Latent-Variable Models

Qinliang Su （苏勤亮）

Sun Yat-sen University

suqliang@mail.sysu.edu.cn

Outline

- Introduction of Latent-Variable Models
- Gaussian Latent-Variable Model
- Gaussian Mixture Model
- Examples of other LVMs

Unsupervised Probabilistic Modeling

- In supervised learning, both regression and classification can be understood as learning *conditional probability distributions*

$$p(y|\boldsymbol{x};\boldsymbol{w})$$

  - In regression, the conditional pdf is assumed of the form

$$p(y|\boldsymbol{x};\boldsymbol{w}) = \mathcal{N}(y;\boldsymbol{w}^T\boldsymbol{x},\sigma^2)$$

  - For classification, the conditional pdf is assumed of the form

$$p(y|\boldsymbol{x}) = \big(\sigma(\boldsymbol{x}\boldsymbol{w})\big)^{y} \cdot \big(1-\sigma(\boldsymbol{x}\boldsymbol{w})\big)^{1-y}$$

$$p(\boldsymbol{y}|\boldsymbol{x}) = \prod_{k=1}^{K}[softmax_k(\boldsymbol{W}\boldsymbol{x})]^{y_k}$$

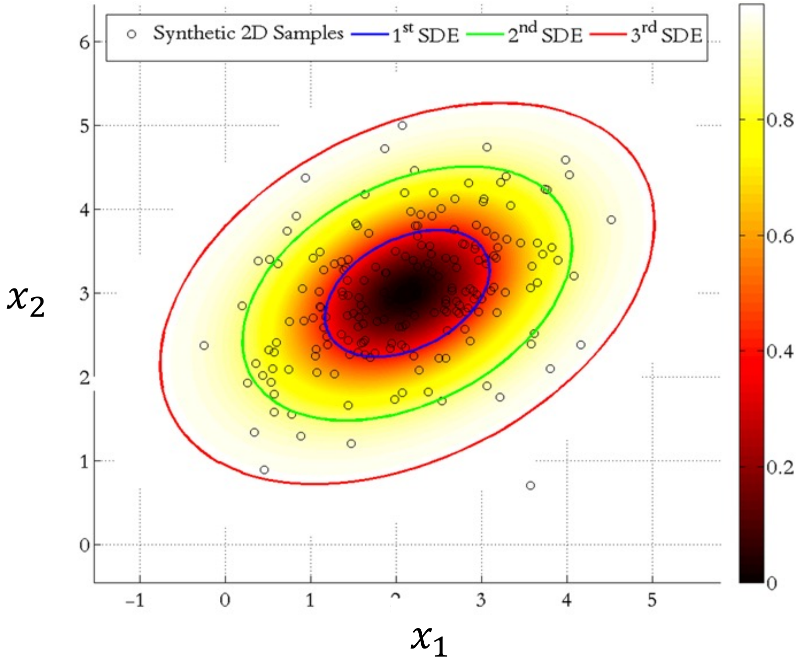
- Unsupervised learning can also be understood from the perspective of learning probability distributions. But it only concerns the distribution of *input data* $\boldsymbol{x}$

$$p(\boldsymbol{x}; \boldsymbol{w})$$

- Modeling $\boldsymbol{x}$ is much difficult than modeling the label y. A naïve way is to restrict $p(\boldsymbol{x}; \boldsymbol{w})$ to the Gaussian form

$$p(\boldsymbol{x};\ \boldsymbol{w}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$$

  - $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are optimized to describe the data points $\{\boldsymbol{x}^{(n)}\}_{n=1}^{N}$ best

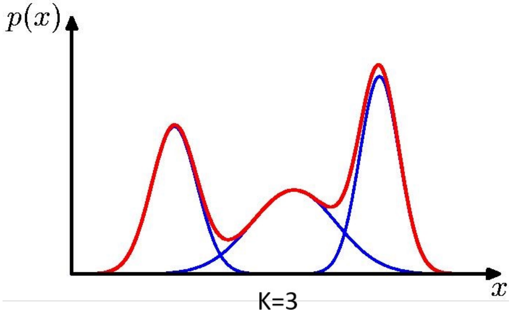


Obviously, the representational ability of the model is very limited

How does Latent Variables Arise?

- Reason 1: Building expressive models using the composition of simple models
  - Suppose there exists a simple categorical distribution $p(z) = Cat(K, \boldsymbol{\pi})$ and a Gaussian distribution $p(x) = \mathcal{N}(x|\mu, \sigma^2)$
  - By using them separately, only simple statistical relations can be modelled
  - But if we composite them as $p(x, z) = p(x|z)p(z)$, the induced marginal distribution $p(x)$ could be much more expressive

$$p(x) = \sum_{z} p(x|z)p(z) = \sum_{k=1}^{K} \pi_z \mathcal{N}(x|\mu_z, \sigma_z^2)$$

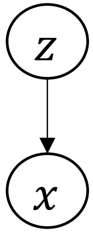


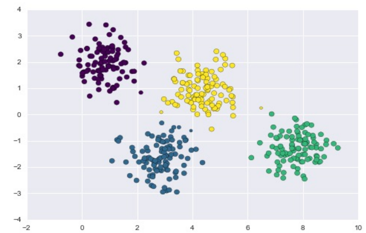
  - Theoretically, it is able to represent any complex distribution

- Reason 2: hidden structures in the data
  1) Data with hidden cluster structure

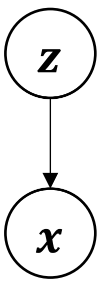
$$z_n \sim Cat(K, \boldsymbol{\pi})$$

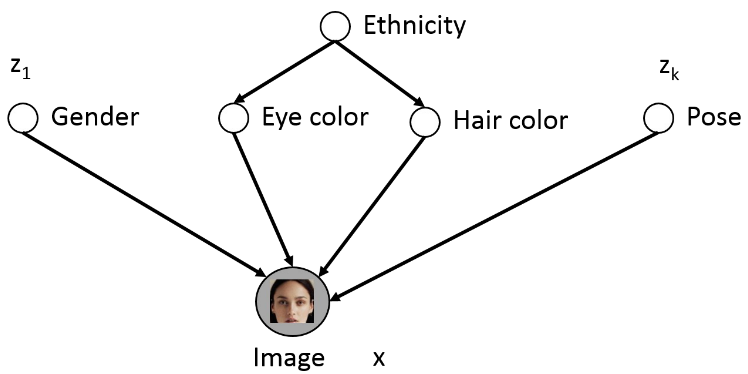
$$x_n \sim \mathcal{N}(x|\boldsymbol{\mu}_{z_n}, \boldsymbol{\Sigma}_{z_n})$$





  2) Topic model for documents
  3) Image Modeling





➢ In the examples above, the latent variables $\boldsymbol{z}$ often correspond to high-level features

➢ *If the latent structure is respected, more interpretable models could be obtained*

LVMs in General Form

- LVMs: a probabilistic model with latent variables

$$p(\boldsymbol{x}, \boldsymbol{z})$$

  - $\boldsymbol{x}$ is the random variable of interest
  - $\boldsymbol{z}$ is the latent variable (nuisance variable)

➢ Sometimes, there may exist multiple latent variables $\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_K$

$$p(\boldsymbol{x}, \boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_K)$$

- The probabilistic model *w.r.t.* the interested variable $\boldsymbol{x}$ is

$$p(\boldsymbol{x}) = \int_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z} \quad \text{or} \quad p(\boldsymbol{x}) = \int_{\boldsymbol{z}_1 \cdots \boldsymbol{z}_K} p(\boldsymbol{x}, \boldsymbol{z}_1, \cdots, \boldsymbol{z}_K) d\boldsymbol{z}_1 \cdots d\boldsymbol{z}_K$$

Outline

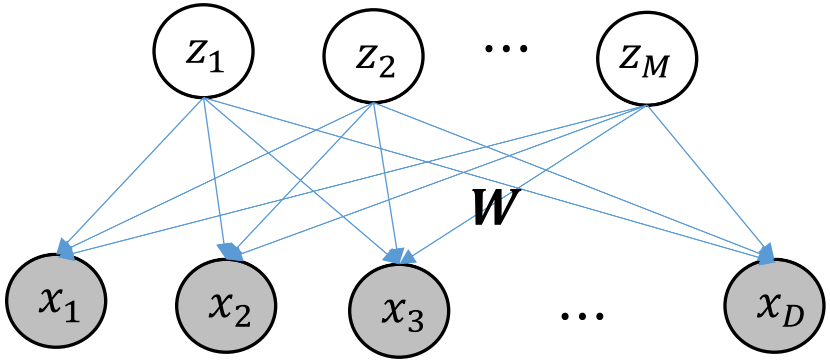
- Introduction of Latent-Variable Models
- Gaussian Latent-Variable Model
- Gaussian Mixture Model
- Examples of other LVMs

- Assuming both of the prior and conditional pdfs are independent Gaussian

Prior distribution: $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}; \mathbf{0}, \boldsymbol{I})$

Likelihood function: $p(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu}, \sigma^2\boldsymbol{I})$

- Actually, the model describes how data samples $\boldsymbol{x}$ are generated



$$\boldsymbol{z} = [z_1, \cdots, z_M] \ \& \ \boldsymbol{x} = [x_1, \cdots, x_D]$$

Training Objective

- Given the samples $\{\boldsymbol{x}_n\}_{n=1}^N$, the question becomes how to train the model $p(\boldsymbol{x}, \boldsymbol{z})$ to make it able to describe the data best

- The model parameter $\boldsymbol{W}$ can be learned by maximizing the log-likelihood

$$\max_{\boldsymbol{W}} \sum_{n=1}^{N} \log p(\boldsymbol{x}_n)$$

- In LVMs, what we have is the joint pdf

$$p(\boldsymbol{x}_n, \boldsymbol{z}_n) = p(\boldsymbol{x}_n | \boldsymbol{z}_n) p(\boldsymbol{z}_n)$$

$$= \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{W}\boldsymbol{z}_n + \boldsymbol{\mu}, \sigma^2 \boldsymbol{I}) \mathcal{N}(\boldsymbol{z}_n; \boldsymbol{0}, \boldsymbol{I}),$$

But what we need is to optimize $p(\boldsymbol{x}_n)$

Marginal Distribution $p(x)$

- The most direct method is to compute the marginal pdf first

$$p(\boldsymbol{x}_n) = \int_{\boldsymbol{z}_n} p(\boldsymbol{x}_n, \boldsymbol{z}_n) d\boldsymbol{z}_n$$

- Deriving the analytical expression for $p(\boldsymbol{x}_n)$ is impossible in most scenarios due to existence of the integration

- But for the Gaussian case, we can easily obtain it as

$$p(\boldsymbol{x}_n) = \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I})$$

A simple method to derive the marginal distribution

- From the model

$$\mathcal{N}(\boldsymbol{x}_n; \boldsymbol{W}\boldsymbol{z}_n + \boldsymbol{\mu}, \sigma^2 \boldsymbol{I})\mathcal{N}(\boldsymbol{z}_n; \boldsymbol{0}, \boldsymbol{I}),$$

  the data point $\boldsymbol{x}_n$ can be understood as generated from

$$\boldsymbol{x}_n = \boldsymbol{\mu} + \boldsymbol{W}\boldsymbol{z}_n + \boldsymbol{\epsilon}_n$$

  where $\boldsymbol{z}_n \sim \mathcal{N}(\boldsymbol{z}_n; \boldsymbol{0}, \boldsymbol{I})$ *and* $\boldsymbol{\epsilon}_n \sim \mathcal{N}(\boldsymbol{z}_n; \boldsymbol{0}, \sigma^2 \boldsymbol{I})$

- That is, data $\boldsymbol{x}_n$ can be understood as generated from $\boldsymbol{z}_n$ and $\boldsymbol{\epsilon}_n$ as $\boldsymbol{x}_n = \boldsymbol{\mu} + \boldsymbol{W}\boldsymbol{z}_n + \boldsymbol{\epsilon}_n$

> Theorem: A linear combination of Gaussian random variables also follows a Gaussian distribution

- Therefore, $\boldsymbol{x}_n$ also follows a Gaussian distribution

How can a Gaussian distribution be determined?

$\Rightarrow$ Mean & Covariance

- Mean & Covariance

Mean: $\mathbb{E}[\boldsymbol{x}_n] = \boldsymbol{\mu} + \boldsymbol{W}\mathbb{E}[\boldsymbol{z}_n] + \mathbb{E}[\boldsymbol{\epsilon}_n] = \boldsymbol{\mu}$

Covariance: $\mathbb{E}[(\boldsymbol{x}_n - \boldsymbol{\mu})(\boldsymbol{x}_n - \boldsymbol{\mu})^T] = \boldsymbol{W}\mathbb{E}[\boldsymbol{z}_n\boldsymbol{z}_n^T]\boldsymbol{W}^T + \mathbb{E}[\boldsymbol{\epsilon}_n\boldsymbol{\epsilon}_n^T]$

$= \boldsymbol{W}\boldsymbol{W}^T + \sigma^2\boldsymbol{I}$

- Thus, the marginal distribution of $\boldsymbol{x}_n$ is

$$p(\boldsymbol{x}_n) = \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2\boldsymbol{I})$$

Training by Maximizing $\log p(x)$

- Given the training dataset $\{\boldsymbol{x}_n\}_{n=1}^N$, to learn $\boldsymbol{W}, \boldsymbol{\mu}$ and σ^2, what we need to do is to optimize the log-probability

$$\log p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N)$$

- Due to $p(\boldsymbol{x}_n) = \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I})$, we have

$$\log p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N) = \sum_{n=1}^N \log \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I})$$

- It can be further written as

$$\log p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N) = -\frac{ND}{2} \log 2\pi - \frac{N}{2} \log \det(\boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I})$$

$$-\frac{1}{2} \sum_{n=1}^N (\boldsymbol{x}_n - \boldsymbol{\mu})^T (\boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I})^{-1} (\boldsymbol{x}_n - \boldsymbol{\mu})$$

- By setting $\frac{\partial \log p(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_N)}{\partial \boldsymbol{\mu}} = 0$, we obtain

$$\boldsymbol{\mu} = \frac{\sum_{n=1}^{N} \boldsymbol{x}_n}{N}$$

$$\frac{\partial \ln \det(\boldsymbol{X})}{\partial \boldsymbol{X}} = (\boldsymbol{X}^{-1})^T$$

$$\frac{\partial \ln trace(\boldsymbol{X}^{-1}\boldsymbol{B})}{\partial \boldsymbol{X}} = -(\boldsymbol{X}^{-1}\boldsymbol{B}\boldsymbol{X}^{-1})^T$$

- By denoting $\boldsymbol{\Sigma} = \boldsymbol{W}\boldsymbol{W}^T + \sigma^2\boldsymbol{I}$, we have

$$\frac{\partial \log p(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_N)}{\partial \boldsymbol{\Sigma}} = -\frac{N}{2}\boldsymbol{\Sigma}^{-1} + \frac{1}{2}\sum_{n=1}^{N} \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_n - \boldsymbol{\mu}_n)(\boldsymbol{x}_n - \boldsymbol{\mu}_n)^T\boldsymbol{\Sigma}^{-1}$$

$$= -\frac{N}{2}\boldsymbol{\Sigma}^{-1} + \frac{N}{2}\boldsymbol{\Sigma}^{-1}\boldsymbol{S}\boldsymbol{\Sigma}^{-1}$$

$\Rightarrow$ Thus, it can be derived that $\boldsymbol{\Sigma} = \boldsymbol{S}$

- When $\boldsymbol{\Sigma}$ is restricted to the form $\boldsymbol{\Sigma} = \boldsymbol{W}\boldsymbol{W}^T + \sigma^2\boldsymbol{I}$, it can be derived that

$$\boldsymbol{W} = \boldsymbol{U}(\boldsymbol{\Lambda} - \sigma^2\boldsymbol{I})^{\frac{1}{2}}$$

  - $\boldsymbol{U}$ consists of the top-M eigenvectors of $\boldsymbol{S}$
  - $\boldsymbol{\Lambda}$ is a diagonal matrix with the top-M eigenvalues of $\boldsymbol{S}$

Relation to PCA

- Comparing the expression

$$\boldsymbol{W} = \boldsymbol{U}\left(\boldsymbol{\Lambda} - \sigma^2 \boldsymbol{I}\right)^{\frac{1}{2}}$$

to the principle components of PCA, which are the matrix $\boldsymbol{U}$, we can see that

  - $\boldsymbol{W}$ can be viewed as un-normalized principle components of data $\boldsymbol{x}_n$, *with the* i*-th component scaled by a coefficient* $\sqrt{\lambda_i - \sigma^2}$

Gaussian latent-variable models are called *probabilistic PCA*

Outline

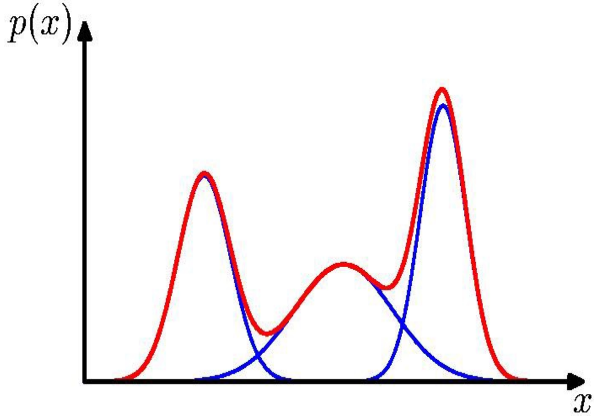
- Introduction of Latent-Variable Models
- Gaussian Latent-Variable Model
- Gaussian Mixture Model
- Examples of other LVMs

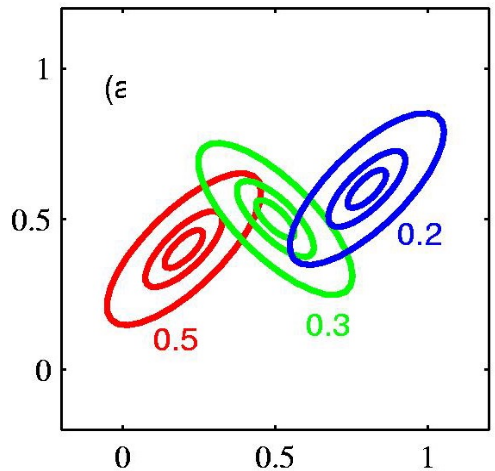
Gaussian Mixture Distributions

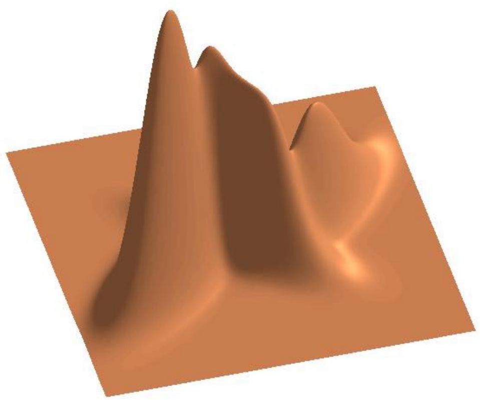
- The distribution expression

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

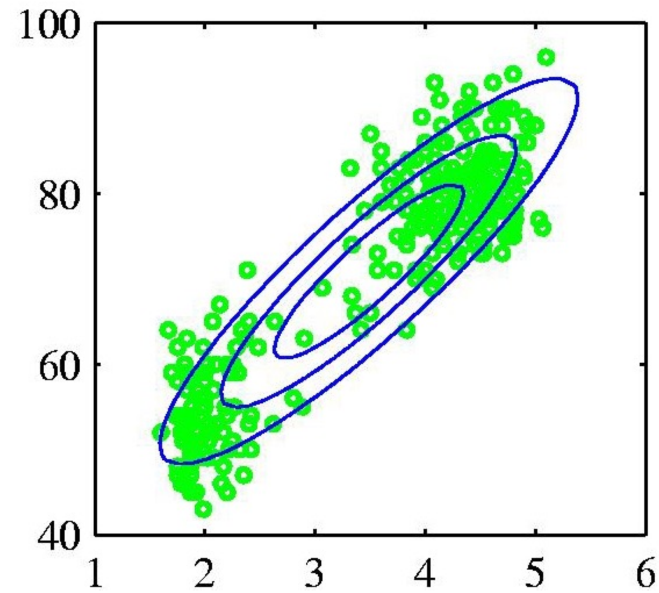
  - K is the number of Gaussian distributions
  - π_k is the weight of the *k*-th distribution with $\sum_{k=1}^{K} \pi_k = 1$
  - $\boldsymbol{\mu}_k$ *and* $\boldsymbol{\Sigma}_k$ are the mean vector and covariance matrix of the k-th Gaussian distribution



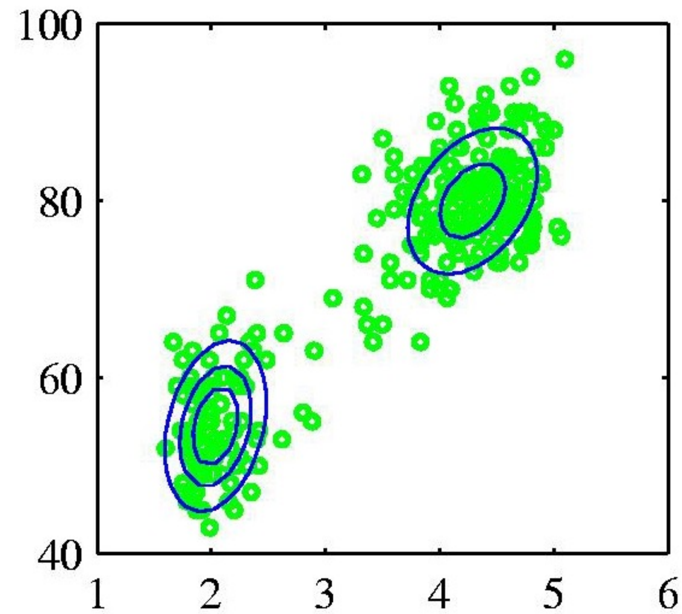




- It is very difficult to model the green points by a Gaussian distribution



- But if we model it with the mixture of two Gaussian distributions, it looks much better



Representing Gaussian Mixture Distribution as LVM

- For a latent-variable model $p(\boldsymbol{x}, \boldsymbol{z})$, if we set its conditional distribution $p(\boldsymbol{x}|\boldsymbol{z})$ and prior distribution $p(\boldsymbol{z})$ as

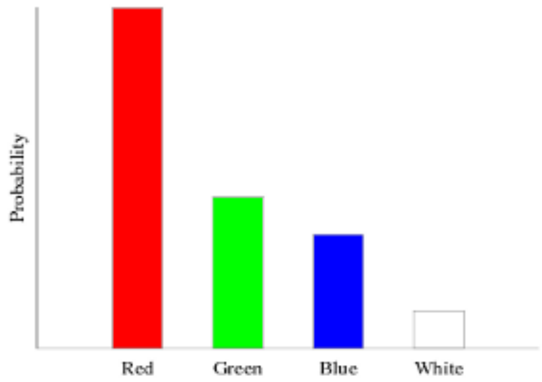
$$p(\boldsymbol{x}|\boldsymbol{z} = \mathbf{1}_k) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

$$p(\boldsymbol{z} = \mathbf{1}_k) = \pi_k$$

  - $\boldsymbol{z}$ can only be a one-hot vector, with $\mathbf{1}_k$ denoting the *k*-th element to be 1
  - $p(\boldsymbol{z} = \mathbf{1}_k) = \pi_k$ actually denotes a categorical distribution, that is,

$$p(\boldsymbol{z}) = Cat(\boldsymbol{z}; \boldsymbol{\pi})$$

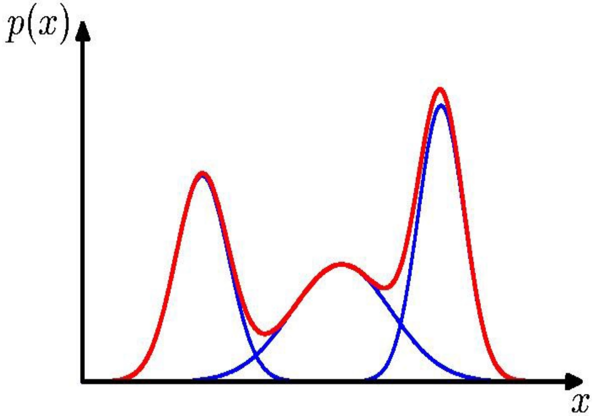
with $Cat(\boldsymbol{z} = \mathbf{1}_k; \boldsymbol{\pi}) = \pi_k$ and $\boldsymbol{\pi} = [\pi_1, \pi_2, \cdots, \pi_K]$



- Due to $p(\boldsymbol{x}) = \sum_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z})$, we can easily see that

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k),$$

which is exactly the Gaussian mixture distribution



Gaussian mixture distributions can be equivalently represented by the latent-variable model

$$p(\boldsymbol{x}, \boldsymbol{z}) = \prod_{k=1}^{K} [\pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)]^{z_k}$$

Training by Maximizing the Marginal

- Given a set of training data $\{\boldsymbol{x}^{(n)}\}_{n=1}^{N}$, the goal is to learn the distribution parameters

$$\{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K} \triangleq \boldsymbol{\theta}$$

- The data points $\boldsymbol{x}^{(n)}$ are assumed *i.i.d*, thus we can write the joint distribution as

The latent-variable form is not used

$$p(\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(N)}) = \prod_{n=1}^{N} \underbrace{\sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}_{p(\boldsymbol{x}^n)}$$

- For probabilistic models, the training objective is *to maximize the log-likelihood function*, that is,

$$\boxed{\log p(\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(N)}) = \sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$$

Maximizing $\log p\left(\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(N)}\right)$

- Substituting the expression of $\mathcal{N}\left(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)$ into it gives

$$\log p\left(\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(N)}\right)$$
$$= \sum_{n=1}^{N} \log\left(\sum_{k=1}^{K} \pi_k \frac{1}{(2\pi)^{D/2}|\boldsymbol{\Sigma}_k|^{1/2}} \exp\left\{-\frac{1}{2}\left(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k\right)^T \boldsymbol{\Sigma}_k^{-1}\left(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k\right)\right\}\right)$$

- To optimize it, we require the *derivatives* of $\log p\left(\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(N)}\right)$ *w.r.t.* the model parameters $\{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$

How to Use the Learned Model?

- After learning the parameters $\{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$, that is, the distribution

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x};\ \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

is known, we can *use it to complete a lot of tasks*

- Example: Given a testing data point $\boldsymbol{x}$, can we use it to determine the probability that an $\boldsymbol{x}$ belongs to the k-th cluster?

$$p(\boldsymbol{x} \in k\text{-}th\ cluster) = \frac{\pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{i=1}^{K} \pi_i \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}$$

- Can we explain the probability in a more principled way?

$$p(\boldsymbol{z} = \mathbf{1}_k | \boldsymbol{x}) = ?$$

$$p(\boldsymbol{z} = \mathbf{1}_k | \boldsymbol{x}) = \frac{p(\boldsymbol{x}, \boldsymbol{z} = \mathbf{1}_k)}{p(\boldsymbol{x})}$$

$$= \frac{p(\boldsymbol{x}, \boldsymbol{z} = \mathbf{1}_k)}{\sum_{i=1}^{K} p(\boldsymbol{x}, \mathbf{1}_i)}$$

$$= \frac{\pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{i=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}$$

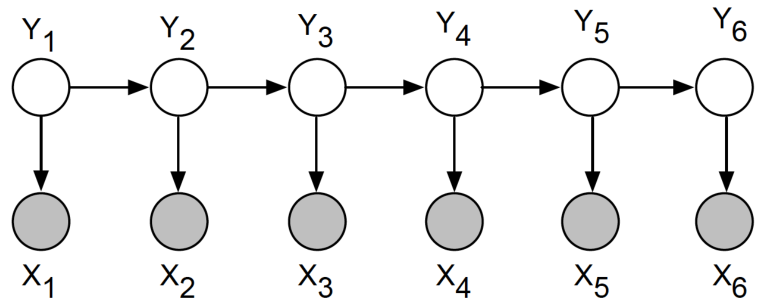
Thus, in the latent-variable model, the posteriori $p(\boldsymbol{z}|\boldsymbol{x})$ indicates the probability that a data instance belongs to different clusters

Outline

- Introduction of Latent-Variable Models
- Gaussian Latent-Variable Model
- Gaussian Mixture Model
- Examples of other LVMs

Application: Hidden Markov Model

- Hidden Markov Model (HMM)



- It is widely used in speech recognition, part-of-speech tagging, localization *etc.*

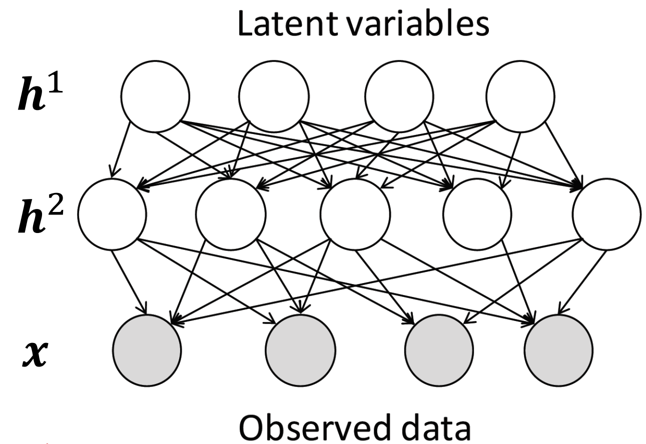
- Joint distribution

$$p(\boldsymbol{y}, \boldsymbol{x}) = p(y_1)p(x_1|y_1)\prod_{t=2}^{T} p(y_t|y_{t-1})p(x_t|y_t)$$

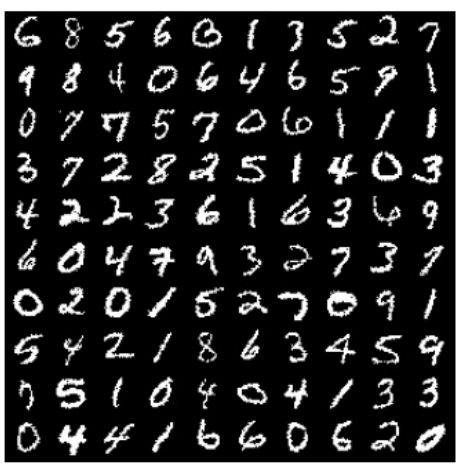
where $p(y_t|y_{t-1})$ is the transition probability; $p(x_t|y_t)$ is the emission probability

Application: Image Modeling

- Sigmoid belief networks (SBN)
  - $h_i^1 \sim Bernoulli(0.5)$
  - $h_j^2 \sim Bernoulli\left(\sigma\left([\boldsymbol{W}_1 \boldsymbol{h}^1 + \mathbf{b}_1]_j\right)\right)$
  - $x_k \sim Bernoulli\left(\sigma([\boldsymbol{W}_2 \boldsymbol{h}^2 + \mathbf{b}_2]_k)\right)$

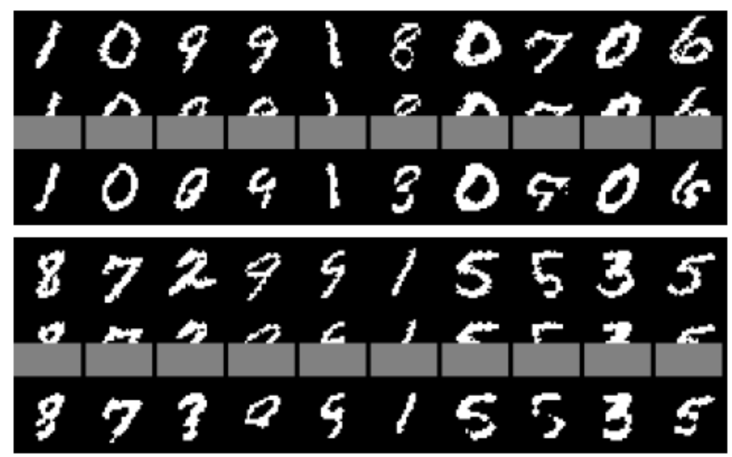


Joint pdf: $p(\boldsymbol{x}, \boldsymbol{h}^2, \boldsymbol{h}^1) = p(\boldsymbol{x}|\boldsymbol{h}^2)p(\boldsymbol{h}^2|\boldsymbol{h}^1)p(\boldsymbol{h}^1)$



Original

Generating



In-painting

Application: Text Modeling

- Topic Model: Latent Dirichlet Allocation (LDA)
  - $\theta \sim Dir(\alpha)$: the distribution of different topics
  - $\varphi_k \sim Dir(\beta)$: the distribution of words for topic
  - $z_n \sim Multinomial(\theta)$: the topic of n-th word
  - $w_n \sim Multinomial(\varphi_{z_n})$: the n-th word