Latent Variable Models

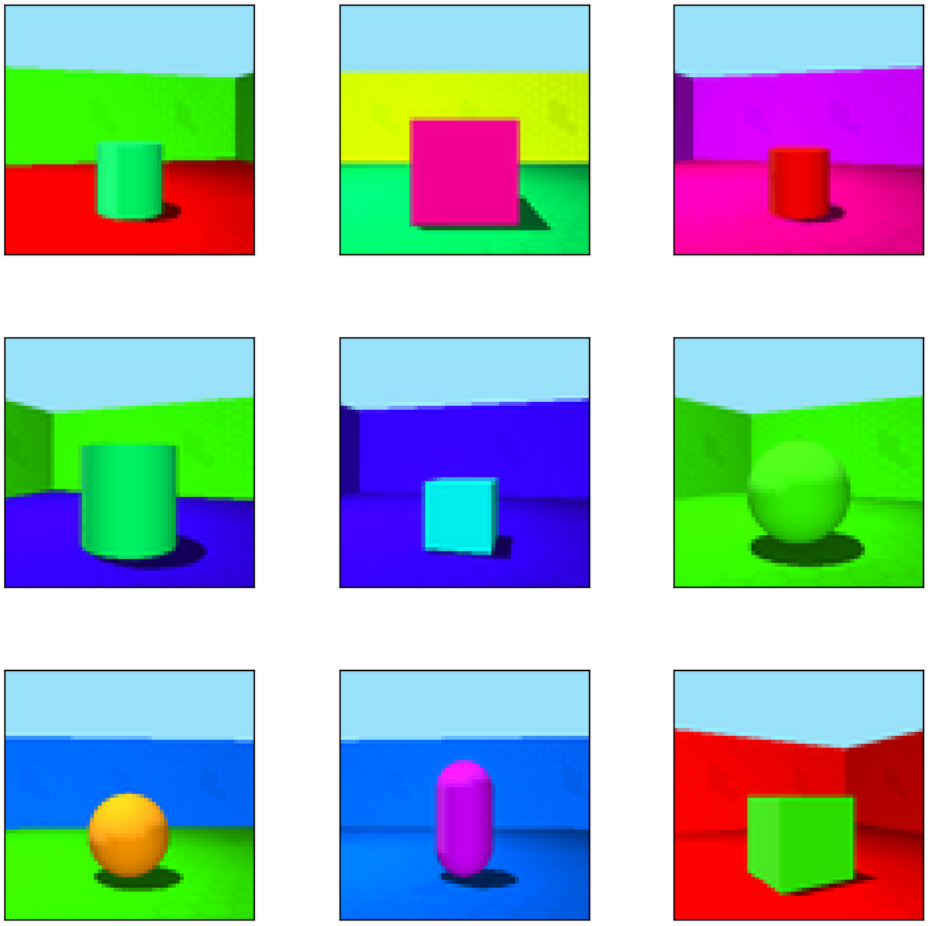
Advanced Statistical Inference

Simone Rossi

Latent variable models

Our objective is to learn the data distribution $p(\boldsymbol{x})$, but we suppose that each data point $\boldsymbol{x}$ is associated with a **latent variable** $\boldsymbol{z}$.

For example, image we want to learn the distribution of images of objects, like the one below:



Each image is a huge vector of pixels $x \in \mathbb{R}^{H \times W \times C}$, where H is the height, W is the width and C is the number of channels (in this case, $64 \times 64 \times 3$).

Each image can be described by a set of 6 latent variables: floor, wall and object color, shape, orientation and scale.

Latent variable models

Latent variables are unobserved variables, we cannot measure them directly, but we can infer them from the observed data.

In math terms, we model our data distribution as a **marginal** of the joint distribution of the observed data and the latent variables:

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}, \boldsymbol{z}) \, \mathrm{d}\boldsymbol{z} \quad \text{or} \quad p(\boldsymbol{x}) = \sum_i p(\boldsymbol{x}, \boldsymbol{z}_i)$$

where $p(\boldsymbol{x}, \boldsymbol{z})$ is the **joint distribution** of the observed data and the latent variables.

Mixture models

Mixture models

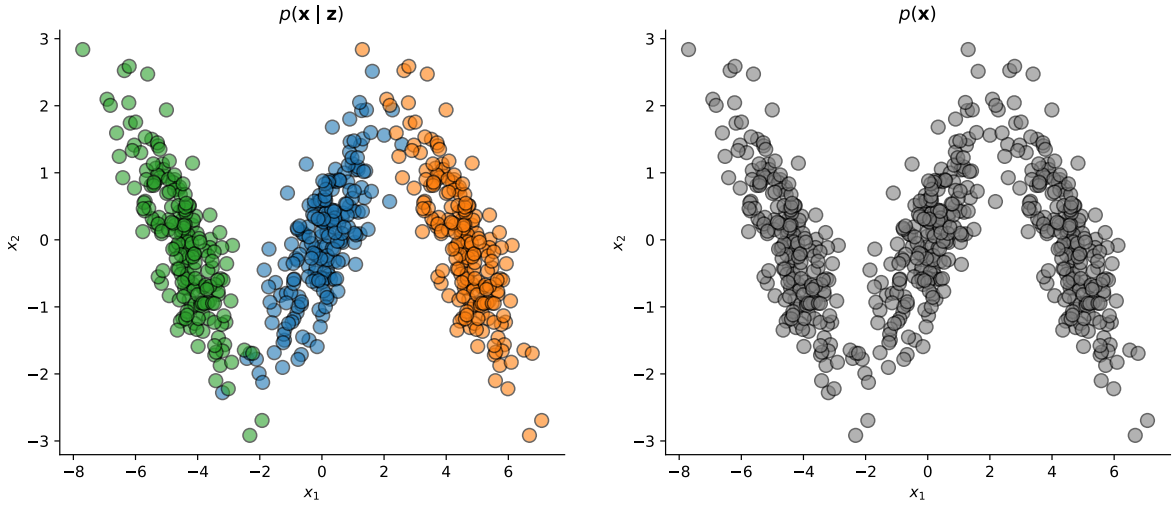
In mixture models, we assume that the data is generated from a mixture of several distributions

$$p(\boldsymbol{x}) = \sum_i p(\boldsymbol{x}, \boldsymbol{z}_i) = \sum_i p(\boldsymbol{x} \mid \boldsymbol{z}_i) p(\boldsymbol{z}_i)$$

where $p(\boldsymbol{z})$ is the **mixture distribution** and $p(\boldsymbol{x} \mid \boldsymbol{z})$ is the **conditional distribution** of the data given the latent variables.

- The latent variables $\boldsymbol{z}$ are often called **clusters**.
- The *likelihood* is often assumed to be Gaussian, i.e., $p(\boldsymbol{x} \mid \boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- The *prior* is often assumed to be a categorical distribution, i.e., $p(\boldsymbol{z}) = \mathrm{Cat}(\boldsymbol{z}; \boldsymbol{\pi})$

Gaussian Mixture Models



Modeling the data distribution

- 1-hot encoding of the **discrete latent variable** $z_k \in \{0, 1\}$, with prior:

$$p(z_k = 1) = \pi_k \quad \text{and} \quad \sum_k \pi_k = 1$$

- The **conditional distribution** of the data given the latent variable is:

$$p(\boldsymbol{x} \mid z_k = 1) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- The **joint distribution** of the data and the latent variable is:

$$p(\boldsymbol{x}, z_k = 1) = p(\boldsymbol{x} \mid z_k = 1)p(z_k = 1) = \pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- The **marginal distribution** of the data is:

$$p(\boldsymbol{x}) = \sum_k p(\boldsymbol{x}, z_k = 1) = \sum_k \pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Posterior distribution

We are interested in the **posterior distribution** of a cluster assignment given the data:

$$\begin{aligned} p(z_k = 1 \mid \boldsymbol{x}) &= \frac{p(\boldsymbol{x} \mid z_k = 1)p(z_k = 1)}{p(\boldsymbol{x})} \\ &= \frac{p(\boldsymbol{x} \mid z_k = 1)p(z_k = 1)}{\sum_j p(\boldsymbol{x} \mid z_j = 1)p(z_j = 1)} \\ &= \frac{\pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} := \gamma(z_k) \end{aligned}$$

$\gamma(z_k)$ is the **responsibility** of cluster k for data point $\boldsymbol{x}$.

Log-likelihood

Given a dataset $\boldsymbol{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$ i.i.d, the **log-likelihood** of the data is:

$$\begin{aligned} \log p(\boldsymbol{X} \mid \boldsymbol{\pi}, \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}) &= \log \prod_{i=1}^{N} p(\boldsymbol{x}_i) \\ &= \sum_{i=1}^{N} \log p(\boldsymbol{x}_i \mid \boldsymbol{\pi}, \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}) \\ &= \sum_{i=1}^{N} \log \sum_{k=1}^{K} p(\boldsymbol{x}_i \mid z_k = 1)p(z_k = 1) \\ &= \sum_{i=1}^{N} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \end{aligned}$$

> **Warning**
>
> We cannot simplify it further, because the log and the sum do not commute. How can we optimize it?

Expectation-Maximization algorithm

We need to maximize the log-likelihood of the data w.r.t. the parameters $\boldsymbol{\pi}$, $\{\boldsymbol{\mu}_k\}$ and $\{\boldsymbol{\Sigma}_k\}$, for $k = 1, \dots, K$

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

- The problem is not convex.
- No closed-form solution! Stationary points depend on the posterior $\gamma(z_{nk})$ for each data point $\boldsymbol{x}_n$.
- ... but we will see that we can write π_k, $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ in terms of $\gamma(z_{nk})$.
- We can find the **local minima** by iterative algorithm: alternate update of the **expected** posterior $\gamma(z_{nk})$ and the **maximized** parameters π_k, $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$.

Expectation-Maximization algorithm

- **E-step**: compute the expected posterior $\gamma(z_{nk})$ given the current parameters $\boldsymbol{\pi}$, $\{\boldsymbol{\mu}_k\}$ and $\{\boldsymbol{\Sigma}_k\}$:
$$\gamma(z_{nk}) = \frac{\mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k}{\sum_{j=1}^{K} \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\pi_j}$$
- **M-step**: update the parameters $\boldsymbol{\pi}$, $\{\boldsymbol{\mu}_k\}$ and $\{\boldsymbol{\Sigma}_k\}$ to maximize the expected log-likelihood, given the current expected posterior $\gamma(z_{nk})$. This can be solved in closed form:
$$N_k = \sum_{n=1}^{N} \gamma(z_{nk}) \qquad \pi_k = \frac{N_k}{N}$$
$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \boldsymbol{x}_n$$
$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top$$

Derivation of the M-step updates

Results of results for Gaussian distributions:

Assume $p(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-\frac{d}{2}} \det(\boldsymbol{\Sigma})^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})\right)$.

Then,

$$\frac{\partial p(\boldsymbol{x})}{\partial \boldsymbol{\mu}} = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})(\boldsymbol{x} - \boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}$$

and

$$\frac{\partial p(\boldsymbol{x})}{\partial \boldsymbol{\Sigma}} = -\frac{1}{2}\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \left(\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}\right)$$

Derivation of the M-step updates ($\boldsymbol{\mu}_k$)

$$\arg\max_{\boldsymbol{\mu}_k} p(\boldsymbol{X} \mid \boldsymbol{\pi}, \{\boldsymbol{\mu}_j\}, \{\boldsymbol{\Sigma}_j\}) = \arg\max_{\boldsymbol{\mu}_k} \sum_{n=1}^{N} \log p(\boldsymbol{x}_n \mid \boldsymbol{\pi}, \{\boldsymbol{\mu}_j\}, \{\boldsymbol{\Sigma}_j\})$$

Compute the partial derivative w.r.t. $\boldsymbol{\mu}_k$:

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\mu}_k} \log p(\boldsymbol{X} \mid \boldsymbol{\pi}, \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}) &= \sum_{n=1}^{N} \frac{1}{p(\boldsymbol{x}_n \mid \boldsymbol{\pi}, \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\})} \frac{\partial p(\boldsymbol{x}_n \mid \boldsymbol{\pi}, \{\boldsymbol{\mu}_j\}, \{\boldsymbol{\Sigma}_j\})}{\partial \boldsymbol{\mu}_k} \\
&= \sum_{n=1}^{N} \frac{1}{p(\boldsymbol{x}_n \mid \boldsymbol{\pi}, \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\})} \frac{\partial \pi_k \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\partial \boldsymbol{\mu}_k} \\
&= \sum_{n=1}^{N} \frac{\pi_k \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)\boldsymbol{\Sigma}_k^{-1} \\
&= \sum_{n=1}^{N} \gamma(z_{nk})(\boldsymbol{x}_n - \boldsymbol{\mu}_k)\boldsymbol{\Sigma}_k^{-1} = 0 \quad \Rightarrow \boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) \boldsymbol{x}_n}{\sum_{n=1}^{N} \gamma(z_{nk})}
\end{aligned}$$

Derivation of the M-step updates ($\boldsymbol{\Sigma}_k$, $\boldsymbol{\pi}$)

Same for the covariance:

1. Write the derivative of the log-likelihood w.r.t. $\boldsymbol{\Sigma}_k$ (use the properties of the Gaussian distribution).
2. Manipulate it to make $\gamma(z_{nk})$ appear.
3. Set it to zero and solve for $\boldsymbol{\Sigma}_k$.

Same for the weights $\boldsymbol{\pi}$, but we have to use the constraint $\sum_{k=1}^{K} \pi_k = 1$.

1. Use Lagrange multipliers to introduce a new variable λ and write the Lagrangian function:

$$\ell(\boldsymbol{\pi}, \lambda) = \log p(\boldsymbol{X} \mid \boldsymbol{\pi}, \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}) + \lambda \left(1 - \sum_{k=1}^{K} \pi_k\right)$$

2. Compute the partial derivative w.r.t. $\boldsymbol{\pi}$ and λ
3. Set them to zero and solve for $\boldsymbol{\pi}$ and λ.

EM for Gaussian Mixture Models

Algorithm:

1. Initialize the parameters $\boldsymbol{\pi}$, $\{\boldsymbol{\mu}_k\}$ and $\{\boldsymbol{\Sigma}_k\}$
2. Repeat until convergence (likelihood does not change significantly):
    1. E-step: compute the expected posterior $\gamma(z_{nk})$ given the current parameters $\boldsymbol{\pi}$, $\{\boldsymbol{\mu}_k\}$ and $\{\boldsymbol{\Sigma}_k\}$:
    2. M-step: update the parameters $\boldsymbol{\pi}$, $\{\boldsymbol{\mu}_k\}$ and $\{\boldsymbol{\Sigma}_k\}$ to maximize the expected log-likelihood, given the current expected posterior $\gamma(z_{nk})$:
    3. Compute the log-likelihood of the data given the current parameters $\boldsymbol{\pi}$, $\{\boldsymbol{\mu}_k\}$ and $\{\boldsymbol{\Sigma}_k\}$

Important:

- EM is guaranteed to converge to a local maximum, not necessarily to the global maximum.
- EM is sensitive to the initialization of the parameters (try different initializations and choose the one that gives the highest log-likelihood)

Gaussian Mixture Models

After the EM algorithm converges, we can the learned parameters to:

1. Generate new data points by sampling from the mixture model.
2. Compute the likelihood of new data points.
3. Compute the posterior distribution of the latent variables given new data points (e.g., clustering).

Gaussian Mixture Models

Pros:

1. Simple and easy to implement.
2. Can model *any* distribution as a mixture of Gaussians.
3. One model can be used for different tasks (e.g., clustering, density estimation, generation).

Cons:

1. Sensitive to the initialization of the parameters, can get stuck in local minima.
2. The number of clusters K must be specified in advance.
3. The number of clusters K must be small, otherwise the model will overfit the data.

Question: Can we be Bayesian about the number of clusters K? **Yes**, it's called **Dirichlet Process Mixture Model** (DPMM)

Dimensionality reduction as a generative model

Linear latent variable models

Linear latent variable models are a class of generative models that assume that the data is generated from a linear combination of latent variables.

We will:

1. Review the Principal Component Analysis (PCA) algorithm.
2. Introduce the (probabilistic) PCA as a linear latent variable model.

Principal Component Analysis

1. Assume we have a dataset $\boldsymbol{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$, where $\boldsymbol{x}_n \in \mathbb{R}^D$.
2. We want to find a low-dimensional representation of the data, i.e., we want to find a mapping $\boldsymbol{x}_n \mapsto \boldsymbol{z}_n$ such that $\boldsymbol{z}_n \in \mathbb{R}^K$, where $K < D$.
3. Choose the mapping such that the variance of the projected data is maximized, i.e., we want to find the mapping that maximizes the variance of the data in the low-dimensional space.

How to find the direction of maximum variance?

- Let's start by projecting the data on 1D subspace $z = \boldsymbol{w}^\top \boldsymbol{x}$, where $\boldsymbol{w} \in \mathbb{R}^D$.
- We only want the direction, so we can assume that $\|\boldsymbol{w}\| = 1$.
- Given data $\boldsymbol{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$, we maximize the variance of the projected data $\boldsymbol{z} = \boldsymbol{X}\boldsymbol{w}$ constrained to $\|\boldsymbol{w}\| = 1$:

$$\arg\max_{\boldsymbol{w}} \|\boldsymbol{z}\|^2 \qquad \text{s.t.} \quad \|\boldsymbol{w}\|^2 = 1$$

Assumes that $\boldsymbol{X}$ is centered (i.e., $\boldsymbol{X}$ has zero mean)

Analytic solution

- Let's use the Lagrange multipliers to solve the constrained optimization problem:

$$\ell(\boldsymbol{w}, \lambda) = \|\boldsymbol{z}\|^2 - \lambda(\|\boldsymbol{w}\|^2 - 1) = \boldsymbol{w}^\top \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{w} - \lambda(\boldsymbol{w}^\top \boldsymbol{w} - 1)$$

- Compute the partial derivative w.r.t. $\boldsymbol{w}$ and λ:

$$\begin{aligned} \frac{\partial \ell}{\partial \boldsymbol{w}} &= 2\boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{w} - 2\lambda \boldsymbol{w} = 0 \quad \Rightarrow \quad \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{w} = \lambda \boldsymbol{w} \\ \frac{\partial \ell}{\partial \lambda} &= -(\boldsymbol{w}^\top \boldsymbol{w} - 1) = 0 \end{aligned}$$

This is an eigenvalue problem, where λ is the eigenvalue and $\boldsymbol{w}$ is the eigenvector of the covariance matrix $\boldsymbol{X}^\top \boldsymbol{X}$.

Analytic solution

But which eigenvalue?

- The eigenvalue λ is the variance of the projected data $\boldsymbol{z}$: $\lambda = \boldsymbol{w}^\top \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{w}$
- If we want to maximize the variance, we need to find the largest eigenvalue of the covariance matrix $\boldsymbol{X}^\top \boldsymbol{X}$.
- The corresponding eigenvector is the direction of maximum variance: $\widehat{\boldsymbol{w}}$

Probabilistic PCA

- PCA can be seen as a linear continuous latent variable model, where the latent variable $\boldsymbol{z}$ is a linear combination of the observed data $\boldsymbol{x}$.
- **Question**: Can we approach PCA from a probabilistic point of view? **Answer**: Yes!

Recall the continuous latent variable model:

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}, \boldsymbol{z}) \, \mathrm{d}\boldsymbol{z} = \int p(\boldsymbol{x} \mid \boldsymbol{z}) p(\boldsymbol{z}) \, \mathrm{d}\boldsymbol{z}$$

PPCA Modeling Assumptions

- The **generative model** works as follows:

$$\boldsymbol{x} = \boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\varepsilon}$$

where:

1. $\boldsymbol{W} \in \mathbb{R}^{D \times K}$ is the weight matrix
2. $\boldsymbol{z} \in \mathbb{R}^{K}$ is the latent variable
3. $\boldsymbol{\mu} \in \mathbb{R}^{D}$ is the mean of the data
4. $\boldsymbol{\varepsilon} \in \mathbb{R}^{D}$ is the noise

How can we treat these variables?

PPCA Modeling Assumptions

- The **latent variable** $\boldsymbol{z}$ is assumed to be a standard Gaussian distribution:

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

- The **noise** $\boldsymbol{\varepsilon}$ is assumed to be a Gaussian distribution with zero mean and covariance matrix $\sigma^2\boldsymbol{I}$:

$$p(\boldsymbol{\varepsilon}) = \mathcal{N}(\boldsymbol{\varepsilon}; 0, \sigma^2\boldsymbol{I})$$

. . .

- The **observed data** $\boldsymbol{x}$ conditioned on the latent variable $\boldsymbol{z}$ is a Gaussian distribution

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

- The **marginal distribution** of the data $\boldsymbol{x}$ is obtained by integrating out the latent variable $\boldsymbol{z}$:

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x} \mid \boldsymbol{z}) p(\boldsymbol{z}) \, \mathrm{d}\boldsymbol{z} = \int \mathcal{N}(\boldsymbol{x}; \boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu}, \sigma^2\boldsymbol{I}) \mathcal{N}(\boldsymbol{z}; 0, \boldsymbol{I}) \, \mathrm{d}\boldsymbol{z}$$

It's a Gaussian distribution $\mathcal{N}(?, ?)$.

PPCA Modeling Assumptions

1. For the mean, we have:

$$\begin{aligned}\mathbb{E}[\boldsymbol{x}] &= \mathbb{E}[\boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\varepsilon}] \\ &= \boldsymbol{W}\mathbb{E}[\boldsymbol{z}] + \boldsymbol{\mu} + \mathbb{E}[\boldsymbol{\varepsilon}] \\ &= \boldsymbol{W} \cdot 0 + \boldsymbol{\mu} + 0 = \boldsymbol{\mu}\end{aligned}$$

. . .

2. For the covariance, we have:

$$\begin{aligned}\mathrm{Cov}(\boldsymbol{x}) &= \mathbb{E}[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^\top] \\ &= \mathbb{E}[(\boldsymbol{W}\boldsymbol{z} + \boldsymbol{\varepsilon})(\boldsymbol{W}\boldsymbol{z} + \boldsymbol{\varepsilon})^\top] \\ &= \mathbb{E}[\boldsymbol{W}\boldsymbol{z}\boldsymbol{z}^\top\boldsymbol{W}^\top + \boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\top + \boldsymbol{W}\boldsymbol{z}\boldsymbol{\varepsilon}^\top + \boldsymbol{\varepsilon}\boldsymbol{z}^\top\boldsymbol{W}] \\ &= \boldsymbol{W}\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top]\boldsymbol{W}^\top + \mathbb{E}[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\top] + \boldsymbol{W}\mathbb{E}[\boldsymbol{z}]\mathbb{E}[\boldsymbol{\varepsilon}^\top] + \mathbb{E}[\boldsymbol{\varepsilon}]\mathbb{E}[\boldsymbol{z}^\top]\boldsymbol{W} \\ &= \boldsymbol{W} \cdot \boldsymbol{I} \cdot \boldsymbol{W}^\top + \sigma^2\boldsymbol{I} + 0 + 0 \\ &= \boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}\end{aligned}$$

PPCA Solution

Now, we need to find the parameters $\boldsymbol{W}$, $\boldsymbol{\mu}$ and σ^2 that maximize the log-likelihood of the data:

$$\begin{aligned}\log p(\boldsymbol{X}; \boldsymbol{W}, \boldsymbol{\mu}, \sigma^2) = \sum_{n=1}^{N} \log p(\boldsymbol{x}_n; \boldsymbol{W}, \boldsymbol{\mu}, \sigma^2) &= \sum_{n=1}^{N} \log \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu}, \boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}) \\ = -\frac{NK}{2}\log(2\pi) - \frac{N}{2}\log\Big(\det\Big(\boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}\Big)\Big) - \\ - \frac{1}{2}\sum_{n=1}^{N}(\boldsymbol{x}_n - \boldsymbol{\mu})^\top(\boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I})^{-1}(\boldsymbol{x}_n - \boldsymbol{\mu})\end{aligned}$$

Again, compute the partial derivative w.r.t. $\boldsymbol{W}$, $\boldsymbol{\mu}$ and σ^2 and set them to zero.

PPCA Solution

Let's define the sample covariance matrix $\boldsymbol{S} = \boldsymbol{X}\boldsymbol{X}^\top$. Solutions for $\boldsymbol{W}$, $\boldsymbol{\mu}$ and σ^2 can be found in closed form, no need to use the EM algorithm.

. . .

1. For the mean, we have: $\boldsymbol{\mu} = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{x}_n$

. . .

2. For the noise variance, we have: $\sigma^2 = \frac{1}{D-K}\sum_{j=K+1}^{D} \lambda_j$, where λ_j are the eigenvalues of $\boldsymbol{S}$

. . .

3. For the weight matrix, we have: $\boldsymbol{W} = \boldsymbol{U}_K(\boldsymbol{\Lambda}_K - \sigma^2\boldsymbol{I})^{1/2}$, where $\boldsymbol{U}_K$ are the first K eigenvectors $\boldsymbol{S}$ and $\boldsymbol{\Lambda}_K$ is the diagonal matrix of the first K eigenvalues.

. . .

Because everything is Gaussian, we can also compute the posterior distribution of the latent variable $\boldsymbol{z}$ given the data $\boldsymbol{x}$:

$$p(\boldsymbol{z}_n \mid \boldsymbol{x}_n) = \mathcal{N}(\boldsymbol{z}_n; \boldsymbol{C}^{-1}\boldsymbol{W}^\top(\boldsymbol{x}_n - \boldsymbol{\mu}), \sigma^2\boldsymbol{C}^{-1})$$

with $\boldsymbol{C} = \boldsymbol{W}^\top\boldsymbol{W} + \sigma^2\boldsymbol{I}$

Is the PPCA solution the same as PCA?

- The PPCA solution is more general than PCA, because it assumes that the data is generated from a Gaussian distribution with noise.
- However, when the noise variance σ^2 goes to zero, the PPCA solution converges to the PCA solution.

Note: The principal components found by PCA and PPCA span the same subspace, but the actual components (i.e., the directions) may differ by an (un-identifiable) rotation matrix $\boldsymbol{R}$