CS 3750 Advanced Machine Learning
Lecture 8

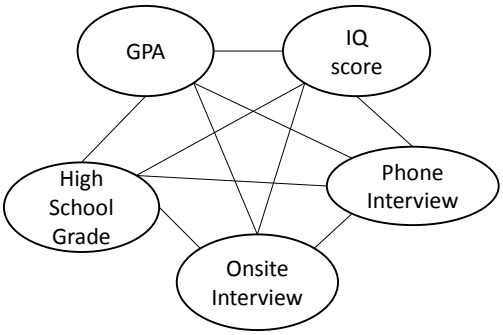
Latent Variable Models

Tianyi Cui
tianyicui@cs.pitt.edu
Sept 20, 2018

Outline

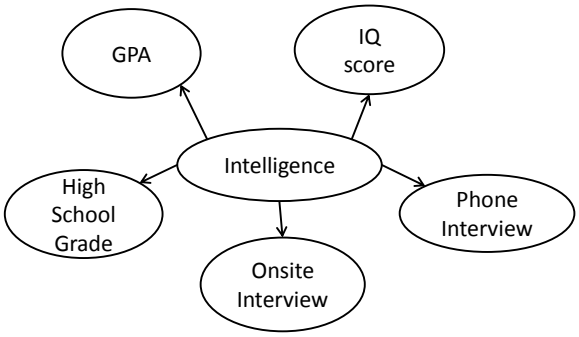
- **Latent Variable Models**
- **Probabilistic Principle Component Analysis**
  - Model Formulation
  - Maximum Likelihood Estimation
  - Expectation Maximum Algorithm for pPCA
- **Cooperative Vector Quantizer Model**
  - Model Formulation
  - Variational Bayesian Learning for CVQ
- **Noisy-OR Component Analyzer**
  - Model Formulation
  - Variational EM for NOCA
- **References**

Latent Variable Models



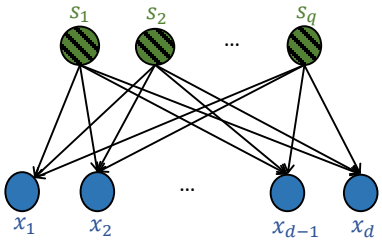
Name	High School Grade	University Grade	IQ score	Phone Interview	Onsite Interview
John	4.0	4.0	120	3/4	?
Helen	3.2	N/A	112	2/4	?
Sophia	3.5	3.6	N/A	4/4	85/100
Jack	3.6	N/A	N/A	3/4	?

Latent Variable Models



Name	High School Grade	University Grade	IQ score	Phone Interview	Onsite Interview
John	4.0	4.0	120	3/4	?
Helen	3.2	N/A	112	2/4	?
Sophia	3.5	3.6	N/A	4/4	85/100
Jack	3.6	N/A	N/A	3/4	?

Latent Variable Models



Latent variables s: q-dimensions

$q < d$

Observed variables x : d-dimensions

Pros:

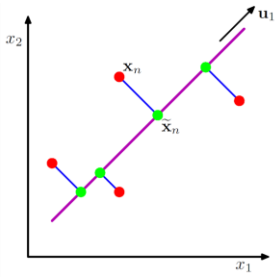
- Dimension reduction
- Explain correlation of observed data at latent variables level
- Inferring state of latent variables

- Cons:
  - Hard to work with

Probabilistic Principle Component Analysis

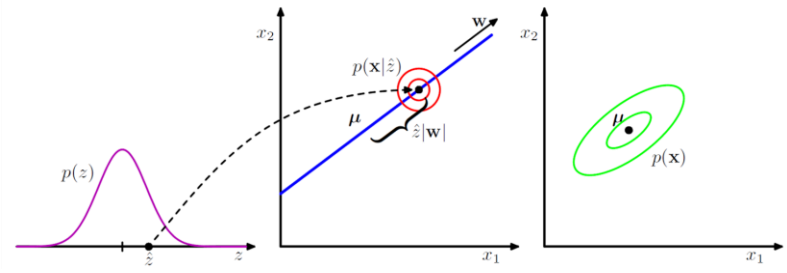
Review: Principle Component Analysis

- Project the data onto a low dimensional space
- Maximum variance of the projected data

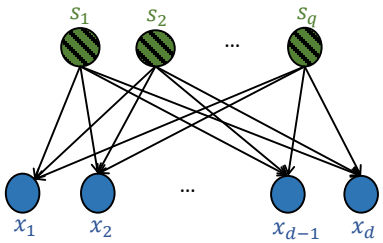


- Limitations:
  - Non-parametric
  - Computational intensive for covariance matrix
  - Missing data

Probabilistic Principle Component Analysis



Probabilistic Principle Component Analysis



Latent variables s: q-dimensions

$s \sim \mathcal{N}(0, I)$

$x|s \sim \mathcal{N}(Ws + \mu, \sigma^2 I)$

$q < d$

Observed variables x : d-dimensions

$x = Ws + \mu + \varepsilon \sim \mathcal{N}(\mu, C_x)$

μ : non zero mean

$\varepsilon \sim \mathcal{N}(0, \sigma^2)$

$C_x = WW^T + \sigma^2$

- Generative Model for PCA
- Take advantage of probabilistic distributions to compute
- Observed variables become independent given latent variables

- Standard PCA: $\sigma^2 \rightarrow 0$

Factor analysis

- **Latent variable model with a linear relationship:**

x ~ Ws + μ + ε

- **W** is a d x q matrix that relates observed variables ***x*** to the latent variables ***s***
- Latent variables: ***s ~ N(0, I)***
- Error (or noise): ***ε ~ N(0, ψ)*** – Gaussian noise
- Location term (mean): ***μ***

Then: ***x ~ N(μ, C_x)***

- where ***C_x=WW^T + ψ*** is the covariance matrix for observed variables ***x***
- the model's parameters ***W, μ*** and ***ψ*** can be found using maximum likelihood estimate

Probabilistic PCA (PPCA)

- A special case of the factor analysis model
  - Noise variances constrained to be equal ($\psi_i=\sigma^2$)

    $\boldsymbol{x \sim Ws + \mu + \varepsilon}$
  - Latent variables: $\boldsymbol{s \sim N(0, I)}$
  - Error (or noise): $\boldsymbol{\varepsilon \sim N(0, \sigma^2 I)}$ (isotropic noise model)
  - Location term (mean): $\boldsymbol{\mu}$

  - $\boldsymbol{x|x \sim N(WX + \mu, \sigma^2 I)}$
  - $\boldsymbol{x \sim N(\mu, C_x)}$
  - where $\boldsymbol{C_x=WW^T + \sigma^2 I}$ is the covariance matrix of $\boldsymbol{x}$

- Normal PCA is a limiting case of probabilistic PCA, taken as the limit as the covariance of the noise becomes infinitesimally small ($\boldsymbol{\psi} = \lim_{\sigma 2 \to 0} \sigma^2 \boldsymbol{I}$)

PPCA (Maximum likelihood PCA)

- Log-likelihood for the Gaussian noise model:
  - $L = -\frac{N}{2}\{d \ln(2\pi) + \ln|\boldsymbol{C_x}| + \mathrm{tr}(\boldsymbol{C_x}^{-1}\mathbf{S})\}$

    $\boldsymbol{C_x=WW^T + \sigma^2}$
- Maximum likelihood estimates for the above:
  - $\boldsymbol{\mu}$: mean of the data
  - $\boldsymbol{S}$ (sample covariance matrix of the observations $\boldsymbol{X}$):

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{X}_n - \boldsymbol{\mu})(\mathbf{X}_n - \boldsymbol{\mu})^{\mathrm{T}}$$

- MLE's for $\boldsymbol{W}$ and $\boldsymbol{\sigma^2}$ can be solved in two ways:
  - closed form (Tipping and Bishop)
  - EM algorithm (Roweis)

Tr(A) = sum of diagonal elements of A

Probabilistic PCA

The likelihood is maximized when:

$$\mathbf{W}_{\mathrm{ML}} = \mathbf{U}_q(\sqrt[2]{\mathbf{\Lambda}_q - \sigma^2\mathbf{I}})\,\mathbf{R}$$

- For $\mathbf{W} = \mathbf{W}_{\mathrm{ML}}$ the maximum $\mathbf{U}_q$ is a d x q matrix where the q column vectors are the principal eigenvectors of **S.**
- $\mathbf{\Lambda}_q$ is a q x q diagonal matrix with corresponding eigenvalues along the diagonal.
- **R** is an arbitrary q x q orthogonal rotation matrix
- Max likelihood estimate for σ^2 is:

$$\sigma^2{}_{\mathrm{ML}} = \frac{1}{d-q}\sum_{j=q+1}^{d}\lambda_j$$

- To find the most likely model given **S**, estimate $\sigma^2{}_{\mathrm{ML}}$ and then $\mathbf{W}_{\mathrm{ML}}$ with $\mathbf{R} = \mathbf{I}$, or you can employ the EM algorithm

Review: General Expectation Maximization (EM)

The key idea of a method:

Compute the parameter estimates iteratively by performing the following two steps:

Two steps of the EM:

1. **Expectation step.** For all hidden and missing variables (and their possible value assignments) calculate their expectations for the current set of parameters Θ'
2. **Maximization step.** Compute the new estimates of Θ by considering the expectations of the different value completions

Stop when no improvement possible

Borrowed from CS2750 Lecture Slides

EM for Probabilistic PCA

Parameters:

μ(mean), W(weighted matrix), σ^2(covariance of noise)

- ML PCA computationally heavy for high dimensional data or large datasets
- $p(s_n, x_n) = (2\pi\sigma^2)^{-d/2} \exp\left\{-\frac{\|x_n - Ws_n - \mu\|^2}{2\sigma^2}\right\} (2\pi)^{-q/2} \exp\left\{-\frac{\|s_n\|^2}{2}\right\}$
- $E[log\ p(\mathrm{X}, S|\mu, W, \sigma^2)]$

$$= -\sum_{n=1}^{N} \{\frac{d}{2}\log(2\pi\sigma^2) + \frac{1}{2}\mathrm{Tr}(E[s_n s_n^T]) + \frac{1}{2\sigma^2}\|x_n - \mu\|^2$$
$$-\frac{1}{\sigma^2}E[s_n]^T W^T(x_n - \mu) + \frac{1}{2\sigma^2}\mathrm{Tr}(E[s_n s_n^T]w^T w)$$

EM for Probabilistic PCA

Parameters:

μ(mean), W(weighted matrix), σ^2(covariance of noise)

- **E-step:** Derive $E[s_n]$, $E[s_n s_n^T]$ from $p(s_n|x_n, \theta)$ using current parameters

$p(s_n) = p(s_n|x_n, \theta) = \frac{p(x_n|s_n,\theta)P(s_n|\theta)}{Z} \sim \mathcal{N}(E[s_n], E[s_n s_n^T])$

  - $E[s_n] = M^{-1}W(x_n - \mu)$
  - $E[s_n s_n^T] = \sigma^2 M^{-1} + E[s_n]E[s_n]^T$
  - $M = W^T\mathrm{W} + o^2 I$

- **M-step:** Maximum expectation

$$\widetilde{w} = \left[\sum_{n=1}^{N}(x_n - \mu)E[s_n]^T\right]\left[\sum_{n=1}^{N}E[s_n s_n^T]\right]^{-1}$$

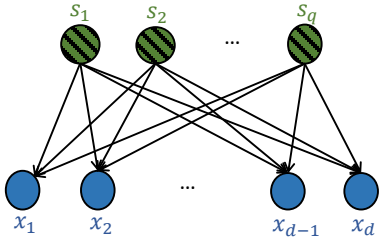
$$\tilde{\sigma}^2 = \frac{1}{Nd}\sum_{n=1}^{N}\{\|x_n - \mu\|^2 - 2E[s_n]^T\widetilde{W}^T(x_n - \mu) + \mathrm{Tr}\big(E[s_n s_n^T]\widetilde{W}^T\widetilde{W}\big)\}$$

Advantages of using EM algorithm in probabilistic PCA models

- **Convergence:**
  - Tipping and Bishop showed (1997) that the only stable local extremum is the *global maximum* at which the true principal subspace is found
- **Complexity:**
  - Methods that explicitly compute the sample covariance matrix have complexities $O(nd^2)$
  - EM algorithm does not require computation of the sample covariance matrix, $O(dnq)$
    - Huge advantage when $q << d$ (# of principal components is much smaller than original # of variabes)

Cooperative Vector Quantizer Model

Cooperative Vector Quantizer (CVQ) Model



Latent variables s: q-dimensions

$s \in \{0,1\}^q, P(s_i|\pi_i) = \pi_i^{s_i}(1-\pi_i)^{1-s_i}$

$$x|s \sim \mathcal{N}\left(\sum_{i=1}^{q} s_i w_i, \sigma^2 I\right)$$

w_i: Gaussian

$q < d$

Observed variables x : d-dimensions

$x = \sum_{i=1}^{q} s_i w_i + \varepsilon$: *Gaussian*

$\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$

$$P(\mathrm{X}, S|\theta) = (2\pi)^{-d/2}\sigma^{-d/2}\exp\left\{-\frac{1}{2\sigma^2}(\mathrm{X}-WS)^T(\mathrm{X}-WS)\right\}\prod_{i=1}^{q}\pi_i^{s_i}(1-\pi_i)^{(1-s_i)}$$

Variational Bayesian Learning of CVQ

Object : Estimate Parameters of the Model: W, π, σ^2

If both x and s are observable, it is easy to calculate

$$\sum_{n=1}^{N}\log\left(P(x_n, s_n|\theta)\right)$$

$$= \sum_{n=1}^{N} - \mathrm{d}\log\sigma - \frac{1}{2\sigma^2}(x_n - Ws_n)^T + \sum_{i=1}^{\mathrm{q}} s_{ni}\log\pi_i + \left(1-s_{n_i}\right)\log(1-\pi_i) + \mathrm{c}$$

- With s unobservable, we need to determine what model structure best describe the data (the model with the highest posterior probability)
- $P(x) = \sum_{\{s\}} P(x, s)$, $\{\mathrm{s}\}$ has 2^q $configurations$
- $\log\left(P(X|\theta)\right) = \sum_{n=1}^{N}\log\left(P(x_n|\theta)\right) = \sum_{n=1}^{N}\log\left(\sum_{\{s\}} P(x_n, s_n|\theta)\right) = \sum_{n=1}^{N}\log\left(\sum_{\{s\}} \boldsymbol{P(s_n|x_n, \theta)}P(x_n|\theta)\right)$
- The computation is exponential, if q is larger, there will be a bottleneck.

Variational Approximation

Object: Find a lower bound for $log\big(P(x_n|\theta)\big)$

$$\log\big(P(X|\theta,\xi)\big) = \log\big(P(\mathrm{X},S|\theta,\xi)\big) - \log\big(P(S|\mathrm{X},\theta,\xi)\big)$$

Average both sides with $Q(S|\lambda)$

$E_{S|\lambda}\log\big(P(X|\theta,\xi)\big)$

$= E_{S|\lambda}\log\big(P(S,X|\theta,\xi)\big) - E_{S|\lambda}\log\big(Q(S|\lambda)\big)$ ... ... $E_{S|\lambda}F(P,Q)$

$+E_{S|\lambda}\log\big(Q(S|\lambda)\big) - E_{S|\lambda}\log\big(P(S|\theta,\xi)\big)$ ... ... $E_{S|\lambda}KL(Q,P)$

$F(P,Q) = \sum_{\{s\}} Q(S|\lambda)\log\big(P(S,X|\theta,\xi)\big) - \sum_{\{x\}} Q(S|\lambda)\log\big(Q(S|\lambda)\big)$

$\mathrm{KL}(Q,P) = \sum_{\{s\}} Q(S|\lambda)\ [\log\big(Q(S|\lambda)\big) - log\big(P(S|X,\theta)\big)]$

Note: if $Q(S)$ is the posterior then the variational EM reduces to the standard EM

Variational EM

Kullback-Leibler (KL) divergence:

- Distance between 2 distribution
- $KL(P|R) = \sum_i P_i \log\frac{P_i}{R_i} \geq 0$ is always positive

$\mathrm{KL}(Q,P) = \sum_{\{s\}} Q(S|\lambda)\ [\log\big(Q(S|\lambda)\big) - log\big(P(S|X,\theta)\big)]$

$= \sum_{\{s\}} Q(S|\lambda)\log\frac{Q(S|\lambda)}{P(S|\mathrm{X},\theta)} \geq 0$

$\log\big(P(X|\theta,\xi)\big) \geq F(P,Q)$

Mean Field Approximation

Object: Choose $Q(H|\lambda)$

Motivation:

- Exact calculation of the posterior is intractable
- When one variable is dependent on many other variables, change of each variable may have limited effect on the original variable.

Haft et al(1997) demonstrated that for any P$(S)$and $Q(S)$, if distribution

$$Q(S) = \prod_{n=1}^{N} Q_n(s_n)$$

One can minimize KL$(Q||P)$ iteratively with respect to $Q_n(s_n)$ while fixing others.

$$Q(S|\lambda) = \prod_{n=1}^{N} Q_n(s_n|\lambda_n)$$

In CVQ model, all latent variables are binary,

$$Q(S|\lambda) = \prod_{n=1}^{N} Q_n(s_n|\lambda_n)$$

$$Q_n(s_n|\lambda_n) = \prod_{i=1}^{q} Q_n(s_{ni}|\lambda_{n_i}) = \prod_{i=1}^{q} \lambda_{n_i}^{s_{ni}}(1-\lambda_{ni})^{1-s_{ni}}$$

Variational EM

$$F(P,Q)$$
$$= -\mathrm{d}\, log\sigma - \frac{1}{2\sigma^2}\left[x^T x - 2\sum_{i=1}^{q}(\lambda_i w_i)x + \sum_{i=1}^{q}\sum_{j=1}^{q}\left[\lambda_i\lambda_j + \delta_{ij}(\lambda_i - \lambda_i^2)\right]w_i^T w_j\right] + \sum_{i=1}^{q}\lambda_i \log\pi_i$$
$$+ (1-\lambda_i)\log(1-\pi_i) + \sum_{i=1}^{q}\lambda_i \log\lambda_i + (1-\lambda_i)\log(1-\lambda_i)$$

$$\frac{\partial}{\partial\lambda_u}F = 0$$

E-step:

Optimize $F(P,Q)$ with respect to $\lambda = \lambda_1, \lambda_2, \ldots, \lambda_N$ while keeping $\Theta(W, \pi, \sigma^2)$ fixed
Iterate a set fixed point equations for all indexes u $= 1 \ldots$ k and for all n

$$\lambda_u = g\left(\frac{1}{\sigma^2}\left(x - \sum_{j\neq u}\lambda_j w_j\right)^T w_u - \frac{1}{2\sigma^2}w_u^T w_u + \log\frac{\pi_u}{1-\pi_u}\right)$$

$$g(x) = \frac{1}{1+\mathrm{e}^{-x}}$$

Variational EM

$$F(P,Q)$$
$$= -\mathrm{d}\, log\sigma - \frac{1}{2\sigma^2}\left[x^T x - 2\sum_{i=1}^{q}(\lambda_i w_i)x + \sum_{i=1}^{q}\sum_{j=1}^{q}\left[\lambda_i\lambda_j + \delta_{i_j}(\lambda_i - \lambda_i^2)\right]w_i^T w_j\right]$$
$$+\sum_{i=1}^{q}\lambda_i \log\pi_i + (1-\lambda_i)\log(1-\pi_i) + \sum_{i=1}^{q}\lambda_i \log\lambda_i + (1-\lambda_i)\log(1-\lambda_i)$$

M-step:

Optimize $F(P,Q)$ with respect to $\Theta(W,\pi,\sigma^2)$ while keeping $\lambda = \lambda_1, \lambda_2, \ldots, \lambda_N$

$$\frac{\partial}{\partial \pi_u}F = 0 \rightarrow \pi_u = \frac{\sum_{n=1}^{N}\lambda_{nu}}{N}$$
$$\frac{\partial}{\partial w_{uv}}F = \sum_{n=1}^{N} -\frac{1}{2\sigma^2}\left[\lambda_{nv}X_{nu} + 2\sum_{j\neq v}\lambda_{nv}\lambda_{nj}W_{uj} + 2\lambda_{nv}W_{uv}\right] = 0 \rightarrow W$$
$$\frac{\partial}{\partial \sigma^2}F = -\frac{\mathrm{d}}{\sigma} + \frac{1}{\sigma^3}[.] = 0 \rightarrow \sigma^2$$

Noisy-OR Component Analyzer

Noisy-OR

- **A generalization of the logical OR**

Assumptions:

- All possible causes U_i for an event X are modeled using nodes (random variables) and their values, with T (or 1) reflecting the presence of the cause , and F (or 0) its absence
- If one needs to represent unknown causes one can add a leak node
- **Parameters:** For each cause U_i define an (independent) probability qi that represents the probability with which the cause does not lead to X = T (or 1), or in other words, it represents the probability that the positive value of variable X is inhibited when U_i is present

$$\mathrm{p}(x = 1|U_1, \dots, U_\mathrm{j}, \neg U_{j+1}, \dots, \neg U_k) = 1 - \prod_{i=1}^{j} q_i$$

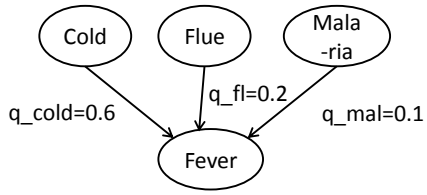
$$p(x = 0|U_1, \dots, U_\mathrm{j}, \neg U_{j+1}, \dots, \neg U_k) = \prod_{i=1}^{j} q_i$$

Please note that the negated causes $\neg U_i$ (reflecting the absence of the cause) do not have any influence on X

Noisy-OR Example

$$\mu(x = 1|U_1, \dots, U_\mathrm{j}, \neg U_{j+1}, \dots, \neg U_k) = 1 - \prod_{i=1}^{j} q_i$$

$$\mu(x = 0|U_1, \dots, U_\mathrm{j}, \neg U_{j+1}, \dots, \neg U_k) = \prod_{i=1}^{j} q_i$$



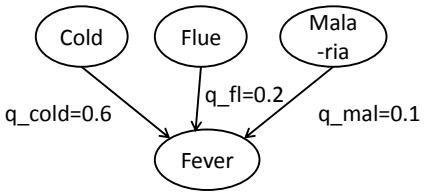
Cold	Flu	Malaria	μ(Fever)	$\mu(\neg Fever)$
F	F	F	0	1
F	F	T	0.9	0.1
F	T	F	0.8	0.2
F	T	T	0.98	$0.02 = 0.2 \times 0.1$
T	F	F	0.4	0.6
T	F	T	0.94	$0.06 = 0.6 \times 0.1$
T	T	F	0.88	$0.12 = 0.6 \times 0.2$
T	T	T	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

Noisy-OR parameter reduction

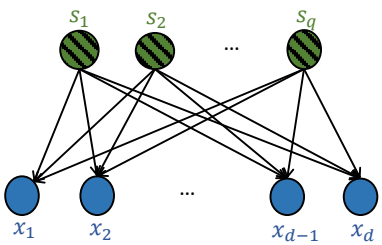
- Please note that in general the number of entries defining the CPT (conditional probability table) grows exponentially with the number of parents;
  - for q binary parents the number is : 2^q
- For the noisy-or CPT the number of parameters is q + 1

Example:
CPT: 8 different combination of values for 3 binary parents

Noisy-or: 3 parameters



Noisy-OR Component Analyzer (NOCA)



Latent variables s: $(q+1)$-dimensions

$s \in \{0,1\}^q, P(s_i|\pi_i) = \pi_i^{s_i}(1-\pi_i)^{1-s_i}$

Loading Matrix: $\boldsymbol{p} = \{p_{ij}\}_{j=1,\dots,d}^{i=1,\dots,q}$

$q < d$

Observed variables x : d-dimensions

$x \in \{0,1\}^d$

$$P(x) = \sum_{(s)} \left(\prod_{j=1}^{d} P(x_j|s)\right)\left(\prod_{i=1}^{q} P(s_i)\right)$$

Variational EM for NOCA

- Similar to what we did for CVQ, we simplify the distribution with a decomposable Q(s)

$$\log\big(P(x|\theta)\big) = \log\left(\prod_{n=1}^{N} P(x_n|\theta)\right) = \sum_{n=1}^{N} \log\left[\sum_{\{s\}} P(x_n, s_n|\theta)\right]$$

$$= \sum_{n=1}^{N} \log\left[\sum_{\{s\}} P(x_n, s_n|\theta, q_n)\frac{Q(s_n)}{Q(s_n)}\right] \geq \sum_{n=1}^{N}\left[\sum_{\{s_n\}} E_{s_n}\log\big(P(x_n, S_n|\theta)\big) - E_{s_n}\log\big(Q(S_n)\big)\right]$$

- $\log(P(x_n, s_n|\theta, q_n)$ still can not be solved easily
  - Noisy-Or is not in exponential family

Variational EM for NOCA

A further lower bound is required

- **Jensen's inequality:** $\boldsymbol{f(a + \sum_j q_j x_j) \geq \sum_j q_i f(a + x_j)}$

$$P(x_j|s) = \left[1 - (1 - p_{0j})\prod_{i=1}^{q}(1 - p_{ij})^{s_i}\right]^{x_j}\left[(1 - p_{0j})\prod_{i=1}^{q}(1 - p_{ij})^{s_i}\right]^{(1-x_j)}$$

Set $\boldsymbol{\theta_{ij} = -\log(1 - p_{ij})}$

$$P(x_j|s) = \exp\left[x_j \log\left(1 - \exp\left\{-\theta_{0j} - \sum_{i=1}^{q}\theta_{ij}s_i\right\}\right) + (1 - x_j)\left(-\theta_{0j} - \sum_{i=1}^{q}\theta_{ij}s_i\right)\right]$$

$P(x_j|s)$ does not factorize for $x_j = 1$

$$P(x_j = 1|s) = \exp[\log\left(1 - \exp\left\{-\theta_{0j} - \sum_{i=1}^{q}\theta_{ij}s_i\right\}\right)]$$

$$= \exp\left[\log\left(1 - \exp\left\{-\theta_{0j} - \sum_{i=1}^{q}\theta_{ij}s_i\frac{q_j(i)}{q_j(i)}\right\}\right)\right] \geq \exp[\sum_{i=1}^{q} q_j(i)\log\left(1 - \exp\left\{-\theta_{0j} - \frac{\theta_{ij}s_i}{q_j(i)}\right\}\right)]$$

$$= \exp[\sum_{i=1}^{q} q_j(i)\,[s_i\log\left(1 - \exp\left\{-\theta_{0j} - \frac{\theta_{ij}}{q_j(i)}\right\}\right) + (1 - s_i)\log(1 - \exp\{-\theta_{0j}\})]]$$

$$= \prod_{i=1}^{q} \exp[q_j(i)s_i[\log\left(1 - \exp\left\{-\theta_{0j} - \frac{\theta_{ij}}{q_j(i)}\right\}\right) - \log((1 - \exp\{-\theta_{0j}\}))] + q_j(i)\log((1$$

Variational EM for NOCA

A further lower bound is required

$$\log\big(P(x|\theta)\big)$$
$$\geq \sum_{n=1}^{N} \Big[\sum_{\{s_n\}} E_{s_n} \log P(x_n, s_n|\theta) - E_{s_n} \log Q(s_n)\Big]$$
$$\geq \sum_{n=1}^{N} \Big[\sum_{\{s_n\}} E_{s_n} \log\Big(\tilde{P}(x_n, s_n|\theta, q_n)\Big) - E_{s_n} \log\big(Q(s_n)\big)\Big]$$
$$= \sum_{n=1}^{N} \Big[\sum_{\{s_n\}} E_{s_n} \log\Big(\tilde{P}(x_n|s_n, \theta, q_n)P\big(s_n|\theta\big)\Big) - E_{s_n} \log\big(Q(s_n)\big)\Big]$$
$$= \sum_{n=1}^{N} \mathcal{F}_n\big(x_n, Q(s_n)\big)$$
$$= \mathcal{F}(x, Q(s))$$

Variational EM for NOCA

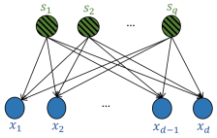
Parameters: $q_n, \theta_{ij}, \theta_{0j}$

- **E-step: update q_n to optimize F_n**
- $q_{nj}(\mathrm{i}) \Leftarrow \langle s_{ni} \rangle_{Q(S_n)} \frac{q_{nj}(\mathrm{i})}{\log\left(1-\mathrm{e}^{-\theta_{0j}}\right)} \Big[\log\big(1 - A^n(\mathrm{i}, j)\big) - \frac{\theta_{ij}}{q_{nj}(\mathrm{i})} \frac{A^n(\mathrm{i}, j)}{1-A^n(\mathrm{i}, j)} -$

Summary

	pPCA	CVQ	NOCA
Latent Variables s-q	$s \sim \mathcal{N}(0, I)$	$s \in \{0,1\}^q$	$s \in \{0,1\}^q$
Observed Variables x-d	$x = Ws + \mu + \varepsilon \sim \mathcal{N}(\mu, C)$ μ : non zero mean $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ $C = WW^T + \sigma^2$	$x = \sum_{i=1}^{q} s_i w_i + \varepsilon$: *Gaussian* $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$	$x \in \{0,1\}^d$
Dependency	$x \mid s \sim \mathcal{N}(Ws + \mu, \sigma^2 I)$	$x \mid s \sim \mathcal{N}(\sum_{i=1}^{q} s_i w_i, \sigma^2 I)$ w_i: Gaussian	Loading Matrix: $\boldsymbol{p} = \{p_{ij}\}_{j=1,\ldots,d}^{i=1,\ldots,q}$
Parameter Estimation Method	ML, EM	Variational EM	Variational EM

- The dimension reduction mapping can also be interpreted as a variational autoencoder.



Reference

- CS2750 Spring 2018 Lecture 17 Slides
- Week2 Slides from Coursera Bayesian Methods for Machine Learning
- http://www.blutner.de/Intension/Noisy%20OR.pdf
- C. Bishop. Pattern recognition and Machine Learning. Chapter 12.2.
- Michael E. Tipping, Chris M. Bishop. Probabilistic Principal Component Analysis, 1999. (TR in 1997)
- Sam Roweis. EM Algorithms for PCA and SPCA. NIPS-1998.
- Singliar, Hauskrecht. Noisy-or Component Analysis and its Application to Link Analysis. 2006
- Jaakkola, Tommi S., and Michael I. Jordan. "Variational probabilistic inference and the QMR-DT network." Journal of artificial intelligence research 10 (1999): 291-322.