Todennäköisyyslaskennan perusaksioomia käyttäen näemme helposti että

$$P(B=1 \mid C=1, D=1) = \frac{P(d^+ \mid b^+, c^+)P(c^+ \mid b^+)P(b^+)}{P(c^+, d^+)},$$

missä arvoasetuksesta $X=1$ on käytetty lyhennettä x^+ (asetuksesta $X=0$ käytetään jatkossa lyhennettä x^-). Bayes-verkkoihin liittyvän ehdollisen riippumattomuuden määritelmän perustella muuttuja D on ehdollisesti riippumaton muuttujasta C, annettuna muuttujan B arvo, joten saamme

$$P(b^+ \mid c^+, d^+) = \frac{P(d^+ \mid b^+)P(c^+ \mid b^+)P(b^+)}{P(c^+, d^+)}.$$

Toisaalta, koska

$$P(b^+ \mid c^+, d^+) = \frac{P(b^+ \mid c^+, d^+)}{P(b^+ \mid c^+, d^+) + P(b^- \mid c^+, d^+)},$$

saamme että

$$\begin{aligned} P(b^+ \mid c^+, d^+) &= \frac{\frac{P(d^+|b^+)P(c^+|b^+)P(b^+)}{P(c^+,d^+)}}{\frac{P(d^+|b^+)P(c^+|b^+)P(b^+)}{P(c^+,d^+)} + \frac{P(d^+|b^-)P(c^+|b^-)P(b^-)}{P(c^+,d^+)}} \\ &= \frac{P(d^+ \mid b^+)P(c^+ \mid b^+)P(b^+)}{P(d^+ \mid b^+)P(c^+ \mid b^+)P(b^+) + P(d^+ \mid b^-)P(c^+ \mid b^-)P(b^-)}. \end{aligned}$$

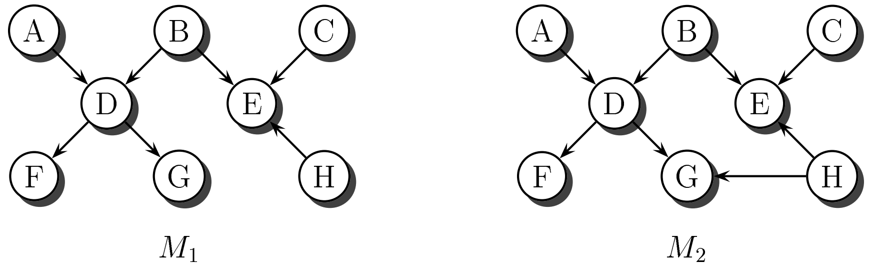
Näemme että koska nimittäjissä esiintyvä termi $P(c^+, d^+)$ ei riipu muuttujan B arvosta (eli on B:n ehdollisen todennäköisyysjakauman suhteen vakio), sen arvoa ei tarvitse laskea, vaan se kumoutuu pois! Haluamamme ehdollisen todennäköisyyden saamiseksi riittää siis laskea kaksi tuloa,

$$\begin{gathered} P(d^+ \mid b^+)P(c^+ \mid b^+)P(b^+) \text{ ja} \\ P(d^+ \mid b^-)P(c^+ \mid b^-)P(b^-), \end{gathered}$$

jotka molemmat sisältävät vain Bayes-verkkomalliin talletettuja parametreja.

Bayes-verkoissa tapahtuvan probabilistisen päättelyn havainnollistamiseksi ajatellaan kuvan 3.5 Bayes-verkkoa tietokoneohjelma-arkkitehtuurina, jossa kukin muuttuja X vastaa moduulia, jonka tehtävänä on laskea ehdollinen todennäköisyysjakauma $P(X \mid \mathbf{S}_2 = \mathbf{s}_2)$, ja oletetaan että Bayes-verkon parametrit on talletettu siten, että kukin moduuli X näkee muotoa $P(X \mid F(X))$ olevat parametrit. Kuten edellä näimme, yllä esitetyssä tapauksessa moduuli B tarvitsee seuraavat luvut voidakseen määrätä ehdollisen todennäköisyysjakauman $P(b^+ \mid c^+, d^+)$:

$$P(d^+ \mid b^+), P(c^+ \mid b^+), P(b^+), P(d^+ \mid b^-), P(c^+ \mid b^-), \text{ ja} P(b^-).$$



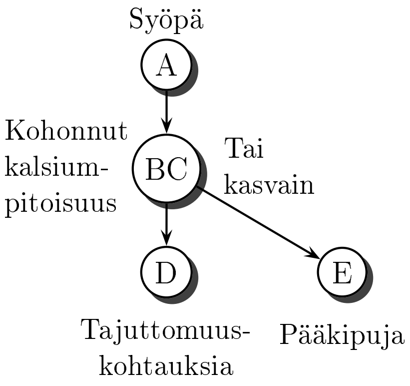
Kuva 3.6: Yksipolkuinen (M_1) ja monipolkuinen (M_2) Bayes-verkkostruktuuri.

Luvut $P(b^+)$ ja $P(b^-)$ on talletettu paikallisesti moduuliin B Bayes-verkkomallin parametreina. Koska $P(c^+ \mid b^+) = P(c^+ \mid a^-, b^+) + P(c^+ \mid a^+, b^+)$, moduli C voi laskea tämän tarvittavan luvun, samoin kuin luvun $P(c^+ \mid b^-)$, nopeasti ja tehokkaasti käyttäen vain paikallisesti talletettua informaatiota (moduuliin C talletettuja Bayes-verkon parametreja). Moduuli D voi vastaavasti laskea arvot $P(d^+ \mid b^+)$ ja $P(d^+ \mid b^-)$. Laskeakseen todennäköisyyden $P(b^+ \mid c^+, d^+)$ modulin B riittää siis lähettää tarvittavien lukujen laskupyyntö moduleille C ja D, ja kertoa saadut vastaukset edellä esitetyllä tavalla halutun ehdollisen todennäköisyysjakauman tuottamiseksi.

Edellä kuvattua viestinvälitysarkkitehtuuriin perustuva menetelmää ehdollisten todennäköisyyksien laskemiseksi voidaan soveltaa kaikissa *yksipolkuisissa (singly connected)* Bayes-verkoissa: yksipolkuisissa verkoissa ei minkään kahden solmun välillä ole kuin yksi suuntaamaton polku, missä suuntaamaton polku tarkoittaa verkon kaaria seuraavaa reittiä, joka voi kulkea suunnattuja kaaria kumpaan suuntaan tahansa. Kuvassa 3.6 yksipolkuinen Bayes-verkkostruktuuri M_1 muutetaan *monipolkuiseksi (multi-connected)* yhdistämällä solmut H ja G, jolloin syntyy kaksi (suuntaamatonta) polkua esimerkiksi solmujen B ja G välille.

Yksipolkuisissa Bayes-verkoissa voidaan ehdollisten todennäköisyysjakaumien laskeminen suorittaa erittäin tehokkaasti, lineaarisessa ajassa Bayes-verkon parametrien lukumäärään nähden. Algoritmin yksityiskohtainen kuvaus löytyy esimerkiksi alan perusoppikirjoista [105, 97]. Maksimitodennäköisyyskonfiguraation etsiminen voidaan niinikään suorittaa samankaltaisella algoritmilla.

Edellä kuvatunlaiseen viestinvälitystekniikkaan perustuvat tehokkaat probabilistiset päättelyalgoritmit perustuvat siihen tosiseikkaan, että yksipolkuisissa verkoissa kukin solmu osittaa verkon kahteen erilliseen osaan, jotka voidaan käsitellä erikseen, ja sama pätee rekursiivisesti jokaiselle verkon solmulle. Monipolkuisissa verkoissa menetelmä ei valitetavasti toimi. Itse asiassa voidaan teoreettisesti osoittaa, että muotoa $P(X = 1 \mid \mathbf{S}_2 = \mathbf{s}_2)$ olevien



$P(A = 1)$	$=$	0.20
$P(BC = 00 \mid A = 1)$	$=$	$0.20 \times 0.80 = 0.16$
$P(BC = 01 \mid A = 1)$	$=$	$0.20 \times 0.20 = 0.04$
$P(BC = 10 \mid A = 1)$	$=$	$0.80 \times 0.80 = 0.64$
$P(BC = 11 \mid A = 1)$	$=$	$0.80 \times 0.20 = 0.16$
$P(BC = 00 \mid A = 0)$	$=$	$0.80 \times 0.95 = 0.76$
$P(BC = 01 \mid A = 0)$	$=$	$0.80 \times 0.05 = 0.40$
$P(BC = 10 \mid A = 0)$	$=$	$0.20 \times 0.95 = 0.19$
$P(BC = 11 \mid A = 0)$	$=$	$0.20 \times 0.05 = 0.01$
$P(D = 1 \mid BC = 11)$	$=$	0.80
$P(D = 1 \mid BC = 01)$	$=$	0.80
$P(D = 1 \mid BC = 10)$	$=$	0.80
$P(D = 1 \mid BC = 00)$	$=$	0.05
$P(E = 1 \mid BC = 01)$	$=$	0.80
$P(E = 1 \mid BC = 11)$	$=$	0.80
$P(E = 1 \mid BC = 00)$	$=$	0.60
$P(E = 1 \mid BC = 10)$	$=$	0.60

Kuva 3.7: Kuvan 3.3 Bayes-verkko muutettuna yksipolkuiseksi verkoksi.

ehdollisten todennäköisyyksien laskeminen [27] tai edes niiden approksimointi [29, 115], ja myös maksimitodennäköisyystilojen etsiminen [124], on erittäin vaikea ongelma monipolkuisten verkkojen tapauksessa: jos näille ongelmille olisi aina polynomisessa ajassa toimiva ratkaisumenetelmä, menetelmää voitaisiin käyttää ratkaisemaan kaikki ns. NP-täydelliset päätösongelmat polynomisessa ajassa, minkä ei yleisesti uskota olevan mahdollista. On kuitenkin syytä muistaa näiden teoreettisten tulosten käsittelevän pahimman tapauksen (worst-case) skenaariota: yhdenkin eksponentiaalisen ajan vaativan tapauksen löytäminen riittää, vaikka kaikki muut tapaukset voitaisiin käsitellä polynomisessa ajassa.

Yleisin lähestymistapa probabilistisen päättelyn toteuttamiseksi monipolkuisissa Bayes-verkoissa on yrittää muuttaa monipolkuinen verkko yksipolkuiseksi, jolloin päättely on tehokasta kuten edellä näimme. Yleisimmin käytetty transformointitekniikka on muuttujien *ryhmittely (clustering)*. Tarkastellaan esimerkiksi kuvassa 3.3 esitettyä Bayes-verkkoa. Ryhmittelemällä muuttujat uudelleen siten että ne muodostavat neljä ryhmää $\{A\}$, $\{B, C\}$, $\{D\}$, ja $\{E\}$ näemme, että jos sulautamme muuttujat B ja C uudeksi muuttujaksi BC, syntyvä Bayes-verkko on yksipolkuinen (katso kuva 3.7). Koska muuttujilla B ja C oli kummallakin 2 mahdollista arvoa, on yhdistetyllä muuttujalla BC 4 mahdollista arvoa; merkitään näitä arvoja symboleilla 00,01,10 ja 11. Syntyneeseen uuteen verkkoon liittyvät todennäköisyydet on helppo laskea alkuperäisen verkon parametriarvoista: muotoa $P(D \mid BC)$ olevat todennäköisyydet vastaavat suoraan muotoa $P(D \mid B, C)$ olevia todennäköisyyksiä, ja ehdollisen riippumattomuuden määritelmän mukaan näemme että $P(E \mid BC) = P(E \mid C)$ ja $P(BC \mid A) = P(B \mid A)P(C \mid A)$.

Edellä kuvatussa yksinkertaisessa tapauksessa oli helppo nähdä kuinka annettu monipolkuinen verkko voitiin muuttaa yksipolkuiseksi yhdistämällä kaksi muuttujaa, mutta monimutkaisemmissa verkoissa tehtävä on huomattavasti vaikeampi eivätkä ratkaisut ole aina yhtä yksinkertaisia. Käytännön kannalta tärkeä kysymys on voidaanko tämä transformaatioprosessi automatisoida — onko mahdollista kehittää algoritmi joka tuottaa mistä tahansa annetusta monipolkuisesta Bayes-verkosta yksipolkuisen Bayes-verkon polynomisessa ajassa? Vastaus on hieman yllättäen kyllä: tällaisia algoritmeja on kehitetty itse asiassa lukuisia (katso esim. [105, 97, 60, 18]). Useimmat esitetyistä menetelmistä perustuvat kaksivaiheiseen algoritmiin, jossa ensimmäisessä vaiheessa annettu Bayes-verkko *moralisoidaan*[2] yhdistämällä solmut joilla on yhteinen seuraaja, ja poistamalla kaarien suunnat. Menetelmän toisessa vaiheessa syntynyt suuntaamaton verkko *kolmioidaan (triangulation)* lisäämällä verkkoon suuntaamattomia kaaria kunnes kaikissa verkossa sijaitsevien yli kolmen kaaren pituisissa silmukoissa on joitakin kahta silmukan solmua yhdistävä kaari. Jos kolmioidussa verkossa esiintyvien maksimaalisten *klikkien* (suurimpien sellaisten verkosta löytyvien solmujoukkojen joissa kaikki solmut on kytketty toisiinsa) solmut yhdistetään yhdeksi muuttujaksi, niin syntyvien yhdistettyjen muuttujien avulla on mahdollista luoda alkuperäisen monipolkuisen Bayes-verkon kanssa ekvivalentti yksipolkuinen verkko. Syntyvää yksipolkuista verkkoa kutsutaan *liittymäpuuksi (junction tree)*. Nimitys johtuu siitä, että liittymäpuun solmusta voi olla kaari toiseen solmuun vain silloin jos solmuja vastaavien klikkien leikkaus ei ole tyhjä. Liittymäpuiden konstruoimisessa käytettävä algoritmi on melko monimutkainen, eikä sen esittäminen tämän raportin yhteydessä ole mielekästä. Algoritmin ksityiskohdat selviävät mm. lähteistä [105, 97, 60, 18].

Liittymäpuihin perustuvat menetelmät antavat siis algoritmin probabilistisen päättelyn suorittamiseksi monipolkuisissa Bayes-verkoissa. Koska syntyvä liittymäpuu on puu, siis yksipolkuinen Bayes-verkko, voidaan probabilistista päättelyä suorittaa syntyneessä verkossa lineaarisessa ajassa verkon parametrien lukumäärään nähden. Tässä yhteydessä on kuitenkin huomattava että syntyvän liittymäpuun parametrien lukumäärä ei ole sama kuin alkuperäisen monipolkuisen Bayes-verkon parametrien lukumäärä, vaan se voi olla paljon suurempi, pahimmassa tapauksessa eksponentiaalisen suuri alkuperäiseen lukumäärään verrattuna. Voidaan itse asiassa osoittaa [142] että sellaisen kolmioinnin löytäminen joka tuottaa minimaalisen määrän liittymäpuuparametreja on NP-täydellinen ongelma. Alkuperäisen ongelman vaikeutta ei siis voi paeta tiedon esitystapaa muuttamalla (yhdistämällä muuttujia), mikä on tietenkin muutenkin itsestään selvää. Becker ja Geiger [7] ovat esittä-

[2]Moralisointi-termi (moralization) johtuu siitä että prosessissa yhdistetään solmut (vanhemmat) joilla on yhteinen seuraaja (lapsi).

neet "hajota-ja-hallitse"-periaatteeseen perustuvan algoritmin, jonka tuottama kolmiointi on aina vähintään tietyn vakioetäisyyden päässä optimaalisesta. Kjærulff [66] on tutkinut stokastisten optimointmenetelmien soveltamista tämän ongelman ratkaisemisessa.

Käytännön sovelluksista saadut kokemukset osoittavat että liittymäpuihin perustuvat algoritmit toimivat yleensä varsin hyvin — useimmat luvussa 4.2 kuvatuista ohjelmistoista perustuvat tähän lähestymistapaan. Pearlin *ehdollistamismenetelmä (conditioning)* [104] on läheistä sukua solmujen yhdistämiseen perustuville tekniikoille: ehdollistamisessa monipolkuista verkkoa ei kuitenkaan eksplisiittisesti muuteta yksipolkuiseksi verkoksi, vaan ongelmia aiheuttavat polut poistetaan implisiittisesti kiinnittämällä (ehdollistamalla) polulla sijaitsevia muuttujia johonkin arvoon. Probabilistisen päättelyn toteuttamiseksi voidaan myös luopua laskennan deterministisyydestä, ja käyttää stokastisia päättelyalgoritmeja. Teoksessa [93] kuvataan kuinka kuinka tämän ongelman ratkaisemisessa voidaan käyttää Boltzmannin koneena tunnettua stokastista neuroverkkoarkkitehtuuria. Muita vaihtoehtoisia lähestymistapoja probabilistisen päättelyn toteuttamiseksi monipolkuisissa Bayes-verkoissa on kuvattu lähteissä [18, 56, 31, 116, 85, 143].

3.3 Bayes-verkkojen rakentaminen

Kuten luvussa 3.1 näimme, Bayes-verkkomallien komponenteille voidaan antaa intuitiivisesti selkeä semanttinen tulkinta: Bayes-verkkostruktuuri kuvaa ongelmakentän kuvaamisessa käytettyjen attribuuttien välisiä riippuvuuksia, Bayes-verkon parametreina käytetyt ehdolliset todennäköisyydet taas riippuvuuksien voimakkuuksia. Tämäntyyppinen tieto on usein helposti saatavilla ongelmakentän asiantuntijoilta, jolloin Bayes-verkkojen konstruointi on helppoa. Bayes-verkkokehittimiä (katso luku 4.2) käyttäen Bayes-verkkojen soveltaminen ei edellytä syvempää Bayes-verkkoteorian ymmärtämistä, jolloin asiantuntijat voivat myös konstruoida Bayes-verkkomalleja suoraan itse, ja siirtyä samalla testaamaan kehitettyjen mallien toimivuutta. Eric Horvitzin mukaan[3] United Airlinesin insinöörit testasivat ensimmäisiä itse kehittämiään Bayes-verkkomalleja jo muutaman tunnin kuluttua siitä kun Bayes-verkkomallien käsite ja niiden konstruoimiseen soveltuvat kehittimet oli heille esitelty.

Toisaalta, kuten luvussa 1.2 todettiin, bayesiläisen mallinnuksen yksi suurimpia vahvuuksia on se, kuinka lähestymistapa mahdollistaa asiantuntijatietämyksen yhdistämisen tilastolliseen oppimiseen esimerkkiaineistosta. Mallien oppiminen jaettiin edellä kahteen vaiheeseen: mallistruktuurin oppiminen

[3]Katso URL: http://www.auai.org/BN-Testimonial.html

ja malliparametrien oppiminen. Kuvaamme seuraavassa kuinka laskea oppimiskriteeri eri Bayes-verkkostruktuureille, ja kuinka kiinnittää Bayes-verkon parametrit, annettuna struktuuri. Samassa yhteydessä selvitämme kuinka asiantuntijatietämys voidaan priorijakaumia käyttäen ottaa huomioon oppimisprosessissa.

3.3.1 Verkkostruktuurin oppiminen

Luvussa 1.2.2 näimme että bayesiläisessä oppimisessa mallistruktuurit evaluoidaan vertailemalla mallien posterioritodennäköisyyksiä

$$P(M \mid \mathcal{D}) \propto P(\mathcal{D} \mid M)P(M),$$

ja luvussa 1.2.4 näimme kuinka asiantuntijatietämys voidaan integroida oppimisprosessiin priorijakaumien kautta. Mallistruktuurin oppimisen päämääränä on siis iteratiivisen etsintäalgoritmin kautta etsiä mallistruktuuri M joka maksimoi tämän todennäköisyyden. Bayes-verkkojen tapauksessa etsintäavaruus on valtava: erilaisten syklittömien suuntamattomien verkkojen lukumäärä saadaan lähteessä [114] esitetystä rekursioyhtälöstä, josta näemme että jo kymmenellä solmulla saadaan aikaan noin 4.2×10^{18} erilaista Bayes-verkkostruktuuria! Tässä valossa ei olekaan lainkaan yllättävää että Bayes-verkkojen oppimisongelman on osoitettu olevan NP-kova ongelma [25]. On kuitenkin jälleen kerran korostettava, että kyseessä on pahimman tapauksen analyysi, joka ei kerro mitään ongelman keskimääräisestä vaikeudesta. Bayes-verkkojen oppimisessa onkin sovellettu hyvällä menestyksellä tunnettuja heuristisia etsintäalgoritmeja — viitteitä käytettyihin algoritmeihin löytyy esimerkiksi teoksista [48, 122].

Jos ongelmakentästä ei ole saatavilla priori-informaatiota, on järkevää olettaa eri struktuureille tasainen priorijakauma, jolloin todennäköisyyttä $P(M)$ voidaan pitää vakiona, ja käyttää oppimiskriteerinä yksinomaan kokonaisuskottavuutta $P(\mathcal{D} \mid M)$. Mikäli priori-informaatiota on saatavilla, voidaan tätä tietämystä käyttää hyväksi priorijakauman $P(M)$ määräämisessä. Koska mahdollisten Bayes-verkkojen määrä on kuitenkin tavattoman suuri, ei ole käytännössä mahdollista määrätä prioritodennäköisyyttä $P(M)$ erikseen kullekin Bayes-verkkostruktuurille. Priori-informaatiota voidaan kuitenkin käyttää hyväksi yhden (tai muutaman) ns. prioriverkon konstruoimisessa. Prioriverkoilla tarkoitetaan verkkostruktuureja joiden prioritodennäköisyden arvellaan olevan hyvin suuri (verrattuna muihin verkkostruktuureihin). Prioriverkkoja voidaan käyttää hyödyksi mallistruktuurin etsimisessä esimerkiksi heurististen etsintäalgoritmien aloituspisteinä, tai varsinaisen priorijakauman estimoinnissa lähteessä [50] esitetyllä tavalla.

Bayes-verkkojen kokonaisuskottavuuden $P(\mathcal{D} \mid M)$ laskemiseksi on tehtävä joitakin teknisiä oletuksia, joista tärkein koskee parametrien priorijakaumaa $P(\theta \mid M)$. Kuten luvussa 3.1 näimme, Bayes-verkkostruktuuriin liitettävät parametrit θ ovat muotoa $P(X \mid F(X))$ olevia ehdollisia todennäköisyyksiä, missä $F(X)$ on muuttujan X edeltäjien joukko. Numeroidaan joukon $F(X)$ kaikki mahdolliset arvokombinaatiot, ja oletetaan että niitä on q_i kappaletta. Merkinnällä θ_{ijk} tarkoitamme seuraavassa parametria $P(X_i = k \mid F(X_i) = j)$. Jos oletamme nyt että priorijakauma $P(\theta_{ij} \mid M)$ on ns. *Dirichlet-jakauma* (ks. esim. [39]) *hyperparametrein*[4] $N'_{ij1}, \ldots, N'_{ijr_i}$, niin voidaan osoittaa [28, 50] että kokonaisuskottavuus voidaan laskea kaavalla (3.4).

Kaavassa (3.4) esiintyvät tilastolliset tunnusluvut N_{ijk} on helppo laskea opetusjoukosta $\mathcal{D}$, ja kokonaisuskottavuus voidaan laskea ajassa $O(Nn^2r)$, missä N on joukon $\mathcal{D}$ koko, n muuttujien lukumäärä, ja r suurin mahdollinen muuttujien arvojen lukumäärä. Käyttäjän tehtäväksi jää määrätä hyperparametrit N'_{ijk}. Mikäli prioritietämystä ei ole, voidaan käyttää tasaista priorijakauma joka saadaan asettamalla $N'_{ijk} = 1$ kaikilla i:n j:n, ja k:n arvoilla. Jos prioritietämystä on saatavilla, antaa kaava (3.4) vihjeen siitä kuinka asiantuntijatietämys voidaan koodata hyperparametreiksi N'_{ijk}: hyperparametrit ja tilastolliset tunnusluvut N_{ijk} ovat käytettävässä kaavassa täysin identtisessä asemassa, joten on helppo päätellä, että jos käyttäjä asettaa hyperparametrin N'_{ijk} arvoksi esimerkiksi viisi, on tämän asetuksen vaikutus kokonaisuskottavuuteen sama kuin jos käyttäjä olisi lisännyt opetusjoukkoon $\mathcal{D}$ viisi esimerkkiä, joissa kussakin muuttujan X_i arvo olisi k, ja muuttujan edeltäjät $F(X_i)$ olisivat arvokombinaatiossa j. Käyttäjä voi siis muotoilla henkilökohtaisen näkemyksensä muuttujan X_i arvojen jakaumasta muuttamalla hyperparametreja N'_{ijk}, ja hyperparametrien määräämisessä käytetyn arvoalueen skaalaa muuttamalla saadaan vaihdeltua prioritietämyksen vaikutuksen suhdetta opetusjoukkoon vaikutukseen verrattuna. Tätä lähestymistapaa kutsutaan *ekvivalentin otoskoon (equivalent sample size)* menetelmäksi: asiantuntija voi painottaa tietämyksensä merkitystä ajattelemalla sitä "prioriopetusjoukkona", joukkona opetusjoukon ulkopuolisia tapauksia joista asiantuntija on oman kokemusperäisen tietämyksensä muodostanut.

Kaavassa (3.4) esitetty menetelmä kokonaisuskottavuuden laskemiseksi pätee diskreeteille muuttujille, kun muuttujien arvot otosjoukossa $\mathcal{D}$ oletetaan jakautuneiksi multinomijakauman mukaan. Vastaavanlainen tulos voidaan kuitenkin itse asiassa johtaa hyvin laajalle joukolle erilaisia jakaumia

[4]Nimitys "hyperparametri" johtuu siitä että kyseessä on "toisen kertaluvun" parametri, yhteistodennäköisyysjakauman määrittelemisessä käytettävän parametrin jakauman määrittelemisessä käytetty parametri.

Kokonaisuskottavuuden laskeminen:

$$P(\mathcal{D} \mid M) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(N'_{ij})}{\Gamma(N'_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(N'_{ijk} + N_{ijk})}{\Gamma(N'_{ijk})}, \tag{3.4}$$

missä

- Γ on ns. Gamma-funktio, tavallisen kertomafunktion yleistys myös muille kuin kokonaisluvuille (minkä ansiosta hyperparametrien ei tarvitse välttämättä olla kokonaislukuarvoisia),
- n on muuttujien lukumäärä,
- q_i on Solmun X_i edeltäjien mahdollisten arvokombinaatioiden lukumäärä.
- r_i on muuttujan X_i arvojen lukumäärä,
- N_{ijk} on niiden tapausten lukumäärä joukossa $\mathcal{D}$, missä muuttujan X_i arvo on k, ja muuttujan X_i edeltäjät $F(X_i)$ ovat arvokombinaatiossa j,
- N'_{ijk} on parametria θ_{ijk} vastaavan priorijakauman määräämisessä käytetty hyperparametri,
- $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$, ja
- $N'_{ij} = \sum_{k=1}^{r_i} N'_{ijk}$.

(ns. eksponentiaaliselle perheelle), myös jatkuville jakaumille, ja jatkuvien ja diskreettien muuttujien muodostamille sekajakaumille [10]. Esimerkki Bayes-verkon kokonaisuskottavuuden laskemisesta jatkuvien jakaumien tapauksessa löytyy lähteestä [40].

Yllä kuvattu lähestymistapa perustuu Bayes-verkkostruktuurien vertailuun niiden posterioritodennäköisyyden (tai käytännössä yleensä kokonaisuskottavuuden) perusteella. Luvussa 3.1 esitetty Bayes-verkkojen määritelmä tarjoaa myös vaihtoehtoisen lähestymistavan tähän ongelmaan: koska kukin Bayes-verkkostruktuuri vastaa tiettyä joukkoa riippumattomuusoletuksia, voidaan opetusjoukosta ensin tilastollisesti päätellä ongelmakentässä vallitsevat riippumattomuudet, ja konstruoida sitten pääteltyjä riippumattomuuksia vastaava Bayes-verkkostruktuuri. Tähän lähestymistapaan perustuvia algoritmeja on esitetty esimerkiksi lähteessä [130]. Täydellisten riippumattomuuksien sijaan voidaan yrittää tilastollisesti päätellä muuttujien osittaisia järjestyksiä siten, että syntyvässä Bayes-verkossa muuttujasta X voi olla kaari muuttujaan Y vain jos X on Y:n edeltäjä kiinnitetyssä järjestyksessä. Tilastollisesti pääteltyjä osittaisia järjestyksiä voidaan myös yhdistää yllä esitettyyn MAP-metriikkaan perustuviin oppimisalgoritmeihin [125]. Osittaisen järjestyksen määräämiseksi voidaan tilastollisten testien sijaan käyttää tietenkin myös priori-informaatiota. Itse asiassa monet MAP-metriikkaa soveltavat oppimisalgoritmit, kuten esimerkiksi Cooperin ja Herskovitsin K2-algoritmi [28], edellyttävät tällaisen osittaisen järjestyksen määräämistä ennen oppimisprosessin alkua.

3.3.2 Bayes-verkon parametrien oppiminen

Suurimman todennäköisyyden parametrit Kun Bayes-verkkostruktuuri M on kiinnitetty, on seuraava vaihe mallin oppimisessa mallin parametrien θ valinta. Kuten luvussa 1.2.3 näimme, bayesiläisessä lähestymistavassa parametrit kiinnitetään MAP-arvoonsa, siis arvoon $\hat{\theta}$ joka maksimoi posterioritodennäköisyyden $P(\theta \mid \mathcal{D}, M)$:

$$\hat{\theta} = \arg\max_{\theta} P(\theta \mid \mathcal{D}, M).$$

Samojen teknisten oletusten vallitessa, joita käytettiin edellisessä luvussa esitetyn kokonaisuskottavuuden kaavan (3.4) johtamisessa, voidaan osoittaa että malliparametrien posterioritodennäköisyys maksimoituu asettamalla

$$\hat{\theta}_{ijk} = \frac{N_{ijk} + N'_{ijk} - 1}{N_{ij} + N'_{ij} - r_i}, \tag{3.5}$$

missä merkinnät ovat kuten edellä kaavassa (3.4). MAP-parametriarvojen laskemisessa ei siis tarvita iteratiivista oppimisalgoritmia, vaan ne voidaan määrätä suoraan käyttäen kaavaa (3.5)!

Suurimman uskottavuuden parametrit On tärkeää huomata, että yllä annettu laskukaava MAP-parametrien määräämiseksi poikkeaa klassisesta tilastotieteestä tutusta *suurimman uskottavuuden (maximum likelihood, ML)* parametriasetuksesta $\tilde{\theta}$,

$$\tilde{\theta}_{ijk} = \frac{N_{ijk}}{N_{ij}} = \frac{N_{ijk}}{\sum_{k=1}^{r_i} N_{ijk}}, \tag{3.6}$$

joka maksimoi datan uskottavuuden $P(\mathcal{D} \mid \theta, M)$. Koska

$$P(\theta \mid \mathcal{D}, M) \propto P(\theta \mid M) P(\mathcal{D} \mid \theta, M),$$

näemme että ML- ja MAP-parametrit ovat samat siinä tapauksessa, että priorijakauma $P(\theta \mid M)$ on tasainen, ja siis vakio. Suurimman uskottavuuden parametrit saadaan siis erikoistapauksena kaavasta (3.5), jos käytämme tasaisia priorijakaumia, eli asetamme kaikkien hyperparametrien N'_{ijk} arvoksi yksi.

Odotusarvoparametrit Parametrien θ asettaminen MAP-arvoihinsa on bayesiläisen perusidelogian mukainen vastaus kysymykseen kuinka kiinnittää parametrit annettuna mallistruktuuri. Mikäli syntyvää mallia (M, θ) kuitenkin käytetään probabilistisessa päättelyssä (eikä siis ainoastaan ongelmakentän analysoinnissa), näemme hiukan yllättäen että MAP-parametreja käyttävälle mallille löytyy vaihtoehto, joka tuottaa tarkempia ennustuksia probabilistisessä päättelyssä. Olkoon nimittäin $P(X_1, \dots, Xn \mid \theta, M)$ mallin (M, θ) tuottama jakauma. Koska emme tiedä varmasti mikä mallin θ tuottama jakauma vastaa ongelmakentän todellista todennäköisyysjakaumaa, arvioimme eri mallien hyvyyttä niiden posterioritodennäköisyyden $P(\theta \mid \mathcal{D}, M)$ mukaan käyttämällä apuna ongelmakentästä saatua otosta $\mathcal{D}$. Yllä esitetyssä MAP-lähestymistavassa valittiin siis parametrit, jotka maksimoivat tämän todennäköisyyden. Tarkempi jakauma saataisiin kuitenkin aikaan ottamalla esimerkiksi kaksi tässä mielessä parasta mallia θ_1 ja θ_2, ja muodostamalla niiden painotettu keskiarvo:

$$P(X_1, \dots, Xn \mid \theta_1, M)P(\theta_1 \mid \mathcal{D}, M) + P(X_1, \dots, Xn \mid \theta_2, M)P(\theta_2 \mid \mathcal{D}, M).$$

Seuraamalla tätä päättelyketjua eteenpäin, näemme että optimaalisessa tapauksessa meidän pitäisi käyttää *kaikkia (äärettömän monia) mahdollisia parametriasetuksia*, jolloin saamme jakauman

$$\int P(X_1, \dots, Xn \mid \theta, M)P(\theta \mid \mathcal{D}, M)d\theta = P(X_1, \dots, Xn \mid M).$$

Voidaan kuitenkin osoittaa että syntynyt jakauma on identtinen jakauman kanssa, joka saadaan käyttäen yhtä mallia jossa kukin parametriarvo on kiinnitetty odotusarvoonsa $E(\theta_{ijk} \mid \mathcal{D}, M)$. Odotusarvomallin käyttäminen vastaa siis, paradoksaalista kyllä, tilannetta jossa emme kiinnitä parametriarvoa lainkaan vaan käytämme kaikkia äärettömän monia parametriasetuksia! Odotusarvot saadaan laskettua kaavasta

$$\hat{\theta}_{ijk} = E(\theta_{ijk} \mid \mathcal{D}, M) = \frac{N_{ijk} + N'_{ijk}}{N_{ij} + N'_{ij}}. \tag{3.7}$$

Odotusarvoihin perustuvien mallien on empiirisesti havaittu tuottavan paljon tarkempia estimaatteja ongelmakentän todennäköisyysjakaumasta, etenkin kun käytettävissä olevan opetusjoukon $\mathcal{D}$ koko on pieni [74, 75, 133].

Kolme laskukaavaa malliparametrien määräämiseksi:

- Suurimman todennäköisyyden parametrit (3.5):

$$\theta_{ijk} = \frac{N_{ijk} + N'_{ijk} - 1}{N_{ij} + N'_{ij} - r_i}.$$

- Suurimman uskottavuuden parametrit (3.6):

$$\theta_{ijk} = \frac{N_{ijk}}{N_{ij}}.$$

- Odotusarvoparametrit (3.7):

$$\theta_{ijk} = \frac{N_{ijk} + N'_{ijk}}{N_{ij} + N'_{ij}}.$$

3.4 Bayes-verkkojen variaatioita

3.4.1 Päätösverkot (influence diagrams)

Luvussa 1.4 näimme kuinka probabilistisessa päätelyssä tuotettuja ehdollisia todennäköisyysjakaumia voidaan käyttää hyväksi päätöksentekoprosessissa päätösteoriana tunnetun formalismin kautta. *Päätösverkko (influence diagrams)* on Bayes-verkko, jossa päätöksentekoon liittyvät vaihtoehdot koodataan erityisinä päätössolmuina, ja päätössolmujen eri arvoihin liittyvä odotettavissa olevan hyödyn määrä lasketaan suoraan probabilistisen päättelyn yhteydessä (katso esim. [123, 101]). Päätössolmujen käsittelymahdollisuus on liitetty mukaan useisiin Bayes-verkkoja manipuloiviin ohjelmistoihin (katso luku 4.2).

3.4.2 "Noisy-or"-malli

Kuten luvussa 3.2 näimme, probabilistisen päättelyn eri muodot voidaan toteuttaa yksipolkuisilla (joko aidosti yksipolkuisilla, tai monipolkuista verkkoa vastaavalla liittymäpuulla) Bayes-verkoilla ajassa joka on suoraan verrannollinen verkkoa vastaavien parametrien määrään. Luvussa 3.1 näimme että nämä parametrit ovat muotoa $P(X \mid F(X))$, missä $F(X)$ on solmun X edeltäjien joukko annetussa verkossa. Tätä muotoa olevia parametreja on eksponentiaalinen määrä solmun X edeltäjien arvojen lukumäärän suhteen, joten jos solmulla on hyvin suuri joukko edeltäjiä, saattaa tarvittavien parametrien lukumäärä olla liian suuri käytännön tarpeita ajatellen[5]. Ns. "noisy-or"-mallissa muotoa $P(X \mid F(X))$ olevia parametreja approksimoidaan kaavan

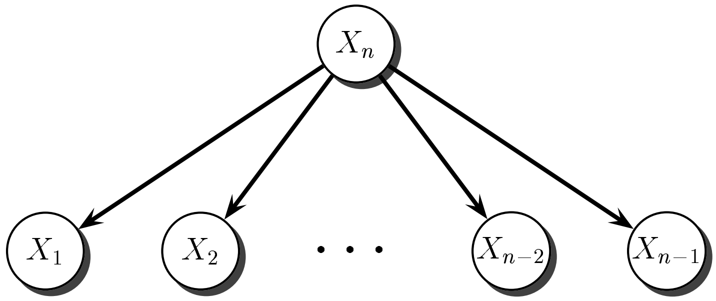
$$P(X \mid Y_1, \ldots, Y_n) = 1 - \prod_{i=1}^{n}(1 - P(X \mid Y_i))$$

mukaan. "Noisy-or"-mallia käyttäen voidaan tarvittavien parametrien lukumäärää pienentää siis huomattavasti: kutakin solmua X kohden riittää tallettaa $|F(X)|$ parametria, missä $|F(X)|$ on solmun edeltäjien lukumäärä. "Noisy-or"-approksimaation on empiirisesti havaittu toimivan useissa käytännön tilanteissa hyvin, vaikka toisinaan sen sisältämät implisiittiset riippumattomuusoletukset eivät pidä paikkaansa jolloin approksimaatio ei ole riittävän tarkka käytännön sovelluksia ajatellen [53, 97].

3.4.3 Jatkuva-arvoiset muuttujat

Tässä raportissa on havainnollisuuden vuoksi keskitytty tarkastelemaan diskreettejä Bayes-verkkoja, joissa erilaisia mahdollisia muuttujien arvoja on vain äärellinen määrä. Koska Bayes-verkko kuitenkin viime kädessä on yhteistodennäköisyysjakauman hajotelman graafinen esitys, joka kertoo kuinka muuttujajoukon yhteistodennäköisyys saadaan laskettua ehdollisten todennäköisyyksien tulona (katso kaava 3.3), ei mikään estä käyttämästä tässä hajotelmassa myös jatkuvia ehdollisia tiheysfunktioita: sekä luvussa 3.2 hahmoteltu probabilistisen päättelyn toteuttamiseen käytetty viestinvälitysalgoritmi [83, 3], että luvussa esitetty Bayes-verkkojen rakentamisessa käytetty mallinvalintakriteeri [40] voidaan määritellä myös jatkuvien muuttujien tapauksessa, ja myös sekatapauksessa jossa käytetään sekä jatkuvia että diskreettejä muuttujia. Malliperheen laajennus tällä tavalla saattaa joissakin tapauksissa aiheuttaa teknisiä ongelmia — yksinkertaisinta jatkuvien muuttujien käsittely on ns. äärellisissä sekajakaumamalleissa, joita käsittelemme luvussa 3.4.5. On myös huomattava että käytännön sovelluksissa reaaliarvoisten muuttujienkaan arvot eivät todellisuudessa ole äärettömän tarkkuuden reaalilukuja, vaan jollakin äärellisellä tarkkuudella esitettyjä numeroita. Niinpä monissa sovelluksissa jatkuvia arvoja voidaankin yhtä hyvin käsitellä suoraan diskreetteinä muuttujina, tai ne voidaan ensin diskretisoida. Bayesiläistä lähestymistapaa diskretisointiin on käsitelty lähteissä [37, 71].

[5] Useimmat käytännön sovelluksissa käytettävät Bayes-verkot ovat suhteellsien harvoja, jolloin tätä ongelmaa ei esiinny.



Kuva 3.8: Naiivi Bayes-mallin Bayes-verkkoesitys. Juurimuuttuja X_n on luokkamuuttuja, jonka suhteen muiden muuttujien riippumattomuusoletus tehdään.

3.4.4 Naiivi Bayes-malli

Luokitteluongelmissa tehtävänä on laskea diskreetin luokkamuuttujan jakauma annettuna muiden muuttujien (mahdollisesti osittainen) arvokombinaatio $\mathbf{S}_2 = \mathbf{s}_2$. Niin sanotussa *Naiivi Bayes*-mallissa muut muuttujat oletetaan riippumattomiksi, jos luokkamuuttujan arvo on tunnettu. Tästä oletuksesta seuraa että muuttujien yhteistodennäköisyysjakauma voidaan esittää muodossa

$$P(X_1, \ldots, X_n) = P(X_1, \ldots, X_{n-1} \mid X_n)P(X_n) = P(X_n) \prod_{i=1}^{n-1} P(X_i \mid X_n),$$

missä muuttujat on oletettu järjestetyiksi siten että luokkamuuttujan indeksi on n. Tämän hajotelman Bayes-verkkoesitys on kuvassa 3.8.

Naiivi Bayes-mallia on sovellettu monissa erilaisissa luokitteluongelmissa varsin hyvällä menestyksellä. Yksi syy mallin suosioon on probabilistisen päättelyn tehokkuus: koska Bayes-verkko on tässä tapauksessa kaksitasoinen puu, on laskenta äärimmäisen tehokasta. Itse asiassa Naiivi Bayes-mallissa tapahtuva probabilistinen päättely voidaan toteuttaa feedforward-neuroverkon tapaisena arkkitehtuurina [95, 94]. Mallin toinen merkittävä etu on se että koska mallin arkkitehtuuri on kiinnitetty, ei mallistruktuurin oppimisongelmaa ole, ja mallin parametrit voidaan asettaa suoraan luvussa 3.3.2 esitettyjä kaavoja käyttäen.

Vaikka Naiivi Bayes-malliin johtava riippumattomuusoletus saattaa tuntua melko epärealistiselta, toimii Naiivi Bayes-malli käytännössä usein yllättävän hyvin [133, 82]. Itse asiassa monet Bayes-verkkojen kaupallisista sovelluksista käyttävät tätä yksinkertaista mallia (esimerkiksi Microsoftin Answer Wizard ja Office Assistant, ks. luku 4.1). Hyvä esimerkki Naiivi Bayes-mallin sovellettavuudesta saatiin "Knowledge Discovery and Data Mining"-konferenssin yhteydessä kesällä 1997 julkistetussa maailmanlaajuisessa ennustuskilpailussa, jossa kaksi kolmesta parhaasta ratkaisusta käytti juuri Naiivi Bayes-mallia [103]. Tämän yksinkertaisen mallin menestykselle voidaan antaa seuraava intuitiivinen selitys: monissa luokitteluongelmissa (esimerkiksi lääketieteellisissä ongelmissa) on muuttujien arvojen määrittäminen kallista, koska se edellyttää esimerkiksi erityisten laboratoriokokeiden suorittamista. Niinpä kahden toisistaan riippuvan muuttujan mukaanottaminen on resurssien tuhlausta, koska jo toinenkin näistä muuttujista sisältää tarvittavan informaation. Vaikuttaa siltä että monet reaalimaailman datajoukot on kerätty resursseja säästäen, minkä seurauksena Naiivi Bayes-mallin riippumattomuusoletus saattaa olla hyvinkin järkevä.

Naiivi Bayes-mallin on toisaalta havaittu luokittelevan hyvin myös sellaisissa (keinotekoisissa) tilanteissa, joissa käytetty datajoukko on konstruoitu sellaiseksi, että Naiivi Bayes-mallin riippumattomuusoletus rikkoutuu varmasti [5]. Mallin luokittelutarkkuus on näissäkin tilanteissa ollut suuri — malli pystyy siis diskriminoimaan suurella tarkkuudella annetut esimerkit vastaaviin luokkiin — mutta jos tarkastellaan mallin antamaa jakaumaa luokkamuuttujalle, nähdään että mallin estimoima jakauma ei ole kovinkaan tarkka. Naiivi Bayes-mallin riippumattomuusoletuksesta seuraa siis ilmeisen konsistetti vinouma (bias) estimoituun luokkamuuttujan jakaumaan. Vinouma estää luokkamuuttujan jakauman tarkan estimoinnin, mutta koska oletuksista seurannut vinouma on kaikissa tilanteissa johdonmukainen, saattaa menetelmän erottelukyky olla edelleen hyvä. Tämä tarkoittaa siis sitä että Naiivi Bayes-malli saattaa olla hyvä vaihtoehto silloin jos luokittelussa riittää erotella annetut tapaukset eri luokkiin, mutta naiivi Bayes-mallin tarjoaman luokkamuuttujan jakauman käyttöön riskianalyysissä kannattaa suhtautua varauksella.

3.4.5 Latentit muuttujat ja äärelliset sekajakaumat

Standardit Bayes-verkkomallit perustuvat siihen oletukseen että mallintamisessa käytettävät muuttujat $X_1, \ldots, X_n$ on valittu siten että ongelmakentälle voidaan konstruoida toimiva malli valittuja muuttujia käyttäen. Mikäli näin ei ole, voidaan tietysti kyseenalaistaa koko tehtävänasettelu, ja valita käytettävät muuttujat uudelleen. Jos muuttujajoukon uudelleenmäärittely ei kuitenkaan tällaisessa tilanteessa ole mahdollista, saattaa Bayes-verkkojen (tai minkä tahansa vain annettuja muuttujia käyttävän malliperheen) soveltaminen tuottaa epätyydyttäviä tuloksia. Seuraava yksinkertainen esimerkki havainnollistaa tätä seikkaa.

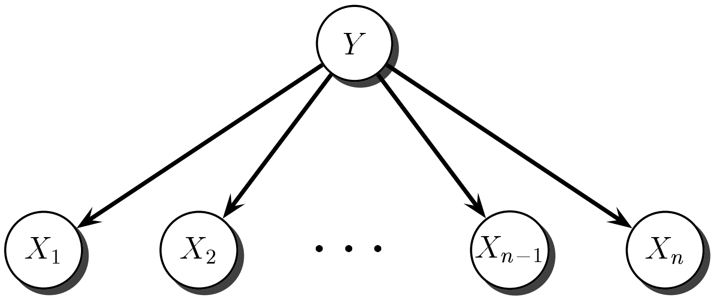
Oletetaan että käytettävissä olevia muuttujia on kaksi, X_1, ja X_2, ja muuttujan X_1 arvo kertoo hukkumiskuoleminen viikottaisen lukumäärän Suomessa, ja muuttujan X_2 arvo Suomessa viikottain myydyn jäätelön määrän. Tilastojen perusteella on helppo nähdä että muuttujilla on selvästi jonkinlainen riippuvuussuhde, joten Bayes-verkkostruktuuri jossa muuttujat eivät ole yhdistettyjä toisiinsa, ei mallinna ongelmakenttää tyydyttävällä tavalla. Jäljelle jää vain kaksi mahdollista Bayes-verkkostruktuuria: $X_1 \to X_2$ ja $X_1 \leftarrow X_2$. Intuitiivisesti ottaen nämä tapaukset näyttäisivät kuitenkin indikoivan että jompi kumpi muuttuja olisi toisen muuttujan syy, mikä ei varmaankaan pidä paikkaansa todellisuudessa. Jos meidän nyt sallitaan lisätä malliimme uusi muuttuja Y, niin näemme että Bayes-verkkorakenne $X_1 \leftarrow Y \to X_2$ tuottaa muuttujien X_1 ja X_2 välille *epäsuoran riippuvuussuhteen*, joka mallintaa tyydyttävästi intuitiivista kuvaamme ongelmakentästä.

Edellisessä esimerkissä käytettyä muuttujaa Y kutsutaan *latentiksi muuttujaksi* tai *piilomuuttujaksi*, koska muuttuja ei ollut mukana alkuperäisessä muuttujajoukossa, ja sen arvoja ei siten ole annettu opetusjoukossa $\mathcal{D}$. Yksinkertaisessa esimerkissämme on käytetylle latentille muuttujalle helppo keksiä semanttinen tulkinta (vuodenaika), mutta aina näin ei ole asianlaita, eikä se bayesiläisen mallintamisen kannalta ole välttämättä tarpeenkaan: latenttien muuttujien arvoja opetusjoukossa voidaan pitää puuttuvana datana, joka marginalisoidaan pois käytetyistä laskukaavoista.

Mikäli latentteja muuttujia on vain yksi, ja latentista muuttujasta on kaari kaikkiin muihin muuttujiin, ja muiden muuttujien välillä ei ole kaaria, kutsutaan syntyvää Bayes-verkkomallia *äärelliseksi sekajakaumamalliksi (finite mixture model)* [35, 136]. Syntyvä yhteistodennäköisyysjakauma on tässä tapauksessa

$$\begin{aligned} P(X_1,\ldots,X_n) &= \sum_y P(X_1,\ldots,X_n,Y=y) \\ &= \sum_y P(X_1,\ldots,X_n \mid Y=y)P(Y=y) \\ &= \sum_y P(Y=y)\prod_{i=1}^{n} P(X_i \mid Y=y), \end{aligned}$$

missä summa käy yli kaikkien mahdollisten piilomuuttujan Y arvojen. Vastaava Bayes-verkkostruktuurin graafinen esitys on kuvassa 3.9. On tärkeää huomata että vaikka syntyvä mallistruktuuri on muodollisesti identtinen Naiivi Bayes-mallin (kuva 3.8) kanssa, on juurimuuttuja sekajaumamallien tapauksessa latentti muuttuja (ja mahdollinen luokkamuuttuja on siis puun lehti siinä missä muutkin muuttujat), joten malli ei tee samoja riippumattomuusoletuksia kuin Naiivi Bayes-malli. Itse asiassa on helppo nähdä että



Kuva 3.9: Äärellisen sekajakaumamallin Bayes-verkkoesitys. Juurimuuttuja Y on latentti piilomuuttuja, jonka suhteen muiden muuttujien riippumattomuusoletus tehdään.

mikä tahansa Bayes-verkkomalli voidaan muuttaa äärelliseksi sekajakaumamalliksi, eli sekajakaumamallit muodostavat universaalin malliperheen.

Bayes-verkkoihin verrattuna sekajakaumamalleilla on se etu, että koska syntyvä Bayes-verkkostruktuuri on yksinkertainen puu, on sekajakaumamalleilla suoritettava päättely laskennallisesti aina erittäin tehokasta. Lisäksi mahdollisia verkkostruktuureja on vain yksi, joten mallistruktuuriavaruus ei ole kovin suuri: ainoa vapaa parametri on latentin muuttujan arvojen lukumäärä, jonka voidaan olettaa olevan vähemmän kuin joukon $\mathcal{D}$ koko. Juurimuuttujan latenttiudesta seuraa kuitenkin että kokonaisuskottavuus $P(\mathcal{D} \mid M)$, joka Bayes-verkkojen tapauksessa lasketaan helposti kaavaa (3.4) käyttäen, ei ole sekajakaumien tapauksessa laskettavissa käytännössä tarkasti: kokonaisuskottavuus saadaan marginalisoimalla yli kaikkien mahdollisten piilomuuttujan arvoista muodostuvien vektorien Z yli,

$$P(\mathcal{D} \mid M) = \sum_{Z} P(\mathcal{D}, Z \mid M),$$

joita on eksponentiaalinen määrä. Sekajakaumamallien kokonaisuskottavuuden laskemiseksi on kuitenkin kehitetty monia tehokkaita approksimaatiomenetelmiä (katso [73, 81, 77, 72, 26]).

Kokonaisuskottavuuden laskemisen lisäksi piilomuuttujaoletuksesta implisiittisesti seuraava puuttuva informaatio tekee myös sekajakaumamallien malliparametrien laskemisen vaikeammaksi kuin normaalien Bayes-verkkojen tapauksessa. Tähänkin tehtävään on kehitetty tehokkaita approksimaatiomenetelmiä, joista yleisin tunnetaan nimella *Expectation-Maximization (EM)* [33, 91, 62, 13, 78]. Koska latenttia piilomuuttujaa Y voidaan ajatella eräänlaisena ”klusterimuuttujana”, joka ryhmittelee annetut esimerkkitapaukset, voidaan EM-algoritmia pitää probabilistisena ryhmittelyalgoritmina. Algoritmin tuottamia tapausryhmiä voidaan soveltaa ongelmakentän probabilistisessa analysoinnissa ja visualisoinnissa (data mining) samalla tavoin kuin esimerkiksi SOM-neuroverkkomallia [67]. Algoritmi tarjoaa siis bayesiläisen lähestymistavan *numeerisena taksonomiana* tunnetulle tutkimusalueelle [43].

Kuten luvussa 2.3 todettiin, sekajakaumamalliperhe tarjoaa yhtenäisen teoreettisen kehikon monille näennäisesti erilaisille lähestymistavoille, kuten esimerkkeihin perustuva päättely [135, 134, 95, 69, 70], ydinestimaattorit, ydinkantafunktiot (radial basis functions), sekä probabilistiset neuroverkot [12], itseorganisoivat kartat [14], sekä piilo-Markov-mallit (hidden Markov models) [126, 30, 108]. Lähteissä [133, 135] raportoidut laajat empiiriset testitulokset osoittavat että mallistruktuurin näennäisestä yksinkertaisuudesta huolimatta sekajakaumallit (ja myös jopa Naiivi Bayes-mallit) tuottavat konsistentisti erittäin hyviä tuloksia vaihtoehtoisilla menetelmillä (kuten esimerkiksi neuroverkot ja päätöspuut) saavutettuihin tuloksiin verrattuna. Erityisen kiinnostavaa on että bayesiläiset mallit näyttävät suoriutuvan hyvin myös tilanteissa, joissa opetusdataa on hyvin vähän: saatavilla olevan datan määrän vähentäminen jopa 90 prosentilla ei näytä juurikaan huonontavan mallien antamia tuloksia [74, 75].

3.4.6 Kvalitatiiviset Bayes-verkot

Kvalitatiivisissa Bayes-verkoissa (qualitative Bayesian networks) [140, 34] luovutaan verkon parametrien (ehdollisten todennäköisyyksien) esittämisestä kvantitatiivisina reaalilukuina, ja siirrytään äärelliseen joukkoon mahdollisia kvalitatiivisia arvoja, esimerkiksi joukkoon $\{-, 0, +\}$. Verkon kaaret kuvaavat tässä tilanteessa kuinka tietyn solmun edeltäjät vaikuttavat vastaavaan muuttujaan X: vähentääkö ('-') vai lisääkö ('+') edeltäjien arvokombinaatio muuttujan X todennäköisyyttä, vai onko arvokombinaatio X:n suhteen irrelevantti ('0'). Näyttää siltä että kvalitatiivisten Bayes-verkkojen tapauksessa on mahdollista suorittaa määrättyjä probabilistisen päättelyn muotoja hyvin tehokkaasti myös monipolkuisten verkkojen tapauksessa, mutta teoreettisia tuloksia asiasta ei vielä juuri ole. Suomenkielinen katsaus kvalitatiivisia verkkoja käsittelevän tutkimuksen nykytilaan löytyy lähteestä [127].

3.4.7 Bayes-verkkojen ja neuroverkkojen yhteyksiä

Kuten edellä on jo mainittu, bayesiläisiä tekniikoita voidaan periaatteessa soveltaa myös muiden malliperheiden kuin Bayes-verkkojen tapauksessa. Mackay [88] ja Neal [96] kuvaavat väitöskirjoissaan bayesiläisten tekniikoiden soveltamista suunnattujen neuroverkkojen oppimisessa. Mackay voitti kehittämällään ohjelmistolla vuonna 1993 järjestetyn laajan aikasarjaennustamiskilpailun [89]. Sekä Mackayn[6] että Nealin[7] kehittämät ohjelmistot ovat vapaasti saatavilla kehittäjiensä kotisivuilta.

Paitsi neuroverkkojen oppimisalgoritmien kehittämisessä, Bayes-verkkojen ja neuroverkkojen välille on luotu myös lukuisia muita yhteyksiä. Michael Jordanin tutkimusryhmä MIT:ssa on tutkinut Bayes-verkkojen ja suunnattujen neuroverkkojen välimuotoa, jossa Bayes-verkkojen muuttujien välisten riippuvuuksien voimakkuuksia kuvataan neuroverkoista tutulla sigmoidfunktiolla. Malliperheestä käytetään nimitystä *sigmoid belief networks* [117]. Bishop kuvaa kirjassaan [13] ydinkantafunktioiden (radial basis functions), probabilististen neuroverkkojen, ydinestimaattoreiden ja sekajakaumallien välisiä yhteyksiä. Myllymäki kuvaa väitöskirjassaan [93] kuinka Bayes-verkoissa suoritettava probabilistinen päättely voidaan suorittaa massiivisesti rinnakkaisella Boltzmannin kone-neuroverkkoarkkitehtuurilla. Bayesiläinen kehikko itseorganisoiville kartoille [67] on kuvattu artikkelissa [14].

[6] URL: http://wol.ra.phy.cam.ac.uk/mackay/homepage.html
[7] URL: http://www.cs.utoronto.ca/~radford/

Luku 4

Bayes-verkkojen sovelluksia

Bayes-verkkojen sovellusesimerkkejä: lääketieteellinen diagnosointi, prosessikontrolli, vikadiagnostiikka, tiedon analysointi. Millä ohjelmistoilla Bayes-verkkomalleja voidaan rakentaa (Bayes-verkkokehittimet): kaupalliset ohjelmistot, tutkimusprototyypit.

Bays-verkkosovellukset voidaan jakaa kahteen ryhmään: yksittäiset Bayes-verkkotekniikkoihin perustuvat sovellukset, ja yleiset Bayes-verkkokehittimet, jotka ovat geneerisiä työkaluja erillisten sovellusten tuottamista varten. Bayes-verkkotutkimukseen on koko sen vasta noin kymmenvuotisen historian aikana liittynyt selkeä pyrkimys kehitettyjen menetelmien kaupalliseen soveltamiseen, ja sovellusten määrä onkin viime vuosina ollut voimakkaassa kasvussa. Lisäkiinnostusta Bayes-verkkojen hyödyntämismahdollisuuksia kohtaan herätti Bill Gatesin lokakuussa 1996 antama lausunto, jonka mukaan Microsoftin kilpailuetu tulevaisuudessa perustuu heidän eritysosaamiseensa Bayes-verkkojen alueella [54]. Microsoftilla onkin yksi maailman johtavia Bayes-verkkojen tutkimusyksiköitä (katso luku 5.2.1).

Luvussa 4.1 esitellään joitakin yksittäisiä Bayes-verkkosovelluksia. Luvussa ei pyritä antamaan tyhjentävää luetteloa kaikista olemassaolevista Bayes-verkkosovelluksista, vaan tyypillisiä esimerkkejä Bayes-verkkotekniikoita käyttävistä sovelluksista erilaisilta sovellusalueilta. Jotkut mainituista sovelluksista on luotu käyttäen kaupallisia Bayes-verkkokehittimiä, joita käsitellään luvussa 4.2, toiset taas ovat käsityönä luotuja räätälöityjä ohjelmistoja. Lisätietoa Bayes-verkkosovelluksista löytyy mm. David Heckermanin laatimasta listasta[1], johon on koottu yli sata viitettä Bayes-verkkojen käytännön sovelluksia raportoiviin artikkeleihin vuosilta 1985–1995. Heckermanin lista on melko laaja, mutta jo hieman vanhentunut. AUAI-järjestö (Association for

[1] URL: ftp://ftp.research.microsoft.com/pub/dtg/david/BN-APPS.TXT

Uncertainty in Artificial Intelligence) on koonnut suppeahkon listan joistakin merkittävimmistä viimeaikaisista sovelluksista. Lista on saatavilla järjeston kotisivulta[2], mistä löytyy myös kokoelma käytännön soveltajilta kerättyjä kommentteja siitä, miksi Bayes-verkot ovat osoittautuneet onnistuneeksi ratkaisuksi heidän sovellusympäristössään.

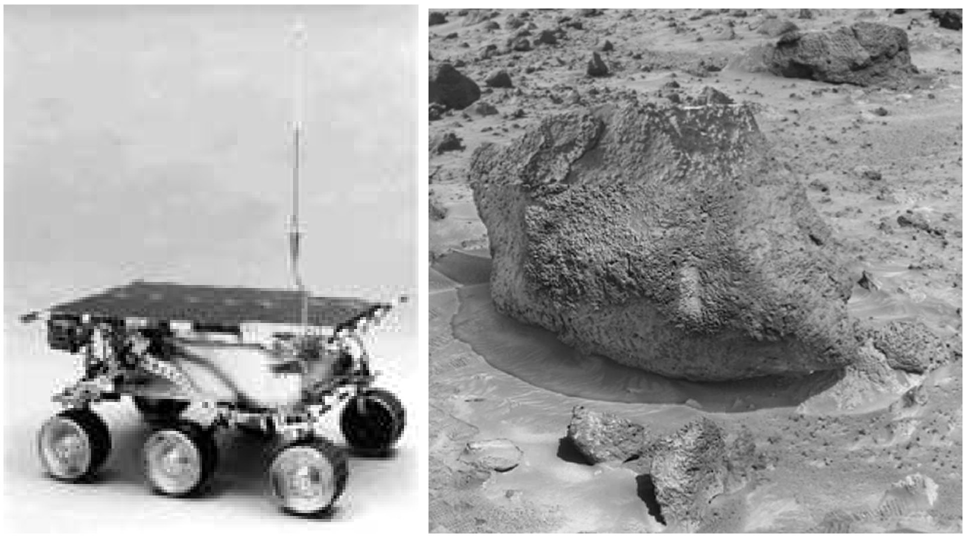
4.1 Sovellusesimerkkejä

Luvussa 3.2 esitettiin kuinka probabilistisen päättelyn eri muotoja voidaan toteuttaa käytännössä Bayes-verkkomalleja hyväksikäyttäen. Laajat julkisesti saatavilla olevilla luokitteluaineistoilla tehdyt vertailut osoittavat, että bayesiläiset mallit tuottavat konsistentisti erittäin hyviä tuloksia vaihtoehtoisilla menetelmillä (kuten esimerkiksi neuroverkot ja päätöspuut) saavutettuihin tuloksiin verrattuna [133, 135, 74, 79, 80]. Kuten luvussa 1.3 näimme, probabilistista päättelyä voidaan soveltaa luokitteluongelmien lisäksi ratkaisemaan hyvin monentyyppisiä ongelmatilanteita, joten Bayes-verkkojen sovellusalueitakin on lukuisia. Seuraavassa joitakin tyypillisimpiä sovellusesimerkkejä sovellusalueittain luokiteltuna.

Lääketieteellinen diagnosointi Lääketieteellinen diagnosointi on perinteisesti ollut yksi probabilististen menetelmien suosituimmista sovellusalueista. Yksi syy tähän varmaankin on se, että probabilististen menetelmien kanssa on tulosten luotettavuutta mahdollista arvioida tekemällä riskianalyysi luvussa 1.4 esitetyllä tavalla. Yksi ensimmäisiä tällaisia ohjelmistoja oli rintasyöpäpotilaiden diagnosointiin suunniteltu PathFinder [51], joka on myöhemmin kaupallistettu BiopSys Medical-yhtiön[3] IntelliPath-nimisenä tuotteena. IntelliPath-ohjelmisto on tällä hetkellä käytössä sadoissa eri sairaaloissa ympäri maailmaa. Muita vastaavia eri sairaaloissa ja tutkimuslaitoksissa aktiivisessa käytössä olleita järjestelmiä ovat mm. ACORN, DXPLAIN ja Iliad[4]. Yksi suurisuuntaisimpia tämän ongelma-alueen hankkeita on Microsoftin OnParenting-ohjelmisto[5], joka on lasten sairauksien diagnosointiin suunniteltu laaja järjestelmä. Järjestelmä käyttää sairauksien diagnosointiin Knowledge Industries-yhtiön kehittämää Bayes-verkkomallia.

Tiedon analysointi (data mining) Koska Bayes-verkkoja käyttäen voidaan tilastollisesta esimerkkiaineistosta konstruoida ongelmakentästä semant-

[2] URL: http://www.auai.org/
[3] URL: http://www.biopsys.cim
[4] URL: http://www-uk.hpl.hp.com/people/ewc/list-main.html
[5] URL: http://www.onparenting.msn.com



Kuva 4.1: NASA:n Pathfinder-projektin Marsin pinnalle lähettämä Sojouner-robotti, ja sen lähettämistä kuvista superresoluutiotekniikalla tuotettu kuva noin metrin kokoisesta kivenlohkareesta ("Yogi") Marsin pinnalla. Kuva löytyy osoitteesta http://mpfwww.jpl.nasa.gov/mpf/high-res.html.

tisesti tulkittavissa oleva malli, voidaan syntyviä malleja käyttää ongelmakentän ominaisuuksien analysoimisessa ja visualisoimisessa. NASAn Ames-tutkimuskeskuksen AutoClass-projektissa[6] on sovellettu bayesiläistä mallintamista menestyksellisesti mm. IRAS tähtikartaston kuvien, Yhdysvaltojen eri lentokenttien, ja eri DNA-sekvenssien analysoimisessa [24].

Kuvankäsittely NASAn tutkimusryhmät ovat kehittäneet menetelmän ns. *superresoluutiokuvien (super-resolution images)* tuottamiseksi. Menetelmässä on kyse useiden samasta kohteesta otettujen huonolaatuisten kuvien yhdistämisestä yhdeksi hyvälaatuiseksi ("superresoluutio"-) kuvaksi. Kuvassa 4.1 on tällainen Sojourner-robotin tuottamista kuvista bayesiläisellä tekniikalla muodostettu superresoluutiokuva kivenlohkareesta Marsin pinnalla.

Superresoluutiokuvia tuottava tekniikka ei perustu suoraan Bayes-verkkoteoriaan, mutta se seuraa puhtaasti edellä esitettyjä bayesiläisen mallintamisen periatteita (katso [23]). Paitsi Pathfinder-projektin aineistosta, NASAn tutkijat ovat tuottaneet superresoluutiokuvia myös Galileo-luotaimen aineistosta, sekä Viking-luotaimen Marsin pinnasta ottamista kuvista[7].

[6]URL: http:/ic-www.arc.nasa.gov/ic/projects/bayes-group/autoclass/
[7]URL: http://ic.arc.nasa.gov/ic/projects/super-res/

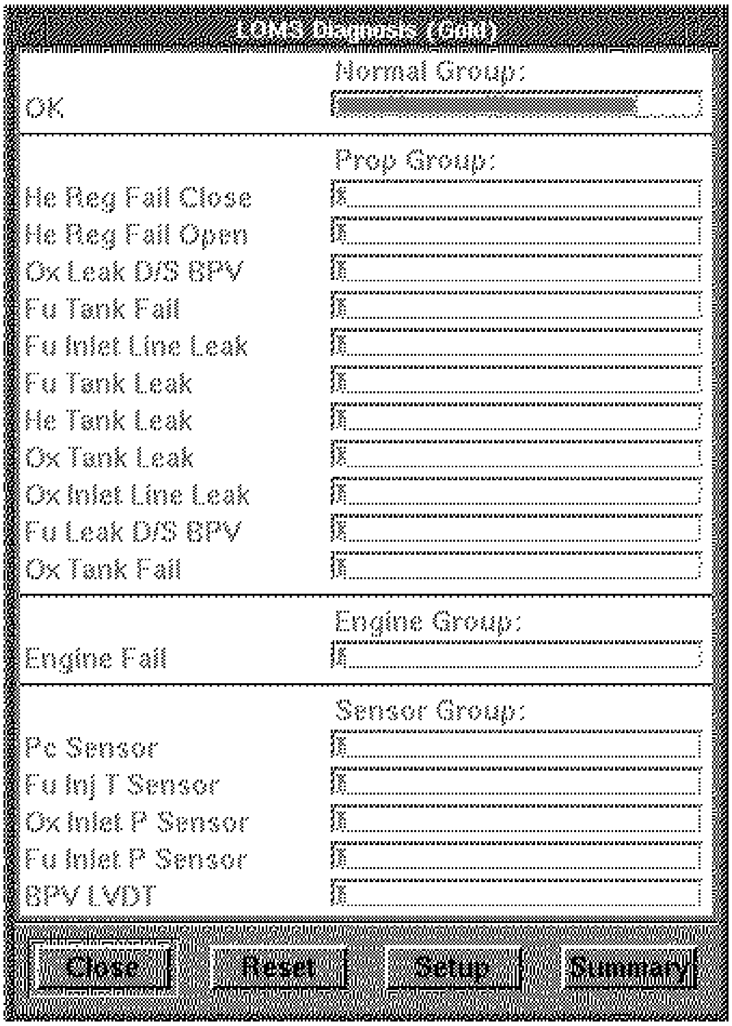
Hahmontunnistus Bayesiläistä mallinnusta on sovellettu hahmontunnistuksessa menestyksellisesti mm. NASA:n tutkimuskeskuksissa LandSat-satelliittien ottamien kuvien alueiden automaattisessa tunnistamisessa [64]. Käytetty Bayes-verkkomalli oli tässä tapauksessa luvussa 3.4 esitetty äärellinen sekajakaumamalli. Tavanomaisten Bayes-verkkojen käyttöä kuvantunnistuksessa käsitellään mm. lähteissä [61, 42]. Bayes-verkoilla on myös läheinen yhteys ns. *piilo-Markov-malleihin (hidden Markov models)* (katso [126, 30, 108]), joita käytetään yleisesti esimerkiksi puheentunnistukseen liittyvien hahmontunnistusongelmien ratkaisemisessa.

Vikadiagnostiikka Microsoftin tutkimusosasto on kehittänyt useita kymmeniä Bayes-verkkoihin perustuvia vikadiagnostiikkajärjestelmiä (troubleshooting wizards), jotka auttavat tietokoneen käyttäjää erilaisissa ongelmatilanteissa[8]. Intel käyttää Bayes-verkkoja prosessoripiirien valmistusprosessin laaduntarkkailussa ja American Airlines ohjelmistovirheiden etsimisessä[9].

Prosessikontrolli Yksi esimerkki prosessikontrolliin liittyvistä sovelluksista on kuvassa 3.4 esitetty ALARM-verkko jota käytetään sairaalapotilaiden tilaa tarkastelevan mittausprosessin ohjaamisessa [8]. NASA on kehittänyt laajan Bayes-verkkoihin perustuvan järjestelmän avaruussukkuloiden lentojen kontrollomiseksi. Järjestelmä paitsi ilmoittaa mahdollisista vikatilanteista, myös ehdottaa päätösteorian mukaisesti parhaat mahdolliset toimenpiteet kriisitilanteissa. Lisäksi ohjelmisto muokkaa aktiivisesti lentojen valvonnassa käytettäviä graafisia näyttöjä siten, että kullakin hetkellä näytetään vain tärkeimmät sukkuloiden tilaa kuvaavat mittaustulokset lentojen kulkua seuraavalle henkilökunnalle, mikä helpottaa teknikkojen työtä, ja nopeuttaa päätöksentekoa ongelmatilanteissa (katso kuva 4.2). VISTA-järjestelmä on ollut käytössä NASA:n Houstonin avaruussukkuloiden lennonjohtokeskuksessa jo useita vuosia. Enemmän tietoa VISTA-projektista löytyy projektin WWW-sivulta[10].

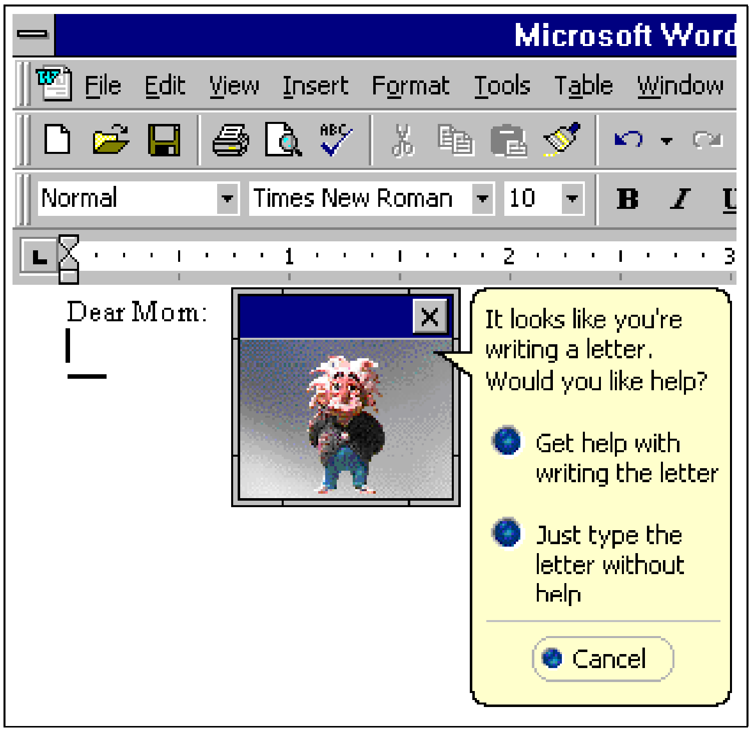
Bayes-verkkoja on sovellettu menestyksellisesti myös mm. lentokoneiden suihkumoottorien suunnittelussa (General Electric[11]), sekä erilaisten asejärjestelmien (Mitre/US Navy[12]) ja miehittämättömien vedenalaisten ajoneuvojen (Lockheed/DARPA[13]) ohjausprosesseissa.

[8] URL: http://support.microsoft.com/support/
[9] URL: http://www.auai.org/BN-Routine.html
[10] URL: http://www.research.microsoft.com/research/dtg/horvitz/vista.htm
[11] URL: http://www3.us.com/kic/kic/
[12] URL: http://www.afit.af.mil/Schools/EN/ENG/LABS/AI/BayesianNetworks/fieldedSystems.html#imsi
[13] URL: http://hugin.dk/lockheed-9604.html



Kuva 4.2: Esimerkki VISTA-järjestelmän muokkaamasta adaptiivisesta graafisesta näytöstä, jonka perusteella avaruussukkuloiden valvontakeskus seuraa lentojen kulkua.

Älykkäät agentit Älykkäät agentit ovat perussovellusohjelmiin liitettäviä itsenäisiä moduleita, jotka seuraavat järjestelmän toimintaa, ja helpottavat sovellusten käyttöä tarjoamalla aktiivisesti apua silloin kun katsovat sen käyttäjän kannalta tarpelliseksi. Microsoft on soveltanut Bayes-verkkotekniikkaa älykkäiden agenttien toteuttamisessa, ja tekniikka on jo käytössä Office'97-ohjelmistossa nimellä *Office Assistant*-nimisenä tuotteena (katso kuva 4.3). Microsoftin pitkän tähtäyksen suunnitelmana on toteuttaa Bayes-verkkotekniikoihin perustuva *adaptiivinen käyttöjärjestelmä*, koodinimeltään *Lumiere*, joka seu-



Kuva 4.3: Microsoftin Office Assistant-agentti.

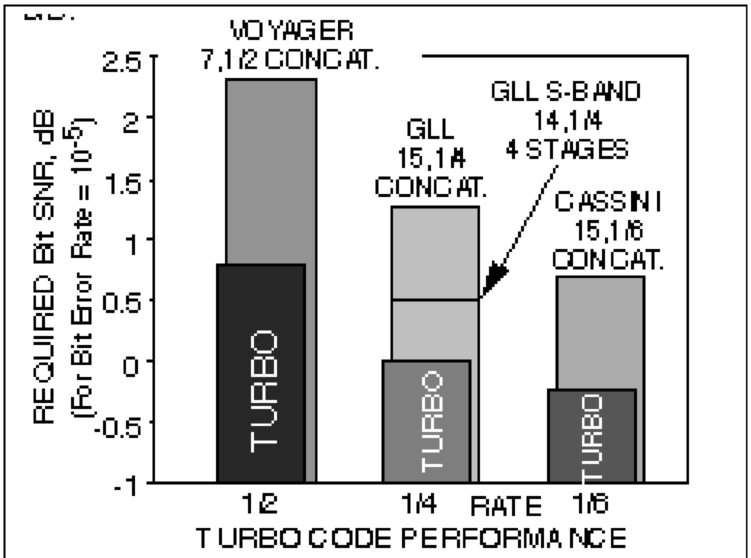
raa tietokoneen käyttäjän toimintaa, ja adaptoituu toimimaan siten että tietokoneen käyttö on mahdollisimman tehokasta. Lisätietoja Lumiere-projektista löytyy projektin WWW-sivulta[14].

Tiedon kompressointi Kuten luvussa 2.4 näimme, bayesiläiselle lähestymistavalle voidaan antaa myös informaatioteoreettinen tulkinta todennäköisyysjakaumien ja koodinpituuksien läheisen yhteyden ansiosta. Niinpä ei olekaan yllättävää että Bayes-verkkoteoriaa voidaan soveltaa myös erilaisten koodausmenetelmien kehittämisessä, ja kehitettyjä koodeja voidaan soveltaa esimerkiksi tiedon kompressoinnissa. Vuonna 1993 esiteltiin uusi vallankumouksellinen koodausmenetelmä, *turbo-koodaus* [11], jonka on empiirisesti havaittu olevan huomattavasti tehokkaampi koodausmenetelmä kuin aikaisemmin käytetyt koodit (katso kuva 4.4[15]).

Teoksessa [90] osoitetaan kuinka turbo-koodaaja voidaan esittää Bayes-verkkomallina. Turbo-koodausta pidetään yhtenä tärkeimmistä koodausteo-

[14] URL: http://www.research.microsoft.com/research/dtg/horvitz/lum.htm

[15] Kuva löytyy WWW-sivulta URL: http://www331.jpl.nasa.gov/public/JPLtcodes.html



	NO. OF STATES	NO. OF ITERATIONS	INTERLEAVER SIZE (bits)
TURBO	16+16	10	16,384
VOYAGER	64	1	8,160
GALILEO			
planned	16,384	1	4,080
redesigned	8,192	4	16,320
CASSINI	16,384	1	10,200

TURBO DECODER COMPLEXITY

Kuva 4.4: Turbo-koodin vertailu avaruusluotaimissa nykyisin käytössä oleviin koodausmenetelmiin. Huomaa että histogrammikuvan asteikko on logaritminen, joten erot eri menetelmien välillä ovat huomattavia.

rian tuottamista tuloksista käytännön sovellusten kannalta. Menetelmä on erityisen kiinnostava avaruustutkimuksen kannalta, koska avaruusluotaimien lähetyskapasiteetti on melko pieni, jolloin tarvitaan tehokkaita tiedon pakkausalgoritmeja.

4.2 Bayes-verkkokehittimet

Koska uusia Bayes-verkko-ohjelmistoja syntyy kiihtyvässä tahdissa, oheinen lista saatavilla olevista Bayes-verkkokehittimistä vanhenee nopeasti. Elektronisesti saatavilla oleva Russell Almondin ylläpitämä lista löytyy Almondin kotisivulta[16], ja toinen vastaavanlainen lista CoSCo-ryhmän kotisivulta[17].

Ohjelmistot on seuraavassa jaoteltu kaupallisiin tuotteisiin ja tutkimusprototyyppeihin, mutta on huomattava että joistakin kaupallisista ohjelmistoista on saatavilla ilmainen versio tutkimus- tai opetuskäyttöön. Lisäksi useimmat kaupalliset yritykset tarjoavat tuotteistaan vapaasti kokeiltavia demo-versioita, jotka on saatavilla alla annettujen WWW-linkkien kautta. On myös todettava, että jotkut ilmaiseksi jaettavat, ja siksi tutkimusprototyypeiksi luokitellut ohjelmistot ovat ohjelmistoteknisesti kaupallisten ohjelmistojen tasolla (esim. Microsoftin MSBN).

Alla olevat ohjelmistot on jaoteltu sen mukaan, voidaanko ohjelmistoja käyttää Bayes-verkkojen oppimiseen datasta, vai tukevatko ohjelmat ainoastaan verkkojen manuaalista konstruomista. Oppimista tukevien ohjelmien lukumäärä ei ole toistaiseksi kovin suuri, mutta tilanne tullee muuttumaan lähitulevaisuudessa radikaalisti kun tuoreet aiheeseen liittyvät tutkimustulokset saadaan toteutettua sovellusasteelle.

4.2.1 Kaupalliset ohjelmistot

Oppimista tukevat ohjelmistot

- Ergo/Cogito
  Lisätietoja: Noetic Systems,
  URL: http://www.noeticsystems.com/ergo.shtml

- MIM
  Lisätietoja: David Edvards,
  URL: http://www.math.auc.dk/~jhb/CoCo/mim.html

- TETRAD
  Lisätietoja: Carnegie-Mellon University,
  URL: http://hss.cmu.edu/HTML/departments/philosophy/TETRAD/ tetrad.html

Manuaaliseen mallin konstruoimiseen perustuvat ohjelmistot

- Baron
  Lisätietoja: KC Associates,
  URL: mailto: kchang@gmu.edu

- Analytica
  Lisätietoja: Lumina Decision Systems,
  URL: http://www.lumina.com/software/index.html

[16] URL: http://bayes.stat.washington.edu/almond/belief.html
[17] URL: http://www.cs.Helsinki.FI/research/cosco/

- Decision Tools Suite
  Lisätietoja: Palisade,
  URL: http://www.palisade.com/

- DPL: Decision Programming Language
  Lisätietoja: Applied Decision Analysis,
  URL: http://www.dpl.adainc.com/

- DX Solution Series
  Lisätietoja: Knowledge Industries,
  URL: http://www.kic.com/

- GRAPHICAL-BELIEF
  Lisätietoja: Russell Almond,
  URL: http://bayes.stat.washington.edu/almond/gb/graphical-belief.html

- HUGIN
  Lisätietoja: Hugin Expert,
  URL: http://www.hugin.dk/

- NETICA
  Lisätietoja: Norsys Software,
  URL: http://www.norsys.com/netica.html

- STRATEGIST
  Lisätietoja: Prevision,
  URL: http://www.prevision.com/strategist.html

- DATA
  Lisätietoja: TreeAge Software,
  URL: http://www.treeage.com/

4.2.2 Tutkimusprototyypit

Oppimista tukevat ohjelmistot

- Belief Network Power Constructor
  Lisätietoja: Jie Cheng,
  URL: http://193.61.148.131/jcheng/bnpc.htm

- BIFROST
  Lisätietoja: Bo Thiesson,
  URL: http://www.sp.dk/~sorenh/software.html

- BKD: Bayesian Knowledge Discoverer
  Lisätietoja: Marco Ramoni,
  URL: http://kmi.open.ac.uk/~marco/projects/bkd/software/
- BNG
  Lisätietoja: Peter Haddawy,
  URL: http://www.cs.uwm.edu/public/dsail/research/bng.html
- BUGS
  Lisätietoja: Bugs-projekti,
  URL: http://www.mrc-bsu.cam.ac.uk/bugs/
- COCO
  Lisätietoja: Jens Henrik Badsberg,
  URL: http://www.math.auc.dk/~jhb/CoCo/cocoinfo.html

Manuaaliseen mallin konstruoimiseen perustuvat ohjelmistot

- BAYES
  Lisätietoja: Carnegie-Mellon University,
  URL: http://www.cs.cmu.edu/afs/cs/project/ai-repository/ai/areas/reasonng/probabl/bayes/0.html
- BAYONNET
  Lisätietoja: Yoichi Motomura,
  URL: http://www.etl.go.jp/etl/suri/motomura/BN/bn-java.html
- JAVABAYES
  Lisätietoja: Fabio Gozman,
  URL: http://www.cs.cmu.edu/~javabayes/Home/
- Bayesian Network Editor
  Lisätietoja: Sreekanth Nagarajan,
  URL: http://eel.engr.orst.edu/~nagarasr/bayesnet/index.html
- BELIEF
  Lisätietoja: Carnegie-Mellon University,
  URL: http://www.cs.cmu.edu/afs/cs/project/ai-repository/ai/areas/reasonng/probabl/belief/0.html
- IDEAL
  Lisätietoja: Rockwell International Science Center,
  URL: http://rpal.rockwell.com/ideal.html
  URL: mailto: ideal-request@rpal.rockwell.com

- MacEvidence
  Lisätietoja: Prakash Shenoy,
  URL: http://stat1.cc.ukans.edu:80/~pshenoy/
- Microsoft Bayes Networks (MSBN)
  Lisätietoja: Microsoft Research,
  URL: http://www.research.microsoft.com/research/dtg/msbn/
- Pulcinella
  Lisätietoja: IRIDIA,
  URL: http://iridia.ulb.ac.be/pulcinella/Welcome.html
- S-ElimBel
  Lisätietoja: Nicholas Thiéry,
  URL: http://www.spaces.uci.edu/thiery/elimbel/
- SPI
  Lisätietoja: Bruce D'Ambrosio,
  URL: http://www.cs.orst.edu/~dambrosi/
- TresBel
  Lisätietoja: Hong Xu,
  URL: http://iridia.ulb.ac.be/Projects/imple.html
- XBaies
  Lisätietoja: Robert Cowell,
  URL: mailto:xbaies-list@stats.ucl.ac.uk
- X-pert
  Lisätietoja: AI Group, Cantabria University,
  URL: http://ccaix3.unican.es/~AIGroup/
- XPress
  Lisätietoja: Mark Stitson,
  URL: mailto:M.Stitson@rhbnc.ac.uk

Yllä olevaan listaan on otettu mukaan vain standardeja Bayes-verkkoja käyttävät sovellukset. Latentteja muuttujia käyttäviä sekajakaumamalleja käyttäviä ohjelmistoja on lukuisia — tunnetuimpia näistä ohjelmistoista ovat:

- AutoClass
  Lisätietoja: NASAn AutoClass-projekti,
  URL: http://ic-www.arc.nasa.gov:80/ic/projects/bayes-group/group/autoclass/

- Snob
  Lisätietoja: Snob-kotisivu,
  URL: http://www.cs.monash.edu.au/~dld/Snob.html

Sekä AutoClass että Snob ovat tutkimusprototyyppejä, joiden toiminta perustuu mallien bayesiläiseen oppimiseen datasta. Linkkeja sekajakaumamallien kaupallisiin sovelluksiin, ja muihin tutkimusprototyyppeihin löytyy David Dowen ylläpitämältä WWW-sivulta[18].

Tilastollisista sovelluksistaan tunnettu SPSS-yhtiö on myös ryhtynyt soveltamaan bayesiläisiä tekniikoita kaupallisissa ohjelmistoissaan. Käytettävä malliperhe ei tosin ole Bayes-verkot, vaan neuroverkkomallit. Yhtiön mainoksen mukaan bayesiläisen lähestymistavan avulla voidaan välttää ylioppiminen, ja mallien oppiminen on myös myös yleensä nopeampaa. Lisätietoja bayesiläisiä neuroverkkoja soveltavasta "Neural Connection 2"-tuotteesta löytyy WWW-sivulta URL: http://www.spss.com/software/Neuro/. Vastaavanlainen vapaasti saatavilla olevilla ohjelmisto on raportoitu Radford Nealin väitöskirjassa [96], ja on saatavilla Nealin kotisivulta[19]. Nealin ohjelmistoa on laajennettu hiljakkoin käsittämään neuroverkkojen lisäksi myös äärellisiä sekajakaumamalleja.

[18] URL: http://www.cs.monash.edu.au/~dld/mixture.modelling.page.html
[19] URL: http://www.cs.utoronto.ca/~radford/

Luku 5

Tietolähteitä

Bayes-verkkojen aktiiviset kaupalliset soveltajat ja niitä markkinoivat yritykset. Bayes-verkkoja koskevaa kirjallisuutta. Tutkimusalan tutkimusryhmät ja järjestöt.

5.1 Kaupallinen sektori

Bayes-verkkojen kaupallistamisen pioneereja on tanskalainen HUGIN-yhtiö, joka soveltaa Aalborgin yliopistossa toimivan, professori Steffen Lauritzenin johtaman tutkimusryhmän tuloksia kaupallisen Hugin-ohjelmiston kehittämisessä. Kuten Bill Gatesin antamasta lausunnosta [54] saattaa arvata, merkittävin kaupallinen tekijä Bayes-verkkojen sovellusalueella on tällä hetkellä eittämättä Microsoft, jonka Bayes-verkkotutkimusryhmä on tutkimusalan suurimpia.

HUGIN-yhtiö on keskittynyt kehittämään ja myymään yhtä tuotetta, Hugin-kehitintä, kun taas Microsoft tuottaa sisäisiä sovelluksia, jotka liitetään yhtiön kaupallisiin tuotteisiin. Monet Bayes-verkkosovelluksia tekevistä yhtiöistä toimivat HUGIN-yhtiön tapaan tuottaen geneerisiä Bayes-verkkokehittimiä, mutta yhä useammat yhtiöt laajentavat toimintaansa myös konsultointialueelle, ja auttavat asiakkaitaan rakentamaan laajoihin järjestelmiin integroitavia räätälöityjä sovelluksia, samalla tavoin kuin Microsoft liittää oman tutkimuskeskuksensa tulokset tuotteisiinsa. Esimerkkejä tällaisista räätälöidyistä sovelluksista löytyy AUAI-järjestön kotivulta[1], samoin kuin lista syistä jotka ovat johtaneet Bayes-verkkojen käyttöönottoon[2].

Bayes-verkkotekniikoiden aktiivisiin kaupallisiin soveltajiin kuuluvat tällä hetkellä seuraavat yritykset:

[1] URL: http://www.auai.org/BN-Routine.html
[2] URL: http://www.auai.org/BN-Testimonial.html

- Applied Decision Analysis, URL: http://www.dpl.adainc.com/
- Decisioneering, URL: http://www.decisioneering.com/
- Hugin Expert, URL: http://www.hugin.dk/
- Knowledge Industries, URL: http://www.kic.com/
- Lumina Decision Systems, URL: http://www.lumina.com/
- Microsoft, URL: http://www.research.microsoft.com/research/dtg/
- Noetic Systems, URL: http://www.noeticsystems.com/
- Norsys Software, URL: http://www.norsys.com/home.html
- Palisade, URL: http://www.palisade.com/
- Prevision, URL: http://www.prevision.com/
- ThinkBank, URL: http://www.Thinkbank.COM/
- TreeAge Software, URL: http://www.treeage.com/
- Ultimode Systems, URL: http://www.ultimode.COM/index.html

5.2 Bayes-verkkotutkimus

5.2.1 Tutkimusryhmiä

Yksi merkittävimmistä Bayes-verkkotutkimusryhmistä on perinteisesti ollut professori Steffen Lauritzenin johtama tutkimusryhmä Aalborgin yliopistossa Tanskassa. Ryhmä on osallistunut aktiivisesti myös HUGIN-ohjelmistotuotteen (ks. luku 4.2) rakentamiseen. Muita merkittäviä tutkimusryhmiä Euroopassa löytyy mm. Suomesta, Belgiasta, ja Espanjasta (ks. allaoleva lista).

Maailmanlaajuisesti suurin tutkimusryhmä tällä hetkellä on luultavasti Microsoftilla, joka on haalinut koko 90-luvun ajan lupaavia tutkijoita tutkimuskeskukseensa. Microsoftin asema lienee vain vahvistumassa tulevaisuudessa, koska tutkimusryhmä on tuottanut jo useita menestyksellisiä Bayes-verkkosovelluksia (ks. luku 4.1), ja yhtiö on lisäksi lisäämässä tutkimusrahoitustaan moninkertaiseksi nykyiseen verrattuna. Yrityksen pääjohtajan Bill Gatesin lausunnot [54] kuvaavat myös vahvaa luottamusta Bayes-verkkotekniikkaan. Toinen merkittävä tekijä tutkimusalueella Yhdysvalloissa on NASA, jolla on ollut lukuisia menestyksellisiä Bayes-verkkoprojekteja.

Monash-yliopiston tutkimusryhmä Australiassa on tehnyt pioneerityötä sekajakaumamallien tutkimuksen alueella, samoin kuin CoSCo-ryhmä Helsingin yliopistossa.

Alla oleva lista sisältää joitakin WWW-linkkejä Bayes-verkkotutkimusta tekeviin ryhmiin Suomessa, Euroopassa, ja muualla maailmassa. Lista sisältää luonnollisesti vain muutaman esimerkin Bayes-verkkotutkimusta tekevistä tutkimusryhmistä; lisää hyödyllisiä WWW-linkkejä (mm. yli 100 nimeä käsittävä luettelo bayesiläistä mallintamista tutkivista henkilöistä[3]) löytyy AUAI-järjestön kotisivulta (ks. luku 5.2.4).

- Helsingin yliopisto
  URL: http://www.cs.Helsinki.FI/research/cosco/
- Aalborgin yliopisto,
  URL: http://www.cs.auc.dk/general/ES/
- Cantabrian yliopisto/Cornell University
  URL: http://ccaix3.unican.es/~AIGroup/
- IRIDIA,
  URL: http://iridia.ulb.ac.be/Projects/imple.html
- NASA
  URL: http://fi-www.arc.nasa.gov/fia/projects/bayes-group/
- Microsoft
  URL: http://www.research.microsoft.com/research/dtg/
- Monash University
  URL: http://www.cs.monash.edu.au/~dld/

5.2.2 Kirjallisuutta

Bayes-verkkokirjallisuuden klassikkona pidetään Pearlin vuonna 1988 julkaisemaa kirjaa [105], joka vielä hieman vanhentuneenakin sopii hyväksi johdatukseksi Bayes-verkkoteoriaan. Pearlin lähestymistapa perustuu pitkälti verkkoteoreettisiin käsitteisiin, kun taas lähes samaan aikaan julkaistu Neapolitanin oppikirja [97] tarjoaa matemaattisesti hieman pragmaattisemman, asiantuntijajärjestelmien rakentamiseen tähtäävän yleisesityksen. Charniakin artikkeli [19] antaa selkeän, ei-matemaattisen johdatuksen Bayes-verkkojen peruskäsitteisiin. Teknisempi lyhyt katsaus aiheeseen löytyy artikkelista [129].

[3] URL: http://bayes.stat.washington.edu/bayes_people.html

Edellä mainitut artikkelit, samoin kuin Pearlin ja Neapolitanin kirjat, ovat hieman vanhentuneita mm. siinä mielessä, että niissä ei juurikaan käsitellä Bayes-verkkojen oppimista. Uudempia johdatuksia Bayes-verkkoteoriaan tarjoavat Spirtesin et al. [130], Jensenin [60], Schaferin [118], ja Castillon et al. [18] kirjat, sekä Heckermanin artikkeli [49]. Spirtesin kirja käsittelee kausaalisia Bayes-verkkoja, ja niiden ei-bayesiläistä oppimista. Jensenin ja Schaferin kirjat keskittyvät kuvaamaan probabilistisen päättelyn toteuttamista Bayes-verkoissa, eivätkä käsittele juuri lainkaan Bayes-verkkojen oppimista. Castillon kirja on selkeää luettavaa, mutta Bayes-verkkojen oppimista käsittelevä luku sisältää joitakin epätarkkuuksia. Heckermanin artikkeli on hyvä yleiskatsaus Bayes-verkkotutkimuksen tuoreimpiin tuloksiin. Lauritzenin [84] ja Whittakerin [141] kirjat käsittelevät yleisemmin graafisiin verkkoesityksiin perustuvan mallintamisen teoriaa.

Bayes-verkkojen oppimiseen keskittyvää materiaalia löytyy lähteistä [50, 48, 16, 17]. Suomenkielisiä katsauksia tutkimusalueeseen löytyy tutkielmista [122, 5]. Yleisemmin bayesiläistä mallinmuodostusta käsitellään lähteissä [22, 88, 96]. Näistä Mackay [88] ja Neal [96] keskittyvät bayesiläisen lähestymistavan soveltamiseen neuroverkkojen oppimisessa. Muita neuroverkkojen ja bayesiläisen mallintamisen välisiä yhteyksiä käsitteleviä teoksia ovat [13, 14, 93, 117].

Teoreettisempia katsauksia todennäköisyyslaskennan soveltamistekniikoihin löytyy teoksista [41, 10, 59]. Hyviä johdatuksia päätösteoriaan löytyy lähteistä [32, 9, 38, 86]. Epätäsmällisyyden käsitettä yleisemmin käsitellään teoksissa [119, 102, 138]. Todennäköisyyslaskennan ja sumean logiikan suhdetta tarkastellaan artikkeleissa [21, 20].

Lisää viitteitä Bayes-verkkoihin liittyviin teoksiin löytyy seuraavilta WWW-sivuilta:

- URL: http://www.cs.pitt.edu/~tsamard/bnpointers.html
- URL: http://www.cs.cmu.edu/afs/cs.cmu.edu/user/ldc/WWW/bayes-net-research/bayes-net-research.html
- URL: http://www.Ultimode.com/~wray/graphbib/
- URL: http://www.lis.pitt.edu/~dsl/da-books.html

Elektronisesti luettavia (HTML-formaatissa olevia) johdatuksia Bayes-verkkoteoriaan löytyy mm. seuraavilta WWW-sivuilta:

- URL: http://www.afit.af.mil/Schools/EN/ENG/LABS/AI/BayesianNetworks/

- URL: http://www.cm.cf.ac.uk/Dave/AI2/AI_notes.html
- URL: http://www.research.att.com/~dmac/notes/bayesian.html
- URL: http://yoda.cis.temple.edu:8080/UGAIWWW/lectures97/uncertainty/bindex.html

Lisäksi useiden Bayes-verkkotutoriaalien/kurssien kalvot ovat myös saatavilla elektronisessa muodossa. Näitä löytyy mm. seuraavilta WWW-sivuilta:

- URL: http://www.cs.Helsinki.FI/~myllymak
- URL: http://www.cs.Helsinki.FI/~tirri
- URL: http://www.research.microsoft.com/research/dtg/heckerma/heckerma.html
- URL: http://www.ultimode.COM/~wray/
- URL: http://www.auai.org/auai-tutes.html

5.2.3 Konferensseja ja lehtiä

Yksinomaan Bayes-verkkotutkimukseen keskittynyttä jokavuotista tieteellistä konferenssia ei toistaiseksi ole perustettu, joskin *Uncertainty in Artificial Intelligence (UAI)*-konferenssit ovat käytännössä muodostuneet sellaiseksi — lähes kaikki konferenssissa nykyään julkaistavat artikkelit käsittelevät Bayes-verkkoja. Bayesiläiseen mallinnukseen yleisesti liittyviä asioita käsitteleviä konferensseja ovat mm. *International Workshops on Artificial Intelligence and Statistics*, *Valencia International Meetings on Bayesian Statistics*, *International FLAIRS Conferences (Special Track on Uncertain Reasoning)*, ja *International Workshops on Maximum Entropy and Bayesian Methods.* Koska bayesiläinen mallinnus liittyy läheisesti koneoppimiseen ja muihin tekoälyn osa-alueisiim, julkaistaan alueeseen liittyviä artikkeleita paljon mm. seuraavissa konferensseissa: *Knowledge Discovery and Data mining (KDD)*, *International Conferences on Machine Learning (ICML)*, *International Joint Conferences on Artificial Intelligence (IJCAI)*, ja *National Conferences on Artificial Intelligence (AAAI).* Neuroverkkokonferensseista ovat jokavuotiset *Neural Information Processing Systems (NIPS)*-konferenssit julkaisseet yhä kasvavassa määrin bayesiläiseen mallinnukseen liittyviä artikkeleita. Eurooppalaiset epätäsmällisen tiedon käsittelyyn liittyvät konferenssit, kuten esimerkiksi *International Joint Conferences on Qualitative and Quantitative Practical Reasoning (ECSQARU)*, *European Symposiums on Intelligent*

Techniques, ja *The European Congresses on Intelligent Techniques and Soft Computing (EUFIT)*, ovat keskittyneet pääasiassa ei-bayesiläisten lähestymistapojen käsittelemiseen.

Bayes-verkkotutkijat eivät toistaiseksi ole perustaneet omaa, Bayes-verkkotutkimukselle omistautunutta tieteellistä julkaisua, vaan alan tutkimustulokset ilmestyvät monissa eri tekoälyn osa-alueita käsittelevissä lehdissä, joita ovat mm. *Artificial Intelligence*, *Journal of Artificial Intelligence Research*, *Journal of Approximate Reasoning*, ja *Machine Learning*.

5.2.4 Järjestöjä

Koska edellä mainittu UAI-konferenssi on muodostunut Bayes-verkkotutkimusalueen tärkeimmäksi vuosittaiseksi tapaamiseksi, on konferenssin järjestävästä *Association for Uncertainty in Artificial Intelligence (AUAI)*-organisaatiosta luonnollisesti muovautunut eräänlainen Bayes-verkkotutkijoiden kattojärjestö. Muita merkittäviä alan tutkimukseen liittyviä järjestöjä ovat *International Society for Bayesian Analysis (ISBA)*, joka sponsoroi *Valencia International Meeting on Bayesian Statistics*- ja *International Workshop on Maximum Entropy and Bayesian Methods*-kokouksia, sekä *Society for Artificial Intelligence and Statistics*, joka järjestää joka toinen vuosi pidettävän *International Workshop on Artificial Intelligence and Statistics*-kokouksen. Muita tutkimusalueeseen liittyviä järjestöjä on lueteltu alla. Eurooppalaisista organisaatioista on huomautettava, että kuten eurooppalaiset alan konferenssit, myös tieteelliset organisaatiot (kuten esim. ERUDIT, NeuroCOLT, ja MLnet) ovat keskittyneet pääasiassa ei-bayesiläisiin epätäsmällisyyden käsittelytapoihin.

- Association for Uncertainty in Artificial Intelligence (AUAI),
  URL: http://www.auai.org/
- Society for Artificial Intelligence and Statistics,
  URL: http://www.vuse.vanderbilt.edu/~dfisher/ai-stats/society.html
- International Society for Bayesian Analysis (ISBA)
  URL: http://omega.albany.edu:8008/isba/
- ASA Section on Bayesian Statistical Sciences
  URL: http://www.isds.duke.edu/sbss/sbss.html
- European Association for Decision Making
  URL: http://huizen.dds.nl/~eadm/

- Decision Analysis Society
  URL: http://www.fuqua.duke.edu/faculty/daweb/
- Society for Risk Analysis (SRA)
  URL: http://www.sra.org/
- Classification Society of North America (CSNA)
  URL: http://www.pitt.edu/~csna/
- Network of Excellence for Uncertainty Modeling and Fuzzy Technology in the European Union (ERUDIT)
  URL: http://ss-m3.mitgmbh.de/erudit/
- ESPRIT Working Group in Neural and Computational Learning (NeuroCOLT)
  URL: http://www.cs.rhbnc.ac.uk/research/compint/neurocolt/
- ESPRIT Network of Excellence in Machine Learning (MLnet)
  URL: http://omega.gmd.de/ml-archive/MLnet/MLnet.html

Kirjallisuutta

[1] Aha, D., Ed. *Lazy Learning*. Kluwer Academic Publishers, Dordrecht, 1997. Reprinted from Artificial Intelligence Review, 11:1–5.

[2] Aha, D., Kibler, D., and Albert, M. Instance-based learning algorithms. *Machine Learning 6* (1991), 37–66.

[3] Alag, S., and Agogino, A. Inference using message propagation and topology transformation in vector gaussian continuous networks. In *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI–96)* (Portland, Oregon, 1996), pp. 20–27.

[4] Amari, S., Murata, N., Müller, K.-R., Finker, M., and Hua Yang, H. Asymptotic statistical theory of overtraining and cross-validation. *IEEE Transactions on Neural Networks 8*, 5 (February 1997), 985–996.

[5] Antti-Poika, T. Bayes-verkkojen käyttö luokittelussa. Master's thesis, Deparment of Computer Science, University of Helsinki, 1997.

[6] Baxter, R., and Oliver, J. MDL and MML: Similarities and differences. Tech. Rep. 207, Department of Computer Science, Monash University, 1994.

[7] Becker, A., and Geiger, D. A sufficiently fast algorithm for finding close to optimal junction trees. In *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI–96)* (Portland, Oregon, 1996), pp. 81–89.

[8] Beinlich, I., Suermondt, H., Chavez, R., and Cooper, G. The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. In *Proceedings of the Second European Conf. on Artificial Intelligence in Medicine* (London, UK, August 1989), pp. 247–256.

[9] Berger, J. *Statistical Decision Theory and Bayesian Analysis.* Springer-Verlag, New York, 1985.

[10] Bernardo, J., and Smith, A. *Bayesian theory.* John Wiley, 1994.

[11] Berrou, G., Glavieux, A., and Thitimajshima, P. Near Shannon limit error-correcting coding: Turbo codes. In *Proceedings of the 1993 International Conference in Communications* (geneve, May 1993), pp. 1064–1070.

[12] Bishop, C. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.

[13] Bishop, C., Svensén, M., and Williams, C. EM optimization of latent-variable density models. In *Advances in Neural Information Processing Systems 8*, D. Touretzky, M.C.Mozer, and M.E.Hasselmo, Eds. MIT Press, 1996.

[14] Bishop, C., Svensén, M., and Williams, C. GTM: A principled alternative to the self-organizing map. In *Advances in Neural Information Processing Systems 9 (to appear).* MIT Press, 1997.

[15] Breiman, L. Bias, variance, and arcing classifiers. Tech. Rep. TR 460, University of California, Statistics Department, April 1996.

[16] Buntine, W. Operations for learning with graphical models. *Journal of Artificial Intelligence Research 2* (1994), 159–225.

[17] Buntine, W. A guide to the literature on learning graphical models. *IEEE Transactions on Knowledge and Data Engineering 8* (1996), 195–210.

[18] Castillo, E., Gutiérrez, J., and Hadi, A. *Expert Systems and Probabilistic Network Models.* Monographs in Computer Science. Springer-Verlag, New York, NY, 1997.

[19] Charniak, E. Bayesian networks without tears. *AI Magazine 12*, 4 (1991), 50–63.

[20] Cheeseman, P. Probabilistic versus fuzzy reasoning. In Kanal and Lemmer [63], pp. 85–102.

[21] Cheeseman, P. In defense of probability. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence* (1995), vol. 2, pp. 1002–1009.

[22] Cheeseman, P. On Bayesian model selection. In *The Mathematics of Generalization*, D. Wolpert, Ed., vol. XX of *SFI Studies in the Sciences of Complexity*. Addison-Wesley, 1995, pp. 315–330.

[23] Cheeseman, P., Kanefsky, B., Hanson, R., and Stutz, J. Super-resolved surface reconstruction from multiple images. Tech. Rep. FIA-94-12, NASA Ames Research Center, Artificial Intelligence Branch, October 1994.

[24] Cheeseman, P., and Stutz, J. Bayesian classification (AutoClass): Theory and results. In *Advances in Knowledge Discovery and Data Mining*, U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds. AAAI Press, Menlo Park, 1996, ch. 6.

[25] Chickering, D., Geiger, D., and Heckerman, D. Learning Bayesian networks is NP-hard. Tech. Rep. MSR-TR-94-17, Microsoft Research, 1994.

[26] Chickering, D., and Heckerman, D. Efficient approximations for the marginal likelihood of incomplete data given a Bayesian network. In *Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence* (Portland, Oregon, August 1996), E. Horvitz and F. Jensen, Eds., Morgan Kaufmann Publishers, pp. 158–168.

[27] Cooper, G. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence 42*, 2–3 (March 1990), 393–405.

[28] Cooper, G., and Herskovits, E. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning 9* (1992), 309–347.

[29] Dagum, P., and Luby, M. Approximating probabilistic inference in Bayesian belief networks is NP-hard. *Artificial Intelligence 60* (1993), 141–153.

[30] Dean, T., and Wellman, M. *Planning and Control*. Morgan Kaufmann Publishers, San Mateo, CA, 1991.

[31] Dechter, R., and Rish, I. A scheme for approximating probabilistic inference. In *Proceedings of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–97)* (San Francisco, CA, 1997), Morgan Kaufmann Publishers, pp. 132–141.

[32] DeGroot, M. *Optimal statistical decisions.* McGraw-Hill, 1970.

[33] Dempster, A., Laird, N., and Rubin, D. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B 39*, 1 (1977), 1–38.

[34] Druzdzel, M., and Henrion, M. Efficient reasoning in qualitative probabilistic networks. In *Proceedings of the Eleventh National Conference on Artificial Intelligence* (Washington, DC, July 1993), AAAI Press/MIT, Menlo Park, CA, pp. 548–553.

[35] Everitt, B., and Hand, D. *Finite Mixture Distributions.* Chapman and Hall, London, 1981.

[36] Friedman, J. On bias, variance, 0/1–loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery 1*, 1 (1997), 55–78.

[37] Friedman, N., and Goldszmidt, M. Discretizing continuous attributes while learning Bayesian networks. In *Machine Learning: Proceedings of the Thirteenth International Conference* (1996), L. Saitta, Ed., Morgan Kaufmann Publishers, pp. 157–165.

[38] Gärdenfors, P., and Sahlin, N.-E., Eds. *Decision, Probability, and Utility.* Cambridge University Press, New York, 1988.

[39] Geiger, D., and Heckerman, D. A characterization of the Dirichlet distribution through global and local independence. Tech. Rep. MSR-TR-94-16, Microsoft Research, November (revised February 1995) 1994.

[40] Geiger, D., and Heckerman, D. Learning bayesian networks. Tech. Rep. MSR-TR-95-02, Microsoft Research, December 1994.

[41] Gelman, A., Carlin, J., Stern, H., and Rubin, D. *Bayesian Data Analysis.* Chapman & Hall, 1995.

[42] Gong, S., and Buxton, H. Bayesian nets for mapping contextual knowledge to computational constraints in motion segmentation and tracking. In *British machine vision conference* (Guildford, Surrey, UK, September 1993).

[43] Gyllenberg, M., Koski, T., and Verlaan, M. Classification of binary vectors by stochastic complexity. Tech. Rep. A5, University of Turku, Institute for Applied Mathematics, November 1994.

[44] Hassoun, M. *Fundamentals of Artificial Neural Networks.* MIT Press, Cambridge, Massachusetts, 1995.

[45] Hecht-Nielsen, R. *Neurocomputing.* Addison-Wesley Publishing Company, Reading, MA, 1990.

[46] Heckerman, D. Probabilistic interpretation for MYCIN's certainty factors. In Kanal and Lemmer [63], pp. 167–196.

[47] Heckerman, D. A Bayesian approach to learning causal network. In *Uncertainty in Artificial Intelligence: Proceedings of the Eleventh Conference* (Montreal, Canada, 1995), P. Besnard and S. Hanks, Eds., pp. 285–295.

[48] Heckerman, D. A tutorial on learning with Bayesian networks. Tech. Rep. MSR-TR-95-06, Microsoft Research, Advanced Technology Division, One Microsoft Way, Redmond, WA 98052, 1996.

[49] Heckerman, D. Bayesian networks for data mining. *Data Mining and Knowledge Discovery 1*, 1 (1997), 79–119.

[50] Heckerman, D., and Geiger, D., and Chickering, D. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning 20*, 3 (September 1995), 197–243.

[51] Heckerman, D., and Nathwani, B. An evaluation of the diagnostic accuracy of Pathfinder. *Computers and Biomedical Research 25* (1992), 56–74.

[52] Heckerman, D., and Schachter, R. Decision-theoretic foundations for causal reasoning. *Journal of Artificial Intelligence Research 3* (1995), 405–430.

[53] Heckerman, D., and Shwe, M. Diagnosis of multiple faults: A sensitivity analysis. In *Uncertainty in Artificial Intelligence 9*, D. Heckerman and A. Mamdani, Eds. Morgan Kaufmann Publishers, San Mateo, CA, 1993, pp. 80–87.

[54] Helm, L. The future of software may lie in the obscure theories of an 18th century cleric called Thomas Bayes. *Los Angeles Times* (Monday, October 28 1996).

[55] Henrion, M. Uncertainty in artificial intelligence: Is probability epistemologically and heuristically adequate? In *Expert Judgements and*

Expert Systems, J. Mumpower, L. Philipps, O. Renn, and U. V., Eds., NATO ASI Series F: Computer and Systems Science. Springer-Verlar, Berlin, 1987, pp. 106–129.

[56] HENRION, M. An introduction to algorithms for inference in belief nets. In *Uncertainty in Artificial Intelligence 5*, M. Henrion, R. Shachter, L. Kanal, and J. Lemmer, Eds. Elsevier Science Publishers B.V. (North-Holland), Amsterdam, 1990, pp. 129–138.

[57] HENRION, M., PRADHAN, M., DEL FAVERO, B., HUANG, K., PROVAN, G., AND O'RORKE, P. Why is diagnosis using belief networks insensitive to imprecision in probabilities? In *Uncertainty in Artificial Intelligence, Proceedings of the Twelfth Conference* (Portland, Oregon, August 1996), E. Horvits and F. Jensen, Eds., Morgan Kaufmann Publishers, San Francisco, CA, pp. 307–314.

[58] ISOMURSU, P., NISKANEN, V., CARLSSON, C., AND EKLUND, P. Sumean logiikan mahdollisuudet. Tech. Rep. 34/93, Teknologian kehittämiskeskus (TEKES), 1993.

[59] JAYNES, E. Probability theory: The logic of science. Under construction. A draft copy can be dowloaded from 'ftp://bayes.wustl.edu/Jaynes.book'., 1996.

[60] JENSEN, F. *An Introduction to Bayesian Networks.* UCL Press, London, 1996.

[61] JENSEN, F., CHRISTENSEN, H., AND NIELSEN, J. Bayesian methods for interpretation and control in multi-agent vision systems. *Applications of Artificial Intelligence X: Machine Vision and Robotics, SPIE Proceedings Series 1708* (1992).

[62] JORDAN, M., AND JACOBS, R. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation 6* (1994), 181–214.

[63] KANAL, L., AND LEMMER, J., Eds. *Uncertainty in Artificial Intelligence 1.* Elsevier Science Publishers B.V. (North-Holland), Amsterdam, 1986.

[64] KANEFSKY, B., STUTZ, J., CHEESEMAN, P., AND TAYLOR, W. An improved automatic classification of a Landsat/TM image from Kansas (FIFE). Tech. Rep. FIA-94-01, NASA Ames Research Center, Artificial Intelligence Branch, May 1994.

[65] Kass, R., and Raftery, A. Bayes factors. Tech. Rep. 254, Department of Statistics, University of Washington, 1994.

[66] Kjærulff, U. Optimal decomposition of probabilistic networks by simulated annealing. *Statistics and Computing 2* (1992), 7–17.

[67] Kohonen, T. *Self-Organizing Maps.* Springer-Verlag, Berlin, 1995.

[68] Kolodner, J. *Case-Based Reasoning.* Morgan Kaufmann Publishers, San Mateo, 1993.

[69] Kontkanen, P., Myllymäki, P., Silander, T., and Tirri, H. Bayes optimal lazy learning. Tech. Rep. NC-TR-97-031, ESPRIT Working Group on Neural and Computational Learning (NeuroCOLT), 1997.

[70] Kontkanen, P., Myllymäki, P., Silander, T., and Tirri, H. A Bayesian approach for retrieving relevant cases. In *Artificial Intelligence Applications (Proceedings of the EXPERSYS-97 Conference)* (Sunderland, UK, October 1997), P. Smith, Ed., IITT International, pp. 67–72.

[71] Kontkanen, P., Myllymäki, P., Silander, T., and Tirri, H. A Bayesian approach to discretization. In *Proceedings of the European Symposium on Intelligent Techniques* (Bari, Italy, March 1997), pp. 265–268.

[72] Kontkanen, P., Myllymäki, P., Silander, T., and Tirri, H. Comparing stochastic complexity minimization algorithms in estimating missing data. In *Proceedings of WUPES'97, the 4th Workshop on Uncertainty Processing* (Prague, Czech Republic, January 1997), pp. 81–90.

[73] Kontkanen, P., Myllymäki, P., Silander, T., and Tirri, H. On the accuracy of stochastic complexity approximations. In *Proceedings of the Causal Models and Statistical Learning Seminar* (London, UK, March 1997), pp. 103–117.

[74] Kontkanen, P., Myllymäki, P., Silander, T., Tirri, H., and Grünwald, P. Comparing predictive inference methods for discrete domains. In *Proceedings of the Sixth International Workshop on Artificial Intelligence and Statistics* (Ft. Lauderdale, Florida, January 1997), pp. 311–318.

[75] Kontkanen, P., Myllymäki, P., Silander, T., Tirri, H., and Grünwald, P. On predictive distributions and Bayesian networks. In *Proceedings of the Seventh Belgian-Dutch Conference on Machine Learning (BeNeLearn'97)* (Tilburg, the Netherlands, October 1997), W. Daelemans, P. Flach, and A. van den Bosch, Eds., pp. 59–68.

[76] Kontkanen, P., Myllymäki, P., Silander, T., Tirri, H., and Grünwald, P. On predictive distributions and Bayesian networks. Tech. Rep. NC-TR-97-032, ESPRIT Working Group on Neural and Computational Learning (NeuroCOLT), 1997.

[77] Kontkanen, P., Myllymäki, P., and Tirri, H. Comparing Bayesian model class selection criteria by discrete finite mixtures. In *Information, Statistics and Induction in Science* (Proceedings of the ISIS'96 Conference, Melbourne, Australia, August 1996), D. Dowe, K. Korb, and J. Oliver, Eds., World Scientific, Singapore, pp. 364–374.

[78] Kontkanen, P., Myllymäki, P., and Tirri, H. Constructing Bayesian finite mixture models by the EM algorithm. Tech. Rep. NC-TR-97-003, ESPRIT Working Group on Neural and Computational Learning (NeuroCOLT), 1996.

[79] Kontkanen, P., Myllymäki, P., and Tirri, H. Predictive data mining with finite mixtures. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (Portland, Oregon, August 1996), E. Simoudis, J. Han, and U. Fayyad, Eds., pp. 176–182.

[80] Kontkanen, P., Myllymäki, P., and Tirri, H. Some experimental results with finite mixture models. In *Book of Abstracts, First European Conference on Highly Structured Stochastic Systems* (Rebild, Denmark, May 1996), pp. 112–115.

[81] Kontkanen, P., Myllymäki, P., and Tirri, H. Experimenting with the Cheeseman-Stutz evidence approximation for predictive modeling and data mining. In *Proceedings of the Tenth International FLAIRS Conference* (Daytona Beach, Florida, May 1997), D. Dankel, Ed., pp. 204–211.

[82] Langley, P., Iba, W., and Thompson, K. An analysis of Bayesian classifiers. In *Proceedings of the 10th National Conference on Artificial Intelligence* (San Jose, CA, July 1992), MIT Press, pp. 223–228.

[83] Lauritzen, S. Propagation of probabilities, means and variances in mixed graphical association models. *Journal of the American Statistical Association 87* (1992), 1098–1108.

[84] Lauritzen, S. *Graphical Models.* Oxford University Press, 1996.

[85] Li, Z., and D'Ambrosio, B. Efficient inference in bayes nets as a combinatorial optimization problem. *International Journal of Approximate Reasoning 11*, 1 (1994), 55–81.

[86] Lindley, D. *Making Decisions*, second ed. John Wiley & Sons, London, 1992.

[87] M., W., and Holyoak, K. Predictive and diagnostic learning within causal models: Asymmetries of cue competition. *Journal of Experimental Psychology: General 121* (1992), 222–236.

[88] Mackay, D. *Bayesian Methods for Adaptive Models.* PhD thesis, California Institute of Technology, 1992.

[89] MacKay, D. J. C. Bayesian non-linear modelling for the prediction competition. In *ASHRAE Transactions, V.100, Pt.2* (Atlanta Georgia, 1994), ASHRAE, pp. 1053–1062.

[90] McEliece, R., Mackay, D., and Cheng, J.-F. Turbo decoding as an instance of Pearl's "belief propagation" algorithm. *International Journal of Selected areas in Communication* (1997). (to appear).

[91] McLachlan, G., and Thriyambakam, K., Eds. *The EM Algorithm and Extensions.* John Wiley & Sons, New York, 1997.

[92] Moody, J., and Darken, C. Fast learning in networks of locally-tuned processing units. *Neural Computation 1* (1989), 281–294.

[93] Myllymäki, P. *Mapping Bayesian Networks to Stochastic Neural Networks: A Foundation for Hybrid Bayesian-Neural Systems.* PhD thesis, Report A-1995-1, Department of Computer Science, University of Helsinki, December 1995.

[94] Myllymäki, P., and Tirri, H. Bayesian case-based reasoning with neural networks. In *Proceedings of the IEEE International Conference on Neural Networks* (San Francisco, March 1993), vol. 1, IEEE, Piscataway, NJ, pp. 422–427.

[95] Myllymäki, P., and Tirri, H. Massively parallel case-based reasoning with probabilistic similarity metrics. In *Topics in Case-Based Reasoning*, S. Wess, K.-D. Althoff, and M. Richter, Eds., vol. 837 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 1994, pp. 144–154.

[96] Neal, R. *Bayesian Learning for Neural Networks*. No. 118 in Lecture Notes in Statistics. Springer-Verlag, New York, NY, 1996.

[97] Neapolitan, R. *Probabilistic Reasoning in Expert Systems*. John Wiley & Sons, New York, NY, 1990.

[98] Oliver, J., and Baxter, R. MML and Bayesianism: Similarities and differences. Tech. Rep. 206, Department of Computer Science, Monash University, December 1994.

[99] Oliver, J., and Hand, D. Introduction to minimum encoding inference. Tech. Rep. 205, Department of Computer Science, Monash University, July 1994.

[100] Oliver, J., and Hand, D. On pruning and averaging decision trees. In *Machine Learning: Proceedings of the Twelfth International Conference* (1995), A. Prieditis, Ed., Morgan Kaufmann Publishers, pp. 430–437.

[101] Oliver, R., and Smith, J. *Influence diagrams, belief nets and decision analysis*. John Wiley & Sons, 1990.

[102] Paris, J. *The Uncertain Reasoner's Companion: A Mathematical Prespective*, vol. 39 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, Cambridge, 1994.

[103] Parsa, I. KDD-97 knowledge discovery and data mining tools competition. *Knowledge Discovery Nuggets 97*, 19 (1997). An electronic journal, available at `http://www.kdnuggets.com/`.

[104] Pearl, J. Fusion, propagation and structuring in belief networks. *Artificial Intelligence 29* (1986), 241–288.

[105] Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[106] Pearl, J., and Verma, T. A theory of inferred causation. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference* (1991), A. Allen, R. Fikes, and E. Sandewall, Eds., Morgan Kaufmann Publishers, San Mateo, CA, pp. 441–452.

[107] Poggio, T., and Girosi, F. Networks for approximation and learning. *Proceedings of the IEEE 78*, 9 (1990), 1481–1497.

[108] Poland, W. *Decision Analysis with Continuous and Discrete Variables: A Mixture Distribution Approach.* PhD thesis, Department of Engineering Economic Systems, Stanford University, 1994.

[109] Rissanen, J. Stochastic complexity. *Journal of the Royal Statistical Society 49*, 3 (1987), 223–239 and 252–265.

[110] Rissanen, J. *Stochastic Complexity in Statistical Inquiry.* World Scientific Publishing Company, New Jersey, 1989.

[111] Rissanen, J. Fisher information and stochastic complexity. *IEEE Transactions on Information Theory 42*, 1 (January 1996), 40–47.

[112] Rissanen, J. Information theory and neural nets. In *Mathematical Perspectives on Neural Networks*, P. Smolensky, M. Mozer, and D. Rumelhart, Eds. Lawrence Erlbaum Associates, 1996, ch. 16.

[113] Rissanen, J. Oppivien ja älykkäiden järjestelmien sovellukset 1994–1998, tutkimusprojektien arviointi. Tech. Rep. Teknologiaohjelmaraportti 2/97, Teknologian kehittämiskeskus (TEKES), 1997.

[114] Robinson, R. Counting unlabeled asyclic graphs. In *Combinatorial Mathematics*, C. Little, Ed., no. 622 in Lecture Notes in Mathematics. Springer-Verlag, 1977.

[115] Roth, D. On the hardness of approximate reasoning. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence* (1993), vol. 1, pp. 613–618.

[116] Santos, E., and Shimony, S. Belief updating by enumerating high-probability independence-based arguments. In *Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–94)* (Seattle, Washington, July 1994), R. Lopez de Mantaras and D. Poole, Eds., Morgan Kaufmann Publishers, San Francisco, CA, pp. 506–513.

[117] Saul, L., Jaakkola, T., and Jordan, M. Mean field theory for sigmoid belief networks. *Journal of Artificial Intelligence Research 4* (1996), 61–76.

[118] Schafer, G., Ed. *Probabilistic Expert Systems.* Society for Industrial and Applied mathematics (SIAM), 1996.

[119] Schafer, G., and Pearl, J., Eds. *Readings in Uncertain Reasoning.* Morgan Kaufmann Publishers, San Mateo, CA, 1990.

[120] Schwarz, G. Estimating the dimension of a model. *Annals of Statistics 6* (1978), 461–464.

[121] Scott, D. *Multivariate Density Estimation. Theory, Practice, and Visualization.* John Wiley & Sons, New York, 1992.

[122] Seppänen, M. Bayes-verkon muodostaminen datasta. Master's thesis, Department of Computer Science, University of Helsinki, 1996.

[123] Shachter, R. Probabilistic inference and influence diagrams. *Operations Research 36*, 4 (July-August 1988), 589–604.

[124] Shimony, S. Finding MAPs for belief networks is NP-hard. *Artificial Intelligence 68* (1994), 399–410.

[125] Singh, M., and Valtorta, M. An algorithm for construction of Bayesian network structures from data. In *Uncertainty in Artificial Intelligence, Proceedings of the Ninth Conference* (1993), E. Horvits and F. Jensen, Eds., Morgan Kaufmann Publishers, San Francisco, CA, pp. 259–265.

[126] Smyth, P., Heckerman, D., and Jordan, M. Probabilistic independece networks for hidden Markov probability models. Tech. Rep. A.I. Memo 1565, C.B.C.L Memo 132, M.I.T, February 1996.

[127] Sorsa, M. Kvalitatiiviset bayesverkot. Master's thesis, Department of Computer Science, University of Helsinki, 1996.

[128] Specht, D. Probabilistic neural networks. *Neural Networks 3* (1990), 109–118.

[129] Spiegelhalter, D., Dawid, P., Lauritzen, S., and Cowell, R. Bayesian analysis in expert systems. *Statistical Science 8*, 3 (1993), 219–283.

[130] Spirtes, P., Glymour, C., and Scheines, R., Eds. *Causation, Prediction and Search.* Springer-Verlag, 1993. Out of print. Can be downloaded from 'URL: http://hss.cmu.edu/html/departments/philosophy/TETRAD.BOOK/book.html'.

[131] Stanfill, C., and Waltz, D. Toward memory-based reasoning. *Communications of the ACM 29*, 12 (1986), 1213–1228.

[132] Tikhonov, A. On solving incorrectly posed problems and method of regularization. *Doklady Akademii Nauk USSR 151* (1963), 501–504.

[133] Tirri, H. *Plausible Prediction by Bayesian Inference.* PhD thesis, Report A-1995-1, Department of Computer Science, University of Helsinki, June 1997.

[134] Tirri, H., Kontkanen, P., and Myllymäki, P. A Bayesian framework for case-based reasoning. In *Advances in Case-Based Reasoning*, I. Smith and B. Faltings, Eds., vol. 1168 of *Lecture Notes in Artificial Intelligence.* Springer-Verlag, Berlin Heidelberg, November 1996, pp. 413–427.

[135] Tirri, H., Kontkanen, P., and Myllymäki, P. Probabilistic instance-based learning. In *Machine Learning: Proceedings of the Thirteenth International Conference* (1996), L. Saitta, Ed., Morgan Kaufmann Publishers, pp. 507–515.

[136] Titterington, D., Smith, A., and Makov, U. *Statistical Analysis of Finite Mixture Distributions.* John Wiley & Sons, New York, 1985.

[137] Vapnik, V. *The Nature of Statistical Learning Theory.* Springer-Verlag, New York, 1995.

[138] Walley, P. Measures of uncertainty in expert systems. *Artificial Intelligence 83* (1996), 1–58.

[139] Watson, I., and Marir, F. Case-based reasoning: A review. *The Knowledge Engineering Review 9*, 4 (1994), 327–354.

[140] Wellman, M. Fundamental concepts of qualitative probabilistic networks. *Artificial Intelligence 44*, 3 (August 1990), 257–304.

[141] Whittaker, J. *Graphical Models in Applied Multivariate Statistics.* John Wiley & Sons, 1990.

[142] Yannakis, M. Computing the minimal fill-in is NP-complete. *SIAM Journal of Algebraic Discrete Methods 2* (1981), 77–79.

[143] Zhang, L., and Poole, D. Sidestepping the triangulation problem in Bayesian net computations. In *Proceedings of the Eight Annual Conference on Uncertainty in Artificial Intelligence* (Palo Alto, CA, July 1992), Morgan Kaufmann Publishers, pp. 360–367.