d = \uparrow$ и $Y \notin \boldsymbol{Z}$, то // *висхідний шлях активний*
для кожної вершини $Z \in Pa_Y$ // Pa_Y *– батьки вершини* Y
$\boldsymbol{L} = \boldsymbol{L} \cup \{(Z, \uparrow)\}$ // *додаємо батьків вершини* Y *в* $\boldsymbol{L}$
для кожної вершини $Z \in Ch_Y$ // Ch_Y *– діти вершини* Y
$\boldsymbol{L} = \boldsymbol{L} \cup \{(Z, \downarrow)\}$ // *додаємо дітей вершини* Y *в* $\boldsymbol{L}$
інакше якщо $d = \downarrow$, то
якщо $Y \notin \boldsymbol{Z}$, то // *низхідний шлях активний*
для кожної вершини $Z \in Ch_Y$ // Ch_Y *діти вершини* Y
$\boldsymbol{L} = \boldsymbol{L} \cup \{(Z, \downarrow)\}$ // *додаємо дітей вершини* Y *в* $\boldsymbol{L}$
якщо $Y \in \boldsymbol{A}$, то // *v-структура активна*
для кожної вершини $Z \in Pa_Y$ // Pa_Y батьки Y
$\boldsymbol{L} = \boldsymbol{L} \cup \{(Z, \uparrow)\}$ // *додаємо батьків вершини* Y *в* $\boldsymbol{L}$

повернути множину вершин $\boldsymbol{R}$, досяжних активними шляхами

Додаток 1. Моделювання байєсівської мережі Monty Hall

```
from pgmpy.models import BayesianModel
from pgmpy.factors.discrete import TabularCPD

# Defining the network structure
model = BayesianModel([('Contestant', 'Host'), ('Prize', 'Host')])

# Defining the CPDs
cpd_c = TabularCPD(variable='Contestant', variable_card=3,
                   values=[[0.33, 0.33, 0.33]])
cpd_p = TabularCPD(variable='Prize', variable_card=3,
                   values=[[0.33, 0.33, 0.33]])
cpd_h = TabularCPD(variable='Host', variable_card=3,
                    values=[[0, 0, 0, 0, 0.5, 1, 0, 1, 0.5],
                            [0.5, 0, 1, 0, 0, 0, 1, 0, 0.5],
                            [0.5, 1, 0, 1, 0.5, 0, 0, 0, 0]],
                    evidence=['Contestant','Prize'],
                    evidence_card=[3, 3])

# Associating the CPDs with the network structure
model.add_cpds(cpd_c, cpd_p, cpd_h)

# check_model check for the model structure and the associated CPD and
returns True if everything is correct otherwise throws an exception
model.check_model()
print(model.get_cpds('Contestant'))
print(model.get_cpds('Prize'))
print(model.get_cpds('Host'))

# Infering the posterior probability
from pgmpy.inference import VariableElimination

infer = VariableElimination(model)
posterior_p = infer.query(variables=['Prize'],
                          evidence={'Contestant': 0, 'Host': 2})
print(posterior_p['Prize'])
+---------+--------------+
| Prize   |   phi(Prize) |
+=========+==============+
| Prize_0 |       0.3333 |
+---------+--------------+
| Prize_1 |       0.6667 |
+---------+--------------+
| Prize_2 |       0.0000 |
+---------+--------------+

posterior_p = infer.query(variables=['Prize'],
                          evidence={'Contestant': 0, 'Host': 1})
print(posterior_p['Prize'])
+---------+--------------+
| Prize   |   phi(Prize) |
+=========+==============+
| Prize_0 |       0.3333 |
+---------+--------------+
| Prize_1 |       0.0000 |
+---------+--------------+
| Prize_2 |       0.6667 |
+---------+--------------+
```

Додаток 2. Моделювання байєсівської мережі Insurance

```
from pgmpy.models import BayesianModel
from pgmpy.factors.discrete import TabularCPD

# Defining the network structure
model = BayesianModel([('Focused', 'Good_student'),('Age','Good_student'),
       ('Age', 'Driver_quality'), ('Extra_training', 'Driver_quality'),
       ('Focused', 'Driver_quality'),('Driver_quality','DrivingHistory'),
       ('Focused', 'Vehicle_size'),('Vehicle_year', 'Accident'),
       ('Vehicle_size','Accident'), ('Driver_quality', 'Accident'),
       ('Antilock_brakes', 'Accident'), ('Airbag','Accident'),
       ('Vehicle_year', 'Airbag'), ('Vehicle_year', 'Antilock_brakes'),
       ('Accident', 'Cost'), ('Vehicle_year', 'Cost')])

# Defining the CPDs
cpd_focused
cpd_age
cpd_goodstud
cpd_vehiclesize
cpd_extratraining
cpd_driverquality
cpd_drivinghistory
cpd_vehicleyear
cpd_airbag
cpd_antilockbrakes
cpd_accident
cpd_cost

# Associating the CPDs with the network structure
model.add_cpds(cpd_focused, cpd_age, cpd_goodstud, cpd_vehiclesize,
cpd_extratraining, cpd_driverquality, cpd_drivinghistory,cpd_vehicleyear,
cpd_airbag, cpd_antilockbrakes, cpd_accident,cpd_cost)

# check_model check for the model structure and the associated CPD and
returns True if everything is correct otherwise throws an exception
model.check_model()
print(model.get_cpds('Focused'))
print(model.get_cpds('Good_student'))
print(model.get_cpds('Age'))
print(model.get_cpds('Driver_quality'))
print(model.get_cpds('Extra_training'))
print(model.get_cpds('DrivingHistory'))
print(model.get_cpds('Vehicle_size'))
print(model.get_cpds('Vehicle_year'))
print(model.get_cpds('Antilock_brakes'))
print(model.get_cpds('Airbag'))
print(model.get_cpds('Accident'))
print(model.get_cpds('Cost'))


# Infering the posterior probability
from pgmpy.inference import VariableElimination
infer=VariableElimination(model)
```

```
# Distribution for Focused
prior_p=infer.query(variables=['Focused'])
print(prior_p['Focused'])
+-----------+----------------+
| Focused   |   phi(Focused) |
+===========+================+
| Focused_0 |         0.7000 |
+-----------+----------------+
| Focused_1 |         0.3000 |
+-----------+----------------+

# Distribution for Good Student
prior_p=infer.query(variables=['Good_student'])
print(prior_p['Good_student'])
+----------------+--------------------+
| Good_student   |   phi(Good_student) |
+================+====================+
| Good_student_0 |             0.1375 |
+----------------+--------------------+
| Good_student_1 |             0.8625 |
+----------------+--------------------+

# Distribution for Age
prior_p=infer.query(variables=['Age'])
print(prior_p['Age'])
+-------+------------+
| Age   |   phi(Age) |
+=======+============+
| Age_0 |     0.2500 |
+-------+------------+
| Age_1 |     0.7500 |
+-------+------------+

# Distribution for Driver Quality
prior_p=infer.query(variables=['Driver_quality'])
print(prior_p['Driver_quality'])
+------------------+----------------------+
| Driver_quality   |   phi(Driver_quality) |
+==================+======================+
| Driver_quality_0 |               0.4647 |
+------------------+----------------------+
| Driver_quality_1 |               0.5353 |
+------------------+----------------------+

# Distribution for Extra Training
prior_p=infer.query(variables=['Extra_training'])
print(prior_p['Extra_training'])
+-----------------+---------------------+
| Extra_training  |   phi(Extra_training) |
+=================+=====================+
| Extra_training_0 |               0.1000 |
+-----------------+---------------------+
| Extra_training_1 |               0.9000 |
+-----------------+---------------------+
```

```
# Distribution for Driving History
prior_p=infer.query(variables=['DrivingHistory'])
print(prior_p['DrivingHistory'])
+-----------------+----------------------+
| DrivingHistory  |  phi(DrivingHistory) |
+=================+======================+
| DrivingHistory_0 |               0.3394 |
+-----------------+----------------------+
| DrivingHistory_1 |               0.4303 |
+-----------------+----------------------+
| DrivingHistory_2 |               0.2303 |
+-----------------+----------------------+

# Distribution for Vehicle Size
prior_p=infer.query(variables=['Vehicle_size'])
print(prior_p['Vehicle_size'])
+---------------+--------------------+
| Vehicle_size  |  phi(Vehicle_size) |
+===============+====================+
| Vehicle_size_0 |             0.4100 |
+---------------+--------------------+
| Vehicle_size_1 |             0.5900 |
+---------------+--------------------+

# Distribution for Vehicle Year
prior_p=infer.query(variables=['Vehicle_year'])
print(prior_p['Vehicle_year'])
+---------------+--------------------+
| Vehicle_year  |  phi(Vehicle_year) |
+===============+====================+
| Vehicle_year_0 |             0.6000 |
+---------------+--------------------+
| Vehicle_year_1 |             0.4000 |
+---------------+--------------------+

# Distribution for Antilock Brakes
prior_p=infer.query(variables=['Antilock_brakes'])
print(prior_p['Antilock_brakes'])
+------------------+-----------------------+
| Antilock_brakes  |  phi(Antilock_brakes) |
+==================+=======================+
| Antilock_brakes_0 |                0.4600 |
+------------------+-----------------------+
| Antilock_brakes_1 |                0.5400 |
+------------------+-----------------------+

# Distribution for Airbag
prior_p=infer.query(variables=['Airbag'])
print(prior_p['Airbag'])
+----------+--------------+
| Airbag   |  phi(Airbag) |
+==========+==============+
| Airbag_0 |       0.6200 |
+----------+--------------+
| Airbag_1 |       0.3800 |
+----------+--------------+
```

```
# Distribution for Accident
prior_p=infer.query(variables=['Accident'])
print(prior_p['Accident'])
+------------+-----------------+
| Accident   |   phi(Accident) |
+============+=================+
| Accident_0 |          0.0651 |
+------------+-----------------+
| Accident_1 |          0.1629 |
+------------+-----------------+
| Accident_2 |          0.7720 |
+------------+-----------------+

# Distribution for Cost
prior_p=infer.query(variables=['Cost'])
print(prior_p['Cost'])
+--------+-------------+
| Cost   |   phi(Cost) |
+========+=============+
| Cost_0 |      0.7758 |
+--------+-------------+
| Cost_1 |      0.2242 |
+--------+-------------+

import time
# Posterior distribution for Cost (Fig. 3.3.2)
start_time = time.time()
posterior_p=infer.query(variables=['Cost'],
                        evidence={'Antilock_brakes': 0})
print(posterior_p['Cost'])
print("--- %s seconds ---" % round((time.time() - start_time),4))
+--------+-------------+
| Cost   |   phi(Cost) |
+========+=============+
| Cost_0 |      0.8302 |
+--------+-------------+
| Cost_1 |      0.1698 |
+--------+-------------+
--- 0.07813 seconds ---

# Posterior distribution for Cost (Fig. 3.3.3)
start_time = time.time()
posterior_p=infer.query(variables=['Cost'],
                        evidence={'Antilock_brakes': 1})
print(posterior_p['Cost'])
print("--- %s seconds ---" % round((time.time() - start_time),4))
+--------+-------------+
| Cost   |   phi(Cost) |
+========+=============+
| Cost_0 |      0.7294 |
+--------+-------------+
| Cost_1 |      0.2706 |
+--------+-------------+
--- 0.0937 seconds ---
```

```
# Posterior distribution for Cost (Fig. 3.3.4)
start_time = time.time()
posterior_p=infer.query(variables=['Cost'],
                        evidence={'Antilock_brakes': 0, 'Airbag': 0})
print(posterior_p['Cost'])
print("--- %s seconds ---" % round((time.time() - start_time),4))
+--------+-------------+
| Cost   |   phi(Cost) |
+========+=============+
| Cost_0 |      0.8423 |
+--------+-------------+
| Cost_1 |      0.1577 |
+--------+-------------+
--- 0.0780 seconds ---

# Posterior distribution for Cost (Fig. 3.3.5)
start_time = time.time()
posterior_p=infer.query(variables=['Cost'],
                        evidence={'Antilock_brakes': 1, 'Airbag': 1})
print(posterior_p['Cost'])
print("--- %s seconds ---" % round((time.time() - start_time),4))
+--------+-------------+
| Cost   |   phi(Cost) |
+========+=============+
| Cost_0 |      0.6847 |
+--------+-------------+
| Cost_1 |      0.3153 |
+--------+-------------+
--- 0.0781 seconds ---

# Posterior distribution for Cost (Fig. 3.3.6)
start_time = time.time()
posterior_p=infer.query(variables=['Cost'],evidence={'Antilock_brakes': 0,
                                                     'Airbag': 0, 'Driver_quality': 0})
print(posterior_p['Cost'])
print("--- %s seconds ---" % round((time.time() - start_time),4))
+--------+-------------+
| Cost   |   phi(Cost) |
+========+=============+
| Cost_0 |      0.8567 |
+--------+-------------+
| Cost_1 |      0.1433 |
+--------+-------------+
--- 0.0625 seconds ---

# Posterior distribution for Cost (Fig. 3.3.7)
start_time = time.time()
posterior_p=infer.query(variables=['Cost'], evidence={'Antilock_brakes':1,
                                                      'Airbag': 1, 'Driver_quality': 1})
print(posterior_p['Cost'])
print("--- %s seconds ---" % round((time.time() - start_time),4))
+--------+-------------+
| Cost   |   phi(Cost) |
+========+=============+
| Cost_0 |      0.6475 |
+--------+-------------+
| Cost_1 |      0.3525 |
+--------+-------------+
--- 0.0937 seconds ---
```

Додаток 3. Моделювання байєсівської мережі Credit

```
from pgmpy.models import BayesianModel
from pgmpy.factors.discrete import TabularCPD

# Defining the network structure
model = BayesianModel([('Income', 'Assets'), ('Assets', 'FutureIncome'),
        ('FutureIncome', 'CreditWorthiness'), ('Age', 'PaymentHistory'),
        ('DebtIncomeRatio', 'CreditWorthiness'),('Age','Reliability'),
        ('DebtIncomeRatio', 'PaymentHistory'), ('Income', 'FutureIncome'),
   ('Reliability', 'CreditWorthiness'),('PaymentHistory', 'Reliability')])

# Defining the CPDs
cpd_DebtIncomeRatio
cpd_Age
cpd_Income
cpd_FutureIncome
cpd_PaymentHistory
cpd_Reliability
cpd_CreditWorthiness

# Associating the CPDs with the network structure
model.add_cpds(cpd_DebtIncomeRatio, cpd_Age, cpd_Income, cpd_Assets,
cpd_FutureIncome, cpd_PaymentHistory, cpd_Reliability,
cpd_CreditWorthiness)

# check_model check for the model structure and the associated CPD and
returns True if everything is correct otherwise throws an exception
model.check_model()
print(model.get_cpds('DebtIncomeRatio'))
print(model.get_cpds('Age'))
print(model.get_cpds('PaymentHistory'))
print(model.get_cpds('Reliability'))
print(model.get_cpds('Income'))
print(model.get_cpds('Assets'))
print(model.get_cpds('FutureIncome'))
print(model.get_cpds('CreditWorthiness'))

# Infering the posterior probability
from pgmpy.inference import VariableElimination
infer = VariableElimination(model)

# Distribution for DebtIncomeRatio
prior_p=infer.query(variables=['DebtIncomeRatio'])
print(prior_p['DebtIncomeRatio'])
+-------------------+-----------------------+
| DebtIncomeRatio   |   phi(DebtIncomeRatio) |
+===================+=======================+
| DebtIncomeRatio_0 |                0.5000 |
+-------------------+-----------------------+
| DebtIncomeRatio_1 |                0.5000 |
+-------------------+-----------------------+

# Distribution for Age
prior_p=infer.query(variables=['Age'])
print(prior_p['Age'])
```

```
+-------+------------+
| Age   |   phi(Age) |
+=======+============+
| Age_0 |     0.3333 |
+-------+------------+
| Age_1 |     0.3333 |
+-------+------------+
| Age_2 |     0.3333 |
+-------+------------+

# Distribution for PaymentHistory
prior_p=infer.query(variables=['PaymentHistory'])
print(prior_p['PaymentHistory'])
+------------------+-----------------------+
| PaymentHistory   |   phi(PaymentHistory) |
+==================+=======================+
| PaymentHistory_0 |                0.3833 |
+------------------+-----------------------+
| PaymentHistory_1 |                0.3000 |
+------------------+-----------------------+
| PaymentHistory_2 |                0.3167 |
+------------------+-----------------------+

# Distribution for Reliability
prior_p=infer.query(variables=['Reliability'])
print(prior_p['Reliability'])
+---------------+--------------------+
| Reliability   |   phi(Reliability) |
+===============+====================+
| Reliability_0 |             0.7514 |
+---------------+--------------------+
| Reliability_1 |             0.2486 |
+---------------+--------------------+

# Distribution for Income
prior_p=infer.query(variables=['Income'])
print(prior_p['Income'])
+----------+---------------+
| Income   |   phi(Income) |
+==========+===============+
| Income_0 |        0.3333 |
+----------+---------------+
| Income_1 |        0.3333 |
+----------+---------------+
| Income_2 |        0.3333 |
+----------+---------------+

# Distribution for Assets
prior_p=infer.query(variables=['Assets'])
print(prior_p['Assets'])
+----------+---------------+
| Assets   |   phi(Assets) |
+==========+===============+
| Assets_0 |        0.5333 |
+----------+---------------+
| Assets_1 |        0.1833 |
```

```
+----------+---------------+
| Assets_2 |        0.2833 |
+----------+---------------+

# Distribution for Future Income
prior_p=infer.query(variables=['FutureIncome'])
print(prior_p['FutureIncome'])
+----------------+---------------------+
| FutureIncome   |   phi(FutureIncome) |
+================+=====================+
| FutureIncome_0 |              0.5433 |
+----------------+---------------------+
| FutureIncome_1 |              0.4567 |
+----------------+---------------------+

# Distribution for CreditWorthiness
prior_p=infer.query(variables=['CreditWorthiness'])
print(prior_p['CreditWorthiness'])
+--------------------+-------------------------+
| CreditWorthiness   |   phi(CreditWorthiness) |
+====================+=========================+
| CreditWorthiness_0 |                  0.6344 |
+--------------------+-------------------------+
| CreditWorthiness_1 |                  0.3656 |
+--------------------+-------------------------+

# Reasoning Patterns
# Causal reasoning or prediction
# Posterior distribution for PaymentHistory (Fig. 4.2.1)
posterior_p = infer.query(variables=['PaymentHistory'],
                          evidence={'DebtIncomeRatio' : 1, 'Age' : 0})
print(posterior_p['PaymentHistory'])
+------------------+-----------------------+
| PaymentHistory   |   phi(PaymentHistory) |
+==================+=======================+
| PaymentHistory_0 |                0.1000 |
+------------------+-----------------------+
| PaymentHistory_1 |                0.2000 |
+------------------+-----------------------+
| PaymentHistory_2 |                0.7000 |
+------------------+-----------------------+

# Posterior distribution for PaymentHistory (Fig. 4.2.2)
posterior_p = infer.query(variables=['PaymentHistory'],
                          evidence={'DebtIncomeRatio' : 0, 'Age' : 0})
print(posterior_p['PaymentHistory'])
+------------------+-----------------------+
| PaymentHistory   |   phi(PaymentHistory) |
+==================+=======================+
| PaymentHistory_0 |                0.2000 |
+------------------+-----------------------+
| PaymentHistory_1 |                0.3000 |
+------------------+-----------------------+
| PaymentHistory_2 |                0.5000 |
+------------------+-----------------------+
```

```
# Posterior distribution for PaymentHistory (Fig. 4.2.3)
posterior_p = infer.query(variables=['PaymentHistory'],
                          evidence={'DebtIncomeRatio' : 0, 'Age' : 2})
print(posterior_p['PaymentHistory'])
+------------------+-----------------------+
| PaymentHistory   |   phi(PaymentHistory) |
+==================+=======================+
| PaymentHistory_0 |                0.6000 |
+------------------+-----------------------+
| PaymentHistory_1 |                0.3000 |
+------------------+-----------------------+
| PaymentHistory_2 |                0.1000 |
+------------------+-----------------------+

# Evidential reasoning or explanation
# Posterior distribution for PaymentHistory (Fig. 4.2.4)
posterior_p = infer.query(variables=['PaymentHistory'],
                          evidence={'Reliability' : 0})
print(posterior_p['PaymentHistory'])
+------------------+-----------------------+
| PaymentHistory   |   phi(PaymentHistory) |
+==================+=======================+
| PaymentHistory_0 |                0.4813 |
+------------------+-----------------------+
| PaymentHistory_1 |                0.3703 |
+------------------+-----------------------+
| PaymentHistory_2 |                0.1484 |
+------------------+-----------------------+

# Posterior distribution for PaymentHistory (Fig. 4.2.5)
posterior_p = infer.query(variables=['PaymentHistory'],
                          evidence={'Reliability' : 1})
print(posterior_p['PaymentHistory'])
+------------------+-----------------------+
| PaymentHistory   |   phi(PaymentHistory) |
+==================+=======================+
| PaymentHistory_0 |                0.0872 |
+------------------+-----------------------+
| PaymentHistory_1 |                0.0874 |
+------------------+-----------------------+
| PaymentHistory_2 |                0.8254 |
+------------------+-----------------------+

# Intercausal reasoning
# Posterior distribution for DebtIncomeRatio (Fig. 4.2.6)
posterior_p = infer.query(variables=['DebtIncomeRatio'],
                          evidence={'PaymentHistory' : 2, 'Age' : 0})
print(posterior_p['DebtIncomeRatio'])
+-------------------+------------------------+
| DebtIncomeRatio   |   phi(DebtIncomeRatio) |
+===================+========================+
| DebtIncomeRatio_0 |                 0.4167 |
+-------------------+------------------------+
| DebtIncomeRatio_1 |                 0.5833 |
+-------------------+------------------------+
```

Список рекомендованої літератури

1. Koller, Daphne Probabilistic Graphical Models: Principles and Techniques / Daphne Koller and Nir Friedman. – The MIT Press, 2009.
2. Daphne Koller Probabilistic Graphical Models 1: Representation [Електронний ресурс]//Stanford University. URL: https://www.coursera.org/learn/probabilistic-graphical-models (Дата звернення: 24.05.2019)
3. SamIam: Sensitivity Analysis, Modeling, Inference and More – програмний продукт для моделювання та аналізу байєсівських мереж, розроблений Automated Reasoning Group prof. Adnan Darwiche in University of California, Los Angeles (UCLA). URL: http://reasoning.cs.ucla.edu/samiam/
4. Ankur Ankan, Abinash Panda Mastering Probabilistic Graphical Models Using Python. – Packt Publishing Ltd., 2015.
5. Турчин В.Н. Теория вероятностей и математическая статистика. Учебник для студентов высших учебных заведений. – Днепр, Издательство «Лира». – 2018. – 752 с.

Зміст

Вступ
Розділ 1. Основні поняття теорії ймовірностей
- 1.1. Умовна ймовірність
- 1.2. Незалежні події
- 1.3. Дискретна випадкова величина та її розподіл
- 1.4. Незалежні випадкові величини

Розділ 2. Пода́ння байєсівської мережі
- 2.1. Формула множення для байєсвської мережі
- 2.2. Байєсівська мережа Insurance
- 2.3. Байєсівська мережа Credit

Розділ 3. Імовірнісний висновок
- 3.1. Апріорний та апостеріорний розподіли
- 3.2. Алгоритм Variable Elimination
- 3.3. Імовірнісний висновок в байєсівській мережі Insurance
- 3.4. Імовірнісний висновок в байєсівській мережі Credit

Розділ 4. Моделі міркувань
- 4.1. Моделі міркувань в байєсівській мережі Insurance
- 4.2. Моделі міркувань в байєсівській мережі Credit

Розділ 5. d-відокремлення
- 5.1. Означення d-відокремлення
- 5.2. Алгоритм d-відокремлення

Додаток 1. Моделювання байєсівської мережі Monty Hall
Додаток 2. Моделювання байєсівської мережі Insurance
Додаток 3. Моделювання байєсівської мережі Credit