



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

INFERENCE V BAYESOVSKÝCH SÍTÍCH

INFERENCE IN BAYESIAN NETWORKS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. JOSEF ŠIMEČEK

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. FRANTIŠEK V. ZBOŘIL, CSc.

BRNO 2013

Abstrakt

Tato diplomová práce se zabývá demonstrací různých přístupů k inferencím v Bayesovských sítích. V teoretické části jsou rozebrány základy pravděpodobnosti, základy teorie Bayesovských sítí, inferenční metody a oblasti aplikací Bayesovských sítí. Inferenční metody jsou krátce představeny a doplněny jejich algoritmem. Princip každé metody je uveden na příkladu. Praktická část obsahuje popis implementace, experimenty s demonstračními aplikacemi a shrnutí dosažených výsledků.

Abstract

This master's thesis deals with demonstration of various approaches to probabilistic inference in Bayesian networks. Basics of probability theory, introduction to Bayesian networks, methods for Bayesian inference and applications of Bayesian networks are described in theoretical part. Inference techniques are explained and complemented by their algorithm. Techniques are also illustrated on example. Practical part contains implementation description, experiments with demonstration applications and conclusion of the results.

Klíčová slova

Bayesovské sítě, pravděpodobnostní inference, exaktní inference, Kimův a Pearlův algoritmus posílání zpráv, transformace na rozložitelný model, logické vzorkování, věrohodnostní váhování, Gibbsovo vzorkování

Keywords

Bayesian networks, probabilistic inference, exact inference, Kim and Pearl's message passing algorithm, junction tree algorithm, logic sampling, likelihood weighting, Gibbs sampling

Citace

Josef Šimeček: Inference v Bayesovských sítích, diplomová práce, Brno, FIT VUT v Brně, 2013

Inference v Bayesovských sítích

Prohlášení

Prohlašuji, že jsem tento semestrální projekt vypracoval samostatně pod vedením pana Doc. Ing. Františka V. Zbořila, CSc. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Josef Šimeček
22. května 2013

Poděkování

Chtěl bych poděkovat panu Doc. Ing. Františku V. Zbořilovi za odborné vedení, pomoc, rady a čas, který mi věnoval.

© Josef Šimeček, 2013.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	3
2	Základy pravděpodobnosti	5
2.1	Podmíněná pravděpodobnost	6
2.2	Nezávislost	6
2.3	Podmíněná nezávislost	6
2.4	Věty podmíněné pravděpodobnosti	6
2.5	Bayesova věta	7
2.6	Sdružené rozdělení pravděpodobnosti náhodných veličin	7
3	Bayesovské sítě	8
3.1	Úvod do Bayesovských sítí	8
3.2	Uzly a hodnoty	8
3.3	Struktura sítě	8
3.4	Podmíněné pravděpodobnosti	9
3.5	Markovovská vlastnost	9
3.6	Usuzování s bayesovskými sítěmi	9
3.7	Reprezentace sdruženého rozdělení pravděpodobnosti	9
3.8	d-Separace	11
3.9	Základní struktury bayesovských sítí	12
3.9.1	Sériové spojení	12
3.9.2	Divergující spojení	12
3.9.3	Konvergující spojení	12
3.10	Učení Bayesovských sítí	13
4	Příklady Bayesovských sítí	14
4.1	Rozšířený příklad zemětřesení	14
4.2	Metastázující rakovina	15
5	Inference v Bayesovských sítích	17
5.1	Exaktní inference v řetězcích	17
5.1.1	Dvojuzlová síť	17
5.1.2	Tříuzlová síť	18
5.2	Exaktní inference v polystromech	20
5.2.1	Kimův a Pearlův algoritmus posílání zpráv	21
5.3	Inference s nejistým poznatkem	27
5.3.1	Užití virtuálního uzlu	27
5.3.2	Mnohonásobný nejistý poznatek	28

5.4	Exaktní inference v sítích s vícenásobným spojením	28
5.4.1	Shlukovací metody	29
5.4.2	Stromy spojení	30
5.5	Přibližná inference	35
5.5.1	Logické vzorkování	35
5.5.2	Věrohodnostní váhování	36
5.5.3	Markovský řetězec Monte Carlo – MCMC	37
5.5.4	Gibbsovo vzorkování	38
5.5.5	Hodnocení aproximačních inferenčních algoritmů	39
6	Aplikace	40
6.1	Medicínské aplikace	40
6.1.1	Diagnóza	40
6.1.2	Prognóza	41
6.1.3	Léčba	41
6.1.4	Medicínská aplikace – HEPAR II	41
6.2	Mapování potenciálu nerostných zdrojů	41
6.3	Analýza rizikových kriminogenních faktorů	41
6.4	Forenzní vědy	42
6.5	Řízení rizika terorismu	42
6.6	Validace senzorů	42
6.7	Modelování studenta	43
6.8	Správa silnic a mostů	43
6.9	Podpora rozhodování při složitých průmyslových procesech	44
6.10	Rating společností	44
6.11	Detekce narušení sítě	44
7	Návrh a implementace	45
7.1	Návrh	45
7.2	Implementace	45
7.3	Popis tříd	46
8	Demonstrační aplikace	49
8.1	ALARM	49
8.2	Infekce Mycoplasma v chovech prasat	49
8.3	Rok 2000 – analýza rizik	50
9	Experimenty	51
9.1	Year2000 bez poznatků	51
9.2	MycoF bez poznatků	52
9.3	MycoF s poznatky	54
9.4	ALARM bez poznatků	56
9.5	ALARM s poznatky	56
10	Závěr	59
A	Obsah CD	61

Kapitola 1

Úvod

Bayesovské sítě (BN) jsou grafickým modelem, který zachycuje pravděpodobnostní vztahy mezi velkým počtem proměnných a který provádí pravděpodobnostní usuzování (inferenci) mezi těmito proměnnými. Bayesovské sítě tvoří formální rozhraní pro reprezentaci a sdělování rozhodnutí vycházející z usuzování v nejistotě. Nesou jméno po Thomasi Bayesovi, jednomu ze zakladatelů teorie pravděpodobnosti. Vznikly na základě několika matematických výzkumů v 80. letech dvacátého století. Těmito základy byly především práce o pravděpodobnostních sítích, kauzálních sítích a influenčních diagramech.

Bayesovské sítě byly nejprve známy pod pojmem pravděpodobnostní expertní systémy. Název získaly podle knihy Judei Pearla. Pearl byl průkopníkem pravděpodobnostního přístupu k umělé inteligenci, a dokonce bývá zmiňován jako objevitel BN.

Od jejich vzniku byly učení a inference BN tématem velkého množství výzkumů a bylo na nich odvedeno velké množství práce. V posledních 10 letech se výrazně zvýšilo množství aplikací BN na problémy skutečného světa. Mezi tyto problémy patří diagnóza, předpověď, řízení výroby, vyhledávání informací, predikce a plánování. Téměř ve všech vědeckých a technických oborech je viditelný úspěch BN jako nástroje pro modelování složitých vztahů mezi velkým počtem proměnných a pro usuzování v nich. V poslední době jsou na vzestupu především aplikace z oboru informačních a komunikačních technologií, biomedicíny, genomiky a bioinformatiky [11].

BN reprezentují sdružené rozdělení pravděpodobnosti mezi danými proměnnými. Každá proměnná je v grafu zastupována uzlem. Přímé závislosti mezi proměnnými jsou reprezentovány pomocí orientovaných hran mezi příslušnými uzly. Podmíněné pravděpodobnosti (pravděpodobnosti podmíněné všemi možnými kombinacemi rodičovských uzlů) pro každou proměnnou jsou uloženy v tabulce podmíněných pravděpodobností (CPT), která patří k podmíněnému uzlu [7].

Základním úkolem libovolného inferenčního systému je výpočet podmíněné pravděpodobnosti pro množinu dotazovaných uzlů. Existují dva hlavní typy inferenčních algoritmů – exaktní a přibližný, vyvíjené v průběhu posledních 20 let. Různé varianty těchto algoritmů se vyvíjí z důvodu rozdílné vhodnosti pro odlišné struktury sítě [6].

Cílem této diplomové práce je popsání teorie BN, aplikací BN a hlavně inferenčních metod. Inferenční metody jsou krátce představeny a doplněny jejich algoritmem. Princip každé metody je uveden na příkladu. Dále se zde nacházejí experimenty s implementovanými algoritmy provedené na demonstračních aplikacích.

V kapitole 2 jsou uvedeny základní pojmy z teorie pravděpodobnosti, které se využívají u BN. Nachází se zde informace o podmíněné pravděpodobnosti, věty podmíněné pravděpodobnosti a hlavně Bayesova věta.

V kapitole 3 jsou popsány základy BN, typy uzlů a hodnot, struktura sítě, podmíněné pravděpodobnosti a reprezentace sdruženého rozdělení pravděpodobnosti. Dále jsou zde osvětleny pojmy Markovova vlastnost a d-Separace.

Kapitola 4 obsahuje příklady BN, na kterých je v následující kapitole předváděna pravděpodobnostní inference.

V kapitole 5 je popsána pravděpodobnostní inference. Nachází se zde exaktní a přibližné inferenční metody. Metody jsou uvedeny spolu s jejich algoritmem. Princip algoritmu je přímo ilustrován na příkladu.

V kapitole 6 jsou představeny rozmanité obory, ve kterých se používají BN aplikace.

Kapitola 7 obsahuje pouze stručný návrh aplikace, protože inferenční algoritmy byly popsány v kapitole 5. Dále se zde nachází popis implementovaných tříd.

V kapitole 8 se nachází popis a osvětlení demonstračních aplikací BN.

Kapitola 9 obsahuje experimenty s demonstračními aplikacemi. Nachází se zde výsledky získané jednotlivými inferenčními algoritmy, časové srovnání algoritmů a také slovní popis výsledků experimentů.

Poslední kapitola obsahuje shrnutí výsledků a nástin možných rozšíření aplikace.

Kapitola 2

Základy pravděpodobnosti

Pravděpodobnostní kalkulus byl vytvořen v 17. století pány Fermatem a Pascalem za účelem vypořádání se s fyzickou nejistotou. Tuto nejistotu přinesly hazardní hry. Nedlouho poté se zjistilo, že koncept pravděpodobnosti lze použít i při vypořádávání se s nejistotou, kterou přináší nevědomost [6].

Nechť Ω označuje prostor obsahující n různých prvků:

$$\Omega = \{e_1, e_2, \dots, e_n\}.$$

Funkce, která přiřazuje reálné číslo $P(E)$ pro každou událost $E \subseteq \Omega$, je nazývána pravděpodobnostní funkcí, pokud platí následující podmínky:

1. $0 \leq P(\{e_i\}) \leq 1$ pro $1 \leq i \leq n$.
2. $P(\{e_1\}) + P(\{e_2\}) + \dots + P(\{e_n\}) = 1$.
3. Pro každou událost $E = \{e_{i_1}, e_{i_2}, \dots, e_{i_k}\}$, která není elementární událostí,

$$P(E) = P(\{e_{i_1}\}) + P(\{e_{i_2}\}) + \dots + P(\{e_{i_k}\}).$$

Dvojice (Ω, P) se nazývá pravděpodobnostní prostor.

Nechť (Ω, P) je pravděpodobnostní prostor. Pak

1. $P(\Omega) = 1$.
2. $0 \leq P(E) \leq 1$ pro každou $E \subseteq \Omega$.
3. Pro každé E a $F \subseteq \Omega$ takové, že $E \cap F = \emptyset$,

$$P(E \cup F) = P(E) + P(F).$$

Tyto tři podmínky byly označeny jako axiomy teorie pravděpodobnosti [8]. Třetí axiom lze rozšířit pro případ dvou složených událostí, které nejsou disjunktní:

$$\forall E, F \subseteq \Omega, P(E \cup F) = P(E) + P(F) - P(E \cap F)$$

.

2.1 Podmíněná pravděpodobnost

Koncept podmíněné pravděpodobnosti je klíčový pro účelné aplikování pravděpodobnostního kalkulu.

$$P(E|F) = \frac{P(E \cap F)}{P(F)}$$

Podmíněná pravděpodobnost vypovídá o pravděpodobnosti události E při výskytu události F . Pokud F je událost s nulovou pravděpodobností, pak podmíněná pravděpodobnost není definována. Toto se však netýká rozdělení pravděpodobnosti, které je vždy kladné a dle definice nenulové pro každou událost [6].

2.2 Nezávislost

Dalším důležitým pojmem z pravděpodobnosti je nezávislost. Dvě události E a F jsou pravděpodobnostně nezávislé, zanechává-li podmíněnost na jedné druhou nezměněnu:

$$E \perp\!\!\!\perp F \equiv P(E|F) = P(E)$$

Jedná se o symetrický jev: $E \perp\!\!\!\perp F \equiv F \perp\!\!\!\perp E$ [6].

2.3 Podmíněná nezávislost

Podmíněná nezávislost zobecňuje koncept nezávislosti. Události E a F jsou nezávislé při výskytu nějaké další události G .

$$E \perp\!\!\!\perp F|G \equiv P(E|F, G) = P(E|G)$$

V případě, že G bude prázdná množina, tak se toto pravidlo zredukuje na prostou nezávislost. Podmíněná nezávislost platí, pokud událost G vypovídá o všem, co dělá F s E a samozřejmě i více. Jakmile je G známo, pak F není potřeba znát [6].

2.4 Věty podmíněné pravděpodobnosti

Věta 1 (Totální pravděpodobnost [6]). *Množina událostí $\{A_i\}$ je rozdělením univerza U , platí $\bigcup_i A_i = U$ a pro všechny různé i a j je $A_i \cap A_j = \emptyset$, pak*

$$P(U) = \sum_i P(A_i).$$

Podobně lze rozdělit pravděpodobnost nějaké konkrétní události B namísto celého univerza U . Za výše uvedených podmínek a v případě $\forall i A_i \neq \emptyset$:

$$P(B) = \sum_i P(B \cap A_i).$$

Věta 2 (Řetězové pravidlo). *Pro dané tři události A , B , C v řetězu vlivu (A a C jsou nezávislé díky B),*

$$P(C|A) = P(C|B)P(B|A) + P(C|\neg B)P(\neg B|A)$$

za předpokladu, že jsou definovány podmíněné pravděpodobnosti. Toto dovoluje rozdělit pravděpodobnostní vliv C na A přes různé stavy třetí proměnné. V tomto případě je třetí proměnná binární. Věta lze jednoduše zobecnit pro proměnnou o libovolném počtu stavů [6].

2.5 Bayesova věta

Rovnici podmíněné pravděpodobnosti:

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

kde $P(B) > 0$, lze přepsat do jiného tvaru:

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

takto upravená rovnice se nazývá „product rule“.

Výše uvedené pravidlo (product rule) má dva tvary:

$$\begin{aligned} P(A \cap B) &= P(A|B)P(B) \\ P(A \cap B) &= P(B|A)P(A) \end{aligned}$$

Porovnáním pravých stran a vydělením $P(A)$ se získá vzorec:

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

Tento vzorec je znám pod pojmem Bayesova věta (pravidlo, zákon, teorém). Tento jednoduchý vzorec je základem pro systémy s pravděpodobnostním usuzováním. Obecnější podoba vzorce pro vícehodnotové proměnné má tvar:

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

Bayesova věta hraje hlavní roli v pravděpodobnostním usuzování, protože pomocí ní lze přepočítávat pravděpodobnostní vztahy mezi $P(Y|X)$ a $P(X|Y)$ [12].

2.6 Sdružené rozdělení pravděpodobnosti náhodných veličin

Sdružené rozdělení pravděpodobnosti kompletně specifikuje pravděpodobnost přiřazenou ke všem možnostem v doméně.

Pravděpodobnostní model se skládá z množiny náhodných proměnných, které mohou nabývat s určitou pravděpodobností konkrétních hodnot. Nechť jsou proměnné $X_1 \dots X_n$. Atomickou událostí se nazývá přiřazení konkrétních hodnot ke všem proměnným, jinými slovy kompletní specifikace stavu domény.

Sdružené rozdělení pravděpodobnosti $P(X_1, \dots, X_n)$ přiřazuje pravděpodobnost všem možným atomickým událostem [12].

Kapitola 3

Bayesovské sítě

3.1 Úvod do Bayesovských sítí

Bayesovské sítě (BN) jsou grafickým modelem pro usuzování v nejistotě, ve kterém uzly zastupují spojitě či diskrétní proměnné a hrany reprezentují přímé spojení mezi nimi. Spojení mají obvykle kauzální charakter. Silná stránka BN se právě nachází v těchto spojeních mezi proměnnými.

Uzly v BN reprezentují množinu náhodných proměnných, $X = X_1, \dots, X_i, \dots, X_n$. Prvky z množiny orientovaných hran (spojení) spojují dvojice uzlů ($X_i \longrightarrow X_j$). Tyto hrany značí přímé závislosti mezi proměnnými. V případě diskrétních proměnných je síla vztahu mezi dvěma proměnnými určena podmíněným rozdělením pravděpodobnosti, které má každý uzel přiřazen. Pro hrany platí jediná podmínka, a to že se v grafu nesmí vyskytovat žádná orientovaná kružnice. BN je tedy acyklickým orientovaným grafem.

Existuje celá řada kroků, která musí být při tvorbě BN brána v úvahu [6].

3.2 Uzly a hodnoty

Uzly sítě zastupují zájmové proměnné a mohou nabývat libovolných hodnot. Hodnoty musí být vzájemně výlučné a úplné (pokrývají všechny stavy). To znamená, že proměnná v jeden okamžik může nést informaci pouze o jedné z hodnot. Běžné typy diskrétních uzlů zahrnují [6]:

- uzly s booleovskými hodnotami,
- uzly s výčtem hodnot,
- uzly s číselnou hodnotou.

3.3 Struktura sítě

Struktura sítě (topologie) zachycuje kvalitativní vztahy mezi proměnnými. Dva uzly jsou přímo spojeny, pokud jeden ovlivňuje či způsobuje druhý. Hrana, která tyto uzly spojuje, má orientaci ve směru působení [6].

3.4 Podmíněné pravděpodobnosti

Po určení struktury sítě lze kvantifikovat vztahy mezi propojenými uzly. Kvantifikace se provede specifikováním podmíněného rozdělení pravděpodobnosti pro každý uzel. V případě diskrétních proměnných se jedná o tabulky podmíněných pravděpodobností (CPT).

Pro každý uzel se naleznou všechny možné kombinace hodnot rodičovských uzlů. Každá takováto kombinace se nazývá instancí rodičovské množiny. Pro všechny instance musí být specifikována pravděpodobnost, že synovský uzel nabude každé z jeho hodnot [6].

3.5 Markovovská vlastnost

Modelování BN vyžaduje předpoklad Markovovské vlastnosti: neexistují žádné přímé závislosti v modelovaném systému, které by nebyly explicitně zobrazeny hranami. BN splňující Markovovskou vlastnost se nazývají I-mapy (Independence-maps) [6].

3.6 Usuzování s bayesovskými sítěmi

Proces inference (propagování pravděpodobnosti, usuzování, podmiňování) se provádí pomocí toku informací skrz síť. Informační tok není omezen orientací hran. V pravděpodobnostním systému se počítá posteriorní rozdělení pravděpodobnosti pro množinu dotazovaných uzlů na základě poznatků (známých hodnot) pro nějaké uzly (pozorované uzly).

BN poskytují úplnou reprezentaci rozdělení pravděpodobnosti nad svými proměnnými. Díky úplné reprezentaci lze provést usuzování jakýmkoliv směrem pro libovolnou podmnožinu proměnných.

Usuzování v BN není omezeno pouze na diagnostické, ve skutečnosti existují čtyři typy usuzování [6]:

- diagnostické (od následků k příčinám),
- prediktivní (od příčin k následkům),
- interkauzální (vyloučení, mezi příčinami a jedním efektem),
- kombinované odvozování (kombinace dvou nebo více výše zmíněných přístupů).

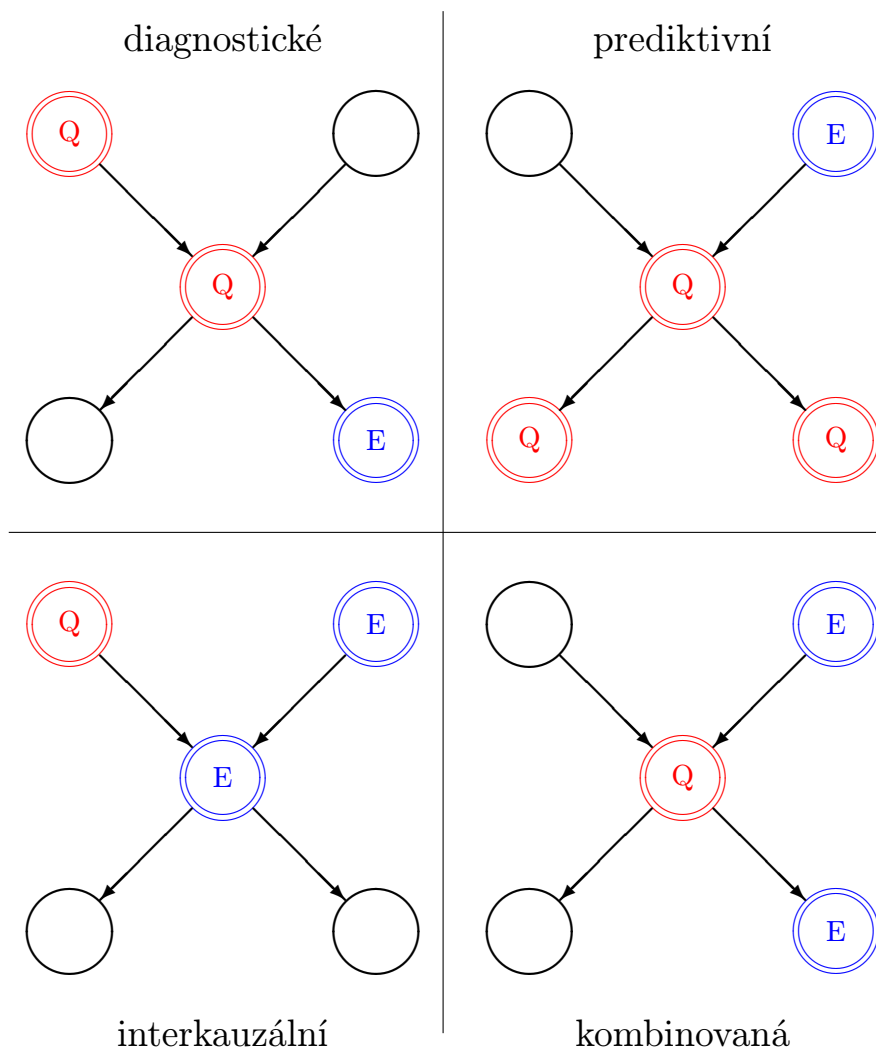
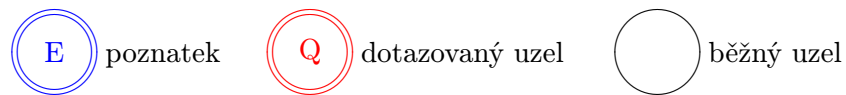
Graficky zobrazené typy usuzování se nachází na obrázku 3.1.

3.7 Reprezentace sdruženého rozdělení pravděpodobnosti

Řídké BN reprezentují rozdělení pravděpodobnosti výpočetně řešitelným způsobem. Řídkou BN se myslí BN, ve které je relativně málo hran, což znamená, že každý uzel má jenom pár rodičovských uzlů.

Nechť existuje BN síť o n uzlech X_1 až X_n . Konkrétní hodnota ve sdruženém rozdělení pravděpodobnosti pro tuto síť je reprezentována $P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n)$, zkráceně zapsáno $P(x_1, x_2, \dots, x_n)$. Řetízkové pravidlo pravděpodobnosti dovoluje faktorizovat sdružené pravděpodobnosti následujícím způsobem [6]:

$$\begin{aligned} P(x_1, x_2, \dots, x_n) &= P(x_1) \times P(x_2|x_1) \times \dots \times P(x_n|x_1, \dots, x_{n-1}) \\ &= \prod_i P(x_i|x_1, \dots, x_{i-1}) \end{aligned} \quad (3.1)$$



Obrázek 3.1: Typy usuzování [6]

Na základě Markovovy podmínky (kapitola 3.5) mají na hodnotu uzlu vliv pouze otcovské uzly, proto lze vztah přepsat na:

$$P(x_1, x_2, \dots, x_n) = \prod_i P(x_i | \text{Parents}(X_i)), \quad (3.2)$$

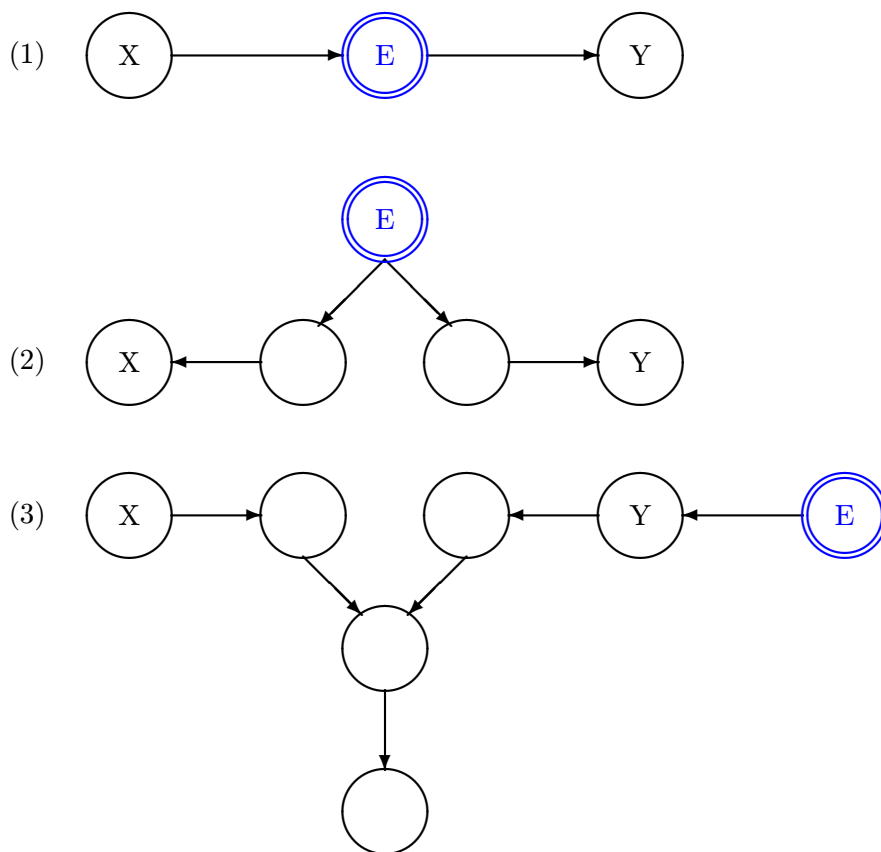
kde $\text{Parents}(X_i) \subseteq \{X_1, \dots, X_{i-1}\}$.

3.8 d-Separace

d-Separace je jednou z důležitých vlastností BN využívaných při inferenci. d-Separace se využívá při tvorbě inferenčního algoritmu, pro samotnou tvorbu a používání BN se nevyužívá.

Množina uzlů E d-separuje dvě množiny uzlů X a Y , pokud každá neorientovaná cesta z uzlu v množině X do uzlu z množiny Y je blokována E . Cesta je blokována množinou uzlů E , pokud na cestě existuje uzel Z , pro který platí jedna z následujících podmínek [11], [8]:

1. Z náleží do množiny E a zároveň na cestě vede jedna hrana (šipka) do Z a druhá ze Z .
2. Z náleží do množiny E a zároveň obě hrany cesty vedou ze Z .
3. Z ani žádný z jeho synovských uzlů není v E a zároveň obě hrany cesty vedou do Z .



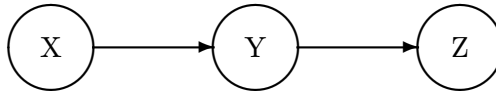
Obrázek 3.2: Ukázka tří situací, ve kterých cesta z X do Y je blokována na základě poznatku E [6]

Nechť X , Y a E jsou množiny uzlů náležících do BN. Pokud každá neorientovaná cesta pro uzel z množiny uzlů X do uzlu z množiny uzlů Y je d-separovatelná E , pak na základě E jsou množiny uzlů X a Y podmíněně nezávislé.

3.9 Základní struktury bayesovských sítí

Na základě definice d-separace existují tři základní typy struktur BN.

3.9.1 Sériové spojení

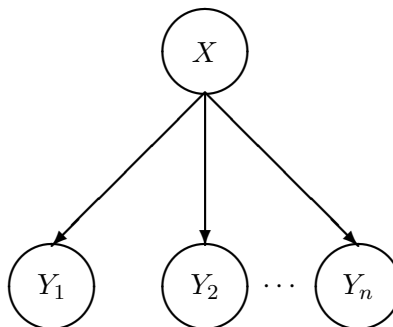


Obrázek 3.3: Sériové propojení [11]

Na obrázku 3.3 má X vliv na Y a následně Y má vliv na Z. Znalost konkrétní hodnoty Z bude ovlivňovat hodnotu pravděpodobnosti Y, která ovlivňuje hodnotu pravděpodobnosti Z. Obdobně poznatek o Z má skrze Y vliv na pravděpodobnost X. V případě, že je známá hodnota Y, je cesta mezi X a Z blokována – X a Z se stanou nezávislými, nebo-li X a Z jsou d-seperované Y [11].

3.9.2 Divergující spojení

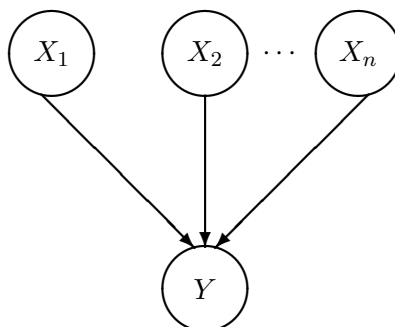
Na obrázku 3.4 na sebe všechny synovské uzly X mohou mít vliv. V případě, že hodnota X je známá, tak tomu tak není – Y_1, Y_2, \dots, Y_n jsou d-separované X [11].



Obrázek 3.4: Divergující spojení [11]

3.9.3 Konvergující spojení

V případě, že v BN na obrázku 3.5 není známa konkrétní hodnota Y, neprobíhá žádné ovlivňování uzlů X_1, \dots, X_n mezi sebou. Uzly X_1, \dots, X_n jsou mezi sebou nezávislé. Situace je přesně opačná než u sériového a divergujícího spojení. Uzly se mohou ovlivňovat v případě, když je poznatek o uzlu Y. Uzly X_1, \dots, X_n se stanou závislými. Jedna příčina ovlivňuje druhou [11].



Obrázek 3.5: Konvergující spojení [11]

3.10 Učení Bayesovských sítí

Bayesovské sítě dosáhly značného úspěchu v tisících aplikacích modelování, predikce a kontroly. Mnoho z těchto sítí bylo vytvořeno tradičním procesem znalostního inženýrství. Strukturu i pravděpodobnosti určili experti na daný obor. Tento proces je však limitován nedostatkem odborné znalosti a také časem a cenou návrhu a validace BN. V oborech obsahujících značné množství dostupných dat je výhodné mít možnost automatického učení, které může nahradit nebo doplnit znalostní inženýrství.

Byly vynalezeny rozmanité techniky pro tvorbu BN. Kauzální vytváření BN je v současné době důležitým odvětvím dolování z dat.

Existují dva učící problémy u BN. Učení struktury BN a učení hodnot podmíněných pravděpodobností [12].

Kapitola 4

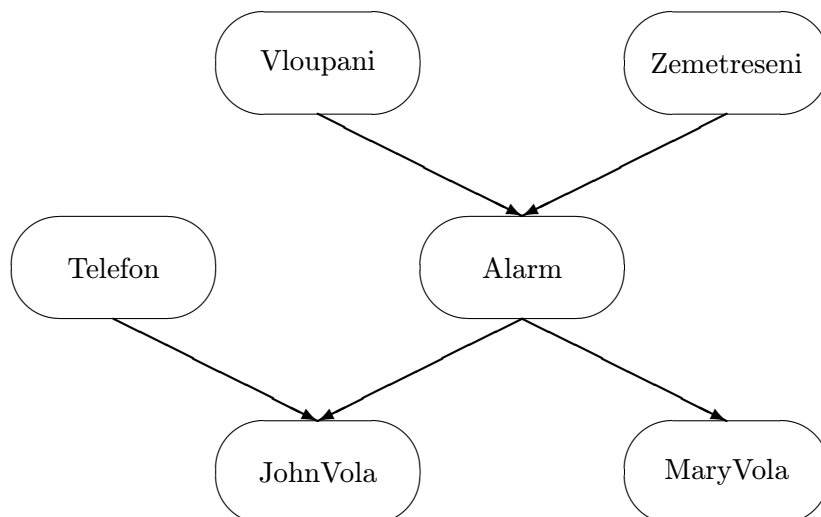
Příklady Bayesovských sítí

V této kapitole se nachází dva příklady, na kterých je v následující kapitole předváděna pravděpodobnostní inference.

4.1 Rozšířený příklad zemětřesení

Příklad: Máte nainstalován nový alarm proti loupeži. Ten spolehlivě detekuje vloupání, ale také se může spustit v případě výskytu slabého zemětřesení. Dva sousedi, John a Mary, slíbili zavolat, když uslyší alarm. John volá vždy, když uslyší alarm, ale občas si plete alarm se zvukem, který vydává zvonící telefon. Na druhou stranu Mary má ráda hlasitou hudbu, tudíž někdy alarm neslyší.

Všechny uzly jsou booleovského charakteru. Nabývají hodnot T (true, pravda) a F (false, nepravda). Tento BN model předpokládá, že John a Mary si nevšimají loupeže přímo a ani nezaznamenávají slabé zemětřesení. Dále neexistuje explicitní reprezentace hlasité hudby, která zabraňuje Mary slyšet alarm. Zvonění telefonu, které máte Johna, je explicitně modelováno [6].



Obrázek 4.1: Rozšířený příklad zemětřesení [6]

<i>Vloupání</i>		<i>Zemetreseni</i>		<i>Telefon</i>	
$P(V = T)$	$P(V = F)$	$P(Z = T)$	$P(Z = F)$	$P(T = T)$	$P(T = F)$
0,01	0,99	0,02	0,98	0,05	0,95

<i>Alarm</i>			
V	Z	$P(A = T V, Z)$	$P(A = F V, Z)$
T	T	0,95	0,05
T	F	0,94	0,06
F	T	0,29	0,71
F	F	0,001	0,999

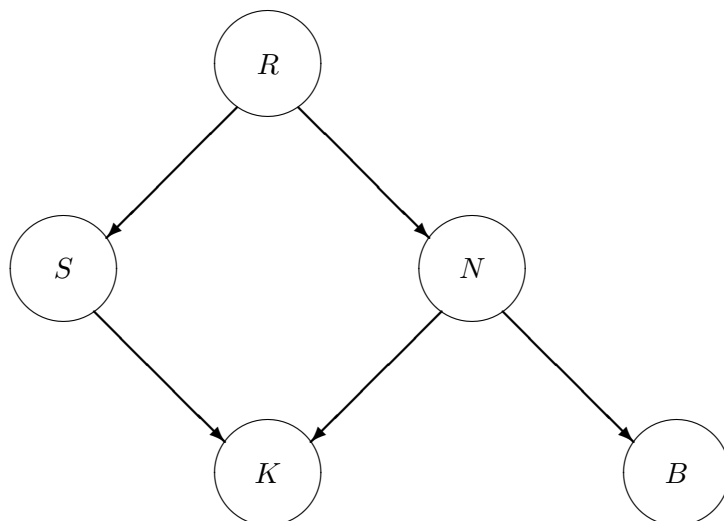
<i>JohnVola</i>			
T	A	$P(J = T T, A)$	$P(J = F T, A)$
T	T	0,95	0,05
T	F	0,5	0,5
F	T	0,9	0,1
F	F	0,01	0,99

<i>MaryVola</i>		
A	$P(M = T A)$	$P(M = F A)$
T	0,7	0,3
F	0,01	0,99

Tabulka 4.1: Tabulky CPT pro rozšířený příklad zemětřesení [6]

4.2 Metastázující rakovina

Příklad: Metastázující rakovina je možnou příčinou nádorů mozku a je také vysvětlením pro zvýšenou hodnotu vápníku v séru. Jak nádor mozku, tak zvýšená hladina vápníku v séru může vysvětlit pacientovo upadnutí do kómatu. Nádory mozku mají navíc spojení se silnými bolestmi hlavy [6].



Obrázek 4.2: Příklad metastázující rakoviny [6]

<i>Rakovina</i>		
	$P(R = T)$	$P(R = F)$
	0,2	0,8

<i>Serum</i>		
R	$P(S = T R)$	$P(S = F R)$
T	0,8	0,2
F	0,2	0,8

<i>Nador</i>		
R	$P(N = T R)$	$P(N = F R)$
T	0,2	0,8
F	0,05	0,95

<i>BolestHlavy</i>		
N	$P(B = T N)$	$P(B = F N)$
T	0,8	0,2
F	0,6	0,4

<i>Koma</i>			
S	N	$P(K = T S, N)$	$P(K = F S, N)$
T	T	0,8	0,2
T	F	0,8	0,2
F	T	0,8	0,2
F	F	0,05	0,95

Tabulka 4.2: Tabulky CPT pro příklad metastázující rakoviny [6]

Kapitola 5

Inference v Bayesovských sítích

Základním úkolem libovolného pravděpodobnostního inferenčního systému je výpočet podmíněné pravděpodobnosti pro množinu dotazovaných uzlů na základě poznatků o síti. Poznatkem se rozumí přiřazení konkrétní hodnoty uzlu.

Tato kapitola se zaměřuje na hlavní typy inferenčních algoritmů – exaktní a přibližný, které byly vyvinuty v průběhu posledních 20 let. Jsou zde uvedeny a na příkladu ilustrovány jejich jednotlivé varianty. Různé algoritmy jsou totiž vhodné pro rozdílné struktury sítě [6].

Sítě, které mají podobu řetězce, vyžadují pouze opakované aplikování Bayesovy věty. Inference v jednoduchých stromových strukturách lze provést pomocí lokálních výpočtů a přeposílání informací mezi uzly. V případě, že dva uzly propojuje více než jedna cesta, jsou inferenční algoritmy složitější. V takových sítích se využívá shlukování uzlů. Pro některé sítě nelze použít exaktní inferenci, protože je pro tyto sítě výpočetně neproveditelná. V těchto případech se musí použít přibližný inferenční algoritmus.

5.1 Exaktní inference v řetězcích

5.1.1 Dvojuzlová síť

Nejjednodušším případem BN je síť o dvou uzlech $X \rightarrow Y$.

Jestliže je známa konkrétní hodnota rodičovského uzlu, například $X = x$, pak posteriorní pravděpodobnost (důvěra v) Y lze určit přímým odečtením $P(Y|X = x)$ z tabulky podmíněných pravděpodobností.

Jestliže je známa konkrétní hodnota synovského uzlu, například $Y = y$, pak inferenční úloha aktualizování důvěry pro X je provedena jednoduchým aplikováním Bayesovy věty.

$$\begin{aligned} P(X = x|Y = y) &= \frac{P(Y = y|X = x)P(X = x)}{P(Y = y)} \\ &= \alpha P(x)\lambda(x) \end{aligned}$$

kde

$$\alpha = \frac{1}{P(Y = y)}$$

$P(x)$ je priorní pravděpodobnost a $\lambda(x) = P(Y = y|X = x)$ je věrohodnost. Pro poznatky není potřeba znát priorní pravděpodobnost. Normalizační konstantu α lze spočítat díky tomu, že celková důvěra pro všechny hodnoty proměnné X musí být rovna jedné [6].

Příklad: Chripka \longrightarrow VysTeplota

Představme si jednoduchý model, ve kterém chřipka způsobuje vysokou teplotu [6]. Priorní pravděpodobnost máme $P(\text{Chripka} = T) = 0,05$ a hodnoty v tabulce podmíněných pravděpodobností jsou $P(\text{VysTeplota} = T | \text{Chripka} = T) = 0,9$ a $P(\text{VysTeplota} = T | \text{Chripka} = F) = 0,2$. Dále známe poznatek $\text{VysTeplota} = T$. Výpočet tohoto diagnostického usuzování je poté následující:

$$\begin{aligned}
 P(\text{Chripka} = T | \text{VysTeplota} = T) &= \alpha P(\text{Chripka} = T) \lambda(\text{Chripka} = T) \\
 &= \alpha \times 0,05 \times 0,9 \\
 &= 0,045\alpha \\
 P(\text{Chripka} = F | \text{VysTeplota} = T) &= \alpha P(\text{Chripka} = F) \lambda(\text{Chripka} = F) \\
 &= \alpha \times 0,95 \times 0,2 \\
 &= 0,19\alpha
 \end{aligned}$$

Následně α lze spočítat ze vztahu

$$P(\text{Chripka} = T | \text{VysTeplota} = T) + P(\text{Chripka} = F | \text{VysTeplota} = T) = 1 = 0,045\alpha + 0,19\alpha$$

vyjádření α

$$\alpha = \frac{1}{0,045 + 0,19}$$

aktualizovaná důvěra

$$\begin{aligned}
 P(\text{Chripka} = T | \text{VysTeplota} = T) &= \frac{0,045}{0,045 + 0,19} = 0,1915 \\
 P(\text{Chripka} = F | \text{VysTeplota} = T) &= \frac{0,19}{0,045 + 0,19} = 0,8085
 \end{aligned}$$

5.1.2 Tříuzlová síť

Tutíž metodu lze aplikovat i na síť, ve které se nachází tři uzly, $X \rightarrow Y \rightarrow Z$.

Pokud je známa hodnota kořenové uzlu, $X = x$, probíhá inference ve směru orientace hran a zahrnuje jednoduché aplikování řetízkového pravidla s využitím nezávislosti v síti.

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

V případě, že poznatkem je hodnota listového uzlu, $Z = z$, diagnostická inference k získání $P(X | Z = z)$ vyžaduje aplikování Bayesovy věty a řetízkového pravidla [6].

$$\begin{aligned}
 P(X = x | Z = z) &= \frac{P(Z = z | X = x) P(X = x)}{P(Z = z)} \\
 &= \frac{\sum_{Y=y} P(Z = z | Y = y, X = x) P(Y = y | X = x) P(X = x)}{P(Z = z)} \\
 &= \frac{\sum_{Y=y} P(Z = z | Y = y) P(Y = y | X = x) P(X = x)}{P(Z = z)} (Z \perp\!\!\!\perp X | Y) \\
 &= \alpha P(x) \lambda(x)
 \end{aligned}$$

kde

$$\lambda(x) = P(Z = z|X = x) = \sum_{Y=y} P(Z = z|Y = y)P(Y = y|X = x)$$

Příklad: Chripka \rightarrow VysTeplota \rightarrow Teplomer

Předchozí příklad jsme rozšířili o proměnnou *Teplomer*, která reprezentuje hodnoty odečítané z teploměru (hodnota *Teplomer* = *T* znamená, že byla na teploměru odečtena vysoká teplota) [6]. Možné nepřesnosti v odečítání na teploměru jsou reprezentovány následovně:

- $P(\text{Teplomer} = T | \text{VysTeplota} = T) = 0,95$ (5% šance na falešně negativní odečítání)
- $P(\text{Teplomer} = T | \text{VysTeplota} = F) = 0,15$ (15% šance na falešně pozitivní odečítání)

Uvažme, že jsme odečetli hodnotu *Teplomer* = *T*. Výpočet $P(\text{Chripka} | \text{Teplomer} = T)$ bude následující:

$$\begin{aligned} \lambda(\text{Chripka} = T) &= P(\text{Teplomer} = T | \text{VysTepl} = T) \times \\ &\quad P(\text{VysTepl} = T | \text{Chripka} = T) \\ &\quad + P(\text{Teplomer} = T | \text{VysTeplota} = F) \times \\ &\quad P(\text{VysTepl} = F | \text{Chripka} = T) \\ &= 0,95 \times 0,9 + 0,15 \times 0,1 \\ &= 0,855 + 0,015 = 0,87 \\ P(\text{Chripka} = T | \text{Teplomer} = T) &= \alpha P(\text{Chripka} = T) \lambda(\text{Chripka} = T) \\ &= \alpha 0,05 \times 0,87 = 0,0435\alpha \\ \lambda(\text{Chripka} = F) &= P(\text{Teplomer} = T | \text{VysTepl} = T) \times \\ &\quad P(\text{VysTepl} = T | \text{Chripka} = F) \\ &\quad + P(\text{Teplomer} = T | \text{VysTeplota} = F) \times \\ &\quad P(\text{VysTepl} = F | \text{Chripka} = F) \\ &= 0,95 \times 0,2 + 0,15 \times 0,8 \\ &= 0,19 + 0,12 = 0,31 \\ P(\text{Chripka} = F | \text{Teplomer} = T) &= \alpha P(\text{Chripka} = F) \lambda(\text{Chripka} = F) \\ &= \alpha 0,95 \times 0,31 = 0,2945\alpha \end{aligned}$$

Vyjádření α

$$\alpha = \frac{1}{0,0435 + 0,2945}$$

dosazením získáme následující výsledky

$$\begin{aligned} P(\text{Chripka} = T | \text{Teplomer} = T) &= \frac{0,0435}{0,0435 + 0,2945} = 0,1287 \\ P(\text{Chripka} = F | \text{Teplomer} = T) &= \frac{0,2945}{0,0435 + 0,2945} = 0,8713 \end{aligned}$$

Inference v řetězcích je jednoduchá. Jedná se pouze o aplikování Bayesovy věty a předpokladů podmíněné nezávislosti v řetězcích.

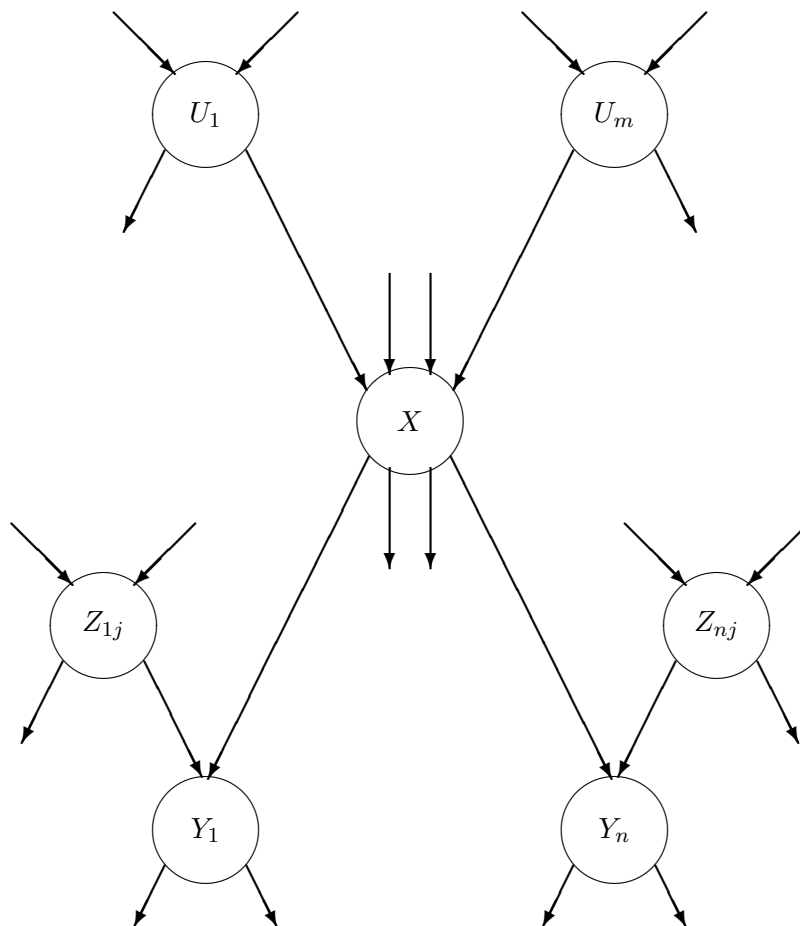
5.2 Exaktní inference v polystromech

Následující část je zaměřena na provádění exaktní inference v případech, kdy má síť podobu jednoduché struktury nazývané polystrom. V polystromech se nachází maximálně jedna cesta mezi všemi dvojicemi uzlů. Z tohoto důvodu se o polystromech mluví jako o jednoduše propojených sítích.

Nechť X je dotazovaný uzel a E je množina poznatků neobsahující X . Inferenční úlohou je v tomto případě vypočet důvěry $P(X|E)$, zjednodušeně zapsáno $Bel(X)$ [6].

Na obrázku 5.1 je zachycen polystrom, který obsahuje uzel X . Polystrom dále pro uzel X obsahuje rodičovské uzly (U_i), synovské uzly (Y_j) a ostatní rodičovské uzly synovských uzlů (Z_{lj}). Aktualizace důvěry uzlu X musí zahrnout všechny poznatky ze všech ostatních částí sítě. V následující síti lze poznatky rozřadit na:

- prediktivní podporu X , od uzlů s poznatkami připojených k X skrze rodičovské uzly U_1, \dots, U_m ,
- diagnostickou podporu X , od uzlů s poznatkami připojených k X skrze synovské uzly Y_1, \dots, Y_n .



Obrázek 5.1: Polystrom [6]

5.2.1 Kimův a Pearlův algoritmus posílání zpráv

Zásadní části algoritmu se nachází níže jako algoritmus 1. Odvození hlavních kroků algoritmu je podrobně uvedeno v [8]. Zjednodušeně řečeno se jedná o opakované aplikování Bayesovy věty a užití podmíněných nezávislostí zakódovaných ve struktuře sítě.

Hlavní myšlenkou algoritmu je lokální aktualizace $Bel(X)$ pomocí tří typů parametrů – $\lambda(X)$, $\pi(X)$ a tabulek podmíněných pravděpodobností. Parametry $\lambda(X)$ a $\pi(X)$ se počítají na základě λ a π zpráv obdržných od rodičovských uzlů uzlu X , respektive od synovských uzlů. λ a π zprávy jsou také rozepisovány z X , aby i sousední uzly mohly provést aktualizaci důvěry.

Algoritmus 1. *Kimův a Pearlův algoritmus posílání zpráv [6]*

Algoritmus vyžaduje tři typy parametrů, které je potřeba udržívat.

- *Aktuální síla prediktivní podpory π , ke které přispívá každé vstupní spojení $U_i \rightarrow X$:*

$$\pi_X(U_i) = P(U_i | E_{U_i \setminus X})$$
kde $E_{U_i \setminus X}$ jsou všechny poznatky propojené do U_i mimo té přes X .
- *Aktuální síla diagnostické podpory λ , ke které přispívá každé výstupní spojení $X \rightarrow Y_j$*

$$\lambda_{Y_j}(X) = P(E_{Y_j \setminus X} | X)$$
kde $E_{Y_j \setminus X}$ jsou všechny poznatky propojené do Y_j skrze jejich otce mimo X .
- *Pevně stanovené tabulky podmíněných pravděpodobností $P(X | U_i, \dots, U_n)$.*

Tyto parametry se používají při lokální inferenci. Lokální inference se provádí pomocí následujících třech kroků, které lze provést v libovolném pořadí.

1. Aktualizace důvěry

Aktualizace důvěry pro uzel X je aktivována zprávami přicházejícími buď ze synovských, nebo z rodičovských uzlů.

V případě aktivace uzlu X se prozkoumají $\pi_X(U_i)$, $\lambda_{Y_j}(X)$ a provede se

$$Bel(x_i) = \alpha \lambda(x_i) \pi(x_i),$$

kde

$$\lambda(x_i) = \begin{cases} 1 & \text{pokud poznatek je } X = x_i \\ 0 & \text{pokud je poznatek pro jiné } x_j \\ \prod_j \lambda_{Y_j}(x_i) & \text{jinak} \end{cases} \quad (5.1)$$

$$\pi(x_i) = \sum_{u_1, \dots, u_n} P(x_i | u_1, \dots, u_n) \prod_i \pi_X(u_i) \quad (5.2)$$

a α je normalizující konstanta ze vztahu $\sum_{x_i} Bel(X = x_i) = 1$

2. Propagace zdola nahoru

Uzel X počítá nové hodnoty λ zpráv a zasílá je otcovským uzlům.

$$\lambda_X(u_i) = \sum_{x_i} \lambda(x_i) \sum_{u_k: k \neq i} P(x_i | u_1, \dots, u_n) \prod_{k \neq i} \pi_X(u_k)$$

3. Propagace shora dolů

Uzel X počítá nové hodnoty π zpráv a posílá je synovským uzlům.

$$\pi_{Y_j}(x_i) = \begin{cases} 1 & \text{pokud } x_i \text{ je poznaťek} \\ 0 & \text{pokud je poznaťek pro jiné } x_j \\ \alpha [\prod_{k \neq j} \lambda_{Y_k}(x_i)] \sum_{u_1, \dots, u_n} P(x_i | u_1, \dots, u_n) \prod_i \pi_X(u_i) & \\ = \frac{\alpha \text{Bel}(x_i)}{\lambda_{Y_j}(x_i)} & \text{jinak} \end{cases} \quad (5.3)$$

Algoritmus vyžaduje následující inicializaci:

- Nastavení všech λ hodnot, λ zpráv a π zpráv na 1.
- Kořenové uzly: V případě, že uzel W nemá žádné rodičovské uzly, tak se nastaví hodnota $\pi(W)$ na hodnotu $P(W)$.

Algoritmus posílání zpráv lze použít na výpočet důvěry pro všechny uzly v síti, a to dokonce dříve než jsou nějaké poznatky k dispozici.

Uzel W může nabývat hodnoty $\{w_1, w_2, \dots, w_n\}$. V případě získání konkrétní hodnoty proměnné $W = w_i$ se nastaví hodnota $\lambda(W) = (0, 0, \dots, 0, 1, 0, \dots, 0)$, kde 1 je na i té pozici.

Zprávy π se posílají ve směru hran, od rodičovského uzlu k synovskému $\pi_{Prijemce}(Odesilatel)$. Zprávy λ se posílají proti orientaci hran, od synovského uzlu k rodičovskému $\lambda_{Odesilatel}(Prijemce)$ [6].

Příklad: V tomto příkladu bude ukázán inferenční výpočet na rozšířeném příkladu zemětřesení (kapitola 4.1) pomocí Kimova a Perlova algoritmu posílání zpráv.

Nejprve provedeme inferenci bez poznatků. Následně přidáme poznatek pro uzel M , a to $M = T$. V příkladu je použit zjednodušený zápis například $\lambda(V) = (1; 1)$, první číslo patří k hodnotě T a druhé k hodnotě F . Dále zápis $A.T$ značí, že se jedná o hodnotu T uzlu A .

Jako první musíme inicializovat λ hodnoty:

$$\begin{aligned} \lambda(V) &= (1; 1) \\ \lambda(Z) &= (1; 1) \\ \lambda(A) &= (1; 1) \\ \lambda(T) &= (1; 1) \\ \lambda(J) &= (1; 1) \\ \lambda(M) &= (1; 1) \end{aligned}$$

dále hodnoty λ a π zpráv:

$$\begin{aligned} \lambda_A(V) = (1; 1) & \quad \text{a} \quad \pi_A(V) = (1; 1) \\ \lambda_A(Z) = (1; 1) & \quad \text{a} \quad \pi_A(Z) = (1; 1) \\ \lambda_J(T) = (1; 1) & \quad \text{a} \quad \pi_J(T) = (1; 1) \\ \lambda_J(A) = (1; 1) & \quad \text{a} \quad \pi_J(A) = (1; 1) \\ \lambda_M(A) = (1; 1) & \quad \text{a} \quad \pi_M(A) = (1; 1) \end{aligned}$$

následuje nastavení π hodnot všech kořenových uzlů:

$$\begin{aligned}\pi(V) &= P(V) = (0, 01; 0, 99) \\ \pi(Z) &= P(Z) = (0, 02; 0, 98) \\ \pi(T) &= P(T) = (0, 05; 0, 95)\end{aligned}$$

uvažme příchod zpráv $\pi_A(V)$ a $\pi_A(Z)$:

$$\begin{aligned}\pi_A(V) &= \pi(V) = (0, 01; 0, 99) \\ \pi_A(Z) &= \pi(Z) = (0, 02; 0, 98)\end{aligned}$$

$$\begin{aligned}\pi(A_T) &= P(A_T|V_T, Z_T)\pi_A(V_T)\pi_A(Z_T) \\ &\quad + P(A_T|V_T, Z_F)\pi_A(V_T)\pi_A(Z_F) \\ &\quad + P(A_T|V_F, Z_T)\pi_A(V_F)\pi_A(Z_T) \\ &\quad + P(A_T|V_F, Z_F)\pi_A(V_F)\pi_A(Z_F) \\ &= (0, 95)(0, 01)(0, 02) + (0, 94)(0, 01)(0, 98) \\ &\quad + (0, 29)(0, 99)(0, 02) + (0, 001)(0, 99)(0, 98) \\ &= 0, 00019 + 0, 009212 + 0, 005742 + 0, 0009702 \\ &= 0, 0161142 \\ \pi(A_F) &= P(A_F|V_T, Z_T)\pi_A(V_T)\pi_A(Z_T) \\ &\quad + P(A_F|V_T, Z_F)\pi_A(V_T)\pi_A(Z_F) \\ &\quad + P(A_F|V_F, Z_T)\pi_A(V_F)\pi_A(Z_T) \\ &\quad + P(A_F|V_F, Z_F)\pi_A(V_F)\pi_A(Z_F) \\ &= (0, 05)(0, 01)(0, 02) + (0, 06)(0, 01)(0, 98) \\ &\quad + (0, 71)(0, 99)(0, 02) + (0, 999)(0, 99)(0, 98) \\ &= 0, 09838858\end{aligned}$$

uvažme příchod zprávy $\pi_M(A)$:

Vzhledem k tomu, že uzel A má více synovských uzlů, musí se příchozí π zprávy násobit λ zprávami od ostatních synovských uzlů.

$$\begin{aligned}\pi_M(A_T) &= \pi(A_T)\lambda_J(A_T) \\ &= 0, 0161142 \cdot 1 = 0, 0161142 \\ \pi_M(A_F) &= \pi(A_F)\lambda_J(A_F) \\ &= 0, 9838858 \cdot 1 = 0, 9838858\end{aligned}$$

$$\begin{aligned}
\pi(M_T) &= P(M_T|A_T)\pi_M(A_T) + P(M_T|A_F)\pi_M(A_F) \\
&= (0, 7)(0, 0161142) + (0, 01)(0, 9838858) \\
&= 0, 0211188 \\
\pi(M_F) &= P(M_F|A_T)\pi_M(A_T) + P(M_F|A_F)\pi_M(A_F) \\
&= (0, 3)(0, 0161142) + (0, 99)(0, 9838858) \\
&= 0, 9788812
\end{aligned}$$

uvažme příchod zprávy $\pi_J(P)$ a $\pi_J(A)$:

$$\begin{aligned}
\pi_J(T) &= \pi(T) = (0, 05; 0, 95) \\
\pi_J(A_T) &= \pi(A_T)\lambda_M(A_T) \\
&= 0, 0161142 \cdot 1 = 0, 0161142 \\
\pi_J(A_F) &= \pi(A_F)\lambda_M(A_F) \\
&= 0, 9838858 \cdot 1 = 0, 9838858
\end{aligned}$$

$$\begin{aligned}
\pi(J_T) &= P(J_T|T_T, A_T)\pi_J(T_T)\pi_J(A_T) \\
&\quad + P(J_T|T_T, A_F)\pi_J(T_T)\pi_J(A_F) \\
&\quad + P(J_T|T_F, A_T)\pi_J(T_F)\pi_J(A_T) \\
&\quad + P(J_T|T_F, A_F)\pi_J(T_F)\pi_J(A_F) \\
&= (0, 95)(0, 05)(0, 0161142) + (0, 5)(0, 05)(0, 9838858) \\
&\quad + (0, 9)(0, 95)(0, 0161142) + (0, 01)(0, 95)(0, 9838858) \\
&= 0, 04848712 \\
\pi(J_F) &= P(J_F|T_T, A_T)\pi_J(T_T)\pi_J(A_T) \\
&\quad + P(J_F|T_T, A_F)\pi_J(T_T)\pi_J(A_F) \\
&\quad + P(J_F|T_F, A_T)\pi_J(T_F)\pi_J(A_T) \\
&\quad + P(J_F|T_F, A_F)\pi_J(T_F)\pi_J(A_F) \\
&= (0, 05)(0, 05)(0, 0161142) + (0, 5)(0, 05)(0, 9838858) \\
&\quad + (0, 1)(0, 95)(0, 0161142) + (0, 99)(0, 95)(0, 9838858) \\
&= 0, 95151288
\end{aligned}$$

V tomto okamžiku jsou všechny zprávy rozposlány. Podívejme se, jak vypadají pravděpodobnosti všech uzlů v případě, že není k dispozici žádný poznatek. Pravděpodobnosti kořenových uzlů již známe $P(V) = (0, 01; 0, 99)$, $P(Z) = (0, 02; 0, 98)$ a $P(T) = (0, 05; 0, 95)$.

$$\begin{aligned}
P(A_T) &= \alpha \lambda(A_T) \pi(A_T) \\
&= \alpha(1)(0,0161142) = 0,0161142\alpha \\
P(A_F) &= \alpha \lambda(A_F) \pi(A_F) \\
&= \alpha(1)(0,9838858) = 0,9838858\alpha \\
P(A_T) &= \frac{0,0161142\alpha}{0,0161142\alpha + 0,9838858\alpha} = 0,0161142 \\
P(A_F) &= \frac{0,9838858\alpha}{0,0161142\alpha + 0,9838858\alpha} = 0,9838858
\end{aligned}$$

Stejným způsobem vypočítáme pravděpodobnosti hodnot pro uzly J a M . Získáme tyto pravděpodobnosti $P(J) = (0,04848712; 0,95151288)$ a $P(M) = (0,0211188; 0,9788812)$.

Přidáme poznatek $M = T$

$$\begin{aligned}
\lambda(M_T) &= 1; \pi(M_T) = 1; Bel(M_T) = 1 \\
\lambda(M_F) &= 0; \pi(M_F) = 0; Bel(M_F) = 0
\end{aligned}$$

vypočítáme $\lambda_M(A)$:

$$\begin{aligned}
\lambda_M(A_T) &= P(M_T|A_T)\lambda(M_T) + P(M_F|A_T)\lambda(M_F) \\
&= (0,7)(1) + (0,3)(0) \\
&= 0,7 \\
\lambda_M(A_F) &= P(M_T|A_F)\lambda(M_T) + P(M_F|A_F)\lambda(M_F) \\
&= (0,01)(1) + (0,99)(0) \\
&= 0,01
\end{aligned}$$

$$\begin{aligned}
\lambda(A_T) &= \lambda_J(A_T)\lambda_M(A_T) = (1)(0,7) = 0,7 \\
\lambda(A_F) &= \lambda_J(A_F)\lambda_M(A_F) = (1)(0,01) = 0,01
\end{aligned}$$

odeslání π zpráv všem synovským uzlům mimo toho, ze kterého přišla λ zpráva, tudíž v našem případě pouze $\pi_J(A)$:

$$\begin{aligned}
\pi_J(A_T) &= \pi(A_T)\lambda_M(A_T) \\
&= 0,0161142 \cdot 0,7 = 0,01127994 \\
\pi_J(A_F) &= \pi(A_F)\lambda_M(A_F) \\
&= 0,9838858 \cdot 0,01 = 0,009838858
\end{aligned}$$

$$\begin{aligned}
\pi(J_T) &= P(J_T|T_T, A_T)\pi_J(T_T)\pi_J(A_T) \\
&\quad + P(J_T|T_T, A_F)\pi_J(T_T)\pi_J(A_F) \\
&\quad + P(J_T|T_F, A_T)\pi_J(T_F)\pi_J(A_T) \\
&\quad + P(J_T|T_F, A_F)\pi_J(T_F)\pi_J(A_F) \\
&= (0,95)(0,05)(0,01127994) + (0,5)(0,05)(0,009838858) \\
&\quad + (0,9)(0,95)(0,01127994) + (0,01)(0,95)(0,009838858) \\
&\doteq 0,01032 \\
\pi(J_F) &\doteq 0,0106
\end{aligned}$$

λ zpráva uzlu T se neposílá, protože hodnoty λ uzlu J se neměnily. Zbývají odeslat λ zprávy pro uzly V a Z :

$$\begin{aligned}
\lambda_A(V_T) &= [P(A_T|V_T, Z_T)\pi_A(Z_T) + P(A_T|V_T, Z_F)\pi_A(Z_F)]\lambda(A_T) \\
&\quad + [P(A_F|V_T, Z_T)\pi_A(Z_T) + P(A_F|V_T, Z_F)\pi_A(Z_F)]\lambda(A_F) \\
&= [(0,95)(0,02) + (0,94)(0,98)](0,7) + [(0,05)(0,02) + (0,06)(0,98)](0,01) \\
&\doteq 0,65874 \\
\lambda_A(V_F) &= [P(A_T|V_F, Z_T)\pi_A(Z_T) + P(A_T|V_F, Z_F)\pi_A(Z_F)]\lambda(A_T) \\
&\quad + [P(A_F|V_F, Z_T)\pi_A(Z_T) + P(A_F|V_F, Z_F)\pi_A(Z_F)]\lambda(A_F) \\
&= [(0,29)(0,02) + (0,001)(0,98)](0,7) + [(0,71)(0,02) + (0,999)(0,98)](0,01) \\
&\doteq 0,01468 \\
\lambda(V_T) &= \lambda_A(V_T) = 0,65874 \\
\lambda(V_F) &= \lambda_A(V_F) = 0,01468
\end{aligned}$$

$$\begin{aligned}
\lambda_A(Z_T) &= [P(A_T|V_T, Z_T)\pi_A(V_T) + P(A_T|V_F, Z_T)\pi_A(V_F)]\lambda(A_T) \\
&\quad + [P(A_F|V_T, Z_T)\pi_A(V_T) + P(A_F|V_F, Z_T)\pi_A(V_F)]\lambda(A_F) \\
&= [(0,95)(0,01) + (0,29)(0,99)](0,7) + [(0,05)(0,01) + (0,71)(0,99)](0,01) \\
&\doteq 0,214654 \\
\lambda_A(Z_F) &= [P(A_T|V_T, Z_F)\pi_A(V_T) + P(A_T|V_F, Z_F)\pi_A(V_F)]\lambda(A_T) \\
&\quad + [P(A_F|V_T, Z_F)\pi_A(V_T) + P(A_F|V_F, Z_F)\pi_A(V_F)]\lambda(A_F) \\
&= [(0,94)(0,01) + (0,001)(0,99)](0,7) + [(0,06)(0,01) + (0,999)(0,99)](0,01) \\
&\doteq 0,0171691 \\
\lambda(Z_T) &= \lambda_A(Z_T) = 0,214654 \\
\lambda(Z_F) &= \lambda_A(Z_F) = 0,0171691
\end{aligned}$$

Nakonec dopočítáme pravděpodobnosti hodnot pro všechny uzly (způsob výpočtu je

uveden výše):

$$\begin{aligned}
Bel(V) &= (0, 311; 0, 689) \\
Bel(Z) &= (0, 203; 0, 797) \\
Bel(A) &= (0, 534; 0, 466) \\
Bel(T) &= (0, 05; 0, 95) \\
Bel(J) &= (0, 493; 0, 507) \\
Bel(M) &= (1; 0)
\end{aligned}$$

5.3 Inference s nejistým poznatkem

Předpokládá se, že poznatek je výsledkem přímého pozorování hodnoty proměnné. Důvěra uzlu je pro hodnotu poznatku nastavena na 1 a pro všechny ostatní hodnoty na 0. Toto je případ konkrétního poznatku, kdy proměnná nabývá konkrétní hodnoty. Takovýto poznatek vložený do algoritmu posílání zpráv je zastoupen nulovým vektorem, ve kterém je na pozici hodnoty poznatku 1. Jakmile je vložen konkrétní poznatek, uzel se uzavře a není možné, aby se jeho hodnota změnila.

Nicméně inferenční algoritmy by měly být schopny vypořádat se s poznatkem, které jsou spojeny s nejistotou. Například v příkladu zemětřesení (kapitola 4.1) uvažme tuto situaci: Zavolá vám sousedka Mary, která hlásí, že slyší alarm. Dále kolega ve vaší kanceláři si myslí, že v rádiu slyšel o malém zemětřesení v oblasti vašeho domu, ale je si jistý jen na 80%. Jak lze reprezentovat nejistotu pozorování? Nejistou informaci lze reprezentovat přijetím nového rozdělení pravděpodobnosti přes nejistou proměnnou. To by ve výše zmíněném případě znamenalo nějakým způsobem nastavení $Bel(Z = T)$ na hodnotu 0,8. Nicméně však není žádoucí uzavření hodnoty důvěry, proto se pro reprezentaci nejistoty pozorování používají mechanismy, které toto nezpůsobí [6].

5.3.1 Užití virtuálního uzlu

Nejjednodušší případ nejistého pozorování obsahuje booleovský (mající dvě hodnoty) uzel X s uniformním rozložením priorní pravděpodobnosti $P(X = T) = P(X = F) = 0,5$. Zastoupení nejistoty se provede připojením virtuálního synovského uzlu V , který bez ohledu na pozorovaný uzel obsahuje hodnoty $\{T, F\}$. Nejistota pozorování uzlu X je zachycena v CPT uzlu V . V případě 80% jistoty pozorování bude CPT obsahovat hodnoty $P(V = T|X = T) = 0,8$ a $P(V = T|X = F) = 0,2$ (tabulka není kompletně specifikována, chybí informace o $P(V = F|X)$). Konkrétní poznatek je vložen jako $V = T$. Nyní již lze k inferenčnímu výpočtu použít Bayesovu větu:

$$\begin{aligned}
Bel(X = T) &= \alpha P(V = T|X = T)P(X = T) \\
&= \alpha 0,8 \times 0,5 \\
Bel(X = F) &= \alpha P(V = T|X = F)P(X = F) \\
&= \alpha 0,2 \times 0,5
\end{aligned}$$

Na základě $Bel(X = T) + Bel(X = F) = 1$ se získá $\alpha = 2$, a tudíž $Bel(X = T) = 0,8$ a $Bel(X = F) = 0,2$.

Toto byla situace, ve které mají hodnoty rodičovského uzlu stejnou priorní pravděpodobnost. Na následujících příkladech je ukázána změna hodnota důvěry v případě, že hodnoty nemají stejnou pravděpodobnost výskytu. Věrohodnost pozorování je u všech příkladů stejná, a to 80%.

Uvažme příklad zemětřesení (kapitola 4.1), kde $P(Z = T) = 0,02$ a $P(Z = F) = 0,98$.

$$\begin{aligned} Bel(Z = T) &= \alpha P(V = T|Z = T)P(Z = T) \\ &= \alpha 0,8 \times 0,02 \\ Bel(Z = F) &= \alpha P(V = T|Z = F)P(Z = F) \\ &= \alpha 0,2 \times 0,98 \end{aligned}$$

Na základě $Bel(Z = T) + Bel(Z = F) = 0,016\alpha + 0,196\alpha = 1$ se získá $\alpha \doteq 4,72$, tudíž $Bel(Z = T) = 0,075$ a $Bel(Z = F) = 0,925$.

V dalším příkladu uvažme hodnoty $P(Z = T) = 0,9$ a $P(Z = F) = 0,1$.

$$\begin{aligned} Bel(Z = T) &= \alpha P(V = T|Z = T)P(Z = T) \\ &= \alpha 0,8 \times 0,9 \\ Bel(Z = F) &= \alpha P(V = T|Z = F)P(Z = F) \\ &= \alpha 0,2 \times 0,1 \end{aligned}$$

Na základě $Bel(Z = T) + Bel(Z = F) = 0,72\alpha + 0,02\alpha = 1$ se získá $\alpha \doteq 1,35$, pak tedy $Bel(Z = T) = 0,973$ a $Bel(Z = F) = 0,027$.

Jak lze vyčíst z uvedených příkladů, hodnota důvěry se nezmění přímo na hodnotu věrohodnosti pozorování (to pouze v případě, že hodnoty nabývají se stejnou pravděpodobností), ale podle poměru věrohodností ($P(V = T|Z = T) : P(V = T|Z = F)$) se změní výše uvedeným způsobem.

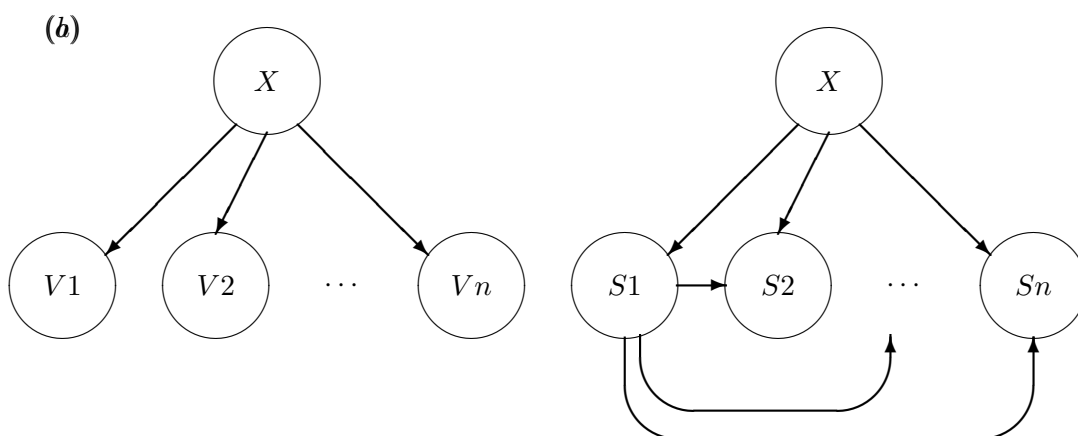
5.3.2 Mnohonásobný nejistý poznatek

Tato situace nastává ve chvíli, kdy existuje více nejistých pozorování stejné proměnné. Například existuje více svědků stejného zločinu. Jeden virtuální synovský uzel zastupuje jedno pozorování. Nicméně pokud nejsou pozorování nezávislá, musí být závislosti reprezentovány explicitně namodelováním se skutečnými uzly. Toto se například týká případu, ve kterém jeden svědek poví ostatním svědkům, jak situaci viděl on [6]. Grafická reprezentace těchto dvou případů je zobrazena na obrázku 5.2.

5.4 Exaktní inference v sítích s vícenásobným spojením

Ve většině případů je však struktura BN spíše graf než pouhý strom. To znamená, že alespoň dva uzly v neorientovaném grafu (struktura BN po odstranění orientace hran) jsou spojeny více než jednou cestou. Taková to síť se nazývá vícenásobně spojená a vyskytuje se v případě, kdy nějaká proměnná má na jinou vliv skrze více než jeden kauzální mechanismus.

Na obrázku 4.2 je zachycena síť metastázující rakoviny. Dvě příčiny K, kterými jsou S a N, sdílí společný otcovský uzel R. V této BN se nachází neorientovaná smyčka přes uzly R, S, K a N. V případě, že se v síti vyskytuje neorientovaná smyčka, Kimův a Pearlův algoritmus posílání zpráv nepracuje správně. Důvod je prostý. Stejný poznatek se dostane k jednomu uzlu dvěma cestami, tudíž se započítá dvakrát.



Obrázek 5.2: Mnohonásobný nejistý poznatek pro stejný uzel lze reprezentovat virtuálním synovským uzlem za každý nejistý poznatek: (a) v případě, že jsou pozorování nezávislá (b) explicitním namodelováním závislostí pozorování [6]

V následujících kapitolách budou ukázány principy a metody, které se používají k inferenci v sítích s vícenásobným spojením. Jedná se o metody exaktní, kam patří shlukovací metody a jejich speciální případ stromy spojení, a přibližné (aproximační), kam spadá logické vzorkování, věrohodnostní váhování a Gibbsovo vzorkování.

5.4.1 Shlukovací metody

Shlukovací inferenční algoritmy transformují původní BN slučováním uzlů na pravděpodobnostně ekvivalentní BN, jež má strukturu polystromu. Transformací se odstraní vícenásobné cesty mezi dvěma uzly, přes které mohl procházet poznatek. V příkladu rakoviny (kapitola 4.2) se tato transformace provede vytvořením nového uzlu Z , který bude kombinací uzlů S a N . Nový uzel má čtyři možné hodnoty $\{TT, TF, FT, FF\}$, které odpovídají všem kombinacím stavů uzlů N a S . Tabulky CPT pro transformovanou síť se spočítají z originálních tabulek následujícím způsobem:

$$\begin{aligned} P(Z|R) &= P(S, N|R) \quad \text{dle definice } Z \\ &= P(S|R)P(N|R) \quad S \text{ a } N \text{ jsou nezávislé díky } R \end{aligned}$$

podobně

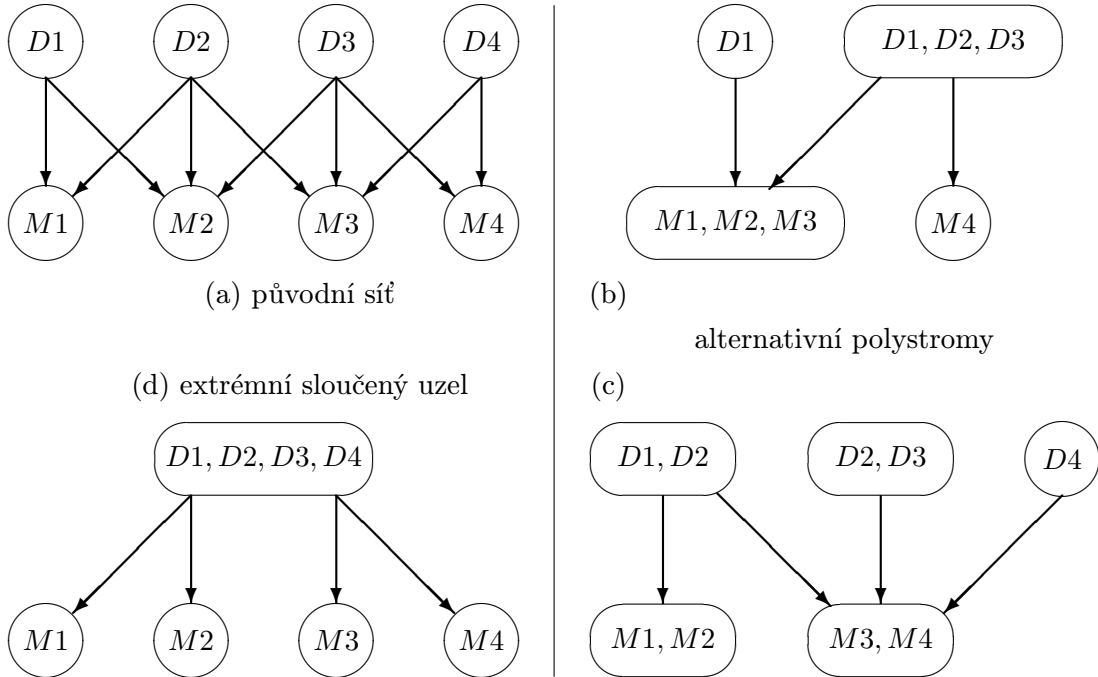
$$\begin{aligned} P(B|Z) &= P(B|S, N) = P(B|N) \quad B \text{ je nezávislé na } S \text{ díky } N \\ P(C|Z) &= P(C|S, B) \quad \text{dle definice } Z \end{aligned}$$

Ukázka transformované tabulky CPT pro uzel Z se nachází na další straně.

Vždy existuje možnost transformování vícenásobně spojené sítě na polystrom. V extrémních případech lze všechny nelistové uzly spojit do jednoho složeného uzlu. Jak lze vidět na obrázku 5.3, není přesně dáno, které uzly se mají shlukovat, proto se síť může transformovat více způsoby.

Z				
R	$P(Z = TT R)$	$P(Z = TF R)$	$P(Z = FT R)$	$P(Z = FF R)$
T	0,16	0,64	0,04	0,16
F	0,01	0,19	0,04	0,76

Tabulka 5.1: Transformovaná tabulka CPT pro uzel Z



Obrázek 5.3: Původní vícenásobně spojenou BN (a) lze transformovat shlukováním na strom (d) nebo na rozdílné polystromy (b),(c) [6]

Velikost tabulky CPT pro zvyšující se počet sloučených uzlů roste exponenciálně, proto je lepší mít menší shluky. V praxi platí čím hustěji propojená síť, tím větší shluky [6]. Exaktní shlukovací algoritmus provádí inferenci ve dvou fázích:

- transformace sítě na polystrom,
- provedení inference na tomto polystromu.

5.4.2 Stromy spojení

Výše uvedená shlukovací metoda se zdá poměrně nahodilá, protože není přesně určeno, které uzly se mají sloučit. Transformace na strom spojení (někdy označována jako transformace na rozložitelný model) představuje metodickou a efektivní metodu shlukování.

Algoritmus 2. Transformace na strom spojení [6]

1. *Moralizace: propojení všech otcovských uzlů, odstranění orientace hran \rightarrow moralizovaný graf.*

2. *Triangulace: přidání hran tak, aby každá kružnice délky > 3 měla tětivu (hrana mezi dvěma sousedními uzly) \rightarrow triangulovaný graf.*
3. *Tvorba nových struktur: nalezení největších klik v triangulovaném grafu, z těchto klik se stanou složené uzly. Pro strom spojení platí podmínka: pokud se uzel nachází ve dvou klikách, tak se tento uzel nachází všude na cestě mezi těmito klikami. Nové struktury (kliky) nelze propojit libovolně. Strom spojení musí být propojen tak, aby byl maximální kostrou grafu, ve kterém je ohodnocením hrany velikost průniku přilehlých uzlů ($|\{R, S, N\} \cap \{S, N, K\}| = |\{S, N\}| = 2$).*
4. *Vytvoření oddělovačů: každá hrana stromu spojení má připojený oddělovač, který se skládá z průniku přilehlých uzlů.*
5. *Výpočet nových parametrů: každý uzel a oddělovač ve stromu spojení má přiřazenou tabulku nad konfiguracemi obsažených proměnných. Všechny hodnoty všech tabulek jsou nastaveny na 1.*
Pro každý uzel X v původní síti se ve stromu spojení vybere jeden uzel Y , který obsahuje X a všechny rodičovské uzly X . Hodnoty v tabulce uzlu Y se roznásobí $P(X|Parents(X))$.
6. *Aktualizace znalostí: Poznatek je přidán do sítě a je propagován pomocí algoritmu posílání zpráv.*

Pseudokódy jednotlivých fází lze nalézt v [4].

Příklad: Ukažme si, jak pracuje tento algoritmus na příkladu metastázuující rakoviny (kapitola 4.2).

V tomto příkladu mají uzly S a N společný synovský uzel K , nicméně není mezi nimi přímé spojení, proto musí být přidáno. Do grafu se nemusí přidávat další hrany, protože se v něm nevyskytuje žádná klika obsahující více jak 3 uzly. Graf je již triangulován (obrázek 5.4(a)).

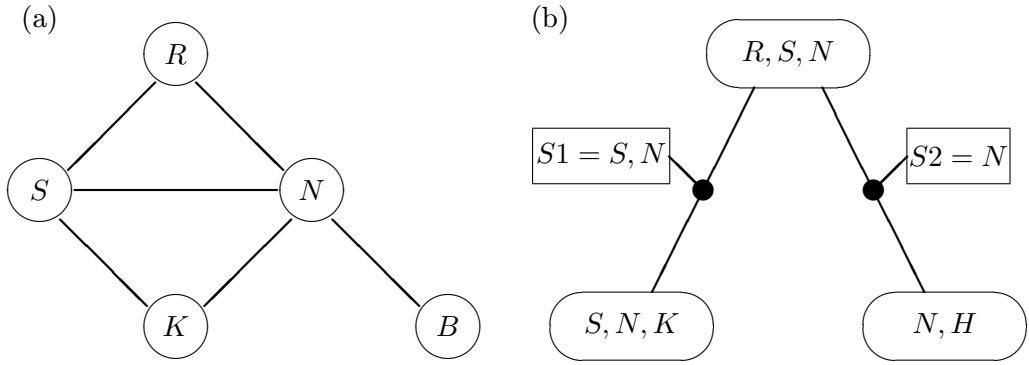
Klika v neorientovaném grafu je definována jako množina uzlů, ve které je každá dvojice uzlů propojena hranou. Maximální klika je takový podgraf, který nelze zvětšit přidáním libovolného uzlu. V příkladu rakoviny lze kliky odečíst z moralizovaného grafu. Nachází se zde tři složené uzly $\{R, S, N\}$, $\{S, N, K\}$ a $\{N, B\}$ (obrázek 5.4(a)) [6]. Zvolíme $\{R, S, N\}$ jako kořenový uzel. Do kořenového uzlu se sbíhají všechny zprávy a také se z něj následně zprávy rozesílají.

Základní potenciály jednotlivých klik ve stromu spojení:

$$\begin{aligned}\phi(R, S, N) &= P(R)P(S|R)P(N|R) \\ \phi(S, N, K) &= P(K|S, N) \\ \phi(N, B) &= P(B|N)\end{aligned}$$

Na základě toho, že kliky musí obsahovat všechny informace kvůli marginalizaci proměnné z kliky, musí být zajištěna konzistence. Nechť V a W jsou dvě sousedící kliky a S je separátor hrany spojující tyto kliky.

$$\sum_{V \setminus S} \phi(V) = \phi(S) = \sum_{W \setminus S} \phi(W)$$



Obrázek 5.4: (a) na obrázku se nachází moralizovaný (triangulovaný) graf metastázující rakoviny, (b) na obrázku se nachází strom spojení příkladu metastázující rakoviny [6]

Absorbe zprávy $V \rightarrow W$:

$$\phi^*(W) = \phi(W) \frac{\phi^*(S)}{\phi(S)}, \text{ kde } \phi^*(S) = \sum_{V \setminus S} \phi(V)$$

Distribuce zprávy opačným směrem:

$$\phi^{**}(V) = \phi(V) \frac{\phi^{**}(S)}{\phi^*(S)}, \text{ kde } \phi^*(V) = \phi(V) \text{ a } \phi^{**}(S) = \sum_{W \setminus S} \phi^*(W)$$

Tímto je zajištěna konzistence:

$$\sum_{V \setminus S} \phi^{**}(V) = \phi^{**}(S) = \sum_{W \setminus S} \phi^*(W)$$

Nastavení počátečních hodnot pro tabulku $\phi(R, S, N)$:

	S=T		S=F	
	N=T	N=F	N=T	N=F
R=T	1	1	1	1
R=F	1	1	1	1

Roznásobení $P(R) = (0, 2; 0, 8)$:

	S=T		S=F	
	N=T	N=F	N=T	N=F
R=T	0,2	0,2	0,2	0,2
R=F	0,8	0,8	0,8	0,8

Roznásobení $P(S|R)$:

	S=T		S=F	
	N=T	N=F	N=T	N=F
R=T	0,16	0,16	0,04	0,04
R=F	0,16	0,16	0,64	0,64

Roznásobení $P(N|R)$:

	S=T		S=F	
	N=T	N=F	N=T	N=F
R=T	0,032	0,128	0,008	0,032
R=F	0,008	0,152	0,032	0,608

Stejným způsobem získáme tabulky pro $\phi(S, N, K)$ a $\phi(N, B)$:

	B=T	B=F
N=T	0,8	0,2
N=F	0,6	0,4

		K=T	K=F
S=T	N=T	0,8	0,2
	N=F	0,8	0,2
S=F	N=T	0,8	0,2
	N=F	0,05	0,95

Nyní máme počáteční tabulky pro všechny kliky v síti. V další fázi budeme zasílat zprávy kořenovému uzlu (sběr informací).

$\phi(S1)$ získáme sumací proměnné K v tabulce $\phi(S, N, K)$:

S=T	N=T	1
	N=F	1
S=F	N=T	1
	N=F	1

Všechny hodnoty v tabulce separátoru $\phi(S1)$ jsou rovny 1, předešlé byly taktéž 1, tudíž vynásobení $\phi(R, S, N)$ poměrem $\frac{\phi^*(S1)}{\phi(S1)}$ s $\phi(R, S, N)$ nic neudělá.

N=T	1
N=F	1

$\phi(S2)$ s potenciálem kořenového uzlu taktéž nic neudělá. Sběr informací do kořenového uzlu byl touto zprávou dokončen. Dalším krokem bude distribuce zpráv z kořenového uzlu. Odeslání zprávy pro S, N, K , určení hodnoty separátoru $\phi(S1)$ z $\phi(R, S, N)$:

S=T		S=F	
N=T	N=F	N=T	N=F
0,04	0,28	0,04	0,64

Nastavení nové hodnoty $\phi(S, N, K)$:

		K=T	K=F
S=T	N=T	0,032	0,008
	N=F	0,224	0,056
S=F	N=T	0,032	0,008
	N=F	0,032	0,608

Určení nové hodnoty $\phi(S2)$ sumací proměnných R a S z $\phi(R, S, N)$:

N=T	N=F
0,08	0,92

Nastavení nové hodnoty $\phi(N, B)$:

	B=T	B=F
N=T	0,064	0,016
N=F	0,552	0,368

Po této aktualizaci všechny kliky obsahují kompletní informace. Z každé kliky lze vyjádřit libovolnou proměnnou.

Ukažme si marginalizaci proměnné S z kliky $\{R, S, N\}$. V tabulce $\phi(R, S, N)$ budeme nejprve sčítat přes proměnnou R :

		S=T		S=F	
		N=T	N=F	N=T	N=F
R	0,04	0,28	0,04	0,64	

Dále přes proměnnou N :

	S=T,N	S=F,N
R	0,32	0,68

Zjištěná pravděpodobnost $P(S) = (0,32; 0,68)$.

5.5 Přibližná inference

Ve většině případů je exaktní inference v BN výpočetně složitá. Pro malé a střední sítě (asi do 40 uzlů) je nejefektivnější exaktní algoritmus využívající stromů spojení v praxi dostatečně dobrý. Pro větší a hustěji propojené sítě musí být použita přibližná (aproximační) inference.

Jedním z přístupů provádění pravděpodobnostní inference ve vícenásobně spojených sítích je stochastická simulace. Stochastická simulace využívá síť ke generování velkého množství případů na základě rozdělení pravděpodobností v síti. Posteriorní pravděpodobnost cílového uzlu je určena použitím těchto vygenerovaných případů. Na základě zákona velkých čísel určená pravděpodobnost konverguje k pravděpodobnosti exaktní [6].

5.5.1 Logické vzorkování

Nejjednodušším vzorkovacím algoritmem je logické vzorkování (LS). Jsou generovány případy náhodně zvolených hodnot pro každý uzel. Generování je váhováno v závislosti na pravděpodobnosti výskytu hodnoty. Uzly se prochází od kořenových uzlů až k listům. Váhou je tedy priorní pravděpodobnost, nebo položka z tabulky CPT, která je dána hodnotami rodičovských uzlů. K určení $P(X|E)$ pomocí hodnoty vzorku $P'(X|E)$ stačí spočítat poměr případů, kdy jsou X a E pravdivé ku případům, kdy je pravdivé pouze E.

Algoritmus 3. *Algoritmus logického vzorkování*

Inicializace:

- Pro každou hodnotu x_i uzlu X vytvoř sumární proměnnou $Count(x_i|e)$
- Vytvoř sumární proměnnou $Count(e)$
- Inicializuj všechny sumární proměnné na 0

Pro každé kolo simulace

1. Pro všechny kořenové uzly zvol váhově hodnotu v závislosti na priorní pravděpodobnosti
2. Opakuj, dokud nejsou dosaženy listové uzly
Zvol náhodné hodnoty pro synovské uzly na základě podmíněných pravděpodobností daných známými hodnotami rodičovských uzlů
3. Uprav sumární proměnné
Pokud případ zahrnuje $E=e$
 $Count(e) \leftarrow Count(e) + 1$
Pokud případ obsahuje $X = x_i$ i $E = e$
 $Count(x_i, e) \leftarrow Count(x_i, e) + 1$

Aktuální určení posteriorní pravděpodobnosti

$$P'(X = x_i|E = e) = \frac{Count(x_i, E = e)}{Count(E = e)}$$

Příklad: Ukažme si na příkladu metastázující rakoviny (kapitola 4.2) jedno kolo simulace LS.

- Jediným kořenovým uzlem v síti je uzel R, který má priorní pravděpodobnost $P(R) = (0, 2; 0, 8)$. Generátor náhodných čísel generuje čísla v rozsahu 0 až 1. Každé vygenerované číslo větší jak 0,2 značí zvolení $R=F$. Uvažujme tento případ.
- Dalším krokem je vážené zvolení hodnot pro uzly S a N. Využijeme k tomu položky z tabulek CPT, kterými jsou $P(S|R = F)$ a $P(N|R = F)$. Uvažme, že $S = T$ a $N = F$, byly náhodně zvoleny.
- Jako poslední se na základě $P(K|S = T, N = F)$ a $P(B|N = F)$ váhově zvolí hodnoty pro K a B. Uvažme vybrání $K=F$ a $B=T$.
- V tomto kole simulace máme hodnoty $R = F, S = T, N = T, K = F$ a $B = T$.
- Pokud nemáme o síti žádný poznatek, či se poznatek vyskytuje v našem vygenerovaném vzorku, například $K = F$, přičteme 1 ke všem sumárním proměnným hodnot, které se ve vzorku nachází, a celkovému počtu běhů, v opačném případě vzorek zahodíme.

Problém LS algoritmu nastává v případě malé pravděpodobnosti poznatků E. Většina vygenerovaných případů je zahozena, a tedy se nezapočítává do počtu běhů [6].

5.5.2 Věrohodnostní váhování

Jedná se o modifikaci LS algoritmu, která překonává problém málo pravděpodobného poznatku. Vzorkovaná hodnota pro každý uzel má vždy hodnotu poznatku. Nicméně počítání počtu běhů stejným způsobem jako v LS by vedlo k úpravě posteriorní pravděpodobnosti, která by už neodpovídala BN modelu. Místo přičítání hodnoty jedna k počtu běhů se podle CPT určí, jaká je věrohodnost, že kombinace hodnot poznatkových uzlů byla dána stavem jejich rodičovských uzlů, a tato věrohodnost se přičte k počtu běhů [6].

Algoritmus 4. *Algoritmus věrohodnostního váhování*

Inicializace:

- Pro každou hodnotu x_i uzlu X vytvoř sumární proměnnou $Count(x_i, e)$
- Vytvoř sumární proměnnou $Count(e)$
- Inicializuj všechny sumární proměnné na 0

Pro každé kolo simulace

1. *Pro všechny kořenové uzly*

Pokud je pro kořenový uzel poznatek, E_j

vyber hodnotu poznatku, e_j

věrohodnost($E_j = e_j$) $\leftarrow P(E_j = e_j)$

nebo zvol váhově hodnotu v závislosti na priorní pravděpodobnosti

2. Opakuj, dokud nejsou dosaženy listové uzly

pokud je poznatek o synovském uzlu, E_j

vyber hodnotu poznatku, e_j

věrohodnost($E_j = e_j$) = $P(E_j = e_j | \text{vybrané hodnoty rodičovských uzlů})$

nebo zvol náhodné hodnoty pro synovské uzly použitím podmíněných pravděpodobností daných známými hodnotami rodičovských uzlů

3. Uprav sumární proměnné

Pokud případ obsahuje $E = e$

pak $\text{Count}(e) \leftarrow \text{Count}(e) + \Pi_j \text{věrohodnost}(E_j = e_j)$

Pokud případ obsahuje $X = x_i$ a zároveň $E = e$

pak $\text{Count}(x_i, e) \leftarrow \text{Count}(x_i, e) + \Pi_j \text{věrohodnost}(E_j = e_j)$

Určení posteriorní pravděpodobnosti

$$P'(X = x_i | E = e) = \frac{\text{Count}(x_i, E = e)}{\text{Count}(E = e)}$$

Příklad: Podívejme se, jak tento algoritmus bude pracovat na příkladu metastázující rakoviny (kapitola 4.2), kde máme poznatek $N = T$.

- Určíme hodnotu pro kořenový uzel R na základě priorní pravděpodobnosti $P(R)$ a náhodného čísla (číslo se generuje vždy, když určujeme hodnotu nějaké proměnné). Uvažujeme výběr $R = F$.
- Dále určíme hodnotu pro uzel S podle $P(S | R = F)$. V závislosti na vygenerovaném čísle uvažme, že jsme získali hodnotu $S = T$.
- O uzlu N máme poznatek, že nabývá hodnoty T . V CPT odečteme hodnotu $P(N = T | R = F) = 0,05$, tudíž tento běh se bude počítat jako 0,05 úplného běhu.
- Jako poslední určíme hodnoty pro K a B . Uvažme $K = T$ a $B = T$.
- V tomto kole simulace máme věrohodnost 0,05 a následující kombinaci hodnot: $R = F$, $S = T$, $N = T$, $K = T$ a $B = T$.
- Zvýšíme všechny sumární proměnné hodnot, které se ve vzorku nachází, o hodnotu věrohodnosti (0,05).

5.5.3 Markovský řetězec Monte Carlo – MCMC

Logické vzorkování i věrohodnostní váhování generují vzorek jednotlivě, začínají od začátku. MCMC na druhou stranu generuje vzorek provedením náhodné změny do předchozího vzorku. Provádí se to náhodným vzorkováním hodnoty pro jeden z uzlů, o kterém nemáme poznatek. Vzorkování je podmíněno aktuálními hodnotami uzlů v Markovovu blanketu. Markovovův blanket se sestává z rodičovských uzlů, synovských uzlů a ostatních rodičovských uzlů synovských uzlů.

V další části bude popsán konkrétní případ MCMC nazvaný Gibbsovo vzorkování [6].

5.5.4 Gibbsovo vzorkování

Algoritmus Gibbsova vzorkování začíná v libovolném stavu (hodnoty proměnných, o kterých je znám poznatek, jsou nastaveny na hodnotu poznatku) a generuje následující stav náhodným vzorkováním hodnoty pro jednu z nepozorovaných (o které nemáme poznatek) proměnných Z_i . Vzorkování Z_i je prováděno podmíněně na aktuálních stavech proměnných v Markovově blanketu proměnné Z_i , který lze zapsat $mb(Z_i)$.

Jak je možné, že Gibbsovo vzorkování vrací odpovídající odhad posteriorní pravděpodobnosti? Vzorkovací proces se usadí v dynamickém ekvilibriu, ve kterém dlouhodobý podíl času stráveného v každém stavu je přímo úměrný jeho posteriorní pravděpodobnosti. Tato pozoruhodná vlastnost vyplývá z pravděpodobnosti přechodu, že proces přejde z jednoho stavu do jiného, definovaného podmíněným rozdělením, jež je dáno Markovovým blanketem vzorkované proměnné [12].

Algoritmus 5. *Algoritmus Gibbsova vzorkování [12]*

Parametry a inicializace:

- X – množina všech uzlů v síti
- E – množina uzlů, ke kterým je k dispozici poznatek
- Z – množina uzlů, o kterých nejsou poznatky
- Z_i – konkrétní proměnná z množiny Z ,
- N – počet iterací,
- x – okamžitý stav sítě, všechny proměnné, pro které máme poznatek, jsou nastaveny na hodnotu poznatku,
- pro každou hodnotu x_i uzlu z X vytvoř sumární proměnnou $Count(x_i)$,
- vytvoř sumární proměnnou počtu běhů $Count(e)$,
- inicializuj všechny sumární proměnné na 0.

Inicializuj x náhodnými hodnotami proměnných z Z

Pro každou iteraci, kde jejich celkový počet je roven N proved

Pro každý $Z_i \in Z$ proved

Urči hodnotu Z_i v x vzorkováním z $P(Z_i|mb(Z_i))$

*Zvyš hodnoty všech sumárních proměnných, které jsou obsaženy v x ,
a sumární proměnné počtu běhů*

Určení posteriorní pravděpodobnosti:

$$P'(X = x_i) = \frac{Count(x_i)}{Count(e)}$$

$P(Z_i|mb(Z_i))$ se získá ze vzorce

$$P(Z_i|mb(Z_i)) = P(Z_i|Parents(Z_i))\prod_{Z_j \in CH_{Z_i}} P(Z_j|Parents(Z_j))$$

Příklad: Ukažme si jednu iteraci Gibbsova vzorkování na příkladu metastázující rakoviny (kapitola 4.2). Uvažme, že máme poznatky $K = T$ a $B = F$ a aktuální stav sítě má následující podobu $[R = T, S = F, N = F, K = T, B = F]$.

- Proměnná R je vzorkována podle aktuálních hodnot $mb(R)$. V tomto případě vzorkujeme podle $P(R|S = F, N = F)$. Uvažme, že výsledkem je $R = F$. Nový stav sítě má podobu $[R = F, S = F, N = F, K = T, B = F]$. Inkrementujeme sumární proměnné pro hodnoty všech proměnných, které se nachází v aktuálním stavu sítě.
- Proměnná S je vzorkována podle aktuálních hodnot $mb(S)$. V tomto případě vzorkujeme podle $P(S|R = F, N = F, K = T)$. Uvažme, že výsledkem je $S = T$. Nový stav sítě má podobu $[R = F, S = T, N = F, K = T, B = F]$. Inkrementujeme sumární proměnné pro hodnoty všech proměnných, které se nachází v aktuálním stavu sítě.
- Proměnná N je vzorkována podle aktuálních hodnot $mb(N)$. V tomto případě vzorkujeme podle $P(N|R = F, S = T, K = T, B = F)$. Uvažme, že výsledkem je $N = F$. Nový stav sítě má podobu $[R = F, S = T, N = F, K = T, B = F]$. Inkrementujeme sumární proměnné pro hodnoty všech proměnných, které se nachází v aktuálním stavu sítě.

5.5.5 Hodnocení aproximačních inferenčních algoritmů

V zájmu porovnání a určení kvality algoritmů je potřeba změřit kvalitu řešení. Jednou z možných mír je Kullback–Leiblerova divergence (relativní entropie) mezi skutečným rozložením P a určeným rozložením P' uzlu s i stavy.

$$KL(P, P') = \sum_i P(i) \log \frac{P(i)}{P'(i)}$$

Hodnoty P a P' by měly být stejné, proto by KL-divergence měla být v ideálním případě rovna nule. Jestliže je $P'(i) = 0$, tak KL-divergence je rovna nekonečnu, pokud je ale $P(i) = 0$, tak je rovna nule.

V případě více dotazovaných uzlů se počítá sdružená KL-divergence přes všechny dotazované uzly. Nechť X a Y jsou dotazované uzly a Z je poznaitek, pak pro výpočet bude použit následující vzorec $KL(P(X, Y|Z), P'(X, Y|Z))$ [6]. Přímý výpočet sdružené KL-divergence je však příliš výpočetně náročný [5], proto se často se pro určení chybovosti používá průměr či suma KL-divergencí pro jednotlivé dotazované uzly, což ale není úplně přesné [6]. Nicméně při dalším výskytu v této práci se pod pojmem KL-divergence uvažuje suma KL-divergencí jednotlivých dotazovaných uzlů.

Kapitola 6

Aplikace

Usuzování pomocí BN spočívá v aktualizaci posteriorního rozdělení pravděpodobnosti jedné či více proměnných na základě poznatků. V závislosti na tom, jaké poznatky jsou k dispozici, provádí inference jednu z následujících úloh [6]:

- Monitorování/kontrola.

Příklad: Je pacientova hladina cukru v krvi stabilní, nebo potřebuje podat mimořádnou dávku inzulínu?

- Dopředná predikce.

Predikce se dělí na faktickou (založenou na poznatcích) a hypotetickou (predikování efektu intervence).

Příklady: Faktická – Přežije pacient na základě rentgenu a výsledků krevních testů navrhovanou operaci?

Hypotetická – Pokud pacientovi bude podán přípravek proti srážení krve, zvýší se tím jeho šance na přežití?

V současné době jsou BN používány v širokém spektru oblastí, jako je medicína, strojírenství, vědecké obory, vojenský sektor, školství, obchod, finančnictví, výpočetní technika a jiné. Vybrané typy aplikací jsou uvedeny níže v této kapitole, další aplikace lze nalézt v [6, 11, 10, 8].

6.1 Medicínské aplikace

Medicína je bezesporu oblast, ve které jsou BN aplikace nejvíce rozšířené. Složitá doména, kde obrovským množstvím znalostí disponují zkušení lékaři. Počátky spojení BN a lékařských aplikací sahají již do 80. let dvacátého století. Využitelnost BN v této oblasti spočívá ve využití k diagnóze, prognóze či ke zvolení léčby [6].

6.1.1 Diagnóza

Lékařská diagnóza často bývá zjednodušována na vytvoření hypotézy pro každou chorobu na základě poznatků o pacientovi. Usuzováním se vybírá nejvíce pravděpodobná hypotéza. Formálně lze toto zapsat:

$$diagnosa = \max_i P(D_i|E).$$

Kde $P(D_i|E)$ je pravděpodobnost choroby D_i na základě množiny poznatků o pacientovi E . Množina E představuje pozorované symptomy, příznaky a výsledky laboratorních testů.

6.1.2 Prognóza

Prognóza pokouší predikovat budoucí stav pacienta na základě množiny poznatků o pacientovi a nasazené léčby. Formálně zapsáno:

$$prognóza = P(O|E, T).$$

Kde E je množina poznatků o pacientovi, jako jsou symptomy, příznaky a výsledky laboratorních testů. T zastupuje zvolenou léčbu a proměnná O může představovat očekávanou délku života, kvalitu života či šíření nákazy.

6.1.3 Léčba

Rozšíření BN, influenční diagramy, umožňují modelovat rozhodování. Takto lze při léčbě modelovat rozdílné rozhodnutí za přítomnosti neurčitých proměnných a preferencí. Usuzováním nad všemi možnými způsoby léčby má za následek výběr nejvíce optimální léčby.

6.1.4 Medicínská aplikace – HEPAR II

Projekt HEPARII je systém pro diagnózu jaterních onemocnění, který je zaměřen na snížení počtu prováděných biopsií. Model BN zachycuje přes 70 proměnných. V modelu je pokryto 11 různých jaterních onemocnění a 61 lékařských poznatků o pacientovi [10].

6.2 Mapování potenciálu nerostných zdrojů

Klasifikace a přidělení využití krajiny je jedním z hlavních strategických problémů ve většině zemí našeho světa. Při takovémto rozhodnutí mezi sebou soupeří rozdílné požadavky zainteresovaných subjektů, a proto se vyžadují spolehlivé informace o potenciálu zdrojů. Tento typ informace umožňuje rozhodovacím orgánům určit socioekonomické užítky pro rozdílné využití krajiny. Potenciál pro různé typy zdrojů vyskytující se na zemském povrchu je obecně jednodušší určit než pro ty zdroje, které se vyskytují pod ním. V mnoha situacích informace o potenciálu pro podpovrchově vyskytující se zdroje nepatří mezi vstupy pro rozhodování využití krajiny. Z tohoto důvodu je mnoho pozemků, kde by se měly vyskytovat nerosty, vystaveno důkladnějšímu průzkumu nerostných ložisek.

Oblasti s nerostným potenciálem jsou charakterizovány geologickými znaky, které geneticky a prostorově souvisí s typem ložisek nerostných surovin. Znaky jsou někdy přímo viditelné, ale ve většině případů je jejich přítomnost vyvozena na základě jednoho či více souborů s geografickými daty, které jsou zpracovány a analyzovány pro získání, vylepšení a reprezentaci znaků pomocí predikčních map. Mapování nerostného potenciálu zahrnuje integraci predikčních map s cílem zařadit oblast jako neplodnou či oblast s nerostným ložiskem v závislosti na druhu nerostu [10].

6.3 Analýza rizikových kriminogenních faktorů

V zaznamenávané historii je zločin přetrvávajícím společenským problémem. S rostoucí urbanizací tento problém nabral jak na rozsahu, tak na složitosti. K zajištění efektivní

kontroly kriminality při limitovaných zdrojích je zapotřebí aktivní přístup donucovacích orgánů. Při přidělování zdrojů se využívá analýza vzorce zločinu. Od doby, kdy jsou města i vzorce zločinu svou povahou dynamické, je zapotřebí systém, který neumí pouze správně predikovat, ale který se také dovede přizpůsobovat změnám různých parametrů v čase. Klíčovým prvkem tohoto přístupu je použití existujících záznamů k sestavení počátečního modelu a pokračování ve vylepšování na základě nových případů. Systém poté poskytuje model pro analýzu rizikových kriminogenních faktorů, který se přizpůsobuje změnám ve vzorcích trestných činů. Schopnost aktualizace počátečního stavu modelu na základě nových dat dělá z BN přístupu přirozenou volbu pro analýzu rizikových kriminogenních faktorů [10].

6.4 Forenzní vědy

Forenzní vědy lze považovat za disciplínu, která se snaží poskytovat odborné znalosti jak při vyšetřování, tak soudních řízeních při určování toho, co se v minulosti stalo. Některé případy jsou jedinečné, neopakovatelné a zůstávají neznámé alespoň v určitém stupni. Nicméně minulé události mohly zanechat zřetelný poznaček. V tomto kontextu se jedná o důkaz, který může být objeven a zkoumán. Například na krev získanou na místě činu může být použita analýza DNA. Výsledky lze následně porovnat s výsledky analýzy DNA provedené na vzorku krve od podezřelého.

Forenzní vědy mohou profitovat z širokého spektra důmyslných metod a analytických technik pro zkoumání rozličných důkazů. Obvykle ale existuje řada faktorů, které omezují důsledky, které mohou být z výsledků vyvozeny. Kupříkladu není jisté, zda krevní skvrnu na místě činu zanechal pachatel. V jiném případě může být krevní skvrna znehodnocena.

V současné době je nezbytné napovědět právníkům, porotě a soudcům o důležitosti různých náleží v případě. Forenzní vědci potřebují kvalifikovat, a pokud je možno, tak i kvantifikovat stavy dostupných znalostí a zároveň se opírat o vhodné prostředky při potýkání se s nejistotou, která má vliv na ucelené vyhodnocení důkazů [10].

6.5 Řízení rizika terorismu

Události v posledních letech zdůrazňují význam potřeby efektivního nástroje pro řízení rizika terorismu. Armáda USA definuje antiterorismus jako defenzivní postoj proti teroristickým hrozbám. Antiterorismus zahrnuje podporu informovanosti o možných hrozbách, odrazení útočníků, vypracování bezpečnostních opatření, přerušení útoku a také zmírňování a řízení následků útoku. Tyto aktivity jsou prováděny vládou a soukromými bezpečnostními službami ve vojenských, civilních a komerčních sítích po celém světě. Klíčovým elementem efektivní protiteroristické strategie je vyhodnocení rizika teroristického útoku pro individuální síť či aktiva. Stanovení hrozby teroristického útoku vyžaduje kombinaci informací z více rozdílných zdrojů, z nichž většina obsahuje vnitřní a neredukovatelné nejistoty.

Zájem o vylepšení metod hodnocení rizika terorismu se rapidně zvýšil v závislosti s útokem na budovy Světového obchodního centra 11. září 2001. Z důvodu nedílné nejistoty řízení rizika terorismu je přirozenou doménou aplikací BN [10].

6.6 Validace senzorů

Kontroléry přijímají informaci, zpracují ji a poskytnou výstup nebo učiní rozhodnutí. Lze si představit, co může způsobit rozhodnutí nebo příkaz na základě chybné informace. Příkla-

dem může být jednotka intenzivní péče v nemocnici, přístroje trvale za různých podmínek monitorující životní funkce pacientů. Je možnost, že se špatně mohou odečíst hodnoty. V případě, že se toto stane, mohou nastat dvě různé situace. Senzor nezaznamená žádnou změnu teploty, i když ta se zvýší na nebezpečnou úroveň. Druhou možností je oznámení nebezpečné teploty senzorem, i když ve skutečnosti je normální. První případ může způsobit vážné škody na pacientově zdraví. Druhá situace může způsobit nouzové ošetření, které může pacientův stav taktéž zhoršit.

Algoritmus na validaci senzorů založený na BN nejprve vytvoří model závislosti mezi senzory (reprezentován BN). Samostatná validace se provádí ve dvou fázích. V první fázi jsou možné chyby detekovány porovnáním aktuálních hodnot senzorů s predikovanou hodnotou pro odpovídající senzor. V druhé fázi jsou skutečné chyby izolovány vytvořením další BN. Tento algoritmus byl například použit k validaci teplotních senzorů plynové turbíny v generátoru elektrické energie [10].

6.7 Modelování studenta

Intelligentní výukové systémy (ITS) se snaží simulovat chování lidského učitele přizpůsobením se studentovi. Klíčovou součástí ITS je model studenta, který poskytuje znalost o každém konkrétním studentovi. Tudíž chování systému lze přizpůsobit studentovým potřebám. V modelu studenta je úroveň znalosti látky určena z předchozích dat o studentovi a také ze studentova chování při interakci se systémem. Modelování studenta je komplexní úloha obsahující neurčitost. Není zcela jasné, jak se člověk učí a ani jak reprezentovat znalosti pomocí počítače. Navíc informace dostupné k vybudování modelu a jeho následné aktualizaci jsou velmi limitovány.

Existuje více přístupů pro modelování studenta. V posledních letech se ale BN staly jednou z upřednostňovaných metod [10].

6.8 Správa silnic a mostů

Systémy správy infrastruktury jsou vytvářeny kvůli poskytování informací a účelné analýze dat tak, aby inženýr nebo projektant mohl učinit vhodnější, levnější a obhajitelné rozhodnutí pro udržení pozemních komunikací. V systémech správy silnic je obvykle obsaženo velké množství nejistých vzájemně provázaných atributů založených na informacích rozdílné kvality.

Správa silnic je obecně charakterizována a rozvinuta ve dvou úrovních:

- úrovni celkové silniční sítě (neobsahuje podrobné technické informace),
- projektové úrovni (v případech, kdy je potřeba provést nějaká práce na určitém úseku komunikace).

Mezi hlavní rozdíly u rozhodování na těchto dvou úrovních patří stupeň a rozsah, pro které jsou rozhodnutí prováděna, a také typ a množství dat potřebných k tomuto rozhodnutí. Rozhodování u silničních sítí se týká plánovaných a strategických otázek celé silniční sítě. Projektová úroveň se zabývá otázkami technické a ekonomické správy silnic.

Správa mostů zahrnuje všechny činnosti související s návrhem, výstavbou, správou a obnovou mostů. Systémy správy mostů jsou poměrně novým přístupem, který byl vyvinut po úspěšné implementaci a používání systémů správy silnic [10].

6.9 Podpora rozhodování při složitých průmyslových procesech

Za účelem určení počátku poruchy procesu je potřeba rychlý a flexibilní nástroj pro podporu rozhodování ve vysoce automatizovaných systémech. To zahrnuje analýzu podmínek procesu a doporučení výhodnějších akcí.

Diagnostické schopnosti, transparentní vysvětlení rozhodnutí a důvěryhodná podpora rozhodování jsou očekávány od uživatelů, kterými jsou výrobní inženýři, operátoři a údržbáři. Díky existenci množství diagnostických nástrojů prvního stupně je cíl zaměřen na poskytnutí podpory rozhodování při výrobním procesu.

Od doby, kdy nejsou všechny podmínky měřitelné či spočítatelné v reálném čase, byly BN shledány jako efektivní a flexibilní nástroj v kompletním výrobním procesu [10].

6.10 Rating společností

Rating společnosti je posouzení společnosti jako cíle investice. Poskytuje důležité informace nejenom pro investory, ale také společností samotným, protože má zásadní vliv na cenu akcií.

Je zveřejňován pravidelně a v případech, kdy je vyžadován ekonomy ratingových agentur, bezpečnostními firmami a obchodními věstníky. Rating je primárně založen na současných finančních údajích o výkonnosti podniku, ale často i neoficiálních nebo vnitřních informacích, podmínkách na trhu a datech minulých. Rating je díky tomuto všemu objektivním úsudkem ekonomického experta. Různí ekonomové mohou stejné společnosti přiřadit rozdílné hodnocení [10].

6.11 Detekce narušení sítě

Rozšíření používání internetu dramaticky přispělo k růstu počtu hrozeb, které se na něm nachází. Za účelem hledání lepší ochrany se počítačová bezpečnost a především systémy pro odhalení průniku do počítačové sítě (NIDS) staly tématem výzkumu a zájmu v boji proti těmto hrozbám. NIDS jsou typem počítačového softwaru, který je schopen rozeznat legitimního uživatele od těch, kteří způsobují hrozby. Vzhledem k rostoucí složitosti a množství útoků jsou NIDS prováděny automatizovaně, tudíž NIDS software automaticky monitoruje použití systému kvůli zjištění chování, které překračuje bezpečnostní politiku. NIDS jsou rozděleny na základě své působnosti na:

- detektory zneužití sítě,
- detektory anomálií.

Detektory zneužití sítě se vypořádávají s hrozbami, které dopředu znají. Tyto systémy se vypořádávají s úplnou útočnou základnou a jejich práce se skládá z dohlížení na celou příchozí komunikaci, kde hledají jakékoliv sekvence nacházející se v jejich znalostní bázi.

Systémy detekce anomálií jsou více ambiciózní díky tomu, že se snaží objevit nové neznámé hrozby. Tyto systémy modelují legitimní užití systému za účelem získání konkrétní míry možných odchylek od běžného stavu sítě. Každá odchylka, která je zhodnocena jako dostatečně významná, je považována za anomálii. Taková to anomálie je následně oznámena lidskému operátorovi [11].

Kapitola 7

Návrh a implementace

V této kapitole je stručně popsán návrh aplikace a v implementační části jsou popsány jednotlivé třídy, ze kterých se program skládá.

7.1 Návrh

Cílem práce bylo navrhnout a implementovat aplikace či applety pro demonstraci různých přístupů k inferencím pro zvolené úlohy.

Za tímto účelem byla navržena aplikace s grafickým uživatelským rozhraním, která dokáže načítat bayesovské sítě ve formátu XBN (standard popsán v [3]) a která umožňuje tvorbu vlastní sítě. Jak u vytvořených, tak u načtených sítí lze upravovat počty hodnot, kterých může uzel nabývat, a také lze upravovat tabulky CPT. Kvůli dobrému přehledu je síť zobrazena jako graf, ve kterém je možno s uzlem libovolně manipulovat.

Do sítě lze vkládat konkrétní poznatek pro libovolný uzel. Inferenčními algoritmy, které byly zvoleny pro implementaci, jsou exaktní:

- Kimův a Pearlův algoritmus posílání zpráv,
- stromy spojení,

a přibližné:

- logické vzorkování,
- věrohodnostní váhování,
- Gibbsovo vzorkování.

U každého z přibližných algoritmů lze zadávat počty iterací. Výsledky se po provedení inference zobrazí do nového panelu, aby bylo možno porovnávat, jaký vliv měl konkrétní poznatek nebo jak se liší výsledky jednotlivých algoritmů.

7.2 Implementace

Výše uvedený návrh byl implementován v jazyce **Java**. V této části textu se nachází pouze stručný popis jednotlivých tříd. Detailnější popis tříd a jejich metod byl vygenerován pomocí utility `javadoc` a nachází se na příloženém CD.

Manuál k vytvořené aplikaci se nachází taktéž na CD.

7.3 Popis tříd

BayesNode

Třída reprezentující uzel (proměnnou) v síti. Obsahuje tabulku CPT, informace o rodičovských a synovských uzlech, počet hodnot a jejich popisky. Dále se zde nachází souřadnice uzlu, na které se uzel zobrazí, a informace o poznatku.

Arc

Třída Arc je explicitní reprezentací hrany mezi uzly.

BayesNet

Třída BayesNet reprezentuje síť. Obsahuje uzly sítě a také hrany mezi nimi.

XBNReader

Třída obsahující metodu `getNet`, která načítá síť ze souboru ve formátu XBN.

TableModel

Tato třída je rozšířením `DefaultTableModel` a využívá se ve třídě `EditCPT`. Model je upraven tak, aby šly zadávat pouze číselné hodnoty. Navíc je zamezena editace buněk, které nenesou informaci o pravděpodobnosti (obsahují popisky hodnot rodičovských uzlů).

ButtonTabComponent

Převzatá třída od společnosti Oracle. Třída slouží k přidání a obsluze zavíracího tlačítka na panel, který je obsažen v instanci třídy `JTabbedPane`.

ShowResults

Třída rozšiřuje `JPanel`. Na instance této třídy se zobrazí výsledky, jednotlivé pravděpodobnosti, které určí zvolený infereční algoritmus.

ShowNet

Třída `ShowNet` rozšiřuje třídu `JPanel` a obsahuje metody, které se starají o zobrazení sítě. Dále má metody pro přidání uzlu, přidání hrany spojující uzly, pohyb uzlu a také překreslování hran a jejich šipek při přesunu uzlu.

IntTextDocument

Třída, která je rozšířením `PlainDocument`, se stará o to, aby bylo možné zapsat (do labelu) pouze celočíselnou hodnotu.

EnterEvidence

`EnterEvidence` je rozšířením `JDialog`. Tato třída slouží pro vybrání (vlození), či zrušení poznatku o uzlu.

EditNode

Třída je rozšířením `JDialog` a slouží pro upravení uzlu. Pomocí metod této třídy lze nastavit název uzlu, počet jeho hodnot a jejich popisky.

EditCPT

`EditCPT` je rozšířením třídy `JDialog`. Pomocí metod této třídy se upravují pravděpodobnosti v tabulce CPT. Ty jsou zobrazeny v `JTable`, která využívá jako model instanci třídy `TableModel`.

PerlMessPassAlg

Tato třída má metody, které provádí inferenční výpočet Perlovým a Kimovým algoritmem posílání zpráv. Pro reprezentaci λ , π , λ -zpráv a π -zpráv byly zvoleny kontejnery `Map`. Algoritmus se spouští metodou `initNet`, další důležitou metodou je `updateTree`, pomocí které se přidává do algoritmu poznatek.

Sampling

Třída obsahující metody a kontejnery, které jsou společné pro všechny vzorkovací algoritmy.

LogicSampling

Třída, která obsahuje metody pro provádění logického vzorkování. Mezi metody patří `initNet`, která určí kořenové a uspořádá nekořenové uzly, a `simulation`, která provádí zadaný počet iterací algoritmu logického vzorkování.

LikelihoodWeighting

Třída má stejné metody jako `LogicSampling` s tím rozdílem, že provádí algoritmus věrohodnostního váhování.

GibbsSampling

Třída obsahující metody, které provádí inferenci Gibbsovým vzorkováním. Nachází se zde metody `prepareSimulation`, která nalezne nepozorované proměnné (není o nich poznatek) a generující první vzorek, a `simulation`, která provádí zadaný počet iterací algoritmu Gibbsova vzorkování.

JunctionGraphArc

Třída reprezentující ohodnocenou hranu ve stromu spojení.

Potential

Třída reprezentující jednu položku v množině potenciálů. Nese hodnotu potenciálu pro obsažnou kombinaci hodnot proměnných. Dále obsahuje metody, které provádí operace s potenciálem.

JunctionTree

Třída obsahuje metody pro provádění algoritmu stromy spojení. Metody `moralize`, `triangulate`, `makeTree` a `setTree` vytvoří ze zadané BN strom spojení. Nastavení počátečních potenciálů se provádí metodou `initialPotentials`, sběr poznatků a jejich následnou zpětnou distribuci obsluhují metody `collectEvidence` a `distributeEvidence`. Výpočet jednotlivých pravděpodobností pro všechny uzly provádí metoda `marginalizeAllNodes`.

JunctionTreeClique

Třída reprezentující kliku ze stromu spojení. Obsahuje metody obsluhující výše zmíněné fáze, jako je inicializace stromu, sběr a distribuce poznatků.

KLDivergence

Třída sloužící pro výpočet KL-divergence. Kvůli zajištění vysoké přesnosti je použit typ `BigDecimal`. Vysoká přesnost je potřeba z důvodu, aby byla suma pravděpodobností hodnot uzlu rovna přesně jedna. Při použití typu `double` mohla vyjít KL-divergence záporná, což si odporuje s její definicí.

ComputeProb

Tato třída je rozšířením třídy `SwingWorker` a stará se o veškeré inferenční výpočty. Všechny výpočty jsou prováděny na pozadí za účelem interaktivního GUI za každé situace. Výsledky jsou po dopočítání zobrazeny na instanci třídy `ShowResults`.

BayesNodeProbPane

Třída rozšiřuje `JPanel` a slouží pro zobrazení vypočtených pravděpodobností uzlu. Pravděpodobnost je vypsána jak do labelu, tak je i pro lepší orientaci zobrazena graficky pomocí instancí `JProgressBar`.

BayesGUI

`BayesGUI` je rozšířením `JFrame` a je hlavní třídou (třídou s metodou `main`). Tato třída tvoří základ grafického uživatelského rozhraní. Obsahuje instanci třídy `JMenuBar` s položkami jednotlivých menu, dále toolbar, na kterém se nachází tlačítka na načtení, uložení sítě a také se zde jsou tlačítka, která slouží ke tvoření vlastní sítě. Na většině plochy hlavního okna se nachází instance třídy `JTabbedPane`, která obsahuje právě jednu instanci třídy `ShowNet` a po provedení inference může obsahovat instance třídy `ShowResults`.

Kapitola 8

Demonstrační aplikace

V této kapitole se nachází popis BN, které byly použity k experimentům s jednotlivými inferenčními metodami.

8.1 ALARM

ALARM je diagnostická aplikace využívající se k prozkoumání inferenčních metod BN. ALARM je systém pro monitorování pacienta. Na základě dostupných poznatků určuje pravděpodobnosti pro různé diagnózy. Cílem monitorujícího systému ALARM je poskytnutí textové zprávy, která uživatele informuje o možných problémech.

V systému ALARM jsou zastoupeny tři typy proměnných. Diagnózy a další kvalitativní informace jsou na nejvyšší úrovni sítě. Tyto uzly nemají žádné rodičovské uzly a jsou uvažovány jako vzájemně nezávislé. Všechny uzly mají přiřazeny množiny vzájemně výlučných a úplných hodnot zastupujících výskyt, absenci či vážnost konkrétního onemocnění.

Měření zastupují libovolnou kvalitativní informaci. Všechny spojitě proměnné jsou zastupovány množinami diskrétních intervalů dělících rozsah proměnné. Mezilehlé proměnné jsou usuzovacími entitami, které nelze měřit přímo.

Vztahy v síti jsou zachyceny od diagnóz k poznatkům. Existují 3 důvody, proč tomu tak je:

- medicínská doména zahrnuje známé kauzální vztahy,
- většina medicínských knih používá tuto formu popisu,
- reprezentace je méně komplikovaná než kauzálním směrem, lze provést více předpokladů nezávislosti.

Alarm obsahuje statistická data a také množství subjektivních poznatků [2].

BN systému ALARM se nachází v souboru `ALARM.xbn`.

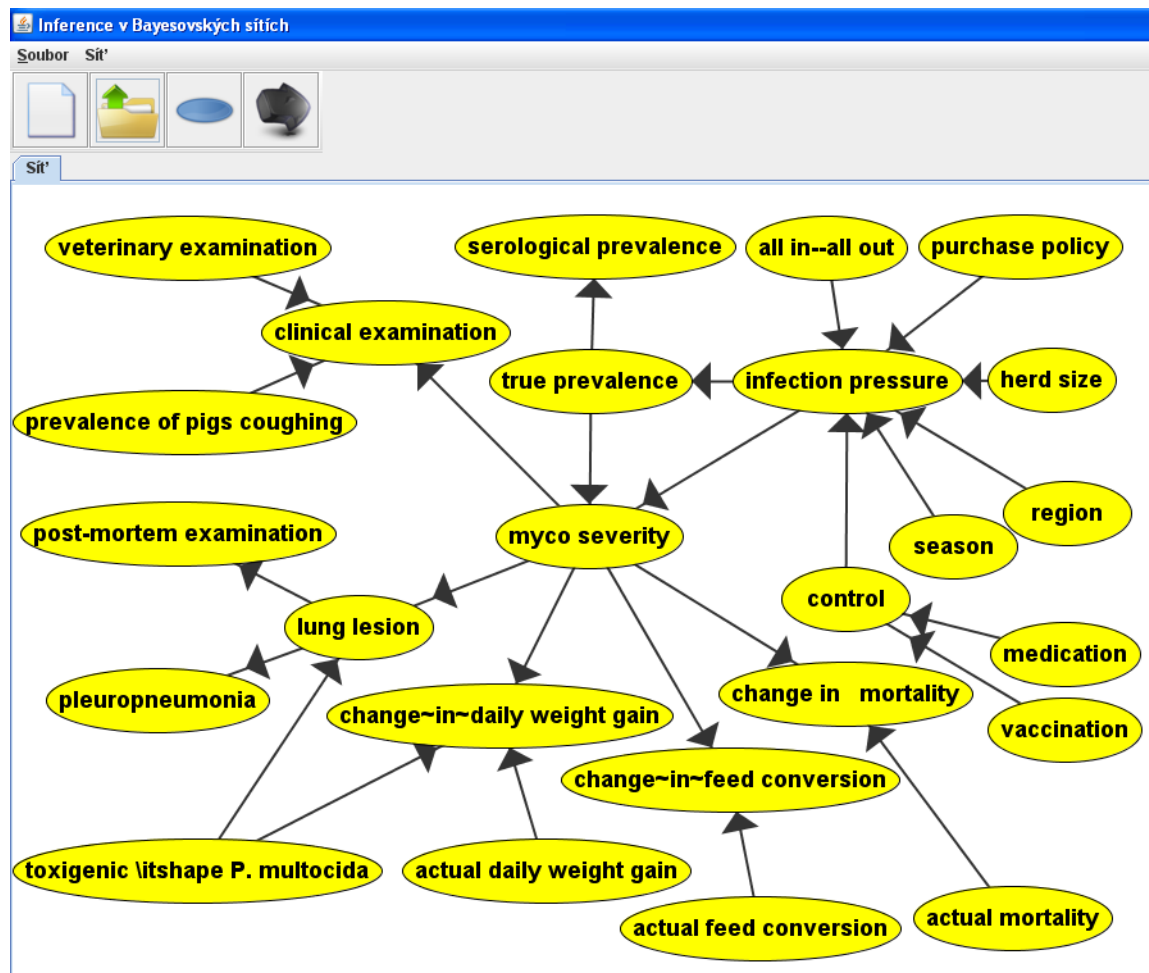
8.2 Infekce Mycoplasma v chovech prasat

Jedná se o biologický model rizikových faktorů ovlivňujících infekci Mycoplasma u jatečných prasat, kde výskyt infekce je založen na veterinárních vyšetřeních. Bakterie Mycoplasma je spojena s prasečím enzootickým zápallem plic, který způsobuje ekonomické ztráty v produkci vepřového masa po celém světě.

Model je určen na základě biologické neurčitosti vážnosti infekce Mycoplasma a vydedukovaných stochastických vlivů na produktivitu chovu. Podmíněné pravděpodobnosti BN jsou založeny na publikované literatuře a názorech expertů.

Pomocí modelu lze spočítat, jaký vliv mají rizikové faktory a dostupné poznatky na vážnost infekce Mycoplasma. Některé faktory jsou jednoduše dostupné, kupříkladu velikost stáda, další potřebují speciální přístupy, mezi tyto patří vzorky sér či pitva [9].

Příslušná BN se nachází v souboru MycoF.xbn (alternativní verze v souboru Myco.xbn).



Obrázek 8.1: BN MycoF zobrazená pomocí implementované aplikace

8.3 Rok 2000 – analýza rizik

Bayesovská síť modeluje problémy roku 2000 ve finančnictví, dopravě, průmyslu a na akciových trzích v USA. Model je vytvořen k predikování vlivu různých scénářů na akciové trhy v USA [3].

Síť se nachází v souboru Year2000.xbn.

Kapitola 9

Experimenty

V této kapitole jsou uvedeny některé z experimentů, které byly provedeny s BN z předchozí kapitoly.

Experimenty byly prováděny na PC s procesorem Intel Core 2 Duo T5670 (2M Cache, 1.80 GHz, 800 MHz FSB) a s pamětí 2x1GB (PC2-5300).

Výsledky experimentů se nacházejí v tabulkách, které obsahují pouze zkratky použitých inferenčních algoritmů. Tytéž zkratky jsou užity i u popisu výsledků experimentů. Význam zkratk je následující:

- JTC – algoritmus stromy spojení,
- Pearl – Kimův a Pearlův algoritmus posílání zpráv,
- LS [iterace] – algoritmus logického vzorkování, [iterace] – číslo udávající počet iterací algoritmu,
- WS [iterace] – algoritmus věrohodnostního váhování, [iterace] – číslo udávající počet iterací algoritmu,
- GS [burnout]_[iterace] – algoritmus Gibbsova vzorkování, [burnout] – číslo udávající počet počátečních iterací, které se nezapočítávají, [iterace] – číslo udávající počet iterací algoritmu.

Každý experiment obsahuje srovnání implementovaných metod z hlediska času a KL-divergence, která je počítána přes všechny dotazované uzly. KL-divergence je počítána mezi výsledky vypočtenými JTC (skutečné rozložení) a výsledky získanými příslušnou metodou.

9.1 Year2000 bez poznatků

Síť obsahuje 8 uzlů, pro které se určují pravděpodobnosti.

V tabulce 9.1 se nachází pravděpodobnosti hodnot všech uzlů vypočtené různými metodami. Jelikož se jedná o vícenásobně spojenou síť, Pearl nefunguje správně. Na uzlech Utilities a US Stock se nefunkčnost projevila nejvíce. U LS 10 000 a LS 100 000 lze pozorovat, že na základě zvyšujícího se počtu vzorků se určené pravděpodobnosti hodnot přibližují k pravděpodobnostem vypočtených JTC.

V tabulce 9.2 se nachází srovnání metod z hlediska času a z hlediska KL-divergence.

Year2000	JTC[%]	Pearl[%]	LS 10 000[%]	LS 100 000[%]	GS 0_10 000[%]
US Banks					
Working	58,600	58,600	58,450	58,629	58,420
Reduced	23,300	23,300	23,760	23,170	23,840
Not Working	18,100	18,100	17,790	18,201	17,740
Telecom					
Working	67,000	67,000	67,350	67,111	67,119
Reduced	21,000	21,000	20,650	20,974	20,910
Not Working	12,000	12,000	12,000	11,915	11,971
Transportation					
Working	65,800	65,725	65,110	65,820	65,768
Moderate	16,740	18,399	17,100	16,634	16,530
Severe	9,730	10,981	9,820	9,827	10,270
Failure	7,730	4,895	7,970	7,719	7,433
Air Travel					
Working	45,000	45,000	44,360	45,078	45,050
Reduced	33,000	33,000	33,350	33,054	33,100
Not Working	22,000	22,000	22,290	21,868	21,850
Electricity					
Working	60,000	60,000	59,450	60,211	60,300
Reduced	30,000	30,000	30,410	29,861	29,660
Not Working	10,000	10,000	10,140	9,928	10,040
Rail					
Working	58,000	58,000	57,560	57,810	57,474
Reduced	23,000	23,000	22,950	23,140	23,180
Not Working	19,000	19,000	19,490	19,050	19,346
Utilities					
Working	70,500	65,610	70,580	70,643	70,423
Moderate	12,100	17,770	11,820	12,044	12,100
Severe	7,850	11,780	7,850	7,811	7,630
Failure	9,550	4,840	9,750	9,502	9,848
US Stocks					
Up	46,168	39,775	45,940	46,199	45,431
Down	39,660	48,123	39,940	39,807	40,560
Crash	14,172	12,101	14,120	13,994	14,009

Tabulka 9.1: V tabulce se nacházejí různými metodami vypočtené pravděpodobnosti hodnot pro všechny uzly v síti.

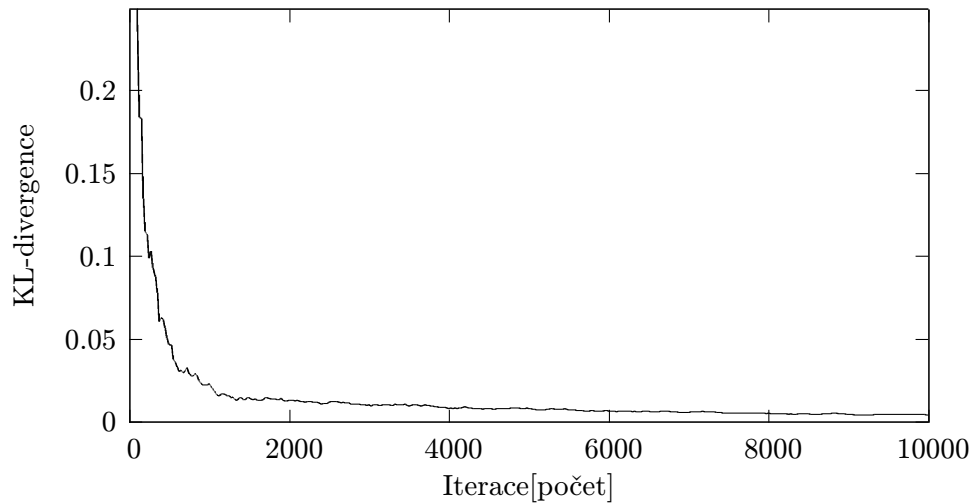
9.2 MycoF bez poznatků

Síť obsahuje 25 uzlů, pro které se určují pravděpodobnosti.

Na obrázcích 9.1 a 9.2 se nacházejí grafy, na kterých je zachycen vývoj KL-divergence v závislosti na počtu iterací algoritmu logického vzorkování respektive Gibbsova vzorkování. V tabulce 9.3 se nachází srovnání metod z hlediska času a z hlediska KL-divergence.

Year2000 bez poznatků		
algoritmus	KL-divergence	čas[ms]
JTC	0	312
Pearl	0,060 18	109
LS 10 000	0,000 52	203
LS 100 000	0,000 08	2 078
GS 0_10 000	0,010 20	1 500

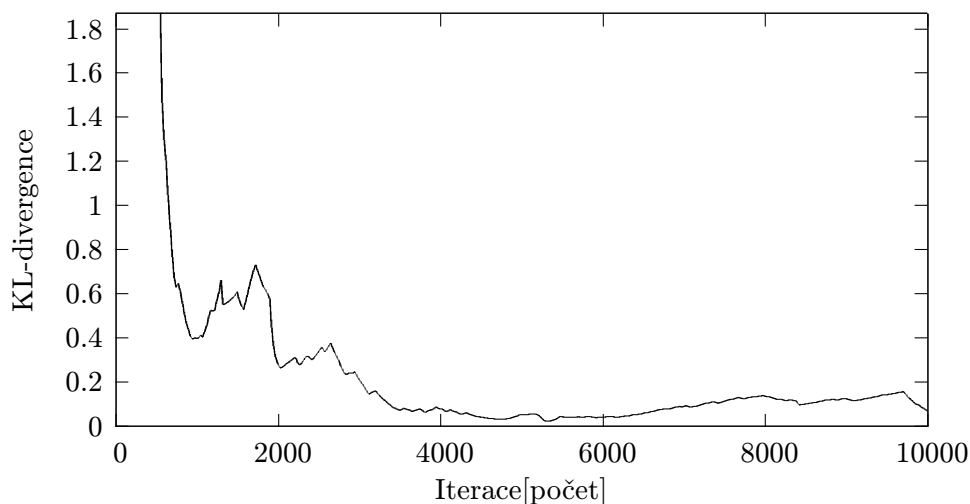
Tabulka 9.2: Tato tabulka zachycuje KL-divergence a dobu výpočtu pravděpodobností příslušnou inferenční metodou.



Obrázek 9.1: Graf zachycuje vývoj KL-divergence v závislosti na počtu iterací algoritmu logického vzorkování pro MycoF bez poznatků.

MycoF bez poznatků		
algoritmus	KL-divergence	čas[ms]
JTC	0	3313
Pearl	0,283 49	94
LS 10 000	0,001 64	781
LS 100 000	0,000 29	6640
GS 0_10 000	0,031 09	7190

Tabulka 9.3: V tabulce se nacházejí různými metodami vypočtené pravděpodobnosti hodnot pro všechny uzly v síti.



Obrázek 9.2: Graf zachycuje vývoj KL-divergence v závislosti na počtu iterací algoritmu Gibbsova vzorkování pro MycoF bez poznatků.

9.3 MycoF s poznatky

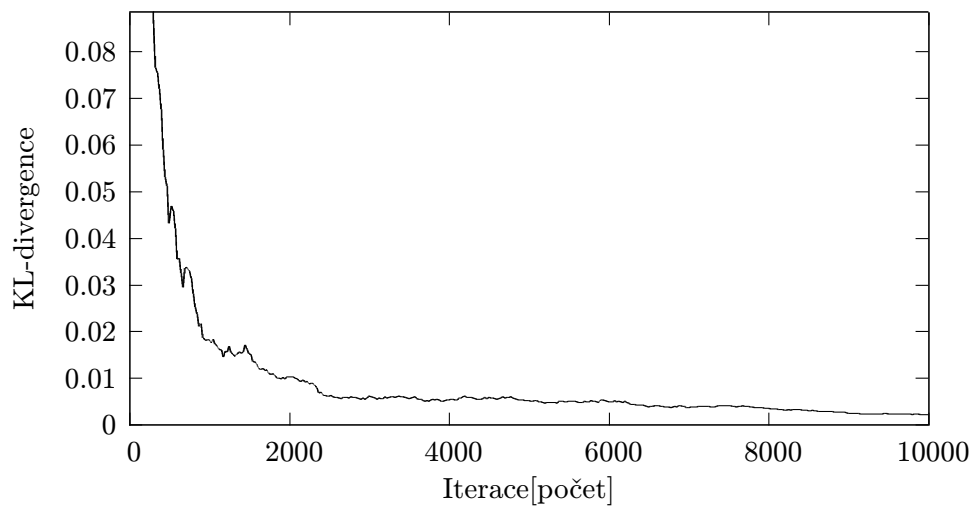
Do sítě MycoF bylo vloženo následujících 6 poznatků: season=Dec, herd size=1001-3000, actual mortality=3%, region=Jutland, vaccination=Yes a medication=Yes. Pravděpodobnosti se počítaly pro všechny nepoznatkové uzly (19 uzlů).

Na obrázcích 9.3 a 9.4 se nacházejí grafy, na kterých je zachycen vývoj KL-divergence v závislosti na počtu iterací algoritmu věrohodnostního váhování respektive Gibbsova vzorkování.

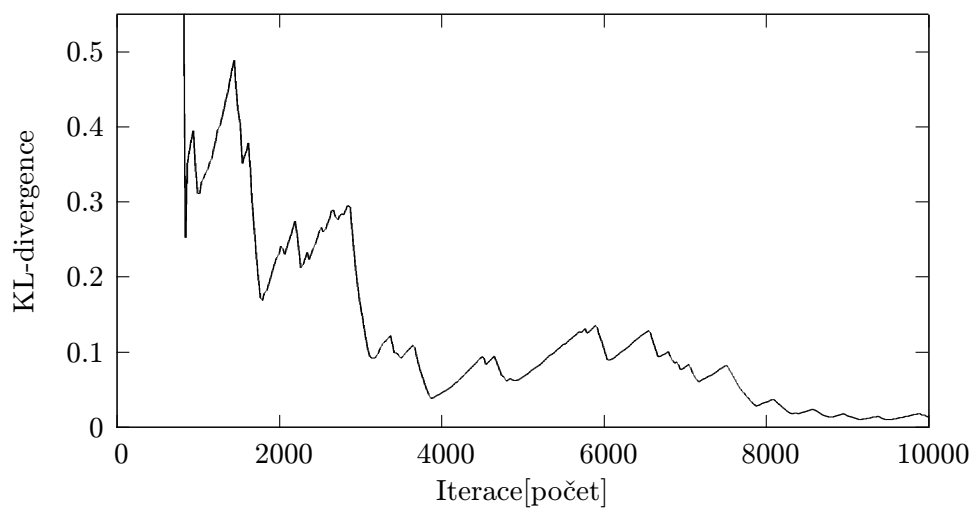
V tabulce 9.4 se nachází srovnání metod z hlediska času a z hlediska KL-divergence. U LS 10 000 se projevuje problém málo pravděpodobných poznatků, většina vygenerovaných vzorků je zahozena, a proto má výpočet touto metodou v porovnání s ostatními metodami mnohonásobně delší dobu výpočtu.

MycoF s poznatky		
algoritmus	KL-divergence	čas[ms]
JTC	0	3 594
Pearl	0,140 02	140
LS 10 000	0,002 31	818 047
LW 10 000	0,002 22	781
LW 100 000	0,000 16	7 094
GS 0_10 000	0.013 40	5 781

Tabulka 9.4: V tabulce se nacházejí různými metodami vypočtené pravděpodobnosti hodnot pro všechny uzly v síti.



Obrázek 9.3: Graf zachycuje vývoj KL-divergence v závislosti na počtu iterací algoritmu věrohodnostního váhování pro MycoF s poznatky.



Obrázek 9.4: Graf zachycuje vývoj KL-divergence v závislosti na počtu iterací algoritmu Gibbsova vzorkování pro MycoF s poznatky.

9.4 ALARM bez poznatků

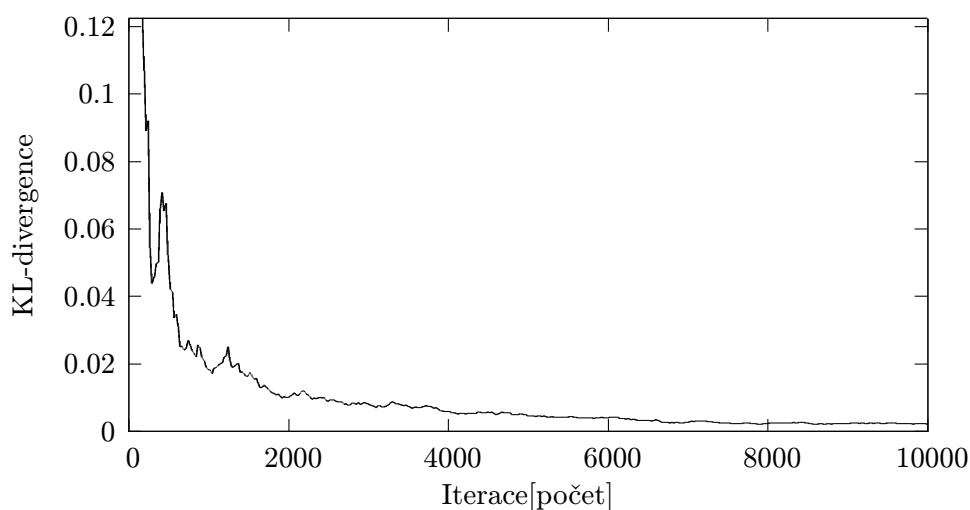
Síť obsahuje 37 uzlů, pro které se určují pravděpodobnosti.

Na obrázcích 9.5 a 9.6 se nacházejí grafy, na kterých je zachycen vývoj KL-divergence v závislosti na počtu iterací algoritmu logického vzorkování respektive Gibbsova vzorkování. V tabulce 9.5 se nachází srovnání metod z hlediska času a z hlediska KL-divergence.

ALARM bez poznatků		
algoritmus	KL-divergence	čas[ms]
JTC	0	349 078
Pearl	0,149 66	109
LS 10 000	0,002 18	1 875
LS 100 000	0,000 33	15 750
GS 0_10 000	0,321 15	18 860

Tabulka 9.5: V tabulce se nacházejí různými metodami vypočtené pravděpodobnosti hodnot pro všechny uzly v síti.

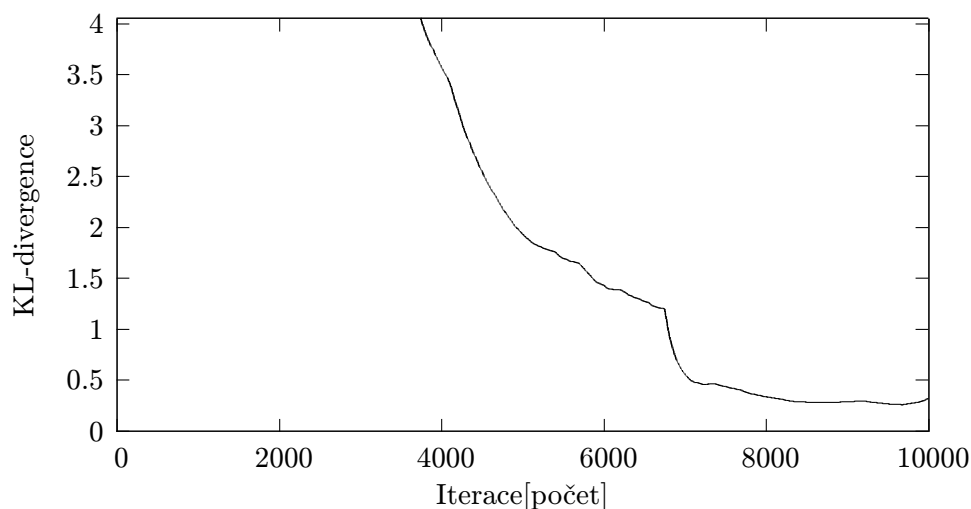
U JTC je velmi dlouhá doba výpočtu. Její hlavní příčinou je velikost klik, které jsou složeny z mnoha uzlů. Díky tomu mají kliky a separátory obrovské tabulky, a proto výpočty s nimi trvají déle. V tomto případě je nejlepší variantou užití metody LS.



Obrázek 9.5: Graf zachycuje vývoj KL-divergence v závislosti na počtu iterací algoritmu logického vzorkování pro Alarm bez poznatků.

9.5 ALARM s poznatky

Do sítě ALARM bylo vloženo následujících 5 poznatků: BP=normal, HR=normal, FiO2=low, Press=high a MINVOLSET=low (poznatky byly čerpány z [2]). Pravděpodobnosti byly počítány pro všechny nepoznatkové uzly (32 uzlů).



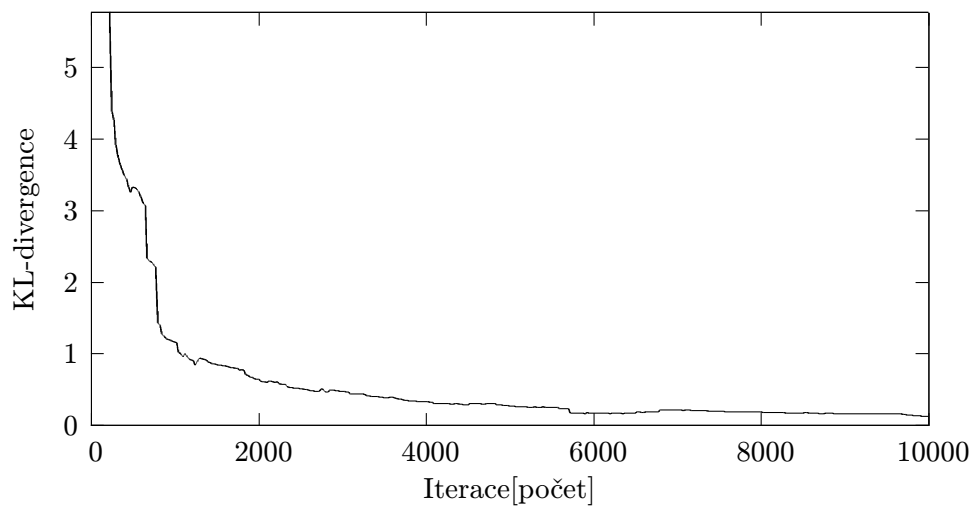
Obrázek 9.6: Graf zachycuje vývoj KL-divergence v závislosti na počtu iterací algoritmu Gibbsova vzorkování pro Alarm bez poznatků.

ALARM s poznatky		
algoritmus	KL-divergence	čas[ms]
JTC	0	339 547
Pearl	nelze	nelze
LS 10 000	časově náročné	časově náročné
LW 10 000	0,125 49	2 125
LW 100 000	0,009 08	16 657
GS 0_10 000	0,207 94	13 078

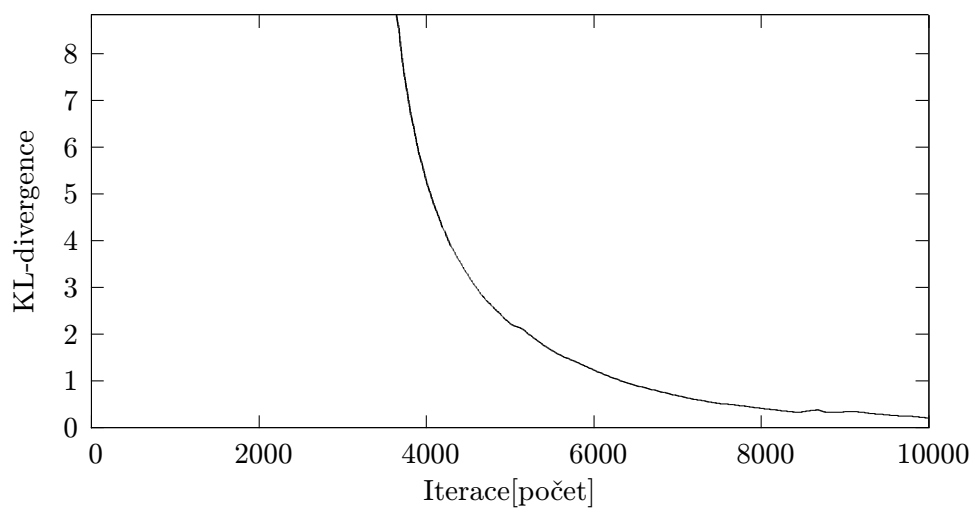
Tabulka 9.6: V tabulce se nacházejí různými metodami vypočtené pravděpodobnosti hodnot pro všechny uzly v síti.

Na obrázcích 9.7 a 9.8 se nacházejí grafy, na kterých je zachycen vývoj KL-divergence v závislosti na počtu iterací algoritmu věrohodnostního váhování respektive Gibbsova vzorkování. V tabulce 9.6 se nachází srovnání metod z hlediska času a z hlediska KL-divergence.

U JTC je velmi dlouhá doba výpočtu. Její hlavní příčinou je velikost klik, které jsou složeny z mnoha uzlů. Díky tomu mají kliky a separátory obrovské tabulky, a proto výpočty s nimi trvají tak dlouho. Kvůli zadaným poznatkům Pearl selhává úplně, protože se zacyklí posílání zpráv mezi jednotlivými uzly. Na základě málo pravděpodobných poznatků nelze výsledky LS 10 000 určit v rozumném čase, proto byl test ukončen. Z porovnání dvou zbylých metod GS a LW vychází lépe použít LW, které má menší KL-divergenci.



Obrázek 9.7: Graf zachycuje vývoj KL-divergence v závislosti na počtu iterací algoritmu věrohodnostního váhování pro **Alarm** s poznatky.



Obrázek 9.8: Graf zachycuje vývoj KL-divergence v závislosti na počtu iterací algoritmu Gibbsova vzorkování pro **Alarm** s poznatky.

Kapitola 10

Závěr

Cílem této práce bylo demonstrovat různé přístupy k inferencím v Bayesovských sítích. V teoretické části práce byly vybrané inferenční metody představeny společně s jejich algoritmem. Každá metoda byla ilustrována na příkladu.

V praktické části práce byly implementovány následující inferenční metody: Kimův a Pearlův algoritmus posílání zpráv, stromy spojení, logické vzorkování, věrohodnostní váhování a Gibbsovo vzorkování. Aplikace kromě výběru inferenční metody umožňuje načítat Bayesovské sítě ve formátu XBN a obsahuje nástroje pro tvorbu vlastní sítě.

S inferenčními metodami byly provedeny experimenty na několika sítích s různými počty skrytých proměnných. V experimentech měl Kimův a Pearlův algoritmus posílání zpráv nejrychlejší dobu výpočtu. Demonstrační BN jsou však vícenásobně spojené, proto výpočty pravděpodobností tímto algoritmem nebyly přesné. Metoda stromy spojení dosahovala dobrých výsledků na sítích s menším počtem uzlů. Rychlost algoritmu stromů spojení byla hlavní měrou ovlivňována velikostí klik. Čím je větší velikost kliky, tím déle výpočet trvá. Velmi tedy záleží na algoritmu, který trianguluje graf a vytváří kliky.

Algoritmus logického vzorkování je použitelný pouze v případě, kdy není poznatek o uzlu nebo jsou poznatky velmi pravděpodobné. Takováto situace ale v reálném světě většinou nenastává. Místo logického vzorkování lze užít věrohodnostního váhování, které pro síť bez poznatků má tentýž algoritmus jak logické vzorkování. Proto při experimentech bez poznatků bylo testováno pouze logické vzorkování. Věrohodnostní váhování dosáhlo dobrých výsledků (malá KL-divergence, rychlost) hlavně na síti ALARM.

Výsledky získané inferencí pomocí Gibbsova vzorkování měly oproti věrohodnostnímu váhování větší KL-divergenci a taktéž výpočet pravděpodobností trval delší dobu.

Experimenty ukázaly, že pro různé struktury sítí se hodí odlišné typy algoritmů. Nelze tedy o některém z implementovaných algoritmů říci, že je nejlepší. Každý se hodí na jinou strukturu sítě.

Další možný vývoj vidím v optimalizaci stávajících metod za účelem snížení času potřebného k výpočtu pravděpodobností. Přidání některé z dalších inferenčních metod, které jsou v [12] pouze zmíněny, a taktéž podpora dalších formátů souborů s BN, které jsou zmíněny v [1], by z aplikace udělaly více komplexní nástroj.

Literatura

- [1] Bayesian Network Repository [online].
<<http://www.cs.huji.ac.il/~galel/Repository/>>, [cit. 2013-01-05].
- [2] Beinlich, I. A.; Suermondt, H.; Chavez, R. M.; aj.: The alarm monitoring system: a case study with two probabilistic inference techniques for belief networks. 1989.
- [3] Bengtsson, H.: Bayesian Networks: a Self-contained Introduction with Implementation Remarks. 1999.
- [4] Davis, W. B.: Graphical Model Theory for Wireless Sensor Networks. *Lawrence Berkeley National Laboratory*, 2002.
- [5] Engelen, R. A. V.: Approximating Bayesian Belief Networks by Arc Removal. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997.
- [6] Korb, K. B.; Nicholson, A. E.: *Bayesian artificial intelligence*. CRC Press, 2011, ISBN 978-1-4398-1591-5.
- [7] Koski, T.; Noble, J.: *Bayesian Networks: An Introduction*. Wiley Series in Probability and Statistics, Wiley, 2009, ISBN 9780470684030.
- [8] Neapolitan, R. E.: *Learning Bayesian Networks*. Prentice Hall, 2003, ISBN 978-0130125347.
- [9] Otto, L.; Kristensen, C. S.: A biological network describing infection with *Mycoplasma hyopneumoniae* in swine herds. *Preventive Veterinary Medicine*, 2004: s. 141 – 161, ISSN 0167-5877, doi:10.1016/j.prevetmed.2004.09.005.
URL
<<http://www.sciencedirect.com/science/article/pii/S0167587704001734>>
- [10] Pourret, O.; Naim, P.; Marcot, B. (editoři): *Bayesian networks: a practical guide to applications*. Wiley, 2008, ISBN 978-0-470-06030-8.
- [11] Rebai, A. (editor): *Bayesian Network*. Sciyo, 2010, ISBN 978-953-307-124-4.
- [12] Russell, S.; Norvig, P.: *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2009, ISBN 978-0136042594.

Příloha A

Obsah CD

Příložené CD obsahuje tyto soubory a složky:

- soubor `technickazprava.pdf` – technická (tato) zpráva v elektronické podobě
- složka `latex` – složka obsahující zdrojový tvar technické zprávy (zdrojové soubory `LaTeXu`, obrázky a `Makefile`)
- soubor `manual.pdf` – uživatelská příručka popisující spuštění a ovládání programu
- složka `src` – složka obsahující zdrojové kódy programu
- složka `doc` – složka obsahující dokumentaci vygenerovanou pomocí `javadoc`
- složka `xbn` – složka obsahující demonstrační aplikace BN
- složka `bin` – složka obsahující spustitelný soubor