Grafické pravděpodobnostní modely – učení z dat

Jiří Kléma

Katedra kybernetiky,
FEL, ČVUT v Praze

http://ida.felk.cvut.cz

Struktura přednášky

- Motivace pro učení
  - znalosti obtížně získatelné, data často po ruce a v dostatečném počtu,
- struktura učících dat
  - obvykle postačuje **frekvenční tabulka**,
  - neúplnost dat stěžuje učení,
- **učení parametrů**
  - snazší úloha,
  - algoritmus MLE (+ EM v případě neúplných dat),
  - demonstrace nároků na data
    - známá síť $\longrightarrow$ učící data $\longrightarrow$ nově naučená síť $\longrightarrow$ srovnání s původní,
- **učení struktury**
  - obtížnější úloha,
  - jak hodnotit kvalitu struktury? věrohodnost, bayesovské kritérium, BIC
  - naivní postup, algoritmy K2 a MCMC,
  - jednoduché příklady.

Učení bayesovských sítí z dat

- jak trénovací data vypadají?
  - množina příkladů D obsahuje M současných pozorování všech veličin na objektu,
  - pro rodinu se psem: $d_m = \{RM_m, PN_m, SS_m, PV_m, PS_m\}$, $m = 1 \dots M$,
  - pro jednoduchost neuvažujeme chybějící hodnoty,
- postačující statistikou (reprezentací) je frekvenční tabulka (hyperkrychle)
  - vyjadřuje počet pozorování s konkrétní konfigurací,
  - 2^5 položek $N(rm_1, pn_1, pv_1, ss_1, ps_1)$, ..., $N(\neg rm_1, \neg pn_1, \neg pv_1, \neg ss_1, \neg ps_1)$,
  - reprezentace blízká rozdělení sdružené psti.

$$d_1 = \{rm_1, \neg pn_1, \neg ss_1, pv_1, \neg ps_1\}$$
$$d_2 = \{\neg rm_2, \neg pn_2, \neg ss_2, pv_2, ps_2\}$$
$$\dots$$
$$d_M = \{\neg rm_M, pn_M, \neg ss_M, pv_M, ps_M\}$$

			rm		¬rm	
			pn	¬pn	pn	¬pn
ss	pv	ps	1	56	0	106
		¬ps	0	24	0	45
	¬pv	ps	0	0	0	4
		¬ps	1	9	0	349
¬ss	pv	ps	0	37	d_M $\boxed{5}$ +1	d_2 $\boxed{71}$ +1
		¬ps	2	d_1 $\boxed{16}$ +1	2	30
	¬pv	ps	0	1	0	2
		¬ps	0	6	0	233

Učení parametrů bayesovské sítě z dat

- struktura sítě je dána, hledáme CPTs v jednotlivých uzlech
- odhad parametrů Θ lze provést maximalizací věrohodnostní funkce
  - pro příklad rodiny se psem

$$L(\Theta : D) = \prod_{m=1}^{M} Pr(d_m : \Theta) = \prod_{m=1}^{M} Pr(RM_m, PN_m, SS_m, PV_m, PS_m : \Theta) =$$
$$= \prod_{m=1}^{M} Pr(RM_m : \Theta) Pr(PN_m : \Theta) Pr(SS_m | RM_m : \Theta) \dots Pr(PS_m | PV_m : \Theta)$$

  - pro obecnou bayesovskou síť

$$L(\Theta : D) = \prod_{m=1}^{M} Pr(d_m : \Theta) = \prod_{m=1}^{M} Pr(P_{1m}, P_{2m}, \dots, P_{nm} : \Theta) =$$
$$= \prod_{j=1}^{n} \prod_{m=1}^{M} Pr(P_j | rodice(P_j) : \Theta_j) = \prod_{j=1}^{n} L_j(\Theta_j : D)$$

- za předpokladu nezávislosti parametrů lze věrohodnost dekomponovat
  - příspěvek každého uzlu $L_j(\Theta_j : D)$ určíme (maximalizujeme) nezávisle.

Učení parametrů bayesovské sítě z dat

- pro každý uzel řešíme optimalizaci: $\widehat{\Theta_j} = \arg\max_{\Theta} L_j(\Theta_j : D)$
- demonstrujme pro uzel RM, kde $\Theta_{RM} = \{Pr(rm)\}$
  - nechť N(rm) je počet příkladů, v nichž platí $RM_j = TRUE$
  - maximalizaci L_{RM} provedeme derivací a srovnáním s 0

$$L_{RM}(\Theta_{RM} : D) = \prod_{m=1}^{M} Pr(RM : \Theta_{RM}) = Pr(rm)^{N(rm)}(1 - Pr(rm))^{M-N(rm)}$$

$$\frac{\partial L_{RM}(Pr(rm) : D)}{\partial \Pr(rm)} = 0 \rightarrow Pr(rm) = \frac{N(rm)}{M}$$

- zobecněný vztah pro odhad parametrů je intuitivně zřejmý

$$\widehat{\theta}_{P_j|rodice(P_j)} = \frac{N(P_j, rodice(P_j))}{N(rodice(P_j))} \approx Pr(P_j|rodice(Pj))$$

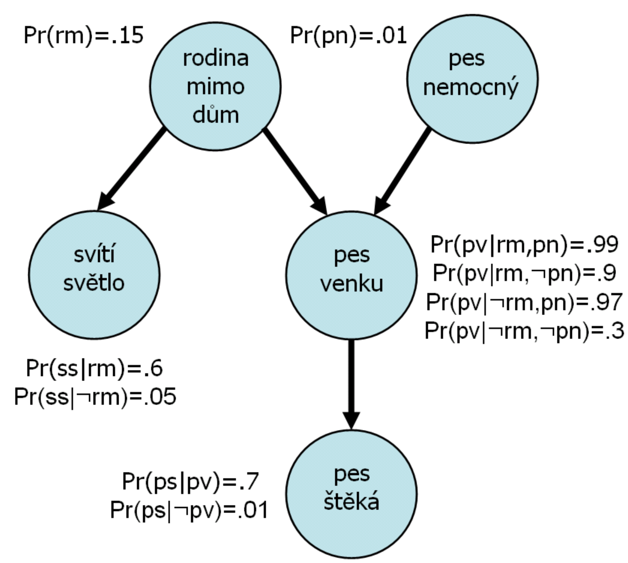
- tento typ odhadu není možný pro řídká nebo neúplná data
  - pro řídká data Dirichlet priors a metoda max aposteriorní psti,
  - pro chybějící data Monte-Carlo vzorkování,
  - nebo EM optimalizace multimodální věrohodnostní funkce.

Učení parametrů z dat – ilustrace nároků na data

1. z existujícího modelu sítě generujeme data
   - model RODINA, různě velká m (počty pozorování),
   - jak se data generují?
     - již dříve Gibbsovo vzorkování,
     - v dané situaci nejsnazší **dopředné vzorkování**, viz dále
2. kvantitativní parametry sítě znáhodníme
   - zůstane pouze struktura sítě,
   - původní CPTs jsou zapomenuty,
3. z dat naučíme parametry znovu
   - úplná pozorování – metoda maximální věrohodnosti (MLE),
   - neúplná pozorování – kombinace MLE odhadu a EM algoritmu,
4. porovnáme původní a naučené CPTs pro různě velká m
   - proč $Pr(rm)$ stanovíme snáze než $Pr(pv|rm, pn)$? viz grafy …

Vzorkování ze známé sítě

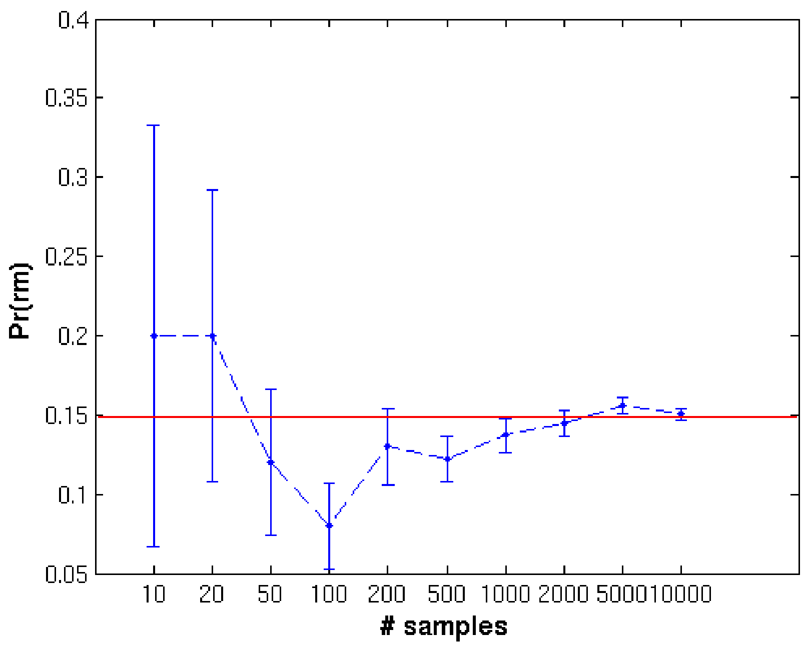
- **dopředné vzorkování** (forward sampling)
  1. uspořádej uzly sítě topologicky
     - hrany musí vést z vrcholu s větším číslem do vrcholu s menším číslem,
     - RODINA: např. RM, PN, SS, PV, PS,
  2. dle uspořádání postupně generuj stavy veličin
     - použij $Pr(P_j|rodice(P_j))$,
     - RODINA: např. $Pr(RM) \rightarrow \neg rm$, $Pr(PN) \rightarrow \neg pn$, $Pr(SS|\neg rm) \rightarrow ss$, $Pr(PV|\neg rm, \neg pn) \rightarrow \neg pv$, $Pr(PS|\neg pv) \rightarrow \neg ps$
  3. krok 2 opakuj pro každý ze vzorků,
- stavy některých veličin mohou být dané (E)
  - pokud je Pr(e) nízká, pak neefektivní,
  - často zamítá částečně nagenerované vzorky,
  - RODINA: znám ps, generuji zbylé veličiny.
- alternativou je mj. Gibbsovo vzorkování.

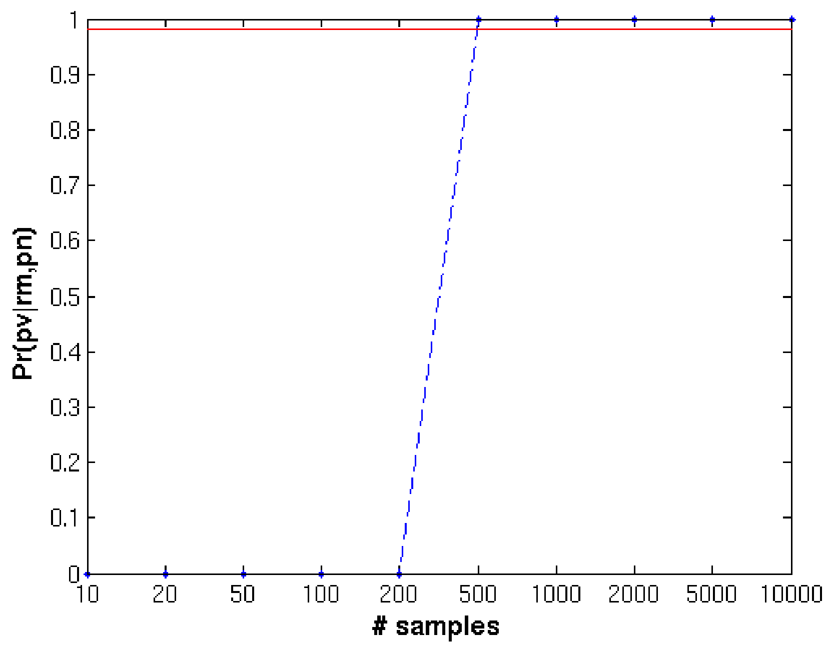


Učení parametrů z dat – úplná pozorování

- Jaká je pst, že rodina bude mimo dům?
  – $Pr(rm) = ?$
- K odhadu lze použít všechny vzorky ...
  – $Pr(rm) = \frac{\sum_{m=1}^{M} \delta(RM^m, rm)}{M}$

- Jaká je podmíněná pst, že je pes venku?
  – $Pr(pv|rm, pn) = ?$
- Podmínka splněna pouze pro 1.5 $^0/_{00}$ vzorků.
  – $Pr(rm) = 0.15$, $Pr(pn) = 0.01$,
  – RM a PN nezávislé veličiny.





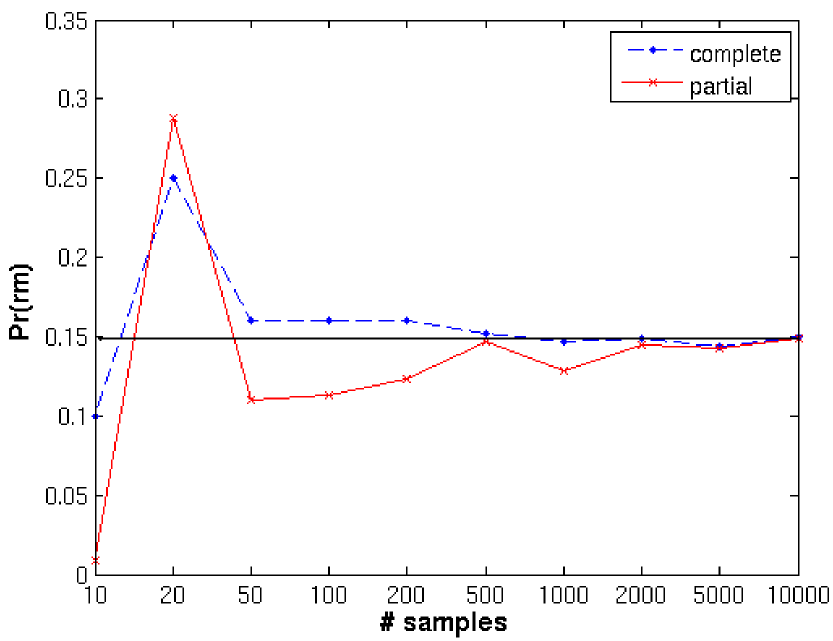
Učení parametrů z dat – neúplná pozorování (50% ztráta)

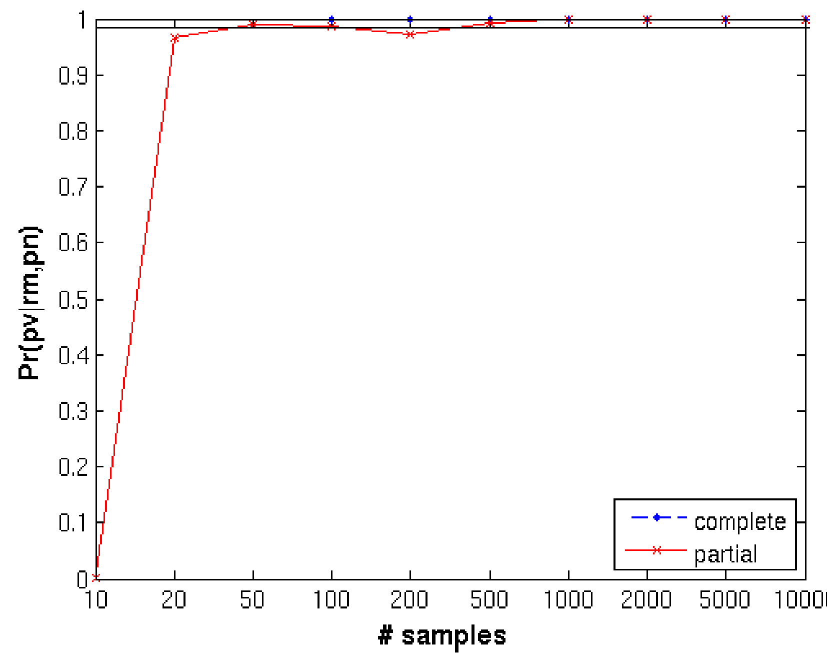
- absence pozorování jsou nezávislé na stavu proměnných
  - jednodušší varianta,
- není vhodné chybějící hodnoty ignorovat
  - ztráta i existujících pozorování,
- MLE odhad kombinujme s **EM** algoritmem:
  1. inicializuj parametry sítě (typicky náhodně),
  2. **E krok:** dopočítej chybějící pozorování z aktuální sítě (inference),
  3. **M krok:** parametry sítě modifikuj dle aktuálních úplných pozorování a kritéria MLE,
  4. opakuj kroky 2 a 3
     (a) po předem daný maximální počet iterací (v experimentu 10),
     (b) dokud nedojde ke konvergenci MLE kritéria (změna log L mezi kroky < 0.001).

Učení parametrů z dat – neúplná pozorování (50% ztráta)

- Jaká je pst, že rodina bude mimo dům?
  - $Pr(rm) = ?$
- Neúplná data = méně informace
  - výrazně delší čas výpočtu,
  - odhad “jen o málo méně přesný”.

- Jaká je podmíněná pst, že je pes venku?
  - $Pr(pv|rm, pn) = ?$
- Neúplná data = méně informace
  - porovnání obtížné.





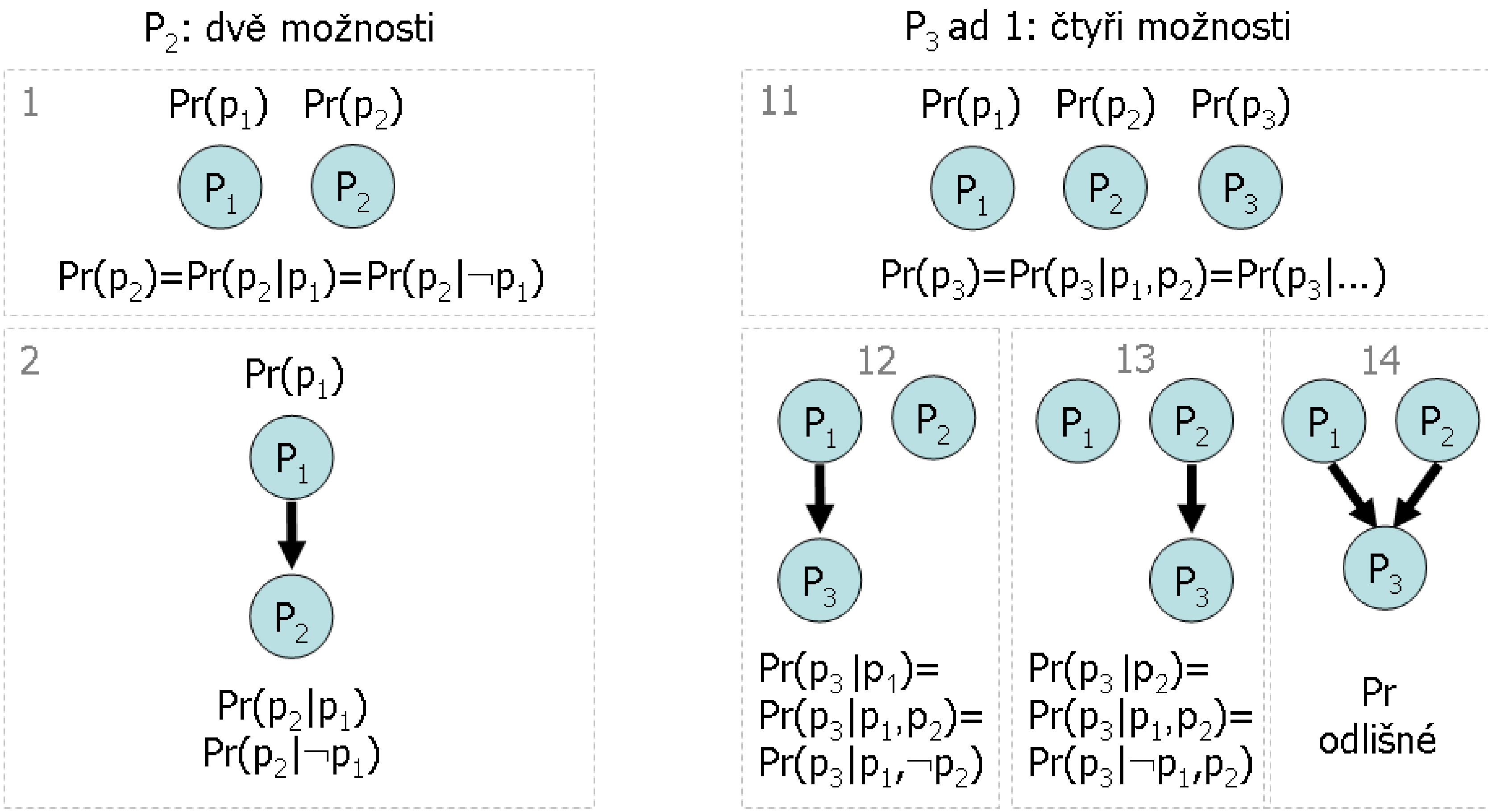
Učení struktury sítě – naivní pohled

- k naučení sítě postačují dva kroky:
  1. definice uspořádání n proměnných,
  2. postupné nalezení podmnožin proměnných, které splňují vztah pro podmíněnou nezávislost $Pr(P_{j+1}|P_1, \ldots, P_j) = Pr(P_{j+1}|rodice(P_{j+1}))$, $rodice(P_{j+1}) \subseteq \{P_1, \ldots, P_j\}$,
- postup lze ilustrovat na příkladu třech proměnných:
  1. zvolíme permutaci π: $\pi(P_1) = 1$, $\pi(P_2) = 2$ a $\pi(P_3) = 3$,
  2. postupně přidáváme uzly do grafu, test podmíněné nezávislosti určuje jak.

Učení struktury sítě – naivní pohled

- k naučení sítě postačují dva kroky:
  1. definice uspořádání n proměnných,
  2. postupné nalezení podmnožin proměnných, které splňují vztah pro podmíněnou nezávislost $Pr(P_{j+1}|P_1, \ldots, P_j) = Pr(P_{j+1}|rodice(P_{j+1}))$, $rodice(P_{j+1}) \subseteq \{P_1, \ldots, P_j\}$,
- postup lze ilustrovat na příkladu třech proměnných:
  1. zvolíme permutaci π: $\pi(P_1) = 1$, $\pi(P_2) = 2$ a $\pi(P_3) = 3$,
  2. postupně přidáváme uzly do grafu, test podmíněné nezávislosti určuje jak.
- není realizovatelné takto jednoduše:
  - pořadí proměnných ovlivňuje výslednou síť – existuje **n!** permutací
    - při nevhodně zvoleném pořadí až úplný graf, nelze ale projít všechna pořadí,
  - testování nezávislosti je také netriviální
    - pro binární veličiny určitě $\mathcal{O}(2^n)$ operací pro jednu permutaci,
    - minimálně musím vyčíslit podmíněné psti $Pr(P_n|P_1, \ldots, P_{n-1})$.

Učení struktury sítě – naivní pohled



Učení struktury sítě – realizace

- založeno na maximalizaci **hodnotící funkce**
  - vyjadřuje jak dobře struktura odpovídá datům,
- přímočará věrohodnostní hodnotící funkce vede na úplně propojenou síť
  - čím více parametrů, tím věrněji modelujeme data,
  - vede na přeučení – zde nevhodné,

$$\log L(G:D) = \log \prod_{m=1}^{M} Pr(d_m : G) = M \sum_{j=1}^{n} (I(P_j : rodice(P_j)^G) - H(P_j))$$

- hodnotící fce často vychází z maximalizace aposteriorní psti (MAP)

$$Pr(G|D) = \frac{Pr(D|G)Pr(G)}{Pr(D)} \rightarrow \log Pr(G|D) = \log Pr(D|G) + \log Pr(G) + c$$

  - na rozdíl od MLE integruje přes všechny parametrizace dané struktury

$$Pr(D|G) = \int Pr(D|G, \Theta_G) \times Pr(\Theta_G|G) d\Theta$$

  - MLE uvažuje jen optimální parametrizaci

$$L(G:D) = Pr(D|G, \widehat{\Theta_G})$$

Učení struktury sítě – realizace

- další častou variantou je Bayesian Information Criterion (BIC)
  - výpočetně jednodušší, heuristické kritérium,
  - jde o analogii MDL principu – nejlepší model je ten, který je kompaktní a dělá málo chyb,
  - uvažujme: q_i ... počet unikátních instanciací rodičů P_i, r_i ... je počet hodnot P_i,
  - v síti je pak: $K = \sum_{i=1}^{n} q_i(r_i - 1)$ nezávislých parametrů,

$$BIC = -\frac{K}{2}\log_2 M + \log_2 L(G:D) = -\frac{K}{2}\log_2 M - M\sum_{i=1}^{n} H(P_i|rodice(P_i)^G)$$

  - první člen: penalta za složitost sítě (K ↑ BIC ↓),
  - druhý člen: věrohodnost sítě
    (síla informačního vztahu mezi uzly a jejich rodiči ↑ $H(|)$ ↓ BIC ↑),
  - jak vyčíslit podmíněnou entropii?
    - N_{ij} ... počet příkladů, kde $rodice(P_i)$ nabývá j-té instanciace hodnot,
    - N_{ijk} ... počet příkladů, kde P_i nabývá k-té hodnoty a $rodice(P_i)$ j-té instanciace hodnot,

$$H(P_i|rodice(P_i)^G) = -\sum_{j=1}^{q_i}\sum_{k=1}^{r_i}\frac{N_{ij}}{M}\frac{N_{ijk}}{N_{ij}}\log_2\frac{N_{ijk}}{N_{ij}} = -\sum_{j=1}^{q_i}\sum_{k=1}^{r_i}\frac{N_{ijk}}{M}\log_2\frac{N_{ijk}}{N_{ij}}$$

Učení struktury sítě – realizace

- nelze aplikovat hodnotící fci na všechny z 2^{n^2} grafů,
- aplikujeme heuristiky a metaheuristiky pro řešení obtížných problémů
  - př. metaheuristiky – **lokální prohledávání**
    - začíná s danou sítí (prázdnou, expertovou, náhodnou),
    - poté sestaví všechny "blízké" sítě, ohodnotí je a přejde na nejlepší,
    - skončí, pokud žádná změna už nevylepší ohodnocení,
  - př. doprovodných heuristik
    - definice "blízké" sítě,
    - řešení problému uváznutí v lokálním minimu nebo na plateaux
      - náhodné restarty, simulované žíhání, TABU search.

Učení struktury sítě – algoritmus K2

- Cooper a Herskovitz (1992), nejblíže naivnímu přístupu,
- výhoda
  - složitost $\mathcal{O}(m, u^2, n^2, r),\ u \leq n \rightarrow \mathcal{O}(m, n^4, r)$
    - m … počet vzorků, n … počet proměnných,
    - r … max počet hodnot proměnné, u … maximální počet rodičů,
- nevýhody
  - na vstupu vyžaduje topologické uspořádání proměnných/uzlů π,
  - hladové prohledávání vede k lokálně optimálnímu řešení.
- pst $Pr(G, D)$ maximalizuje pomocí funkce

$$g(P_i, rodice(P_i)) = \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}!$$

  - q_i … počet unikátních instanciací rodičů P_i, r_i … počet hodnot P_i,
  - N_{ij} … počet příkladů, kde $rodice(P_i)$ nabývá j-té instanciace hodnot,
  - N_{ijk} … počet příkladů, kde P_i nabývá k-té hodnoty a $rodice(P_i)$ j-té instanciace hodnot,

  - kritérium je **separabilní** – lze hodnotit uzel po uzlu.

Učení struktury sítě – algoritmus K2

- algoritmus K2 (π,u,D):

for i=1:n % postupuj dle topologického uspořádání vrcholů π

rodice(P_{π_i})= $\emptyset$ % na počátku je seznam rodičů každého uzlu prázdný

G_{old}=g(P_{π_i},rodice(P_{π_i})) % inicializuj ohodnocení uzlu

while |rodice(P_{π_i})|$\leq$ u % rodičů nesmí být více než zvolené u

$$j^* = \underset{j=1\ldots i-1, P_{\pi_j} \notin rodice(P_{\pi_i})}{\arg\max} g(P_{\pi_i}, rodice(P_{\pi_i}) \cup P_{\pi_j})$$

% $P_{\pi_j^*}$ je rodič maximalizující fci g

% rodič musí mít nižší topologický index, z výpočetních důvodů

% vynecháváme kandidáty, kteří už rodičem uzlu jsou

G_{new}=g(P_{π_i},rodice$(P_{\pi_i}) \cup P_{\pi_j^*}$)

if $G_{new} > G_{old}$ then

$G_{old} = G_{new}$

rodice(P_{π_i})=rodice($P_{\pi_i} \cup P_{\pi_j^*}$)

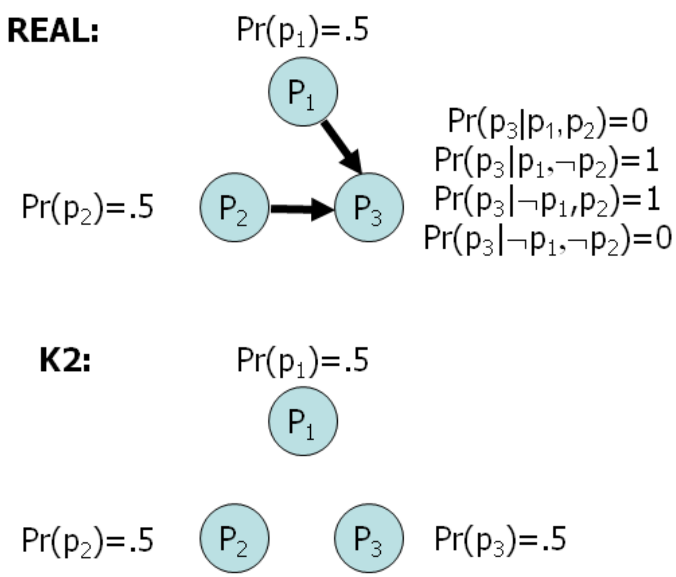
else

STOP % pokud nelze vylepšit, ukonči prohledávání daného uzlu

K2 – lokalita hladového prohledávání

- mějme binární veličiny P_1, P_2, P_3, nechť π={1,2,3} a D jsou dána tabulkou

P_1	P_2	P_3
F	F	F
F	F	F
F	T	T
F	T	T
T	F	T
T	F	T
T	T	F
T	T	F



$g(P_2, \emptyset) = \frac{4!4!}{9!} = \frac{4!}{9\times8\times7\times6\times5} = \frac{1}{630}$

$g(P_2, \{P_1\}) = (\frac{2!2!}{5!})^2 = (\frac{1}{180})^2 = \frac{1}{32400}$

K2: STOP, žádná hrana z P_1 do P_2

$g(P_3, \emptyset) = g(P_2, \emptyset) = \frac{1}{630}$

$g(P_3, \{P_1\}) = (\frac{2!2!}{5!})^2 = (\frac{1}{180})^2 = \frac{1}{32400}$

$g(P_3, \{P_2\}) = g(P_3, \{P_1\})$

K2: STOP, žádná hrana do P_3, přitom

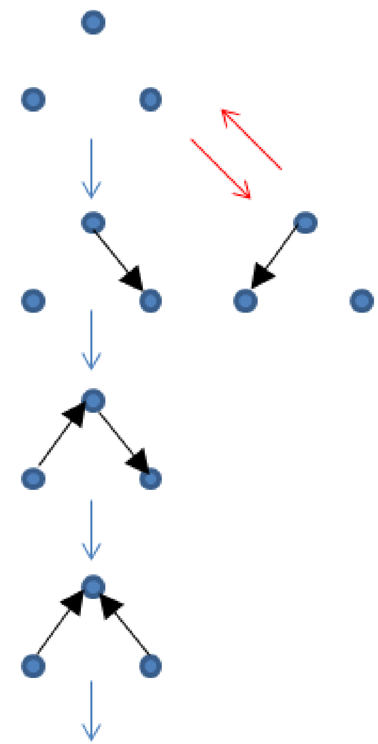
$g(P_3, \{P_1, P_2\}) = (\frac{2!}{3!})^4 = (\frac{1}{3})^4 = \frac{1}{81}$

- drobná vylepšení
  - aplikuj K2 a K2Reverse a vyber lepší řešení
    - K2Reverse vychází z plně propojeného grafu a hrany hladově odebírá,
    - řeší problém výše, není obecným řešením problému,
  - náhodně restartuj algoritmus (různá uspořádání uzlů i počáteční grafy).

Učení struktury sítě – MCMC přístup

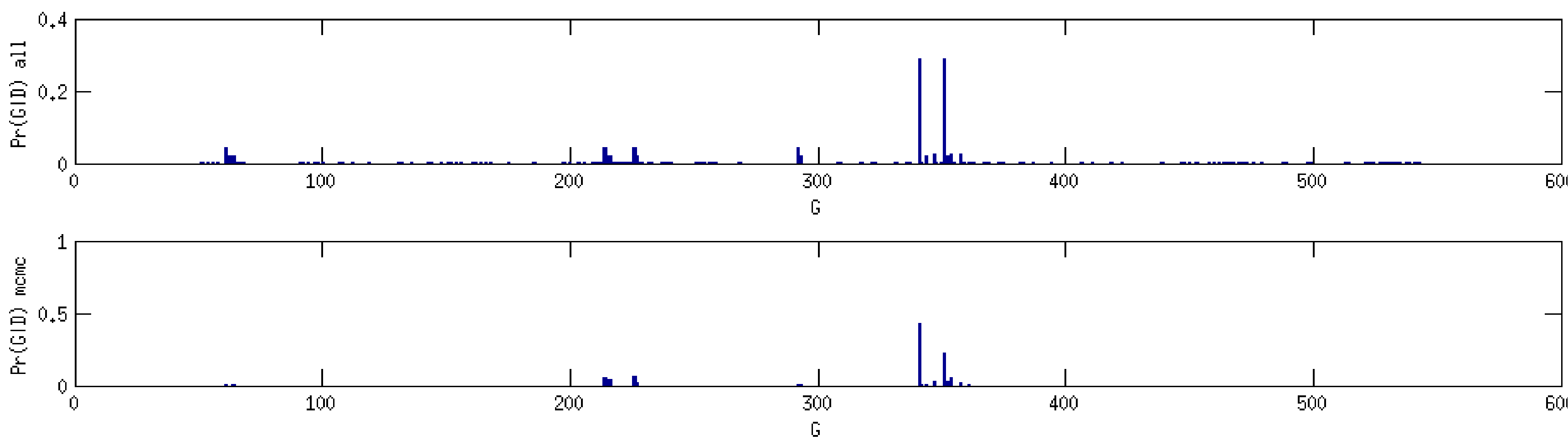
- **MCMC** = Markov chain Monte-Carlo (význam viz slajd pro Gibbsovo vzorkování),
- aplikuje **Metropolis-Hastings** (MH) algoritmus k prohledávání prostoru možných grafů

1. zvol počáteční graf G
   – uživatelsky/informovaně, náhodně, žádné hrany,
2. ohodnoť graf $P(G)$
   – použij vzorky, kritérium například BIC nebo MAP,
3. generuj "souseda" S daného grafu
   – přidej/odeber hranu, změň orientaci hrany,
   – dodržuj omezení na acykličnost grafu,
   – pst přechodu z G do S je dána fcí $Q(G, S)$,
4. ohodnoť souseda $P(S)$,
5. přijmi nebo odmítni přechod na nový graf
   – generuj α z U(0,1) (uniformní rozdělení),
   – if $\alpha < \frac{P(S)Q(G,S)}{P(G)Q(S,G)}$ then akceptuj přechod $G \rightarrow S$,
6. opakuj kroky 3–5, ukončení po daném počtu iterací či dle konvergence.



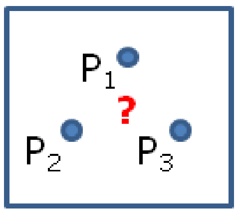
Učení struktury sítě – MCMC přístup

- z četnosti výskytu grafů usuzujeme na jejich aposteriorní pravděpodobnost
  - u náhodného startu ignorujeme počátek posloupnosti,
- výstup lze využít jako bodový i **bayesovský** odhad
  - bodový odhad – uvažuji pouze síť s nejvyšším skóre,
  - bayesovský odhad – uvažuji všechny odhady pravděpodobností přes všechny sítě,
- konvergence
  - teoreticky konverguje v čase polynomiálním s velikostí prostoru grafů,
  - prakticky obtížné pro problémy s více než 10 uzly.

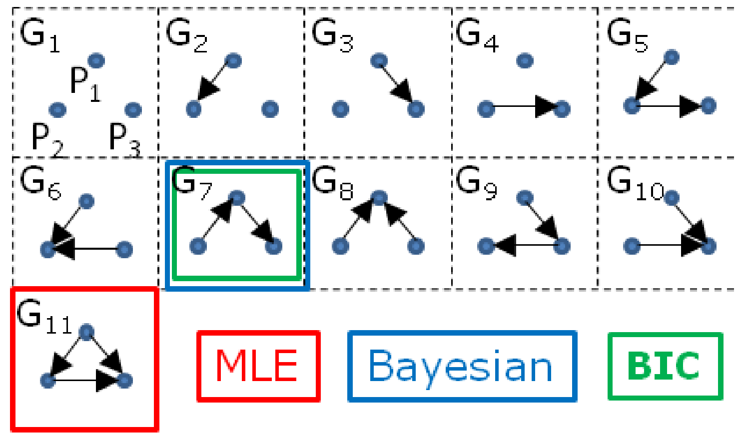


Učení struktury z dat – příklad pro 3DAG

- zvol model se 3 uzly, generuj 16 příkladů,
- úplně prohledej všech 11 reprezentantů tříd markovské ekvivalence,
- použij 3 různá kritéria (věrohodnost, bayesovké a BIC) k identifikaci optimálního modelu.



		p1	¬p1
p2	p3	0	2
	¬p3	3	4
¬p2	p3	0	0
	¬p3	6	1



- hodnotím G_1 postupně pomocí tří kritérií:
  - věrohodnost: nejprve MLE optimální parametry $Pr(p_1) = Pr(p_2) = \frac{9}{16}$, $Pr(p_3) = \frac{1}{8}$

$$\ln L(G_1 : D) = \sum_{m=1}^{16} Pr(d_m : G_1) =$$

$$= 2\ln\left(\frac{7}{16}\frac{9}{16}\frac{1}{8}\right) + 3\ln\left(\frac{9}{16}\frac{9}{16}\frac{7}{8}\right) + 10\ln\left(\frac{9}{16}\frac{7}{16}\frac{7}{8}\right) + \ln\left(\frac{7}{16}\frac{7}{16}\frac{7}{8}\right) = -27.96$$

Učení struktury z dat – příklad pro 3DAG

— ekvivalentní výpočet věrohodnosti přes podmíněnou entropii

$$\ln L(G_1 : D) = -M \sum_{i=1}^{3} H(P_i | rodice(P_i)^{G_1}) =$$

$$= -16\Big[-2\Big(\frac{9}{16}\ln\frac{9}{16} + \frac{7}{16}\ln\frac{7}{16}\Big) - \Big(\frac{1}{8}\ln\frac{1}{8} + \frac{7}{8}\ln\frac{7}{8}\Big)\Big] = -27.96$$

— BIC určíme odečtením penalty za složitost sítě od věrohodnosti

$$BIC(G_1 : D) = -\frac{K}{2}\ln M + \ln L(G_1 : D) = -\frac{3}{2}\ln 16 - 27.96 = -32.12$$

— bayesovské kritérium

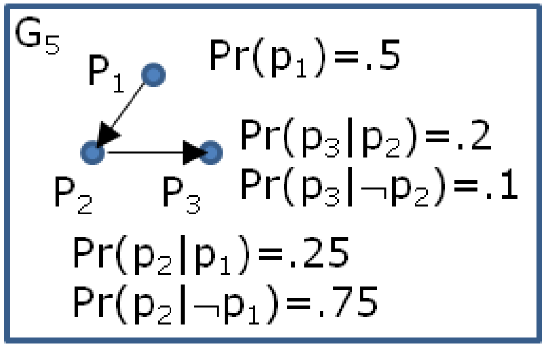
$$\ln Pr(D|G_1) = \ln \prod_{i=1}^{3} g(P_i, rodice(P_i)^{G_1}) = \sum_{i=1}^{3}\sum_{j=1}^{q_i}\sum_{k=1}^{r_i} \ln \frac{(r_i-1)!}{(N_{ij}+r_i-1)!} N_{ijk}! =$$

$$= 2(-\ln 17! + \ln 9! + \ln 7!) - \ln 17! + \ln 2! + \ln 14! = -31.98$$

Přirozený logaritmus je použit pro shodu s Matlab BN Toolboxem.

Změna základu logaritmu neovlivní pořadí modelů při jejich hodnocení.

Učení struktury z dat – příklad pro 3DAG



G_1	G_2	G_3	G_4	G_5
-27.96	-25.59	-26.12	-26.70	-24.33
-31.98	-30.56	-31.78	-32.32	-30.92
-32.12	-31.14	-31.67	-32.25	-31.26
G_6	G_7	G_8	G_9	G_{10}
-25.32	-23.75	-24.64	-24.86	-25.75
-31.03	**-30.36**	-30.56	-31.33	-33.04
-33.64	**-30.68**	-32.96	-31.79	-34.07
G_{11}				
-23.38				
-31.62				
-33.08				

MLE Bayesian **BIC**

- ani jedno z kritérií neodhalilo původní graf
  - u MLE došlo podle očekávání k přeučení,
  - BIC a Bayes trpí nedostatkem příkladů.

Shrnutí

- učení parametrů bayesovské sítě
  - relativně snadné – MLE nebo bayesovsky,
    - oba typy odhadů shodné pro velký počet vzorků, liší se pro počty malé,
  - obtížnější v případě malého počtu dat či chybějících hodnot,
    - nutná počáteční znalost resp. iterativní upřesňování parametry $\leftrightarrow$ hodnoty,
- učení struktury bayesovské sítě
  - je nutné zvolit hodnotící kritérium pro kvalitu struktury
    - věrohodnost, resp. log věrohodnost (v přednášce dva způsoby vyjádření),
    - MAP bayesovské kritérium, BIC odvozené od věrohodnosti,
    - další kritéria – mj. lokální testy podmíněné nezávislosti,
  - prostor kandidátských struktur je obrovský
    - hodnotící kritéria nelze aplikovat na všechny struktury,
    - nelze použít ani naivní postup,
    - K2 – hladové, lokálně optimální prohledávání,
    - MCMC – stochastické prohledávání podobné simulovanému žíhání.

Doporučené doplňky – zdroje přednášky

- Murphy: **A Brief Introduction to Graphical Models and Bayesian Networks.**
  - praktický přehled od autora BN toolboxu,
  - http://www.cs.ubc.ca/~murphyk/Bayes/bayes.html#learn,
- Friedman, Koller: **Learning Bayesian Networks from Data.**
  - Neural Information Processing Systems conference tutorial, slajdy,
  - http://www.cs.huji.ac.il/~nirf/Nips01-Tutorial/,
- Cooper, Herskovits: **A Bayesian Method for the Induction of P.Networks from Data.**
  - teorie + algoritmus K2,
  - www.genetics.ucla.edu/labs/sabatti/Stat180/bayesNet.pdf,
- Heckerman: **A Tutorial on Learning With Bayesian Networks.**
  - teoretický článek, “jde to číst”
  - research.microsoft.com/apps/pubs/default.aspx?id=69588,
- Buntine: **Operations for Learning with Graphical Models.**
  - obecný, úplný a rozsáhlý popis,
  - http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.52.696&rep=rep1&type=pdf.