Učení, adaptace, ladění

Málokdy se podaří napoprvé zkonstruovat složitější síť správně.

Máme-li k dispozici dostatečně velké soubory dat (database of cases), můžeme učením stanovit strukturu sítě i potenciály (tj. pravděpodobnosti příslušné uzlům)

Dostanem-li dodatečně další data, můžeme adaptací modifikovat síť tak, aby lépe reprezentovala nové zkušenosti.

Ladění je proces modifikace parametrů (tj. vstupních pravděpodobností) tak, aby byl splněn nějaký explicitní požadavek na aposteriorní pravděpodobnosti.

(Je to podobné jako trénování neuronové sítě)

Tyto procesy jsou dosud předmětem aktivního výzkumu a vývoje, takže zatím nejsou standardní součástí Bayesovského softwaru kromě jiného i proto, že i když je třeba metoda známá, tak má příliš velkou časovou a prostorovou složitost.

Učení - příklad

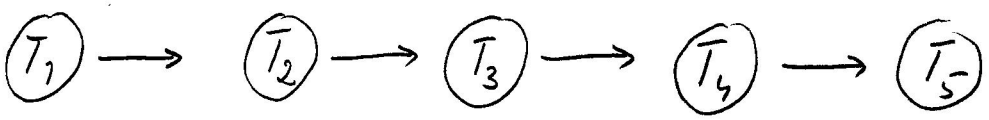
Vytváříme síť pro slova = posloupnosti symbolů a, b délky 5

Z frekvenční analýzy je známo, že:

první 2 písmena	poslední 3 písmena: aaa	aab	aba	abb	baa	bab	bba	bbb
a a	0.017	0.021	0.019	0.019	0.045	0.068	0.045	0.068
a b	0.033	0.04	0.037	0.038	0.011	0.016	0.010	0.015
b a	0.011	0.014	0.010	0.01	0.031	0.046	0.031	0.045
b b	0.050	0.060	0.056	0.057	0.016	0.023	0.015	0.023

Hledáme model pro proměnné - znaky T_1, T_2, T_3, T_4, T_5

Jako první nás asi napadne:



označíme ho M_{simp}

V tomto modelu počítáme podle řetězcového pravidla

$$P^*(\bar{T}_1, \bar{T}_2, \bar{T}_3, \bar{T}_4, \bar{T}_5) = P(\bar{T}_1)\,P(\bar{T}_2|\bar{T}_1)\,P(\bar{T}_3|\bar{T}_2)\,P(\bar{T}_4|\bar{T}_3)\,P(\bar{T}_5|\bar{T}_4)$$

Zřejmě $P(\bar{T}_1 = a) = P(\bar{T}_1 = b) = \frac{1}{2}$

$$P(\bar{T}_2 | \bar{T}_1) = \frac{P(\bar{T}_2, \bar{T}_1)}{P(\bar{T}_1)}$$

např. $P(\bar{T}_2 = a, \bar{T}_1 = a) = P(aa) =$ součet pravděpodobností v řádku aa ve frekvenční tabulce $\doteq 0.3$

$$P(\bar{T}_2 = a | \bar{T}_1 = a) = \frac{0.3}{0.5} = 0.6$$

podobně $P(\bar{T}_2 = a | \bar{T}_1 = b) = 0.4$

$$P(\bar{T}_2 = b | \bar{T}_1 = a) = 0.4$$

$$P(\bar{T}_2 = b | \bar{T}_1 = b) = 0.6$$

Analogicky se napočítají další pravděpodobnosti:

$$P(\bar{T}_2 = a) = 0.5$$

(= součet 1. a 3. řádku frekvenční tabulky)

$P(\bar{T}_3 = a, \bar{T}_2 = a)$ = součet prvků 1.–4. sloupce pro řádky 1. a 3. = 0.121

$$P(\bar{T}_3 = a | \bar{T}_2 = a) = \frac{0.121}{0.5} \doteq 0.24$$

$P(T_3 = a \mid T_2 = b) = 0.74$

$P(T_3 = b \mid T_2 = a) = 0.76$

$P(T_3 = b \mid T_2 = b) = 0.26$

atd.

Z toho nám vyjde tabulka pstí P^*:

první 2 znaky	poslední 3 znaky: aaa	aab	aba	abb	baa	bab	bba	bbb
aa	0.016	0.023	0.018	0.021	0.044	0.067	0.05	0.061
ab	0.03	0.044	0.033	0.041	0.011	0.015	0.012	0.014
ba	0.01	0.016	0.012	0.014	0.029	0.045	0.033	0.041
bb	0.044	0.067	0.052	0.061	0.016	0.023	0.017	0.021

Na pohled je vidět, že tabulky nejsou totožné.

Model M_{Simp} můžeme přijmout, pokud se moc neliší - k tomu ale potřebujeme nějakou míru vzdálenosti – je více možností, ale vezmeme obyčejnou Euklidovskou vzdálenost mezi vektory x a y : $dist_Q(x, y) = \sum_i (x_i - y_i)^2$

(kvadratickou, bez odmocňování)

dostaneme $dist_Q(P, P^*) = 0.000337$

Je to hodně nebo málo? Potřebujeme ještě nějakou hranici pro rozhodování.

Řekněme, že pro žádné 2 pravděpodobnosti nechceme rozdíl větší než 0.004,

Pak rozumná hranice pro rozdíl tabulek je

$$32 \times 0.004^2 = 0.000512,$$

takže model M_{Simp} je z tohoto hlediska akceptovatelný.

Máme ale i jiná kritéria:

síť nesmí být moc velká, aby se s ní dalo rozumně rychle (a v rozumném prostoru počítat).

Definujeme velikost sítě M jako $Size(M) = \sum_{A \in U} Sp(A)$,

kde $U = \{A_1, \ldots, A_m\}$ jsou všechny proměnné sítě,

$pa(A)$ jsou rodiče A

$Sp(A)$ je počet prvků (vstupů) v tabulce pro $P(A|pa(A))$

Máme $Size(M_{Simp}) = 2 + 4 + 4 + 4 + 4 = 18.$

Definujeme míru akceptace jako

$$Acc(P, M^*) = Size(M^*) + k \cdot dist(P, P^*)$$

kde P^* je pravděpodobnost příslušného modelu M^*

k je kladné reálné číslo

Např. pro $k = 10000$ dostaneme pro model M_{Simp}

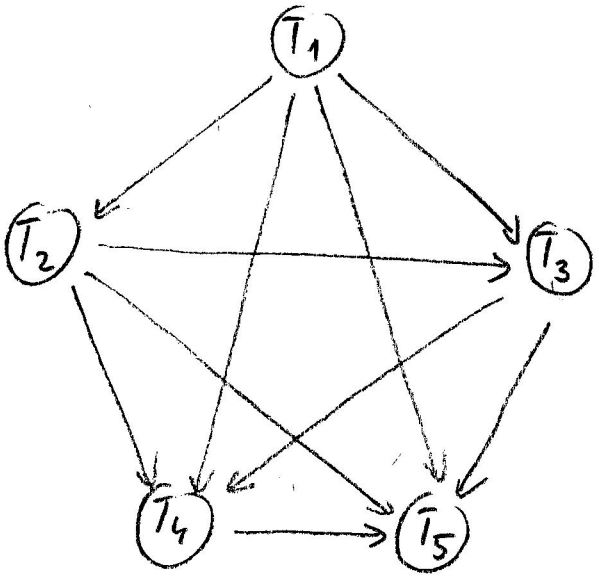
$$Acc(M_{Simp}) = 21.37$$

Úkolem teď bude najít model, který minimalizuje Acc při hranici vzdálenosti 0.0005 (s naší konstantou $k = 10000$).

V principu bychom měli vyhodnotit všechny možné orientované grafy s vrcholy $\bar{T}_1, \bar{T}_2, \bar{T}_3, \bar{T}_4, \bar{T}_5$. Je přirozené se omezit na případ, kdy hrana z $\bar{T}_i$ do $\bar{T}_j$ vede jen když $i < j$.

Začneme s maximálním kokovým grafem a postupně budeme ubírat hrany, ale nesmíme se přitom dostat přes hranici vzdálenosti.

Model M_{Max} :



Označíme $P(word \mid M_{Max})$ pravděpodobnostní rozdělení určené modelem M_{Max}. Platí $P(word \mid M_{Max}) = P(word)$

Dk: $P(word) = P(T_1, T_2, T_3, T_4, T_5) =$

$= P(T_5 \mid T_1, T_2, T_3, T_4)\, P(T_1, T_2, T_3, T_4) =$

$= P(T_5 \mid T_1, T_2, T_3, T_4)\, P(T_4 \mid T_1, T_2, T_3)\, P(T_1, T_2, T_3) =$

$= P(T_5 \mid T_1, T_2, T_3, T_4)\, P(T_4 \mid T_1, T_2, T_3)\, P(T_3 \mid T_1, T_2) \times$
$\times P(T_1, T_2)$

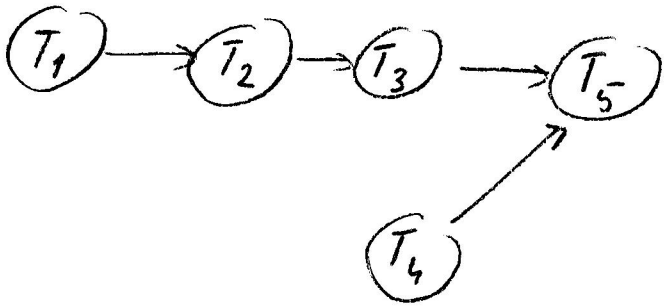
$= P(T_5 \mid T_1, T_2, T_3, T_4)\, P(T_4 \mid T_1, T_2, T_3)\, P(T_3 \mid T_1, T_2)\, P(T_2 \mid T_1)$
$\times P(T_1)$

$= P(word \mid M_{Max})$

tedy distance tohoto modelu od skutečnosti je nulová

$$Acc(M_{Max}) = Size(M_{Max}) = 2 + 4 + 8 + 16 + 32 = 62$$

Výsledkem postupného ubírání hran bude graf M_{Min}:



$$P^*(T_1, T_2, T_3, T_4, T_5) = P(T_1)P(T_4)P(T_2|T_1)P(T_3|T_2)P(T_5|T_3, T_4)$$

musíme dopočítat %

Výsledná tabulka pravděpodobností bude

první 2 znaky	poslední 3 znaky							
	aaa	aab	aba	abb	baa	bab	bba	bbb
aa	0.017	0.021	0.019	0.019	0.045	0.068	0.045	0.068
ab	0.034	0.04	0.037	0.038	0.01	0.015	0.01	0.015
ba	0.011	0.014	0.010	0.010	0.031	0.045	0.03	0.045
bb	0.051	0.062	0.057	0.057	0.015	0.023	0.015	0.023

Pouze na podtržených místech se liší od správné frekvenční tabulky

7.

$$P(\bar{T}_5 = a \mid \bar{T}_3 = a, \bar{T}_4 = a) = \frac{P(\bar{T}_5 = a, \bar{T}_4 = a, \bar{T}_3 = a)}{P(\bar{T}_4 = a, \bar{T}_3 = a)} =$$

$$= \frac{\text{součet sloupce aaa frekvenční tabulky}}{\text{součet sloupce aaa s aab}} = \frac{0.111}{0.111 + 0.135} =$$

$$= \frac{0.111}{0.245} = 0,453 \doteq 0,45$$

$P(\bar{T}_5 \mid \bar{T}_4, \bar{T}_3)$:

$\bar{T}_4$	$\bar{T}_3$ a	$\bar{T}_3$ b
a	(0.45, 0.55)	(0.4, 0.6)
b	(0.5, 0.5)	(0.4, 0.6)

$(\bar{T}_5 = a, \bar{T}_5 = b)$

Vypočteme $dist_Q(P, P^*) = \sum_i (x_i - y_i)^2 =$

$= 0.001^2 + 0.001^2 + 0.001^2 + 0.001^2 + 0.001^2 +$

$+ 0.001^2 + 0.002^2 + 0.001^2 + 0.001^2 =$

$= 0.000008 + 0.000004 = 0.000012$

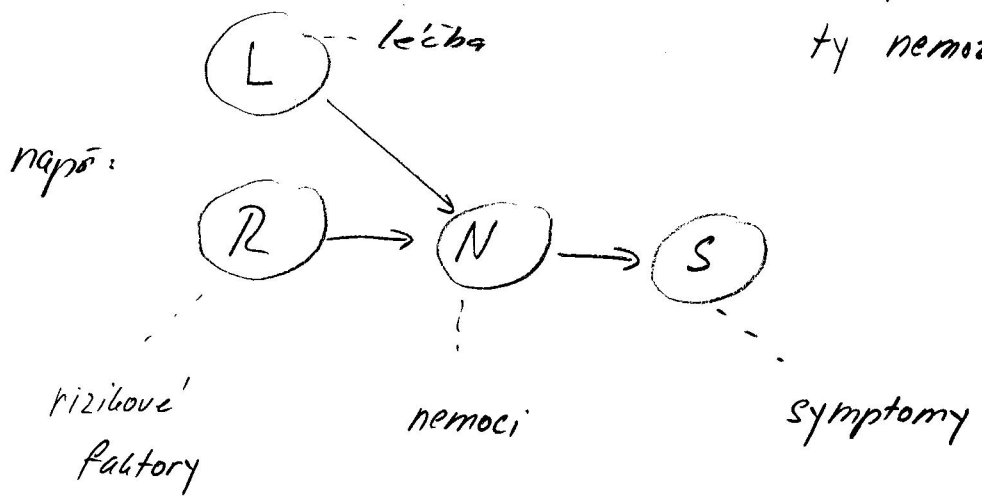
$Acc(M_{Min}) = Size(M_{Min}) + 10000 \times dist_Q(P, P^*) =$

$= 2 + 2 + 4 + 4 + 8 = 20 + 0.12 = 20,12$

Takže vybereme model M_{Min}.

Další možnosti:

určit nějakou hierarchii pro clustery proměnných, která nám pomůže určit možné hrany (nebo spíš vyloučit ty nemožné)



napří:

rozhodně od proměnných z kategorie symptomů nepovedou hrany do rizikových faktorů atd.

No a tohle je vlastně Data-Mining – dát databázi strukturu Bayesovské sítě a tím určit závislosti mezi veličinami. Má to řadu praktických problémů - velkou časovou a prostorovou náročnost, chybějící hodnoty (pozorování) v databázích apod.

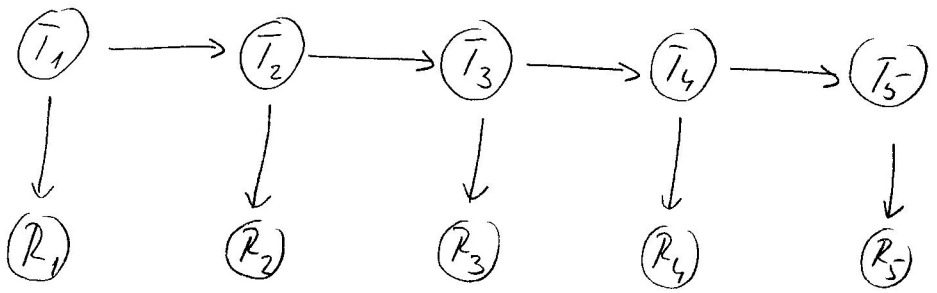
Adaptace

Model, který jsme vybrali v procesu učení, nemusí platit jednou provždy. Během práce se sítí dostáváme nová data a chceme, aby se systém učil i z nich.

Adaptace se týká jak podmíněných pravděpodobností, tak struktury sítě.

Adaptace pravděpodobností – příklad: Přenos symbolů

Máme síť:



I_i jsou vyslané znaky, nabývají hodnot a, b

R_i jsou přijaté znaky, –"– a, b, c

(c slouží jako oddělovač slov – v našem případě vysílané slovo je pětipísmenné, proto se c mezi I_i nevyskytuje, ale v důsledku zkreslení přenosovým kanálem mezi přijatými znaky být může)

Podmíněné pravděpodobnosti $P(R_i|T_i)$ byly odhadnuty na základě zkušeností se statistickými vlastnostmi zkreslení při přenosu:

	$T=a$	$T=b$
$R=a$	0.8	0.15
$R=b$	0.1	0.8
$R=c$	0.1	0.05

Pravděpodobnosti $P(T_{i+1}|T_i)$ se získají z tabulky frekvenční analýzy – to už jsme měli.

Teď si představme, že při každém desátém slově můžeme nějakým způsobem zkontrolovat, co bylo skutečně odesláno.

Tabulku $P(R|T)$ můžeme interpretovat tak, že při nějakém počátečním výběru velikosti 100 byly pozorovány četnosti

	$T=a$	$T=b$
$R=a$	80	15
$R=b$	10	80
$R=c$	10	5

Když nyní získáme další pozorování, dejme tomu jev $(R=a|T=a)$, pak se četnost tohoto jevu zvýší na 81, ostatní četnosti zůstanou stejné a velikost výběru bude 101.

Tedy můžeme modifikovat $P(R=a|T=a) = \frac{81}{101} = 0.8019$

$P(R=b|T=a) = \frac{10}{101} = 0.099$

$P(R=c|T=a) = \frac{10}{101} = 0.099$

(říká se tomu "fractional updating")

Tento postup se může po každém dalším pozorování opakovat – má to ale jednu vadu. Velikost souboru se nám bude rychle

zvětšovat, takže přepočtené pravděpodobnosti budou rychle konvergovat k nějakým číslům, která se dál už téměř nebudou měnit (budou to limity), a tudíž téměř nebudou reagovat na další přidané pozorování.

Aby se tomu zabránilo, udržuje se tzv. efektivní velikost souboru, ze kterého se odhady počítají, označíme ji s^*.

Měla by mít tu vlastnost, že podíl $\frac{s^*-1}{s^*} = q$ bude stále konstantní. (q se nazývá fading factor – faktor slábnutí?)

Zvolíme-li třeba $s^* = 100$, pak $q = 0.99$

Tímto faktorem se budou násobit všechny četnosti i velikost souboru po každém novém pozorování, tj.

velikost souboru $s := sq + 1 = 100 \times 0.99 + 1 = 100$

(efektivní velikost souboru se nemění)

četnost $(R=a \mid T=a) = 80 \times q + 1 = 80 \times 0.99 + 1 = 80.2$

$(R=b \mid T=a) = 10 \times q = 9.9$

$(R=c \mid T=a) = 10 \times q = 9.9$

pravděpodobnosti pak dostaneme vydělením 100

Předpokládejme, že jsme získali jedno zkontrolované slovo a víme, že odesláno bylo baaba a přijato bylo baaca.

Můžeme tedy 3x modifikovat $P(R|a)$ a 2x $P(R|b)$.

četnost $(R=a \mid T=a) = ((80 \times 0.99 + 1) \times 0.99 + 1) \times 0.99 + 1 = 80.6$

$(R=b \mid T=a) = ((10 \times 0.99) \times 0.99) \times 0.99 = 9.7$

$(R=c \mid T=a) = 10 \times 0.99^3 = 9.7$

$(R=a \mid T=b) = (15 \times 0.99) \times 0.99 = 14.7$

$(R=b \mid T=b) = (80 \times 0.99 + 1) \times 0.99 = 79.4$

$(R=c \mid T=b) = (5 \times 0.99) \times 0.99 + 1 = 5.9$

Pravděpodobnosti:

	$T=a$	$T=b$
$R=a$	0.806	0.147
$R=b$	0.097	0.794
$R=c$	0.097	0.059

Takže třeba $P(a|a)$ se mírně zvětšila, protože všechny 3 znaky a se přenesly správně,

$P(b|b)$ se zmenšila, protože ze 2 znaků b se jeden přenesl špatně.

Pravděpodobnosti $P(T_{i+1} | T_i)$ se také dají adaptovat a mohou se k tomu použít všechna přijatá slova, nejen ta kontrolní.

—

Jiná možnost, jak adaptovat pravděpodobnosti, je např. formálně přidat k síti jednu (nebo několik málo) rodičovskou proměnnou, jejíž rozdělení bude reagovat na nová data, a bude se propagovat ke všem proměnným, které jsou na ní závislé.

Adaptace struktury

Bohužel neexistuje žádná silná metoda pro inkrementální adaptaci struktury.

Důvod: změny struktury jsou skokové a nedají se zakládat na jednotlivém případu (třeba na jednom přijatém písmenu), musí být založeny na akumulovaných datech

Jsou 2 způsoby:

akumulovat nová data a čas od času spustit nové učení na celém souboru

ponechat starou síť a vytvořit (naučit) novou na nových datech

Mezi starou a novou sítí se pak buď přepíná, nebo se použijí nějaké váhy.

Mějme modely M_1 a M_2 a váhy w_1, w_2. Když přijde nějaké nové pozorování (evidence), budeme mít v modelu M_1 pravděpodobnosti $P_1(A|e)$, v modelu M_2 $P_2(A|e)$, a zkombinujeme je na $P(A|e) = w_1 P_1(A|e) + w_2 P_2(A|e)$

Interpretace: w_i je pravděpodobnost modelu M_i

Váhy se vždy/po novém pozorování přepočítají:

$$w_i := P(M_i|e) = \frac{P(e|M_i)P(M_i)}{P(e)} = \frac{w_i P_i(e)}{\sum_j w_j P_j(e)}$$

Problém je, že musíme mít uloženo více modelů, což je paměťově náročné, a vytváření každého nového je zase časově náročné

Jak se dá ušetřit čas:

Předpokládá se, že nová síť bude asi dost podobná té staré, může se lišit třeba jen v několika málo hranách.

Vytipujeme si tyto hrany a označíme je za sporné, tj. v síti někdy být mohou, někdy ne.

Vytváření nové struktury pak spočívá vlastně jen v přidávání nebo ubírání těchto sporných hran.

A ještě naivní přístup -

doufáme, že pečlivá adaptace pravděpodobností vykompenzuje i mírně nekorektní strukturu sítě

k adaptaci struktury se přistupuje až když je nevyhnutelná

Ladění

To je složitější, takže ho zatím vynecháme.

V podstatě jde o to vyladit síť tak, aby pravděpodobnosti P(A|e) vyhovovaly nějakým předem daným požadavkům.