(Deep) Generative Models for Unsupervised Learning (Part 1 – Classical Models)

CS698X: Topics in Probabilistic Modeling and Inference

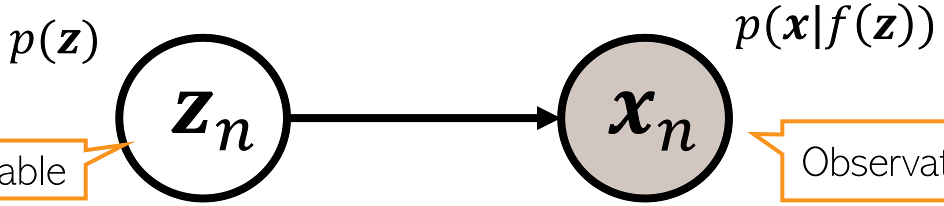
Piyush Rai

Generative Models for Unsupervised Learning

- Many generative models for unsupervised learning have this form

Can be used as a "representation" or "code" or "embedding" (often low-dim) for $\boldsymbol{x}_n$

Latent variable



Observation

There also exist generative models that do not have latent variables (example: NADE)

- Depending on the prior, likelihood, and $\boldsymbol{f}$, various latent factor models arise, e.g.,
  - Factor Analysis and Probabilistic PCA
  - Gamma-Poisson latent factor model
  - Latent Dirichlet Allocation (LDA)
  - Gaussian Process Latent Variable Models (GPLVM) – $\boldsymbol{f}$ is nonlinear modeled by a GP
  - Variational Autoencoders (VAE) - $\boldsymbol{f}$ is nonlinear modeled by a neural net
  - Generative Adversarial Network (GAN) – $\boldsymbol{f}$ is nonlinear modeled by a neural net and the likelihood is only implicitly defined
  - .. and several others..

Among these, VAE and GAN are deep generative models (built using deep neural nets)

Some Classical Models

Factor Analysis and Probabilistic PCA

- Assumption: Latent variables $\boldsymbol{z}_n \in \mathbb{R}^K$ typically assumed to have a Gaussian prior
  - If we want sparse latent variable, can use Laplace or spike-and-slab prior on $\boldsymbol{z}_n$
  - More complex extensions of FA/PPCA use a mixture of Gaussians prior on $\boldsymbol{z}_n$

- Assumption: Observations $\boldsymbol{x}_n \in \mathbb{R}^D$ typically assumed to have a Gaussian likelihood
  - Other likelihood models (e.g., exp-family) can also be used if data not real-valued

- Relationship between $\boldsymbol{z}_n$ and $\boldsymbol{x}_n$ modeled by a noisy linear mapping

$$\boldsymbol{x}_n = \sum_{k=1}^{K} \boldsymbol{w}_k z_{nk} + \epsilon_n = \boldsymbol{W}\boldsymbol{z}_n + \epsilon_n$$

Zero-mean diagonal or spherical Gaussian noise

$$p(\mathbf{z}_n) = \mathcal{N}(\mathbf{z}_n | 0, \mathbf{I})$$
$$p(\mathbf{x}_n|\mathbf{z}_n) = \mathcal{N}(\boldsymbol{x}_n | \mathbf{W}\mathbf{z}_n, \Psi)$$

Diagonal for FA, spherical for PPCA

- Unknowns $\boldsymbol{W}$, $\boldsymbol{z}_n$'s, and $\boldsymbol{\Psi}$ can be learned
  - EM, VI, MCMC

- Inference is mostly straightforward if the model has local conjugacy

Some Other Classical Models

Non-negative priors often give a nice interpretability to such latent variable models (will see some more examples of such models shortly)

- Gamma-Poisson latent factor model

Popular for modeling count-valued data (in text analysis, recommender systems, etc)

  - Assumes K-dim non-negative latent variable $\mathbf{z}_\mathrm{n}$ and D-dim count-valued observations $\mathbf{x}_\mathrm{n}$
  - An example: Each $\mathbf{x}_\mathrm{n}$ is the word-count vector representing a document

$$p(\mathbf{z}_\mathrm{n}) = \prod_{k=1}^{K} \mathrm{Gamma}(\mathrm{z}_\mathrm{nk}|\mathrm{a}_\mathrm{k}, \mathrm{b}_\mathrm{k}))$$
$$p(\mathbf{x}_\mathrm{n}|\mathbf{z}_\mathrm{n}) = \prod_{d=1}^{D} \mathrm{Poisson}(\mathrm{x}_\mathrm{nd}|f(\mathbf{w}_\mathrm{d}, \mathbf{z}_\mathrm{n}))$$

This is the rate of the Poisson. It should be non-negative, $\exp(\mathbf{w}_d^\top \boldsymbol{z}_n)$, or simply $\mathbf{w}_d^\top \boldsymbol{z}_n$ if $\boldsymbol{w}_d$ is also non-negative (e.g., using a gamma/Dirichlet prior on it)

  - This can be thought of as a probabilistic non-negative matrix factorization model

- Dirichlet-Multinomial/Multinoulli PCA
  - Assumes K-dim non-negative latent variable $\mathbf{z}_\mathrm{n}$ and D categorical obs $\mathbf{x}_\mathrm{n} = \{\boldsymbol{x}_{nd}\}_{d=1}^{D}$
  - An example: Each $\mathbf{x}_\mathrm{n}$ is a document with D words in it (each word is a categorical value)

Also sums to 1

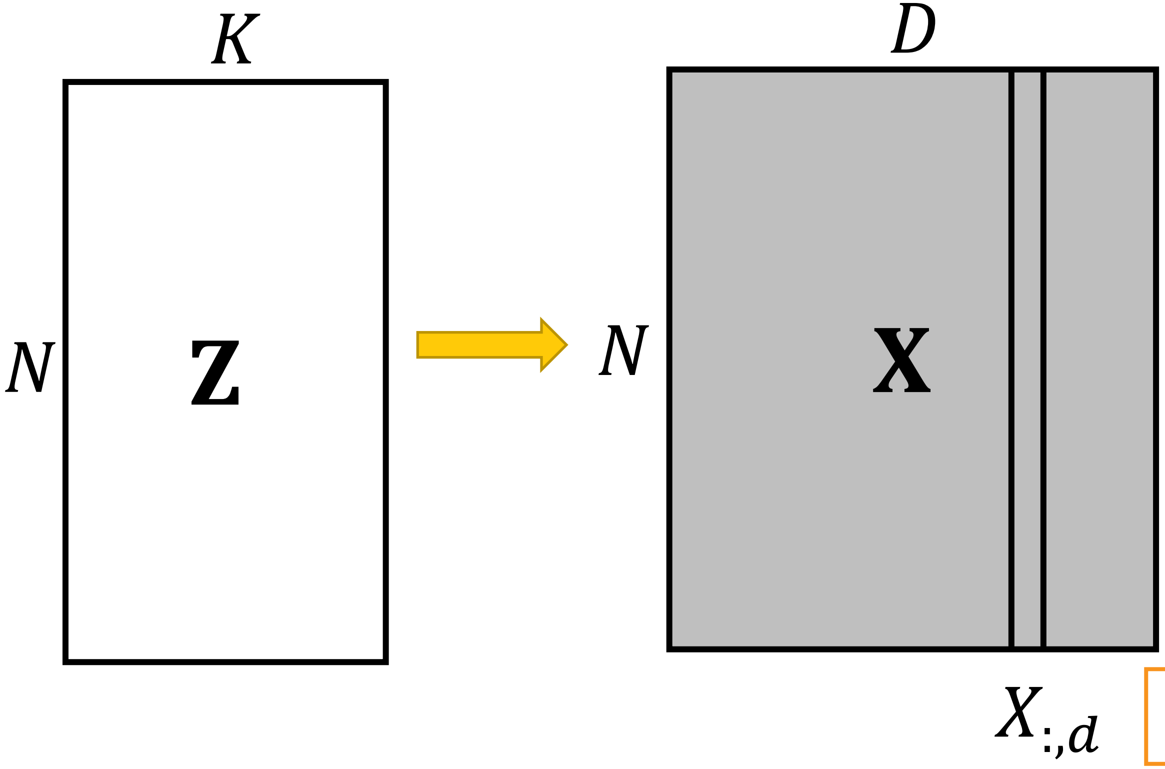
$$p(\mathbf{z}_\mathrm{n}) = \mathrm{Dirichlet}(\mathbf{z}_\mathrm{n}|\boldsymbol{\alpha})$$
$$p(\mathbf{x}_\mathrm{n}|\mathbf{z}_\mathrm{n}) = \prod_{d=1}^{D} \mathrm{Multinoulli}(\mathrm{x}_\mathrm{nd}|f(\mathbf{w}_\mathrm{d}, \mathbf{z}_\mathrm{n}))$$

This should give the probability vector of the multinoulli over x_{nd}. It should be non-negative and should sums to 1

Some Other Classical Models

- Gaussian Process Latent Variable Model* (GPLVM)
  - Uses a GP to define a $\boldsymbol{z}_n$ to $\boldsymbol{x}_n$ (nonlinear) mapping

Similar to learning GP for supervised learning except not only the GP function f_d but also the "inputs" $\mathbf{Z}$ need to be learned



Modeled by a GP

Need D such GPs

$$X_{:,d} = f_d(\mathbf{Z}) + \epsilon_d$$

$N \times 1$ "output"

$N \times K$ "input" matrix

Observation noise

$$p(\mathbf{x}_n|\mathbf{z}_n) = \mathcal{N}(\boldsymbol{x}_n|\mathbf{W}\mathbf{z}_n, \sigma^2\mathbf{I}_D)$$

$$p(\mathbf{W}) = \prod_{d=1}^{D} \mathcal{N}(\boldsymbol{w}_d|0, \mathbf{I}_K)$$

After integrating out $\mathbf{W}$

$$p(\mathbf{X}|\mathbf{Z}, \sigma^2) = \prod_{d=1}^{D} \mathcal{N}(\mathbf{X}_{:,d}|\mathbf{0}, \mathbf{Z}\mathbf{Z}^\top + \sigma^2\mathbf{I}_D)$$

$$= (2\pi)^{-DN/2}|\mathbf{K}_z|^{-D/2}\exp\left(-\frac{1}{2}\mathrm{tr}(\mathbf{K}_z^{-1}\mathbf{X}\mathbf{X}^\top)\right)$$

Do MLE on this to get $\mathbf{Z}$

$N \times N$ kernel matrix defined over $\mathbf{Z}$

*Probabilistic Non-linear Principal Component Analysis with Gaussian Process Latent Variable Models (Lawrence, 2005)

An Aside: Generative Models for Paired Data

- FA/PPCA/other models can be extended to jointly model paired observations, e.g.,
  - Image and its caption
  - EEG data and fMRI data for the same subject
  - Parallel text from two languages
- A popular method is Canonical Correlation Analysis (CCA)
- Its probabilistic version is probabilistic CCA

PCA on paired observations

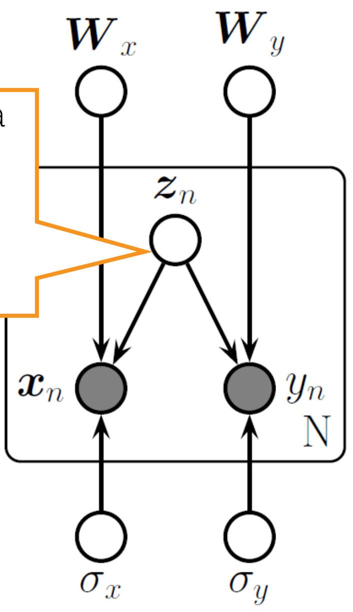
Basically, probabilistic PCA on paired observations

$$p(\mathbf{z}_n) = \mathcal{N}(\mathbf{z}_n|\mathbf{0}, \mathbf{I})$$

$$p(\mathbf{x}_n|\mathbf{z}_n) = \mathcal{N}(\mathbf{x}_n| + \mathbf{W}_x\mathbf{z}_n + \boldsymbol{\mu}_x, \sigma_x^2\mathbf{I})$$

$$p(\mathbf{y}_n|\mathbf{z}_n) = \mathcal{N}(\mathbf{y}_n|\mathbf{W}_y\mathbf{z}_n + \boldsymbol{\mu}_y, \sigma_y^2\mathbf{I})$$

Shared by both data sources – Represents/fused information from both data sources



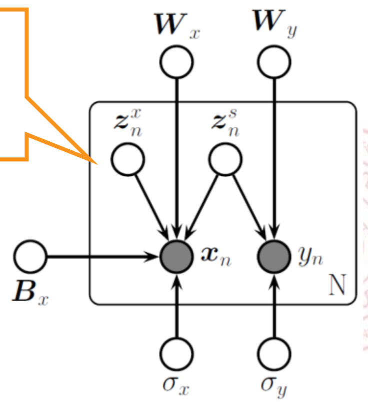
Partial Least Squares (PLS)

Each $\boldsymbol{z_n}$ has two parts, shared and private to $\boldsymbol{x_n}$

$$p(\mathbf{z}_n) = \mathcal{N}(\mathbf{z}_n^s|\mathbf{0}, \mathbf{I})\mathcal{N}(\mathbf{z}_n^x|\mathbf{0}, \mathbf{I})$$

$$p(\mathbf{y}_n|\mathbf{z}_n) = \mathcal{N}(\mathbf{W}_y\mathbf{z}_n^s + \boldsymbol{\mu}_y, \sigma_y^2\mathbf{I})$$

$$p(\mathbf{x}_n|\mathbf{z}_n) = \mathcal{N}(\mathbf{W}_x\mathbf{z}_n^s + \mathbf{B}_x\mathbf{z}_n^x + \boldsymbol{\mu}_x, \sigma_x^2\mathbf{I})$$



Coming Up Next

- Another classical latent variable model for text data
  - Latent Dirichlet Allocation (Topic Model)