Bayesilaisten verkkojen ja approksimoivan inferenssin matemaattiset perusteet teemojen mallinnuksessa

Markus Viljanen
Tietojenkäsittelytieteen laitos,
University of Turku,
majuvi@utu.fi

16. joulukuuta 2013

Tiivistelmä

Tämä LuK-tutkielma käsittelee todennäköisyyslaskennan käyttöä teemojen mallinnuksessa. Aihetta lähestytään kattavalla tutkimuksella mallinnuksen matemaattisista perusteista, joista rakennetaan kaksi suosittua mallia: NBC ja LDA. Molemmille esitetään täyden johtamisen lisäksi pedagoginen Matlab-toteutus. Tämä tutkimus on hyödyllinen kaikille matemaattisesti suuntautuneille, jotka haluavat ymmärtää teemojen mallinuksen diskreettien generatiivisten mallien kautta bayesilaisesta näkökulmasta.

Sisältö

Luku 1

Johdanto

Tietojenkäsittelytieteen käytännön sovelluksissa esiintyy usein seuraava tilanne: on olemassa valtava kokelma asiakirjoja, ja haluttaisiin automaattisesti suorittaa tälle kokoelmalle toiminpide joka luo siihen yksinkertaisen näkymän. Eräs tapa saada aikaan tälläinen näkymä on käyttää lyhyitä selitteitä kullekkin asiakirjalle. Nämä selitteet muun muassa erottavat asiakirjat toisistaan, mahdollistavat niiden samankaltaisuuden vertailun ja ehdot täyttävien asiakirjojen selaaminen selitteen kautta. Kaikkein yksinkertaisin ja ehkä myös intuitiivisin lähestymistapa on määritellä pieni joukku teemoja, ja sitten antaa kullekkin asiakirjalle yksi tai useampi näistä teemoista. On useita tietokokoelmia joihin tätä menetelmää sovelletaan: uutiset, kolumnit ja blogikirjoitukset, asiakirjojen arkistointi, nettisivujen tietorakenne... Kaikille asiakirjoille ei kuitenkaan ole luotu tälläistä luokittelua, tai voi olla ettei asiakirjojen käyttämä luokittelu tietomäärän suuruuden vuoksi täytä sille asetettuja tarpeita paremman näkymän saamiseen. Tällöin haluttaisiin usein luoda kyseinen luokittelu täysin automaattisesti algoritmin avulla.

Tätä tavoitetta lähestytään tässä tutkielmassa diskreettien generatiivisten mallien ja bayesilaisen todennäköisyyslaskennan kautta. Kun tehdään oletus, että asiakirjaan liittyy tekijöitä jotka jollain tekijälle ominaisella tavalla 'tuottavat' sen sisällön, puhutaan generatiivisesta mallista. Esimerkiksi kolikon reiluuden tai epäreiluuden aste on tekijä jonka voidaan katsoa tuottavan kolikon heitoilla saadun sarjan havaittuja heittoja. Samalla tavoin voidaan tehdä mallinnuksessa hivenen yksinkertainen oletus, että asiakirjan aihe vaikuttaa siihen tapaan, jolla asiakirja on ilmaistavissa pitkänä sarjana sanoja. Todennäköisyyslaskenta mahdollistaa, että havainnoista voimme päätellä tämän taustalla olevan tekijän todennäköisyyden. Toisin sanoen, jos teema on kuvattavissa tälläisenä mallina, voimme oppia piilevän teeman, tai mahdollisesti useita teemoja.

Tavoitteen toteuttavista malleista on saatavilla runsaasti kirjallisuutta. Kaikken yksinkertaisimmasta mallista (NBC) on saatavilla useita helppolukuisia selityksiä, ja hieman monimutkaisimmista on olemassa useita tiivistettyjä kuvauksia alan oppikirjoissa. Toiset mallit (LDA) on kuvattu seikkaperäisemmin lähinnä tieteellisissä artikkeleissa. Korkeatasoiset julkaisut olettavat lukijan tietävän matemaattiset periaatteet ja suhteet muihin malleihin, eikä niitä siksi usein esitellä. Tämän vuoksi keskityin tutkimaan perusteita, joita käyttäen pystytään sekä ymmärtämään että rakentamaan mikä tahansa aihetta koskevista malleista.

Luku 2

Bayesilainen oppiminen diskreetille informaatiolle

2.1 Generatiiviseen malliin perustuva inferenssi

Koneoppimisessa on usein tehtävänä muodostaa olemassa olevasta informaatiosta sitä edustava malli, joka sisältää informaatiota koskevat tärkeimmät piirteet. Hyvä malli on yksinkertainen, yleistyvä ja sitä voi käyttää ennustamaan informaatiota tulevaisuudessa. Eräs tapa lähestyä mallin rakentamisen ongelmaa on generatiivinen malli: malli tuottaa informaatiota jollakin määritetyllä tavalla. Oppiminen tarkoittaa mallin tuntemattomien parametrien päättelyä havaitusta informaatiosta. Todennäköisyyslaskentaa hyödyntävässä lähestymistavassa usein keskitytään vielä itse parametrien todennäköisyysjakauman laskemiseen.

Oletetaan esimerkiksi että malli määrittelee $p(D|\theta)$, siis todennäköisyyden jolla informaatio D tuotetaan parametreilla θ. Silloin voidaan käyttää Bayesin sääntöä:[1]

$$p(\theta|D) = \frac{p(D,\theta)}{p(D)} = \frac{p(D|\theta)p(\theta)}{\sum_{\theta'} p(\theta', D)} = \frac{p(D|\theta)p(\theta)}{\sum_{\theta'} p(D|\theta')p(\theta')} \tag{2.1}$$

Tämä malli mahdollistaa tuntemattomien parametrien $\widehat{\theta}$ päättelyn laskemalla millä θ lauseke saa maksimin:

$$\widehat{\theta} = \text{argmax}_{\theta} p(\theta|D) \tag{2.2}$$

Nimittäjä lausekkeessa (2.1) on vakio, eikä riipu muuttujan θ arvosta. Tämä vakio normalisoi todennäköisyydet niin että niiden summa on 1. Siksi lause

yleensä kirjoitetaan ilman normalisaatiovakiota:[1]

$$p(\theta|D) \propto p(D|\theta)p(\theta)$$

Parametrien $\widehat{\theta}$ laskemiseksi sitä ei tarvitse tietää, sillä lauseesta (2.2) ja verrannollisuuden määritelmästä seuraa:

$$\widehat{\theta} = \text{argmax}_\theta p(\theta|D) = \text{argmax}_\theta p(D|\theta)p(\theta)$$

Tätä kutsutaan MAP-arvioksi (*Maximum A Posteriori*) ja $p(D|\theta)$ on posteriori-jakauma. $p(D|\theta)$ määriteltiin mallin yhteydessä, mutta entä $p(\theta)$?

$p(\theta)$ on priori-jakauma. Se sisältää tiedon parametrin θ jakaumasta ennen informaation havaitsemista. Jos parametristä ei ole mitään tietoa, voidaan käyttää tasaista todennäköisyysjakaumaa parametrin θ määrittelyalueella. Tämä tekee kaikista arvoista yhtä todennäköisiä, ja on yleensä järkevä valinta jos todellakin mitään parametrista tai sen luonteesta ei tiedetä. Tällöin jakauman arvon määrittää posteriori-jakauma, sillä silloin $p(\theta|D) \propto p(D|\theta)$. Tätä nimitetään MLE-arvioksi (*Maximum Likelihood Estimate*). Monilla sovellusalueilla hyödytään ihmisen määrittelemästä priorista, mutta tämä lähestymistapa on koneoppimisen kannalta hivenen kyseenalainen. Priorista riippumatta kun informaation määrä kasvaa, lausekkeen arvo riippuu yhä enemmän osasta $p(D|\theta)$, sillä priori ei muutu. Tällöin MAP-arvio lähestyy MLE-arvioita.[1]

Kun parametrien arvo on opittu, on helppoa käyttää mallia tulevan informaation X arvioimiseen. Bayesilainen tapa on marginalisoida θ: [1]

$$p(X|\theta) = \sum_\theta p(X|\theta)p(\theta|D)$$

Tai vaihtoehtoisesti ei-disktreetille muuttujalle saadaan:

$$p(X|\theta) = \int_\theta p(X|\theta)p(\theta|D)$$

Viimeisenä todettakoon, että mallissa $p(\theta|D_1, D_2) \propto p(D_2|\theta)p(\theta|D_1)$. Parametrien bayesilainen analyysi sopii siis hyvin reaaliaikaiseen oppimiseen, jossa ensin opitaan θ joka perustuu ensimmäiseen informaation D_1, jota käytetään priorina seuraavasta informaatiosta D_2:sta oppimiseen.

2.2 Beta-binomial malli ja kolikonheitto

Käsitellään seuraavaksi yksinkertaista esimerkkiä joka auttaa hahmottamaan edellä esitettyjä käsitteitä. Kuvitellaan että havaitaan sarja kolikonheittoja

ja niistä tulee päätellä kolikon reiluuden aste. Tätä yksinkertaista esimerkkiä käytetään usein kirjallisuudessa ja siinä myös mainitaan, että Bayesin alkuperäinen artikkeli käytti samaa esimerkkiä.[1][6]

Tässä tapauksessa generatiivisen mallin määrittely on yksinkertaista:

$$p(D|\theta) = \theta^{N_1}(1-\theta)^{N_0}$$

Missä N_1 on kruunien ja N_0 klaavojen määrä. $N = N_1 + N_0$ on vastaavasti heittojen määrä. Jos informaatiota ei määritellä jonona heittoja $D = (1, 0, 0, 1, \ldots)$, vaan havaitaan kruunien N_1 lukumäärä, saadaan Binomi-jakauma:

$$p(N_1|N, \theta) = \binom{N}{N_1}\theta^{N_1}(1-\theta)^{N-N_1}$$

Priorin $p(\theta)$ määrittelyssä on muistettava että $p(\theta) = 0$ kun $\theta < 0$ tai $\theta > 1$. Tämän lisäksi olisi suotavaa mikäli priori olisi muodossa $p(\theta) = \theta^{X_1}(1-\theta)^{X_0}$ sillä tällöin $p(\theta|D) \propto \theta^{N_1+X_1}(1-\theta)^{N_0+X_0}$. Prioria jolla on sama muoto kuin sen ja MLE:n yhdistelmästä seuraavalla posteriorilla kutsutaan konjukaatti-prioriksi. Se mahdollistaa mallin yksinkertaistamisen sallimalla posteriorin täsmällisen lauseen algebraillisen johtamisen.

Bernoulli-jakauman konjukaatti-priori on Beta-jakauma: [1]

$$\text{Beta}(\theta|\alpha, \beta) = \frac{1}{\text{B}(\alpha, \beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}$$

Beta-jakauman ominaisuudet ovat: [6]

$$\text{mean}[\theta] = \frac{\alpha}{\alpha+\beta} \tag{2.3}$$

$$\text{mode}[\theta] = \frac{\alpha-1}{\alpha+\beta-2} \tag{2.4}$$

$$\text{var}[\theta] = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} \tag{2.5}$$

Tässä parametreja α ja β kutsutaan hyperparametreiksi. Yhdistämällä nämä jakaumat lopulliseksi lausekkeeksi saadaan:

$$\begin{aligned} p(\theta|N_1, N) &\propto p(N_1|\theta, N)p(\theta) \\ &\propto p(N_1|\theta, N)\text{Beta}(\theta|\alpha, \beta) \\ &\propto \theta^{N_1+\alpha-1}(1-\theta)^{N_0+\beta-1} \\ &= \text{Beta}(\theta|N_1+\alpha, N_0+\beta) \end{aligned}$$

Beta-jakauman ominaisuuksia (2.3),(2.4),(2.5) hyödyntämällä saadaan:

$$\widehat{\theta}_{\text{MAP}} = \frac{N_1 + \alpha - 1}{N + \alpha + \beta - 2} \tag{2.6}$$

$$E[\theta|D] = \frac{N_1 + \alpha}{N + \alpha + \beta} \tag{2.7}$$

$$\text{Var}[\theta|D] = \frac{(N_1 + \alpha)(N_0 + \beta)}{(N + \alpha + \beta)^2(N + \alpha + \beta + 1)} \tag{2.8}$$

Seuraavan heiton tulos on helppo ennustaa:

$$\begin{aligned} p(X = 1|D) &= \int_0^1 p(X = 1|\theta)p(\theta|D)\mathrm{d}\theta \\ &= \int_0^1 \theta \text{Beta}(\theta|N_1 + \alpha, N_0 + \beta)\mathrm{d}\theta \\ &= \frac{N_1 + \alpha}{N + \alpha + \beta} \end{aligned}$$

Jos käytetään tasaista prioria, MAP-arviosta tulee MLE-arvio. Oletetaan ettei muuttujaa θ marginalisoida kuten yllä. Tässä tapauksessa:

$$\widehat{\theta}_{\text{MAP}} = \frac{N_1 + 1 - 1}{N + 1 + 1 - 2} = \frac{N_1}{N}$$

ja vastavasti:

$$p(X = 1|\widehat{\theta}_{\text{MAP}}) = \frac{N_1}{N}$$

Jos tässä tapauksessa ei ole havaittu yhtään kruunaa, on todennäköisyys saada kruuna 0. Tapahtuman tulkitaan olevan mahdoton. Tätä voitaneen pitää suurena ylisovittamisena. Sen sijaan bayesilainen menetelmä antaa vastaukseksi $\frac{1}{N+2}$, mikä on varsin usein käytetty *add-to-one smoothing* regularisaatiotapa.[1] Oikein määritelty bayesilainen paradigma siis estää ylisovittamisen.

Vastaavasti voidaan ennustaa seuraavien useiden heittojen tulos:

$$\begin{aligned} p(M_1|D, M) &= \int_0^1 \text{Bin}(M_1, \theta, M)\text{Beta}(\theta|\alpha, \beta) \\ &= \binom{M}{M_1} \frac{1}{\text{B}(\alpha, \beta)} \int_0^1 \theta^{M_1+\alpha-1}(1 - \theta)^{M_0+\beta-1} \end{aligned}$$

Integraali on normalisaatiovakio vastaavalle Beta-jakaumalle, joten:

$$p(M_1|D, M) = \binom{M}{M_1} \frac{\text{B}(M_1 + \alpha, M_0 + \beta)}{\text{B}(\alpha, \beta)}$$

mitä kutsutaan Beta-Binomi-jakaumaksi. [1]

2.3 Dirichlet-multinomial malli

Seuraavassa esitetään lyhyesti ja semanttisesti täysin vastaavasti Dirichlet-multinomial malli, jota käytetään myöhemmin useissa yhteyksissä. Oletetaan että halutaan mallintaa tapahtumia joista jokaisella on useampi kuin kaksi mahdollista arvoa. Esimerkkinä käy sarja nopan heittoja $D = (x_1, x_2, \ldots, x_N)$ missä $x_i = 1 \ldots K$. Sarjalle saadaan seuraava Multinomial-jakauma: [1][6]

$$p(D|\overline{\theta}) = \prod_{i=1}^{K} \theta_i^{N_i} = \mathrm{Mu}(D|\overline{\theta}, n)$$

missä $N_i = \sum_j \mathrm{I}(x_j = i)$, $\overline{\theta} = (\theta_1, \ldots, \theta_K)$ ja $\mathrm{I}(T)$ on indikaattorifunktio, jonka arvo on 1 kun lause T on tosi, ja 0 kun lause T on epätosi.

Konjukaatti-priori on Dirichlet-jakauma: [1]

$$p(\overline{\theta}) = \frac{1}{\mathrm{B}(\overline{\alpha})} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} I(\overline{\theta} \in \mathrm{S}) = \mathrm{Dir}(\overline{\theta}|\overline{\alpha})$$

Posteriori on:

$$p(\overline{\theta}|D) \propto p(D|\overline{\theta}) p(\overline{\theta}) = \frac{1}{\mathrm{B}(\overline{\alpha})} \prod_{i=1}^{K} \theta_i^{N_i + \alpha_i - 1} I(\overline{\theta} \in \mathrm{S}) \propto \mathrm{Dir}(\overline{\theta}|\overline{N} + \overline{\alpha})$$

Jakaumaan pätevät seuraavat ominaisuudet: [6]

$$E[\theta_k|D] = \frac{N_k + \alpha_k}{N + \alpha_0} \tag{2.9}$$

$$\mathrm{mode}[\theta_k] = \frac{N_k + \alpha_k - 1}{N + \alpha_0 - K} \tag{2.10}$$

Luku 3

Bayesilaiset verkot

3.1 Todennäköisyysjakauman esittäminen graafisella mallilla

Oletetaan että mallissa on n satunnaismuuttujaa $X_1, X_2, \ldots, X_n$, joiden arvoja havaitsemme. Tehtävänä on esittää muuttujen todennäköisyysjakauma $p(\overline{x}) = p(X_1 = x_1, \ldots, X_n = x_n)$, käyttää tätä jakaumaa muuttujien parametrien päättelyyn ja tulevaisuudessa havaittavien arvojen ennustamiseen. Oletetaan että jokaisella muuttujalla on K mahdollista tilaa. Silloin todennäköisyysjakauma $p(\overline{x})$ on esitettävissä $O(K^n)$ eri arvon avulla.

Bayesin säännöstä seuraa, että $P(A, B) = P(A|B)p(B)$. Kun tätä kaavaa käytetään toistamiseen, saadaan ketjutussääntö:[6]

$$\begin{aligned} p(X_1, \ldots, X_n) &= p(X_1)p(X_n, \ldots, X_2|X_1) \\ &= p(X_1)p(X_2|X_1)p(X_n, \ldots, X_3|X_2, X_1) \\ &= p(X_1)p(X_2|X_1)p(X_3|X_2, X_1)p(X_n|X_{n-1}, \ldots, X_1) \end{aligned} \tag{3.1}$$

Jos $p(X_i|X_{i-1}, \ldots, X_1)$ kirjoitetaan muuttujien $X_1, \ldots, X_{i-1}$ ehdollisena todennäköisyystaulukkona (CPT), siinä on $O(K * K^{i-1}) = O(K^i)$ arvoa. Siksi ylläoleva täysi todennäköisyysjakauma on ilmaistavissa edelleen $O(K^n)$ arvolla.

Oletetaan seuraavaksi että voidaan tehdä satunnaismuuttujia koskevia riippumattomuuslauseita. Sanotaan esimerkiksi että muuttuja X on ehdollisesti riippumaton muuttujista $Y_1, ..., Y_k$ jos $Z_1, ..., Z_l$:

$$x \perp \overline{y}|\overline{z} \Leftrightarrow p(x, \overline{y}|\overline{z}) = p(x|\overline{z})p(\overline{y}|\overline{z})$$

Tästä seuraa:

$$p(x|\overline{y}, \overline{z}) = \frac{p(x, \overline{y}|\overline{z})}{p(\overline{y}|\overline{z})} = \frac{p(x|\overline{z})p(\overline{y}|\overline{z})}{p(\overline{y}|\overline{z})} = p(x|\overline{z}) \tag{3.2}$$

Avaintekijä ns. suunnatuissa syklittömissä graffeissa *(DAG: Directed Acyclic Graph)* on mahdollisuus järjestää ne niin että jokaisen solmun jälkeläiset tulevat järjestyksessä tämän solmun jälkeen. [1][5] Tätä kutsutaan topologiseksi järjestykseksi. Toisin sanoen jos solmuille on määritelty järjestys $(X_1, X_2, \ldots, X_n)$, mikäli X_i on X_j:n jälkeläinen niin $i > j$. Nimitetään ylläolevia satunnaismuuttujia $\overline{y}$ edeltäjiksi ja satunnaismuuttujia $\overline{z}$ vanhemmiksi, ts. sanotaan että muuttuja x riippuu vain vanhemmistaan $\overline{x}_{\mathrm{pa}}$, ei muista edeltäjistään $\overline{x}_{\mathrm{pred}(i)\setminus\mathrm{pa(i)}}$. Tätä kutsutaan järjestetyksi Markov-ominaisuudeksi.[5]

$$x_i \perp \overline{x}_{\mathrm{pred}(i)\setminus\mathrm{pa(i)}} | \overline{x}_{\mathrm{pa}(i)}$$

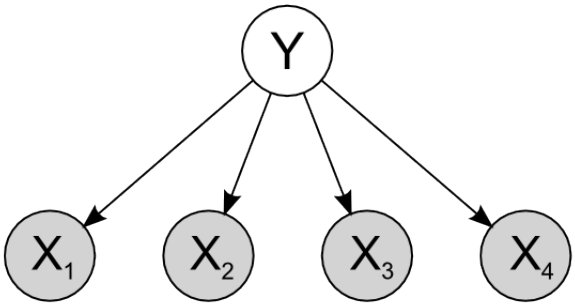
Lausekkeesta (3.2), mutta muuttujien nimet vaihdettuna:

$$p(x_i|\overline{x}_{\mathrm{pred}(i)\setminus\mathrm{pa(i)}}, \overline{x}_{\mathrm{pa}(i)}) = p(x_i|\overline{x}_{\mathrm{pa}(i)})$$

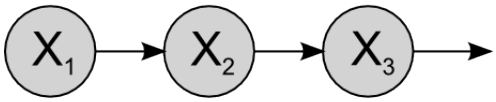
Yksinkertaisella induktiotodistuksella ketjutussääntöä käyttäen saadaan:[5]

$$p(\overline{x}) = p(x_1)p(x_2|\overline{x}_{\mathrm{pa}(2)}) \ldots p(x_n|\overline{x}_{\mathrm{pa}(n)}) = \prod_{i=1}^{n} p(x_i|\overline{x}_{\mathrm{pa}(i)}) \qquad (3.3)$$

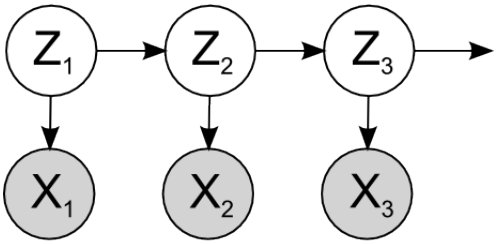
Kun jokaista satunnaismuuttujaa merkitään solmulla graafissa, on Markov-ominaisuudet omaavasta todennäköisyysjakaumasta rakennettu graafi jossa jokaisen solmun todennäköisyys riippuu vain vanhemmistaan. Lause (3.3) mahdollistaa tälläisen graafin todennäköisyysjakauman ilmaisemisen yksittäisten solmujen todennäköisyysjakaumien tulona. Jos graafin jokaisella solmulla on enitään M vanhempaa, solmua koskevalla taululla on $O(K * K^M)$ arvoa. Koska tauluja on n kappaletta, on arvojen määrä $O(n * K^{M+1})$, mikä on yleensä huomattavasti vähemmän kuin $O(K^n)$.[5] Riippumattomuusoletuksien tekeminen siis huomattavasti yksinkertaistaa mallia.



Kuva 3.1: Dokumentti NBC-mallissa: $p(\overline{x}) = p(y) \prod_{i=1}^{4} p(x_i|y)$



Kuva 3.2: Markov-ketju: $p(\overline{x}) = p(x_1) \prod_{i=2}^{N} p(x_i|x_{i-1})$



Kuva 3.3: Piilotettu Markov-ketju: $p(\overline{x}) = \prod_{i=1}^{N} p(x_i|z_i)$

3.2 Inferenssi ja oppiminen

Graafissa solmut esittävät muuttujia. Voi olla että osa muuttujista on havaittu ja osa tuntemattomia. Nimitetään näitä solmuja $\overline{x}_v$ ja $\overline{x}_h$. **Inferenssillä** tarkoitetaan tällöin tuntemattomien muuttujien todennäköisyysjakauman laskemista: [1]

$$p(\overline{x}_h|\overline{x}_v, G) = \frac{p(\overline{x}_h, \overline{x}_v|G)}{p(\overline{x}_v|G)} = \frac{p(\overline{x}|G)}{\sum_{\overline{x}'_h} p(\overline{x}'_h, \overline{x}_v|G)}$$

missä $p(\overline{x}_v|G)$ kutsutaan evidenssin todennäköisyydeksi, mikä on jakauman normalisointivakio.

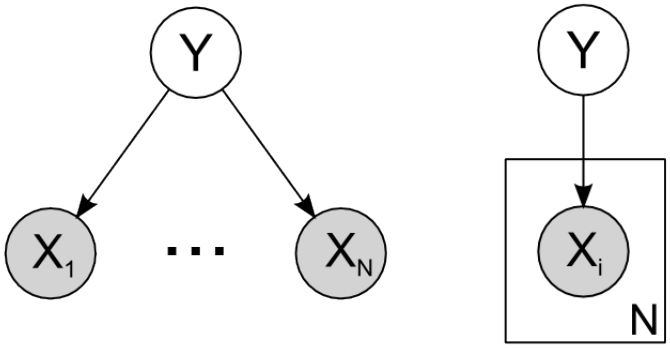
Oppiminen toisaalta yleensä tarkoittaa MAP-arvion tekemistä jakauman parametreistä havaittuun informaation perustuen.[1] Esimerkiksi generatiivisessa mallissa jossa graafin muuttujista on useita havaintoja:

$$\widehat{\theta} = \text{argmax}_\theta p(\theta|D) = \text{argmax}_\theta p(D|\theta)p(\theta) = \text{argmax}_\theta \prod_{i=1}^{N} p(\overline{x}_{i,v}|\theta)p(\theta)$$

Missä $\overline{x}_{i,v}$ on näkyvät muuttujat tapauksessa i.

Inferenssin ja oppimisen konseptuaalinen erottaminen saattaa hämmentää. Missä ovat parametrit θ graafissa? Jos parametri tai useammat parametrit määrittävät muuttujan x_i todennäköisyysjakauman, miksei yksinkertaisesti lisätä niitä muuttujan vanhemmiksi graafissa? Tämä onkin nimenomaan bayesilainen näkökanta, missä tälläistä konseptuaalista eroa ei tehdä.[1] Jotta tällöin voitaisiin inferoida $\overline{x}_h$, on otettava huomioon parametrien θ kaikki mahdolliset arvot. Siksi myös parametrit kuvataan graafissa, ja ne eroavat muuttujista vain siinä mielessä, että parametrien lukumäärä pysyy samana, missä informaation kasvaessa muuttujien määrä kasvaa.

Toinen syy konseptuaaliseen erottamiseen on generatiivisten mallien varsin kohtuullinen oletus: informaatioon sisältyvän saman muuttujan erikseen havaitut arvot ovat toisistaan riippumattomia ja identtisesti jakautuneita, ne riippuvat vain parametrien θ arvosta. Tämä mahdollistaa muuttujien havaittujen arvojen esittämisen käyttäen laatikkomallia. Laatikko on graafisessa mallissa muuttujan rajaava suorakulmio, ja tätä muuttujaa toistetaan mallissa niin monta kertaa kuin laatikon kulmassa lukee.



Kuva 3.4: Laatikkomalli mahdollistaa monimutkaisen graafin esittämisen ilman toistoa

3.3 Markov-ominaisuudet

Bayesilainen verkko määritteli graafisen mallin riippumattomuusoletuksiin *(Ordered Markov property)* perustuen:

$$x_i \perp \overline{x}_{\mathrm{pred}(i)\setminus \mathrm{pa}(i)} | \overline{x}_{\mathrm{pa}(i)} \tag{3.4}$$

Seuraavaksi tarkastellaan koko joukkoa riippumattomuusoletuksia jota arbitraarinen graafi sisältää. Tälläinen graafi voi sisältää myös solmuja joiden muuttujan arvo on havaittu.

Määritelmä lähteestä [1]: Suuntaamaton polku P on d-erotettu joukolla solmuja E jos ja vain jos ainakin yksi seuraavista ehdoista pätee:

1. P sisältää ketjun $s \to m \to t$ tai $s \leftarrow m \leftarrow t$ missä $m \in E$.
2. P sisältää $\wedge$-rakenteen $s \swarrow m \searrow t$ missä $m \in E$.
3. P sisältää $\vee$-rakenteen $s \searrow m \swarrow t$ missä $m \notin E$ ja $\mathrm{children}(m) \cap E = \emptyset$.

Joukko solmuja A on d-erotettu joukosta solmuja B joukolla solmuja E jos ja vain jos kaikki suuntaamattomattomat polut kaikista solmuista $a \in A$ solmuihin $b \in B$ ovat d-erotettu joukolla E.

Suunnatun syklittömän graafin tekemät riippumattomuuslauseet, sen Markov-ominaisuudet *(Directed Global Markov property)*, saadaan määritelmää käyttäen: [1]

$$\overline{x}_A \perp \overline{x}_B | \overline{x}_E \Leftrightarrow A \text{ on d-erotettu } B \text{ joukolla } E. \tag{3.5}$$

Asiaa saattaa selventää visuaalinen kuvaus lähteestä ([1] sivu 325).

Ehdoista d-erottamisen suhteen seuraa *(Directed Local Markov property)*: [1]

$$\overline{x}_A \perp \overline{x}_{\mathrm{nd}(i) \setminus \mathrm{pa}(i)} | \overline{x}_{\mathrm{pa}(i)} \tag{3.6}$$

missä $\mathrm{nd}(i)$ merkitsee kaikkia muita solmuja paitsi somun i seuraajia.

Ilman todistusta todettakoon [2]:

$$3.5 \Leftrightarrow 3.6 \Leftrightarrow 3.4$$

3.4 Yksittäisen solmun todennäköisyysjakauma

Joukko solmuja $\mathrm{mb}(x_i)$ joka erottaa solmun i kaikista muista solmuista $(-i)$ on solmun Markov-peitto *(Markov blanket)*. Käyttäen d-erotuskriteeriä saadaan: [1][5]

$$\mathrm{mb}(x_i) = \overline{x}_{\mathrm{ch}(i)} \cup \overline{x}_{\mathrm{pa}(i)} \cup \overline{x}_{\text{co-pa}(i)}$$

missä $\mathrm{ch}(i)$ merkitsee solmun i lapsia, $\mathrm{pa}(i)$ solmun i vanhempia ja co-pa(i) solmun i lapsien vanhempia. Solmun i todennäköisyysjakaumaksi Markov-peiton suhteen saadaan:

$$p(x_i | \overline{x}_{(-i)}) = \frac{p(x_i, \overline{x}_{(-i)})}{p(\overline{x}_{(-i)})} \propto p(x_i | \overline{x}_{\mathrm{pa}(i)}) \prod_{s \in \mathrm{ch}(i)} p(x_s | \overline{x}_{\mathrm{pa}(i)}).$$

Viimeinen lause seuraa todennäköisyysjakaumasta (3.3) jossa kaikki termit jotka eivät sisällä muuttujaa x_i kumoutuvat osoittajan ja nimittäjän välillä. Tätä lauseketta kutsutaan x_i:n täydelliseksi ehdolliseksi todennäköisyysjakaumaksi (*full conditional*). Sitä tarvitaan Gibbs-näytteenottoon, mikä on seuraavan kappaleen aihe.

Luku 4

Gibbs-näytteenotto

4.1 Todennäköisyysjakaumaan perustuva approksimointi

Luvussa 3. selitettiin miten yksi bayesilaisten verkkojen käyttötarkoitus oli mallin parametrien oppiminen havaittuun informaatioon perustuen, mikä mahdollistaa tämän mallin käytön tulevan informaation ennustamiseen. Voitaisiin siis edetä seuraavasti:

$$\widehat{\theta} = \text{argmax}_\theta p(\theta|D)$$

$$p(D_n|D) \approx p(D_n|\widehat{\theta})$$

Tämä on kuitenkin vain approksimaatio, koska todellinen bayesilainen inferenssi ottaa huomioon kaikki mahdolliset arvot joilla θ voi generoida informaation. Tämä mahdollistaa todellisen ekvivalenssin:

$$p(D_n|D) = \int_\theta p(D_n|\theta)p(\theta|D)\mathrm{d}\theta$$

Missä $p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$ kuten aiemmin. Nyt toisaalta saattaa olla että täytyy laskea $p(D) = \int_\theta p(D|\theta)p(\theta)\mathrm{d}\theta$, koska välttämättä ei haluta laskea lausekketta kaikille mahdollisille D_n arvoille.

Integraalilaskennassa on vaikeaa ja usein mahdotonta saada integraaleja laskettua analyyttisin keinoin. Jos eksaktia arvoa ei saada suoraan laskettua, tarvitaan menetelmä joka approksimoi integraalia riittävän lähelle tätä eksaktia arvoa. Tarvitaan siis menetelmän jolla laskea: [4]

$$E_Z[f(z)] = \sum_{z \in Z} f(z)p(z)$$

Tai vastaavassa tapauksessa jossa data ei ole diskreetti ja lauseke sisältää integraalin:

$$E_Z[f(z)] = \int_{z \in Z} f(z)p(z)$$

Näissä tapauksissa $p(z)$ ei välttämättä ole tasainen jakauma tai edes analyyttisesti arvioitavissa.

Oletetaan että otetaan T näytettä $z_1, \ldots, z_T$ jakaumasta $p(z)$ ja lasketaan $\frac{1}{T}\sum_i f(z_i)$. Tällöin enemmän näytteitä otetaan sellaisesta intervallista jossa $p(z)$ on suurempi. Jos näytteiden näytteenottoavaruus jaetaan tasaisiksi intervalleiksi, on intervallista otettujen näytteiden määrä suoraan suhteessa $p(z)$ arvoon. Arvo $f(z)p(z)$ on siis implisiittisesti otettu huomioon näytteiden 'tiheyksissä'. On määritelty tapa jolla $E_Z[f(z)]$ saadaan arvioitua. Voidaan osoittaa että: [4]

$$\lim_{T \to \infty} \frac{1}{T} \sum_i f(z_i) = E_Z[fz]$$

4.2 Näytteenotto todennäköisyysjakaumasta

Seuraava kysymys on: miten saadaan otettua näyte z jakaumasta $p(z)$? Mikäli pystytään määrittelemään funktion g joka tekee uuteen näytteeseen z_{t+1} siirtymisestä todennäköisen $p(z)$ mukaisesti, on tavoite saavutettu. Funktion tulisi siirtyä edellisten arvojen perusteella eli määritellä $p(z_{t+1}|z_t, \ldots, z_1)$. Tätä kutsutaan Monte Carlo-simulaatioksi. Mikäli todennäköisyys riippuu vain edellisestä näytteestä, $p(z_{t+1}|z_t)$, on kyseessä Markov Chain Monte Carlo-simulaatio *(MCMC)*. Kun funktio g täyttää tietyt ehdot, on sen taattu tuottaa näyte z jakauman $p(z)$ mukaisesti. Gibbs-näytteenotto on eräs tälläinen algoritmi.[4]

Algorithm 1 MCMC näytteiden ottoon

```
z_1 ← random()
for t=1 to T do
    z_{t+1} ← g(z_t)
end for
```

Gibbs-näytteenotossa on määritelty monen satunnaismuuttujan yhteinen jakauma $p(\overline{z}) = p(z_1, \ldots, z_n)$. Sen sijaan että uusi näyte $\overline{z}_{t+1}$ otetaan kerralla edelliseen arvoon $\overline{z}_t$ perustuen, jokainen z^i_{t+1} näytteistetään pienimmästä indeksistä alkaen erikseen perustuen arvoihin $\overline{z}^{(-i)}_{t+1}$. Siis z^i_{t+1} näytteistetään perustuen $z^{j<i}_{t+1}$ ja $z^{j>i}_t$ arvoihin. Kun kaikki n satunnaismuuttujaa on näytteistetty, on saatu uusi näyte $\overline{z}_{t+1}$.[4]

Algorithm 2 Gibbs näytteiden ottoon

$\overline{z}_1 \leftarrow random()$
for t=1 **to** T **do**
  for i=1 **to** n **do**
    $\overline{z}^i_{t+1} \leftarrow p(z_i | z^1_{t+1}, \ldots, z^{i-1}_{t+1}, z^{i+1}_t, \ldots, z^n_t)$
  end for
end for

Luku 5

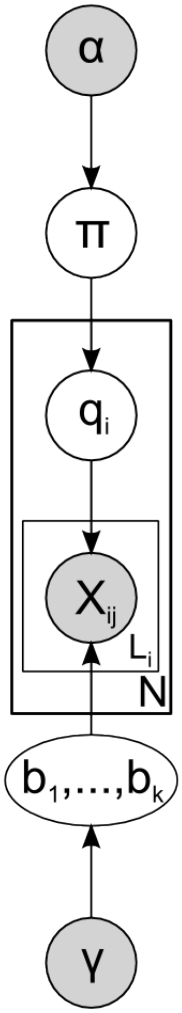
Naive Bayes classifier (NBC)

5.1 Generatiivinen malli

Kaikki perusteet mitä todellisten mallien ymmärtämiseksi ja totetuttamiseksi tarvitaan on nyt käsitelty. Eräs yksikertaisimmista, mutta useimmin käytetyistä, malleista on Naive Bayes luokittelija *(NBC)*. Se toimii seuraavasti. Oletetaan että on N dokumenttia $D = \{\overline{x}_1, \ldots, \overline{x}_N\}$ joista jokainen koostuu sarjasta sanoja $\overline{x_i} = (x_{i1}, x_{i2}, \ldots, x_{iL_i})$ missä L_i on dokumentin i pituus. Oletetaan että on määritelty teemat $T = \{1, \ldots, C\}$ ja jokaiseen $\overline{x}_i$ halutaan liittää teema $q_i \in T$. Tämä on mallin tavoite. Sarja sanoja voidaan vaihtoehtoisesti esittää ns. sanalaatikkona jossa jokainen x_{ij} merkitsee kuinka monta kertaa sana j esiintyy dokumentissa i, tai päälle-pois-mallina jossa jokainen x_{ij} esiintykö sana j dokumentissa i. On kuitenkin hyvin yksinkertainen tehtävä muokata malli vaihtoehtoisiin esitystapoihin.[1]

Mallin perustavanlaatuinen oletus on, että sanat x_{ij} ovat riippumattomia toisistaan kun teema q_i on annettu, ja että jokaiselle teemalla on sille ominainen todennäköisyysjakauma joka generoi nämä sanat. Siis: [1]

$$p(\overline{x}_i|q_i) = p(q_i) \prod_{j=1}^{L_i} p(x_{ij}|q_i)$$



Kuva 5.1: NBC-mallin graafi

5.2 Todennäköisyysjakaumien määrittely ja johtaminen

Määritellään seuraavat todennäköisyysjakaumat: [1]

$$\begin{aligned}\overline{\pi} &\sim \mathrm{Dir}(\alpha\overline{1}_C)\\ \overline{b_k} &\sim \mathrm{Dir}(\gamma\overline{1}_V)\\ q_i &\sim \mathrm{Cat}(\overline{\pi})\\ x_{il}|q_i &\sim \mathrm{Cat}(\overline{b_{q_i}})\end{aligned}$$

Jokainen annetuista teemoista on jakautunut saman kategoria-jakauman mukaisesti: $p(q_i = c) = \pi_c$. Vektori $\overline{\pi}$ määrittelee siis kunkin teeman todennäköisyyden, ja se itse jakautuu Dirichlet-jakauman mukaisesti. Vastaavasti jokaiselle mahdolliselle teemalle $k \in \{1, \ldots, C\}$ on määritelty sille ominainen sanojen $1, \ldots, V$ kategoria-jakauma, siis kaikille sanoille j dokumentissa i: $p(\overline{x}_{ij} = v|q_i = k) = b_{kv}$. Kukin jakauma $\overline{b}_k$ jakautuu vastaavasti Dirichlet-jakauman mukaisesti. Mikäli teemojen tai sanojen jakautumisesta ei tiedetä mitään, on voidaan nämä α ja γ asettaa arvoon 1 kuten tutkielman Matlab-toteutuksessa, jonka seurauksena jokainen vastaava kategoria-jakauma on yhtä todennäköinen. Nyt kun malli on määritelty, täytyy johtaa lauseke joka ilmaisee havaitsemattomien muuttujien todennäköisimmät arvot havaittujen perusteella, eli liittää jokaiseen dokumenttiin todennäköisimmän teeman.

Notaatiotarkoituksessa kerätään $\overline{x}_i$ ja $\overline{b}_k$ joukkoihin:

$$X = \{\overline{x}_1, \ldots, \overline{x}_N\}\, B = \left\{\overline{b}_1, \ldots, \overline{b}_C\right\}$$

Graafisesta mallista todennäköisyysjakaumaksi saadaan:

$$p(\overline{q}, X, B, \overline{\pi}|\alpha, \gamma) = p(\overline{\pi}|\alpha)p(\overline{q}|\overline{\pi})p(B|\gamma)p(X|\overline{q}, B)$$

Päättelyn jatkamiseksi on jokainen termi tässä lausekkeessa määriteltävä:

$$\begin{aligned}p(\overline{\pi}|\alpha) &= \frac{1}{\mathrm{B}(\alpha\overline{1}_C)}\prod_{c=1}^{C}\pi_c^{\alpha-1}\\ p(\overline{b}_k|\gamma) &= \frac{1}{\mathrm{B}(\gamma\overline{1}_V)}\prod_{v=1}^{V}\pi_{kv}^{\gamma-1}\end{aligned}$$

$$p(q_i|\overline{\pi}) = \prod_{c=1}^{C} \pi_c^{I(q_i=c)}$$
$$\Rightarrow p(\overline{q}|\overline{\pi}) = \prod_{i=1}^{N}\prod_{c=1}^{C} \pi_c^{I(q_i=c)} = \prod_{c=1}^{C} \pi_c^{N_{\mathrm{qc}}}$$

$$p(x_{ij}|q_i = k, B) = \prod_{v=1}^{V} b_{kv}^{I(x_{ij}=v)}$$
$$\Rightarrow p(\overline{x}_i|q_i = k, B) = \prod_{j=1}^{L_i}\prod_{v=1}^{V} b_{kv}^{I(x_{ij}=v)} = \prod_{v=1}^{V} b_{kv}^{N_{x_i v}}$$
$$\Rightarrow p(\overline{x}^k|q_i = k, B) = \prod_{i=1}^{N}\prod_{v=1}^{V} b_{kv}^{N_{x_i v}} = \prod_{v=1}^{V} b_{kv}^{N_{kv}}$$

missä olemme määritellet seuraavat lukumäärät:

$$N_{qc} = \sum_{i=1}^{N} I(q_i = c) \;\text{ luokan } c \text{ dokumentit}$$
$$N_{x_i v} = \sum_{j=1}^{L_i} I(x_{ij} = v) \;\text{ sana } v \text{ esiintyy dokumentissa } i$$
$$N_{kv} = \sum_{i=1}^{N}\sum_{j=1}^{L_i} I(q_i = k, x_{ij} = v) \;\text{ sana } v \text{ esiintyy luokassa } k$$

Ja seuraavan funktion: $\mathrm{B}(\overline{\alpha}) = \frac{\prod_{k=1}^{K}\Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{K}\alpha_k)}$.

Konjukaatti-priorit toimivat yhteen vastaavien MLE-arvioiden kanssa:

$$\begin{aligned} p(\overline{\pi}|\overline{q}, \alpha) &\propto p(\overline{q}|\overline{\pi})p(\overline{\pi}|\alpha) \\ &= \frac{1}{\mathrm{B}(\alpha\overline{1}_C)}\prod_{c=1}^{C}\pi_c^{\alpha-1}\prod_{c=1}^{C}\pi_c^{N_{qc}} = \frac{1}{\mathrm{B}(\alpha\overline{1}_C)}\prod_{c=1}^{C}\pi_c^{\alpha+N_{qc}-1} \end{aligned} \tag{5.1}$$

$$\begin{aligned} p(\overline{b}_k|\overline{x}_i, \gamma) &\propto p(\overline{x}_i|\overline{b}_k)p(\overline{b}_k|\gamma) \\ &= \frac{1}{\mathrm{B}(\gamma\overline{1}_V)}\prod_{v=1}^{V} b_{kv}^{\gamma-1}\prod_{v=1}^{V} b_{kv}^{N_{x_i v}} = \frac{1}{\mathrm{B}(\gamma\overline{1}_V)}\prod_{v=1}^{V} b_{kv}^{\gamma+N_{x_i v}-1} \end{aligned} \tag{5.2}$$

$$\Rightarrow p(X|q_i, B) = \prod_{k=1}^{C} p(\overline{x}^k|\overline{b}_k)p(\overline{b}_k|\gamma) = \frac{1}{\mathrm{B}(\gamma\overline{1}_V)} \prod_{k=1}^{C}\prod_{v=1}^{V} b_{kv}^{\gamma+N_{x_i v}-1} \tag{5.3}$$

Joten täydelliseksi todennäköisyysjakaumaksi saadaan:

$$p(\overline{q}, X, B, \overline{\pi}, \alpha, \gamma) \propto \frac{1}{\mathrm{B}(\alpha\overline{1}_C)} \prod_{c=1}^{C} \pi_c^{\alpha+N_{qc}-1} \frac{1}{\mathrm{B}(\gamma\overline{1}_V)} \prod_{k=1}^{C}\prod_{v=1}^{V} b_{kv}^{\gamma+N_{x_i v}-1} \tag{5.4}$$

Voimme itse asiassa marginalisoida muuttujan $\overline{\pi}$. Tällöin sitä ei tarvitse näytteistää: [1]

$$p(\overline{q}, X, B, \alpha, \gamma) \propto \int p(\overline{q}, X, B, \overline{\pi}, \alpha, \gamma)\mathrm{d}\overline{\pi}$$

Tarkastellaan vain muuttujan $\overline{\pi}$ sisältävää termiä:

$$\begin{aligned}\int \prod_{c=1}^{C} \pi_c^{\alpha+N_{qc}-1}\mathrm{d}\overline{\pi} &= \mathrm{B}(\alpha + N_{q1}, \ldots, \alpha + N_{qC}) \int \mathrm{Dir}(\alpha + N_{q1}, \ldots, \alpha + N_{qC})\mathrm{d}\overline{\pi} \\ &= \mathrm{B}(\alpha + N_{q1}, \ldots, \alpha + N_{qC})\end{aligned} \tag{5.5}$$

On saatu yksinkertaisempi lause täydelle todennäköisyysjakaumalle:

$$p(\overline{q}, X, B, \overline{\pi}, \alpha, \gamma) \propto \frac{\mathrm{B}(\alpha + N_{q1}, \ldots, \alpha + N_{qC})}{\mathrm{B}(\alpha\overline{1}_C)} \frac{1}{\mathrm{B}(\gamma\overline{1}_V)} \prod_{k=1}^{C}\prod_{v=1}^{V} b_{kv}^{\gamma+N_{x_i v}-1} \tag{5.6}$$

5.3 Näytteenotto todennäköisyysjakaumasta

Gibbs-näytteenottoa koskevasta kappaleesta seuraa, että tarvitaan seuraava lauseke: [4]

$$p(q_i|\overline{q}_{(-i)}, X_{(-i)}, B, \alpha, \gamma) = \frac{p(\overline{q}, X, B, \alpha, \gamma)}{p(\overline{q}_{(-i)}, X_{(-i)}, B, \alpha, \gamma)}$$

mitä tulee teeman q_i näytteistämiseen. Lausekkeen arvon laskemiseksi tarkastellaan yksitellen osoittajan ja nimittäjän vastaavia termejä, missä dokumentti i ei sisälly nimittäjään. Merkitään dokumentin i poistoa vastaavasta muuttujasta $N \to \overline{N}$.

$$\mathrm{B}(\gamma\overline{1}_V) \quad \text{kumoutuu} \tag{5.7}$$

$$\frac{\mathrm{B}(\alpha + N_{q1}, \ldots, \alpha + N_{qC})}{\mathrm{B}(\alpha \overline{1}_C)} \frac{\mathrm{B}(\alpha \overline{1}_C)}{\mathrm{B}(\alpha + \overline{N}_{q1}, \ldots, \alpha + \overline{N}_{qC})}$$
$$= \frac{\Gamma(\alpha C + N - 1)}{\Gamma(\alpha C + N)} \frac{\prod_c \Gamma(\alpha + N_{qc})}{\prod_c \Gamma(\alpha + \overline{N}_{qc})}$$

Käyttäen hyväksi faktaa $\frac{\Gamma(\alpha)}{\Gamma(\alpha - 1)} = \alpha - 1$ ja $\overline{N}_{qc} = N_{qc}$ kun $q \neq q_i$:

$$= \frac{1}{\alpha C + N - 1} \frac{\Gamma(\alpha + N_{qq_i})}{\Gamma(\alpha + N_{qq_i} - 1)} = \frac{\alpha + N_{qq_i} - 1}{\alpha C + N - 1} \tag{5.8}$$

$$\prod_{k=1}^{C} \prod_{v=1}^{V} b_{kv}^{\gamma + N_{kv} - 1} : \prod_{k=1}^{C} \prod_{v=1}^{V} b_{kv}^{\gamma + \overline{N}_{kv} - 1}$$

Koska $N_{kv} = \overline{N}_{kv}$ kun $k \neq q_i$, muulloin $N_{kv} = \overline{N}_{kv} + N_{x_i v}$ kaikille v. Täten:

$$= \prod_{v=1}^{V} b_{kv}^{\gamma + N_{x_i v} - 1} \tag{5.9}$$

Termeistä (5.8) ja (5.9) seuraa:

$$p(q_i | \overline{q}_{(-i)}, X_{(-i)}, B, \alpha, \gamma) = \frac{\alpha + N_{qq_i} - 1}{\alpha C + N - 1} \prod_{v=1}^{V} b_{kv}^{\gamma + N_{x_i v} - 1} \tag{5.10}$$

Teemakohtaisten jakaumien näytteistäminen on helpompaa koska ne ovat toisistaan riippumattomia:

$$\begin{aligned} p(\overline{b}_k | \overline{x}^k, \gamma) &\propto p(\overline{x}^k | \overline{b}_k) p(\overline{b}_k | \gamma) \\ &= \frac{1}{\mathrm{B}(\gamma \overline{1}_V)} \prod_{v=1}^{V} b_{kv}^{\gamma + N_{kv} - 1} \\ &\propto \mathrm{Dir}(\gamma + N_{k1}, \ldots, \gamma + N_{kV}) \end{aligned}$$

Täytyy siis näytteistää $\overline{b}_k$ Dirichlet-jakaumasta. Mikäli käyettävä ohjelmointikieli ei tue tätä, sen voi näytteistää Gamma-jakaumasta.[4] Kaikille $i = 1, \ldots, V$ tehdään:

$$y_i \sim \mathrm{Gamma}(\gamma + N_{ki}, 1)$$

Josta saadaan jokaiselle $i = 1, \ldots, V$:

$$b_i = \frac{y_i}{\sum_j y_j}$$

Jotka ovat $\overline{b}_k$:n komponentit.

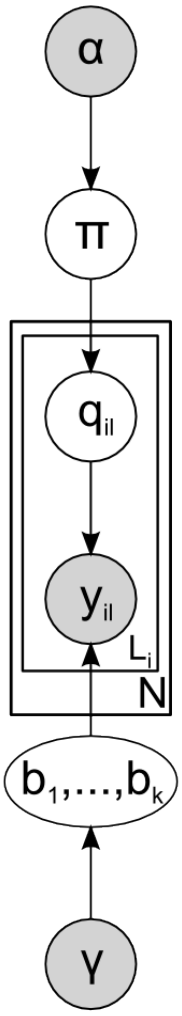
Tähän mennessä on marginalisoitu lauseesta muuttuja $\overline{\pi}$ ja näytetty kuinka jokainen $\overline{q}_i$ ja $\overline{b}_k$ saadaan näytteistettyä todennäköisyysjakaumasta. Kun nämä lausekkeet annetaan Gibbs-näytteenottajalle, saamme Naive Bayes luokittelijan, valvomattoman sellaisen. Jos kuitenkin teemoja on saataville osalle dokumentteja, saadaan luokittelija ottamaan tämä huomioon yksinkertaisesti määrittämällä nämä teemat annetuiksi arvoiksi ja jättämällä niiden näytteistäminen pois iteraatioista. Seuraavassa kappaleessa kuvataan astetta monimutkaisempi malli, minkä jälkeen varsinanen algoritminen toteutus Gibbs-näytteenottajasta molemmille voidaan esitellä.

Luku 6

Latent Dirichlet Allocation (LDA)

6.1 Generatiivinen malli

NBC on hyvä malli dokumenteille jotka voi selkästi jakaa eri teemaluokkiin. Todellisuudessa monesti on kuitenkin niin, että dokumentilla ei ole yhtä selkeästi määriteltyä luokkaa, vaan se sisältää useampia eri teemoja jotka vaikuttavat sen luokitteluun. Jos dokumentissa on esimerkiksi kaksi selkeää teemaa, sen luokittelu NBC:llä yhteen teemoista tarkoittaa, että kun selataan dokumentteja joita kuvataan toisella teemalla, ei nähdä tätä dokumenttia. Latent Dirichlet Allocation (LDA) on ratkaisu tähän ongelmaan. Oletetaan että on joukko dokumentteja $\{\overline{x}_1, \ldots, \overline{x}_N\}$ ja teemoja $T = \{1, \ldots, C\}$ kuten ennen. Nyt sen sijaan että jokaiseen dokumenttiin $\overline{x}_i$ liitetään teema y_i, siihen liitetään todennäköisyysjakauma $\overline{\pi}_i$ teemoja. Tämä saavutetaan käytännössä niin, että jokaiseen erilliseen sanaan $y_{il} \in \overline{x}_i$ liitetään teema q_{il}. Täten jokaisessa dokumentissa jokaisella sanalla on sen generoiva oma teema. Sama oletus sanojen riippumattomuudesta pätee, nyt mallissa on jakauma teemoja dokumentille yhden teeman sijaan. [1][3]



Kuva 6.1: LDA-mallin graafi

6.2 Todennäköisyysjakaumien määrittely ja johtaminen

Määritellään seuraavat todennäköisyysjakaumat:[1][3]

$$\begin{aligned}
\overline{\pi}_i &\sim \mathrm{Dir}(\alpha\overline{1_C}) \\
\overline{b_k} &\sim \mathrm{Dir}(\gamma\overline{1_V}) \\
q_{il} &\sim \mathrm{Cat}(\overline{\pi}_i) \\
y_{il}|q_{il} &\sim \mathrm{Cat}(\overline{b_{q_{il}}})
\end{aligned}$$

Notaatiotarkoituksessa kerätään taas vastaavat vektorit seuraaviin joukkoihin:

$$\Pi, B, Y, Q$$

Mallista täydelliseksi todennäköisyysjakaumaksi saadaan:

$$p(Y, B, Q, \Pi|\alpha, \gamma) = p(\Pi|\alpha)p(Q|\Pi)p(B|\gamma)p(Y|Q, B)$$

Kirjoitetaan lauseke termien tulona kuten NBC:n kohdalla:

$$p(Y, B, Q, \Pi|\alpha, \gamma) \propto \overbrace{\prod_{i=1}^{N} p(\overline{\pi}_i|\alpha)}^{1} \overbrace{\prod_{i=1}^{N}\prod_{j=1}^{L_i} p(q_{ij}|\overline{\pi}_i)}^{2} \overbrace{\prod_{c=1}^{C} p(\overline{b}_c|\gamma)}^{3} \overbrace{\prod_{i=1}^{N}\prod_{j=1}^{L_i} p(y_{ij}|q_{ij}, \overline{b}_{q_{ij}})}^{4} \tag{6.1}$$

Tarkastellaan seuraavaksi lausetta kahdessa osassa MLE:n ja vastaavan konjukaattipriorin kanssa. Integroidaan lausekkeista $\overline{\pi}$ ja B.

Osista (1) ja (2):

$$
\begin{aligned}
p(Q|\alpha) &= \int p(Q,\Pi|\alpha)\mathrm{d}\Pi = \int p(Q|\Pi)p(\Pi|\alpha)\mathrm{d}\Pi \\
&= \prod_i \int \left[\prod_l p(q_{il}|\overline{\pi}_i)\right] p(\overline{\pi}_i|\alpha)\mathrm{d}\overline{\pi}_i \\
&= \prod_i \int \left[\prod_{l=1}^{L_i} \mathrm{Cat}(q_{il}|\overline{\pi}_i)\right] \mathrm{Dir}(\overline{\pi}_i|\alpha\overline{1}_K)\mathrm{d}\overline{\pi}_i \\
&= \prod_i \int \prod_{k=1}^{K} \pi_{ik}^{c_{ik}} \prod_{k=1}^{K} \pi_{ik}^{\alpha-1} \frac{1}{\mathrm{B}(\alpha\overline{1}_K)}\mathrm{d}\overline{\pi}_i \\
&= \prod_i \frac{1}{\mathrm{B}(\alpha\overline{1}_K)} \int \prod_{k=1}^{K} \pi_{ik}^{c_{ik}+\alpha-1}\mathrm{d}\overline{pi}_i \\
&= \prod_i \frac{\mathrm{B}(\alpha\overline{1}_K + \overline{c}_{ik})}{\mathrm{B}(\alpha\overline{1}_K)} = \prod_i \left[\frac{\Gamma(K\alpha)}{\Gamma(K\alpha+L_i)} \prod_k \frac{\Gamma(\alpha+c_{ik})}{\Gamma(\alpha)}\right] \\
&= (\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K})^N \prod_i \frac{\prod_k \Gamma(\alpha+c_{ik})}{\Gamma(K\alpha+L_i)} \qquad (6.2)
\end{aligned}
$$

Osista (3) ja (4):

$$
\begin{aligned}
p(Y|Q,\gamma) &= \int p(Y,B|Q,\gamma)\mathrm{d}B = \int p(Y|B,Q)p(B|\gamma)\mathrm{d}B \\
&= \prod_k \int \left[\prod_{il:q_{il}=k} p(y_{il}|\overline{b}_k)\right] p(\overline{b}_k|\gamma)\mathrm{d}\overline{b}_k \\
&= \prod_k \int \left[\prod_{il:q_{il}=k} \mathrm{Cat}(y_{il}|\overline{b}_k)\right] \mathrm{Dir}(\overline{b}_k|\gamma)\mathrm{d}\overline{b}_k \\
&= \prod_k \int \prod_{v=1}^{V} b_{kv}^{c_{kc}} \prod_{v=1}^{V} b_{kv}^{\gamma-1} \frac{1}{\mathrm{B}(\gamma\overline{1}_K)}\mathrm{d}\overline{b}_k \\
&= \prod_k \frac{1}{\mathrm{B}(\gamma\overline{1}_K)} \int \prod_{v=1}^{V} b_{kv}^{c_{kc}+\gamma-1}\mathrm{d}\overline{b}_k \\
&= \prod_k \frac{\mathrm{B}(\gamma\overline{1}_K + \overline{c}_{vk})}{\mathrm{B}(\gamma\overline{1}_K)} = \prod_k \frac{\Gamma(V\gamma)}{\Gamma(V\gamma+c_k)} \prod_v \frac{\Gamma(\gamma+c_{vk})}{\Gamma(\gamma)} \\
&= (\frac{\Gamma(V\gamma)}{\Gamma(\gamma)^C})^K \prod_k \frac{\prod_v \Gamma(\gamma+c_{vk})}{\Gamma(\gamma)} \qquad (6.3)
\end{aligned}
$$

Missä on määritelty seuraava tärkeä lukumäärä:

$$c_{ivk} = \sum_{l=1}^{L_i} I(q_{il} = k, y_{il} = v)$$

Mikä kuvastaa kertoja joissa sanaan v on liitetty teema k dokumentissa i. Sen kautta on helppo määritellä muut lukumäärät:

$$c_{ik} = \sum_{v} c_{ivk}$$
$$c_{vk} = \sum_{i} c_{ivk}$$
$$c_{iv} = \sum_{k} c_{ivk}$$
$$c_{k} = \sum_{v} c_{vk}$$

6.3 Näytteenotto todennäköisyysjakaumasta

Tarvitaan seuraava lauseke:[4]

$$p(q_{il}|\overline{q}_{-(il)}, \overline{y}_{-(il)}, \alpha, \gamma) = \frac{p(\overline{q}, \overline{y}|\alpha, \gamma)}{p(\overline{q}_{(-il)}, \overline{y}_{(-il)}|\alpha\gamma)}$$

Tehdään kuten NBC:n kohdalla, tarkastellaan todennäköisyysjakauman toisiaan vastaavia termejä osoittajassa ja nimittäjässä. Nimittäjän termeistä on poistettu sana y_{il} ja teema q_{il}. Lauseista (6.2) ja (6.3):

$$p(\overline{q}, \overline{y}|\alpha, \gamma) \propto (\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K})^N \prod_i \frac{\prod_k \Gamma(\alpha + c_{ik})}{\Gamma(K\alpha + L_i)} (\frac{\Gamma(V\gamma)}{\Gamma(\gamma)^C})^K \prod_k \frac{\prod_v \Gamma(\gamma + c_{vk})}{\Gamma(\gamma)} \tag{6.4}$$

Tarkastellaan miten sanan ja teeman il poisto vaikuttaa lukumääriin.

Merkitään lukumäärää josta poisto on tapahtunut $c \to \overline{c}$:

$$
\begin{aligned}
c_{vk} &= \overline{c}_{vk} + 1 \ \text{ kun } v = y_{il}, k = q_{il} \\
c_{vk} &= \overline{c}_{vk} \ \text{ muulloin}
\end{aligned}
$$

$$
\begin{aligned}
c_{ik} &= \overline{c}_{ik} + 1 \ \text{ kun } k = q_{il} \\
c_{ik} &= \overline{c}_{ik} \ \text{ muulloin}
\end{aligned}
$$

$$
\begin{aligned}
c_{k} &= \overline{c}_{k} + 1 \ \text{ kun } k = q_{il} \\
c_{k} &= \overline{c}_{k} \ \text{ muulloin}
\end{aligned}
$$

$$
\begin{aligned}
L_{i} &= \overline{L}_{i} + 1 \ \text{ kun } i \\
L_{i} &= \overline{L}_{i} \ \text{ muulloin}
\end{aligned}
$$

Saadaan seuraava lauseke, jossa on asetettu $v = y_{il}$ ja $k = q_{il}$:

$$
\begin{aligned}
& p(q_{il} | \overline{q}_{-(il)}, \overline{y}_{-(il)}, \alpha, \gamma) \\
&= \frac{\Gamma(c_{vk} + \gamma)}{\Gamma(c_{vk} + \gamma - 1)} \frac{\Gamma(c_k + V\gamma - 1)}{\Gamma(c_k + V\gamma)} \frac{\Gamma(c_{ik} + \alpha)}{\Gamma(c_{ik} + \alpha - 1)} \frac{\Gamma(L_i + K\alpha - 1)}{\Gamma(L_i + K\alpha)} \\
&= \frac{c_{vk} + \gamma - 1}{c_k + V\gamma - 1} \frac{c_{ik} + \alpha - 1}{L_i + K\alpha - 1} \\
&= \frac{\overline{c}_{vk} + \gamma}{\overline{c}_k + V\gamma} \frac{\overline{c}_{ik} + \alpha}{\overline{L}_i + K\alpha}
\end{aligned}
\tag{6.5}
$$

Tämä lauseke oli kaikki mitä Gibbs-näytteenottoon tarvitaan.

Luku 7

Mallien Matlab-toteutus

7.1 Toteutuksen rajoitteet

On käsittelty kaikki matemaattiset perusteet jotka näiden kahden mallin ymmärtämiseen tarvitaan. Seuraavaksi annetaan pedagogisen Matlab-toteutuksen joka konkretisoi esityksen ja mahdollistaa mallien kokeilemisen. Toteutuksen tulee olla mahdollisimman yksinkertainen ja käyttää vain kuvattuja lauseita ja inferenssiperiaatteita. Tärkeä huomio mallin soveltamisesta oikeaan maailmaan on ilmiö nimeltä *numercial underflow*. Tietokoneen muistin säilöttävien lukujen absoluuttisella arvolla on tietty ala- ja yläraja. Esimerkiksi seuraava luku:

$$x = \prod_{i=1}^{N} \prod_{c=1}^{C} \theta_{ic}^{N_c}$$

on erittäin pieni. Se on niin pieni, että dokumenttien määrän kasvaessa sitä ei voida enää tallentaa. Tällöin useat ohjelmat pyöristävät sen nollaksi. Siksi täytyy käyttää seuraavaa logaritmitemppua. Vaihdetaan kaikki pienet arvot x seuraavasti: $y = \log(x)$. Mikäli tarvitaan todellinen x:n arvo verrannollisen x arvon sijaan, normalisoidaan lauseke vastaavalla eksponenttitempulla. [1] Esimerkiksi yllä olevasta x:stä tulee:

$$\log(x) = \sum_{i=1}^{N} \sum_{c=1}^{C} N_c \theta_{ic}$$

7.2 Toteutuksen yksityiskohdat

Data määritellään kokoelmana dokumentteja $\{d_1, \ldots, d_N\}$ missä jokainen dokumentti on sarja sanoja $(w_{i1}, \ldots, w_{iL_i})$. Dokumentin pituus voi siis vaihdetta ja jokainen sana voi ilmetä dokumentissa mielivaltaisen määrän kertoja.

Koska pedagogista esimerkkiä monimutkaistaisi sanaston kerääminen, sanat merkitään numeroin: $w_{ij} \in \{1, \ldots, W\}$, missä W on sanojen määrä sanastossa. Ohjelmissa on funktio `generate_documents` joka automaattisesti luo dokumentit jotka demonstroivat konseptia. Matlabissa dokumentit tallennetaan vektoreita sisältävään taulukkoon. Teemat tallennetaan samalla tavalla, LDA:lle on identtinen sanoja vastaava vektori-taulukko `q{i}(j)` ja NBC:lle taulukko teemoja `q(i)` .

Mallia koskevista luvuista muistetaan että lausekkeissa oli useita lukumäärämuuttujia. Nämä talletaan yhteen muuttujaan c_{kvi} josta summaamalla haluttuun ulottuvuuteen saadaan kaikki muut lukumäärät:

$$c_{vki} = \sum_{l=1}^{L_i} I(q_{il} = k, y_{il} = v)$$

Tämä on siis toteutuksessa $VxKxN$ matriisi. Vastaavat summat voidaan toki tallentaa ja muuttaa erikseen nopeussyistä.

Lasketut todennäköisyydet kullekkin näytteelle talletetaan vektoriin $p_k = (p_1, \ldots, p_k)$ missä $p_i = p(q_i = k|.)$. Tämä oli siis Gibbs-näytteenoton lauseke. Muuttujan q_i arvo on sitten otettu tästä vektorista käyttäen Matlabin sisäistä funktiota `randsample`(.). Kaikki muut totetuksen muuttujat vastaavat edellisissä luvuissa esitettyjä.

Täydellisen toteutuksen kaikki tarvittavat Matlab-tiedostot ovat ladattavissa verkko-osoitteessa `http://users.utu.fi/majuvi/`.

7.3 NBC-algoritmi

Seuraava algoritmi generoi dokumentit, suorittaa NBC-luokittelun ja visualisoi luokittelun tuloksia. Se suorittaa varsinaisen luokittelun kutsumalla funktiota jossa NBC on toteutettu. Suorita ohjelma ajamalla tämä algoritmi:

```
%
% DEMO_NBC: demonstrates Naive Bayes Classifier (NBC)
%
%   => Generates documents as word vectors in a cell array
%   => Visualizes the documents as a bag-of-words bit image
%   => Samples topic and word distributions using Gibbs for NBC
%   => Visualizes the topic distribution as a bit image
%

% generate the documents
documents = 16; topics = 2; words = 5; words_per_doc = 15;
document = generate_documents( documents, topics, words, words_per_doc );
```

```
% visualize the documents
visualize = word_dist(document, words);
figure_distmatrix(visualize, 'Word distribution matrix', 'document', 'word');

% sample the topics
[q] = gibbs_sample_NBC(document, topics, words, 103, 1 ,1);

% visualize the topics
figure_distmatrix(q, 'Topic distribution matrix', 'document', 'topic');
```

NBC:n varsinainen toteutus:

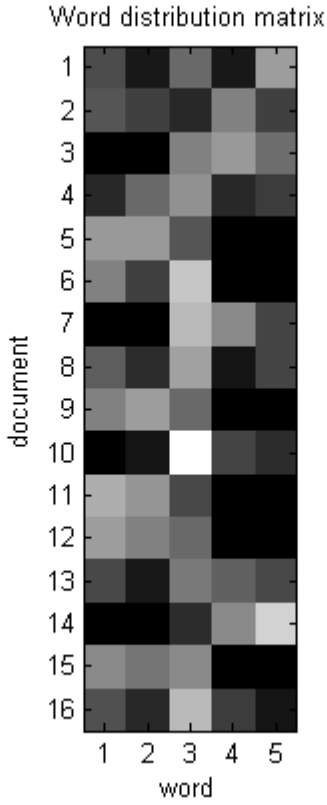
```
function [ q_sample ] =...
gibbs_sample_NBC( document, topics, words, iterations, alpha, lambda )
    %Gibbs_sample_NBC Produces topic allocation for given documents using
    %Gibbs sampling for the Naive Bayes Classifier (NBC) model.
    %
    % Arguments:
    %   document: documents as cell array of vectors of integers
    %   topics: number of topics to allocate
    %   words: number of words in dictionary
    %   iterations: iterations to run
    %   alpha: prior hyperparameter for the topic distribution
    %   lambda: prior hyperparameter for the word distribution
    %
    % Internal variables:
    %   q: N topic allocations
    %   c: KxVxN I(q(ij) == k, y(ij) == v)
    %   q_k: P(q(i) == k | .)
    %
    % Returns:
    %   q_sample: document topic allocations matrix

    documents = size(document,2);
    words_per_doc = [];
    for j=1:documents
        words_per_doc(j) = size(document{j},1);
    end

    q = floor(rand([documents, 1])*topics+1);
    q_sample = zeros([documents, 1]);

    b = zeros([topics,words]);
    for i = 1:topics
        b(i,:) = sample_dirichlet(lambda*ones([1, words]));
    end

    c = [];
    for i=1:documents
        kv = zeros(topics, words);
```

```
    for l=1:words_per_doc(i)
        kv(q(i), document{i}(l))= kv(q(i), document{i}(l)) + 1;
    end
    c(:,:,i) = kv;
end

n_samples = 10;
offset = mod(iterations,n_samples);
block = floor(iterations/n_samples);

fprintf('starting...\n');
for iter = 1:iterations
    for i = 1:documents

        q_i = q(i);

        %c_kvi
        c_vi = sum(c,1);
        c_kv = sum(c,3);
        c_k = sum((sum(c,2)~=0),3);

        p_q = zeros(1,topics);
        for k=1:topics
            p_q(k) = (alpha+c_k(k)-1)/(alpha*topics+documents-1);
            B = 1;
            for v = 1:words
                B = B*(b(k,v)^(lambda+c_vi(1,v,i)-1));
            end
            p_q(k) = p_q(k)*B;
        end

        k = randsample(topics, 1, true, p_q);

        old = c(q_i,:,i);
        c(q_i,:,i) = zeros([1,words]);
        c(k,:,i) = old;

        q(i) = k;
    end

    c_kv = sum(c,3);
    for i = 1:topics
        b(i,:) = sample_dirichlet(c_kv(i,:)+lambda);
    end

    if (mod(iter-offset,block) == 0)
        fprintf('%d/%d\n',iter,iterations);
        q_sample = q_sample + q;
    end
```

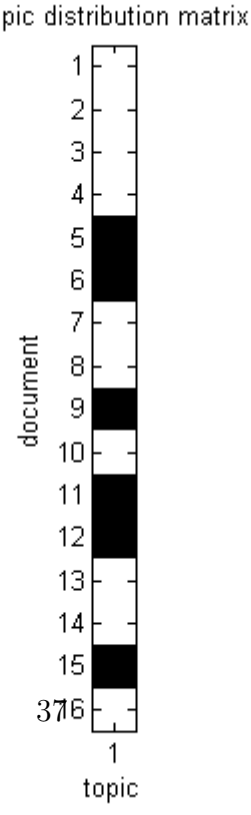
```
    end

    q_sample = round(q_sample./n_samples);

    fprintf('done.\n');
end
```

7.4 NBC:n tulokset

Algoritmi tuottaa seuraavat visualisoinnit:



Kuva 7.1: Sanojen dokumenttikohtainen todennäköisyysjakauma



Kuva 7.2: NBC:n tuottama teemojen jakauma

7.5 LDA-algoritmi

Seuraava algoritmi suorittaa vastaavat toiminpiteet LDA-luokitettelulle:

```
%
% DEMO_LDA: demonstrates Latent Diriclet Allocation (LDA)
%
%   => Generates documents as word vectors in a cell array
%   => Visualizes the documents as a bag-of-words bit image
%   => Samples topic and word distributions using Gibbs for LDA
%   => Visualizes the topic distribution as a bit image
%

% generate the documents
documents = 16; topics = 2; words = 5; words_per_doc = 15;
document = generate_documents( documents, topics, words, words_per_doc );

% visualize the documents
visualize = word_dist(document, words);
figure_distmatrix(visualize, 'Word distribution matrix', 'document', 'word');

% sample the topics
[topic_dist, q] = gibbs_sample_LDA(document, topics, words, 103, 1 ,1);

% visualize the topics
figure_distmatrix(topic_dist, 'Topic distribution matrix', 'document', 'topic');
```

LDA:n varsinainen toteutus:

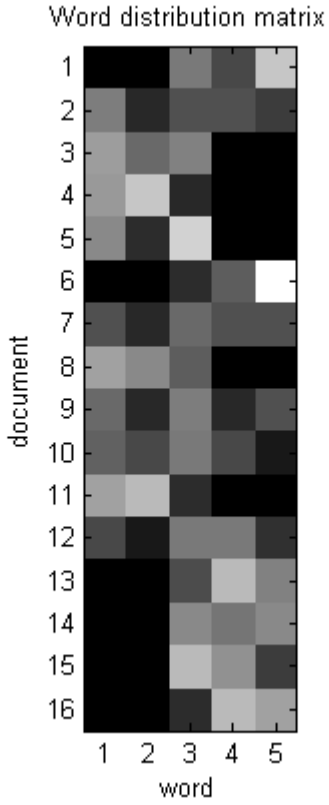
```
function [ q_dist, q_sample ] =...
gibbs_sample_LDA( document, topics, words, iterations, alpha, lambda )
    %Gibbs_sample_LDA Produces topic distribution for given documents using
    %Gibbs sampling for the Latent Dirichlet Allocation (LDA) model.
    %
    % Arguments:
    %   document: documents as cell array of vectors of integers
    %   topics: number of topics to allocate
    %   words: number of words in dictionary
    %   iterations: iterations to run
    %   alpha: prior hyperparameter for the topic distribution
    %   lambda: prior hyperparameter for the word distribution
    %
    % Internal variables:
    %   q: NxV topic allocations
    %   c: KxVxN I(q(ij) == k, y(ij) == v)
    %   q_k: P(q(i) == k | .)
    %
    % Returns:
    %   q_dist: topic distributions cell array
    %   q_sample: word topic allocations matrix
```

```
documents = size(document,2);
words_per_doc = [];
for j=1:documents
    words_per_doc(j) = size(document{j},1);
end

q = {};
q_sample = {};
for i = 1:documents
    q{i} = floor(rand([words_per_doc(i), 1])*topics+1);
    q_sample{i} = zeros([words_per_doc(i), 1]);
end

c = [];
for i=1:documents
    kv = zeros(topics, words);
    for l=1:words_per_doc(i)
        kv(q{i}(l), document{i}(l))= kv(q{i}(l), document{i}(l)) + 1;
    end
    c(:,:,i) = kv;
end

%c_kvi
c_ki = sum(c,2);
c_kv = sum(c,3);
c_k = sum(c_kv,2);

n_samples = 10;
offset = mod(iterations,n_samples);
block = floor(iterations/n_samples);

fprintf('starting...\n');
for iter = 1:iterations
    for i = 1:documents
        for l=1:words_per_doc(i)

            q_il = q{i}(l);
            y_il = document{i}(l);

            %same as c(q_il,y_il,i) = c(q_il,y_il,i) - 1;
            c_ki(q_il,1,i) = c_ki(q_il,1,i) - 1;
            c_kv(q_il,y_il,1) = c_kv(q_il,y_il,1) - 1;
            c_k(q_il) = c_k(q_il) - 1;

            p_q = zeros(1,topics);
            for k=1:topics
                p_q(k) = (c_kv(k,y_il)+lambda) / (c_k(k,1)+words*lambda) *...
                (c_ki(k,1,i)+alpha) / ( words_per_doc(i) + topics*alpha);
```

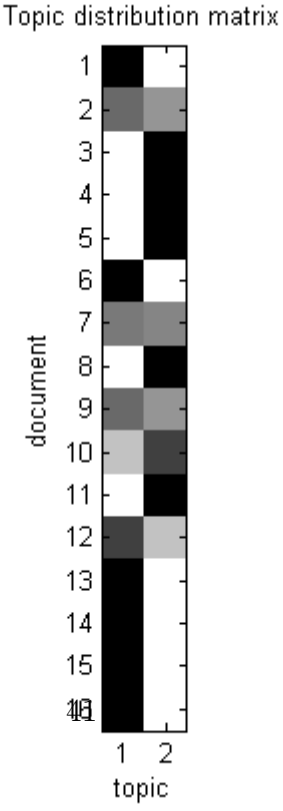
```
                end

                k = randsample(topics, 1, true, p_q);

                %same as c(k,y_il,i) = c(k,y_il,i) + 1;
                c_ki(k,1,i) = c_ki(k,1,i) + 1;
                c_kv(k,y_il,1) = c_kv(k,y_il,1) + 1;
                c_k(k) = c_k(k) + 1;

                q{i}(l) = k;
            end
        end
        if (mod(iter-offset,block) == 0)
            fprintf('%d/%d\n',iter,iterations);
            for j = 1:documents
                q_sample{j} = q_sample{j} + q{j};
            end
        end
    end

    for j = 1:documents
        q_sample{j} = round(q_sample{j}./n_samples);
    end

    q_dist = zeros([documents,topics]);
    for i = 1:documents
        for j = 1:topics
            s = size(find(q_sample{i}==j));
            q_dist(i,j) = s(1,1);
        end
        q_dist(i,:) = q_dist(i,:)./sum(q_dist(i,:));
    end

    fprintf('done.\n');
end
```

7.6 LDA:n tulokset

Algoritmi tuottaa seuraavat visualisoinnit.



Kuva 7.3: Sanojen dokumenttikohtainen todennäköisyysjakauma



Kuva 7.4: LDA:n tuottama teemojen jakauma

7.7 Tulosten tulkinta

Kuten visualisoinnista nähdään, näillä viittä sanaa käyttävillä dokumenteilla jotka tulee jakaa kahteen teemaan, LDA tekee sanan 3 epätäsmällisyydestä paljon paremmin selkoa. Klassisessa luokittelutehtävässä kuten roskaposti vs. tavalliset viestit NBC löytää dokumentin erot nopeasti, mutta semanttisesti hyödyllisen näkökulman saamiseksi teemoihin LDA on parempi. Myös teeman 'läsnäolon astetta' kuvaava todennäköisyys on hyödyllinen, ja siitä voi olla enemmän hyötyä kuin yksinkertaisesta binäriluokittelusta joka olisi voitu saavuttaa muilla menetelmillä.

Luku 8

Yhteenveto

Tavoitteena oli tehdä matemaattinen katsaus tiettyyn menetelmään luokitella dokumentteja teemoihin. Lukijalle ilmeni mitä tämä menetelmä, bayesilainen päättely diskreetin generatiivisen mallin parametreista, oikeastaan tarkoittaa. Tutkielmassa onnistuttiin käymään läpi kaikki matemaattiset perusteet mitä tälläisen mallin ymmärtäminen vaatii. Aiheen laajuuden vuoksi tietyt graafisten mallien riippumattomuussuhteita koskevat teoreemat täytyi esittää ilman todistusta tai laajempaa kontekstia, mutta valtaosa käsittelystä on kuvattu perusteista lähtien. Ainut esitieto mitä lukijalta edellytettiin oli todennäköisyyslaskennan perusteet. Koska keskityttiin teemojen luokitteluparadigmoihin, ei tutkielmassa kuvattu useita mahdollisia inferenssitapoja, vaan se esitteli vain Gibbs-näytteenottajan jotta mallin toimintaa voitaisiin esitellä todellisella datalla.

Käsittely onnistui tavoitteeseen nähden hyvin: esitys tarjoaa kattavan mutta yksinkertaisen johdatuksen menetelmään. Sen lukeminen mahdollistaa tieteellisen kirjallisuuden ymmärtämisen aiheesta ja monimutkaisempien mallien toteuttamisen. Se saattaa myös abstrahoida yhteisen perustan lukijalle joka on entuudestaan tutustunut vastaaviin malleihin. Jos datan määrä pidetään rajoitettuna, voi pedagogisilla Matlab-toteutuksillamme tutkia niiden toimivuutta käytännössä erilaisille datajoukoille. Ne ovatkin erittäin geneerisiä tätä rajoitusta lukuunottamatta. Mutta kuten mallien kuvauksessa todettiin, niiden muuntaminen suurelle datamäärälle toimivaksi on hyvin yksinkertainen tehtävä, mikä olisi tehnyt mallista paljon vähemmän kuvaavan. Usein tälläistä täydellistä konkreettia toteutusta ei tieteellisessä kirjallisuudessa ole kuvattu lainkaan, ja se onkin aloittelijalle tämän tutkielman merkittävin kontribuutio.

Tulevalle oppimiselle voidaan antaa useita suuntaviivoja. Matemaattisista perusteista kiinnostunut voi tutkia laajemmin Bayes-verkkoja ja muita tapoja jotka muuntavat tietyt ominaisuudet graafia koskeviksi väitteiksi. Vastaa-

vasti itse aiheesta kiinnostunut voi laajentaa tutkimustaan muihin menetelmiin: NBC:n ja LDA:n muunnoksia on kirjallisuudessa valtava määrä, useat näistä käsittelevät todellisen datan tuomia haasteita. Niiden ymmärtämisen tulisi nyt olla paljon helpompaa. Vastaavasti samalla tasolla voidaan tutkia eri jakaumien tai datajoukkojen käyttämistä näissä menetelmissä, ja miksei myös muita inferenssimenetelmiä voisi soveltaa malleihin. Mikäli lukijalle heräsi ajatus käytännön sovelluksesta, on mallimme muuntaminen toiselle ohjelmointikielelle logaritmitempun kanssa varsin yksinkertaista. Tehtävän yleisyyden vuoksi näitä sovelluksia on erittäin paljon, ja voitaneen väittää että todellinen maailma on toimivuuden lopullinen mittari.

Kirjallisuutta

[1] K. Murphy, *Machine Learning: A statistical approach*, 2012

[2] D. Koller, N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*, MIT Press, 2009.

[3] Blei, NG, Jordan, *Latent Dirichlet Allocation*, Journal of Machine Learning Research, 2003

[4] Resnik, Hardisty, *Gibbs Sampling for the Uninitiated*, 2010

[5] R. Neopolitan, *Learning Bayesian Networks*, 2004

[6] C. Grinstead, J. Snell, *Introduction to Probability*, American Mathematical Society, 2006