



UNIVERSIDADE  
FEDERAL DO CEARÁ

DEPARTAMENTO  
DE COMPUTAÇÃO

# Aprendizagem de Máquina Probabilística

César Lincoln Cavalcante Mattos

2024

# Agenda

## ① Algoritmo PCA probabilístico

Solução analítica

Solução via algoritmo EM

## ② Análise fatorial

## ③ Tópicos adicionais

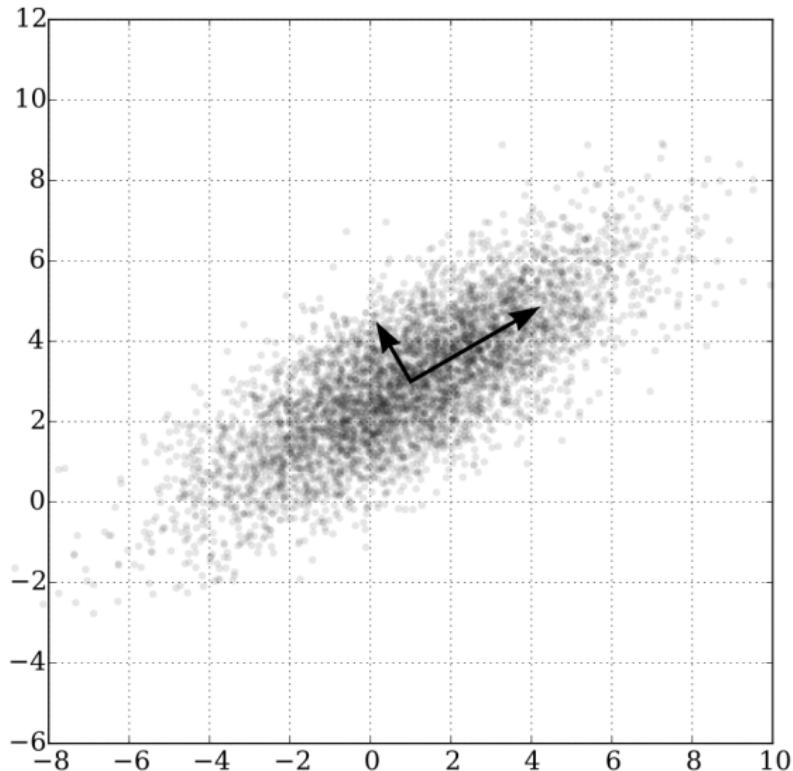
## ④ Referências

# Análise de Componentes Principais

## *Principal Component Analysis (PCA)*

- Método **não-supervisionado** de **projeção linear** de dados.
- Também chamado de **Transformada de Karhunen-Loève**.
- Pode ser usado para redução de dimensionalidade escolhendo-se um subconjunto dos atributos gerados.
- **Considerações:**
  - Os dados são contínuos;
  - A informação está na maneira como os atributos variam.

# Ilustração do conceito de componentes principais



# Análise de Componentes Principais

- Sejam  $N$  padrões  $D$ -dimensionais  $\mathbf{x}_i|_{i=1}^N$ .
- Queremos obter uma projeção  $\mathbf{z}_i \in \mathbb{R}^M$ ,  $M \leq D$ , tal que:

$$\mathbf{z}_i = \mathbf{P}\mathbf{x}_i,$$

em que  $\mathbf{P} \in \mathbb{R}^{M \times D}$  é a **matriz de projeção**.

- **Objetivo:** Maximizar a variância dos dados projetados.

# Análise de Componentes Principais

- Seja  $\mathbf{p}_m \in \mathbb{R}^D$  a  $m$ -ésima linha da matriz de projeção  $\mathbf{P}$ .
- A variância dos dados projetos na componente  $\mathbf{p}_1$ , i.e.,  $z_{i1} = \mathbf{p}_1^\top \mathbf{x}_i$ , é dada por:

$$\sigma_1^2 = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{p}_1^\top \mathbf{x}_i - \mathbf{p}_1^\top \boldsymbol{\mu})^2, \quad \text{em que } \boldsymbol{\mu} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i.$$

- Trabalhando a expressão anterior, temos:

$$\sigma_1^2 = \frac{1}{N-1} \sum_{i=1}^N \mathbf{p}_1^\top (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^\top \mathbf{p}_1$$

$$\sigma_1^2 = \mathbf{p}_1^\top \Sigma \mathbf{p}_1, \quad \text{em que } \Sigma = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^\top.$$

- Note que  $\Sigma \in \mathbb{R}^{D \times D}$  é a matriz de covariância dos dados originais  $\mathbf{x}_i |_{i=1}^N$ .

# Análise de Componentes Principais

- Desejamos maximizar a variância projetada  $\sigma_1^2 = \mathbf{p}_1^\top \Sigma \mathbf{p}_1$ .
- Para isso, limitamos a norma do vetor de projeção:  $\mathbf{p}_1^\top \mathbf{p}_1 = 1$ .
- Temos o seguinte problema de otimização com restrições:

$$\underset{\mathbf{p}_1}{\text{Maximize}} \quad \mathbf{p}_1^\top \Sigma \mathbf{p}_1,$$

$$\text{s.a.} \quad \mathbf{p}_1^\top \mathbf{p}_1 = 1.$$

- Incluímos o multiplicador de Lagrange  $\lambda_1 \geq 0$  para obter um problema sem restrições:

$$\mathcal{L} = \mathbf{p}_1^\top \Sigma \mathbf{p}_1 + \lambda_1(1 - \mathbf{p}_1^\top \mathbf{p}_1).$$

- Sabendo que:  $\frac{\partial \mathcal{L}}{\partial \mathbf{p}_1} = 0$ , temos:

$$2\Sigma \mathbf{p}_1 - 2\lambda_1 \mathbf{p}_1 = 0$$

$$\Sigma \mathbf{p}_1 = \lambda_1 \mathbf{p}_1.$$

# Análise de Componentes Principais

- A condição  $\Sigma p_1 = \lambda_1 p_1$  garante que  $p_1$  é **autovetor** da matriz de covariância  $\Sigma$  dos dados, com **autovalor** correspondente  $\lambda_1$ .
- A variância projetada  $\sigma_1^2$  será agora dada por:

$$\sigma_1^2 = p_1^\top \Sigma p_1 = \lambda_1 p_1^\top p_1 = \lambda_1.$$

- Logo,  $\sigma_1^2$  será maximizada escolhendo-se  $p_1$  igual ao **autovetor correspondente ao maior autovalor**.
- Os demais vetores de projeção correspondem aos demais autovetores da matriz  $\Sigma$ .

# Análise de Componentes Principais

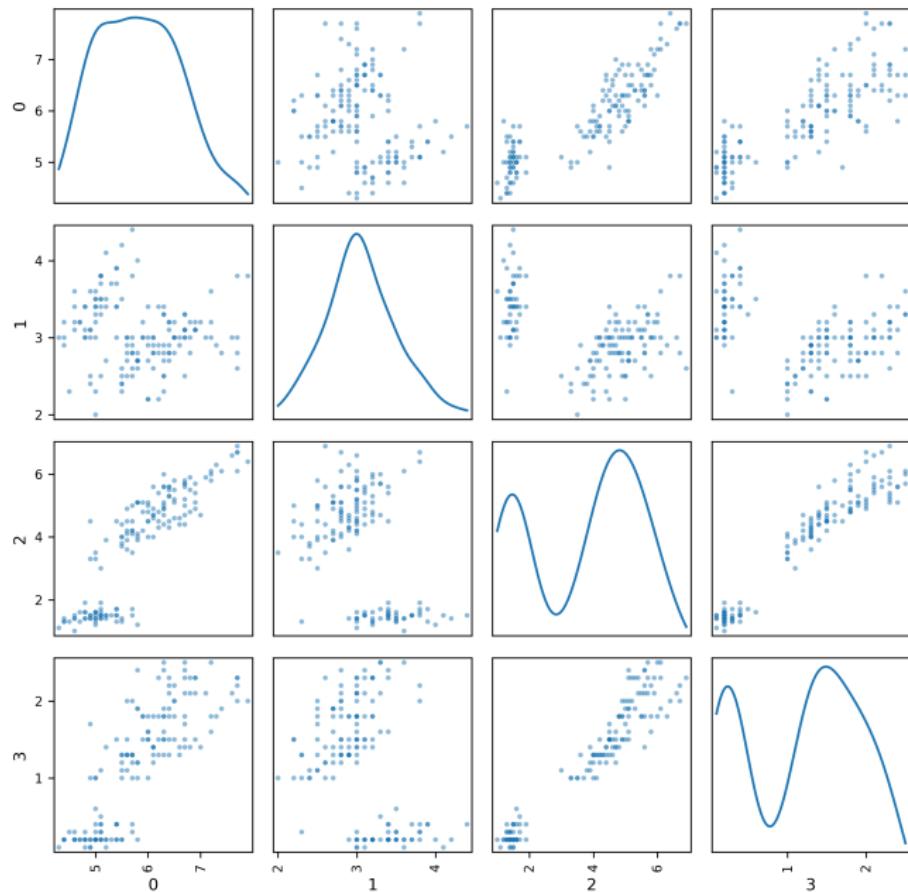
## Algoritmo PCA

- ① Calcule a matriz de covariância dos dados  $\mathbf{x}_i|_{i=1}^N$ :

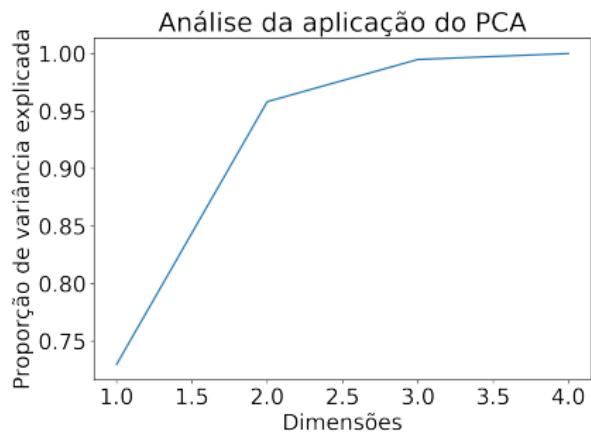
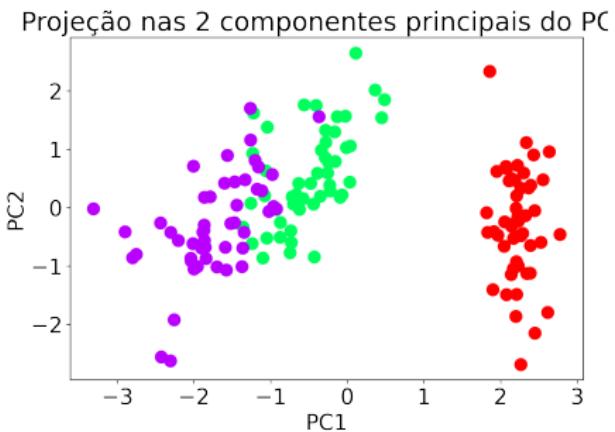
$$\Sigma = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^\top, \quad \text{em que } \boldsymbol{\mu} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i.$$

- ② Encontre os  $M$  autovetores  $\mathbf{p}_m|_{m=1}^M$  da matriz de covariância  $\Sigma$  correspondentes aos  $M$  maiores autovalores  $\lambda_m|_{m=1}^M$ .
  - ③ Os  $M$  autovetores selecionados formarão as linhas da matriz de projeção  $\mathbf{P} \in \mathbb{R}^{M \times D}$ .
  - ④ **Variância explicada:**  $\sum_{m=1}^M \lambda_m$ .
  - ⑤ **Projeção linear dos dados:**  $\mathbf{z}_i = \mathbf{P}\mathbf{x}_i, \quad i \in \{1, \dots, N\}$ .
- 
- **Nota:** Recomenda-se normalizar os dados antes da estimação.

# Exemplo de aplicação do PCA - Íris



# Exemplo de aplicação do PCA - Íris



# Análise de Componentes Principais

## SVD - Singular Value Decomposition

- Método de fatoração de matrizes que generaliza a decomposição de autovetores/autovalores.
- Uma matriz  $M \in \mathbb{R}^{A \times B}$  pode ser decomposta como

$$M = USV^\top.$$

- $S \in \mathbb{R}^{A \times B}$  reúne em sua “diagonal” os valores singulares de  $M$ .
- $U \in \mathbb{R}^{A \times A}$  e  $V^\top \in \mathbb{R}^{B \times B}$  são ortogonais.
- Para uma matriz  $M$  quadrada:
  - $S$  é diagonal com elementos iguais aos autovalores de  $M$ .
  - $U = V$  possui os autovetores de  $M$  nas colunas.

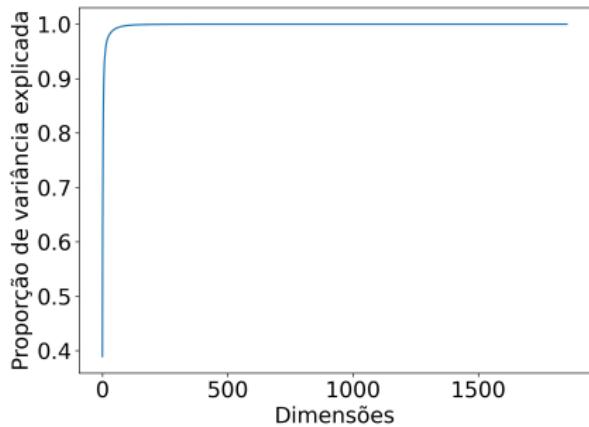
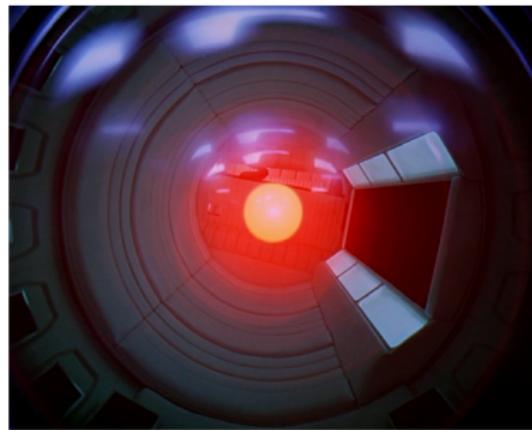
## PCA via SVD

- Seja  $\bar{X} = X - \mu$  e  $\bar{X} = USV^\top$ :  
 $\Sigma \propto \bar{X}^\top \bar{X} = VS^\top U^\top USV^\top = V \tilde{S} V^\top$ , em que  $\tilde{S} = S^\top S$ .
- Por ser uma decomposição espectral, temos  $P = V^\top$ .

# Análise de Componentes Principais

- **Matriz de projeção:**  $P \in \mathbb{R}^{M \times D}$ .
- **Projeção linear dos dados:**  $z_i = P(x_i - \mu)$ .
- **Reconstrução das projeções:**  $\hat{x}_i = \mu + P^\top z_i$ .
- Para  $M < D$ , há a compressão dos dados originais.

# Exemplo de compactação de imagem com PCA



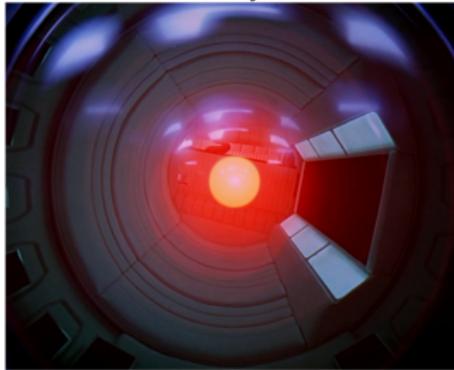
Número de linhas: 500

Número de colunas: 618

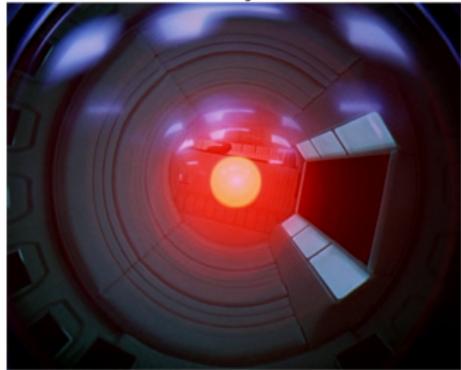
Número de canais: 3

# Exemplo de compactação de imagem com PCA

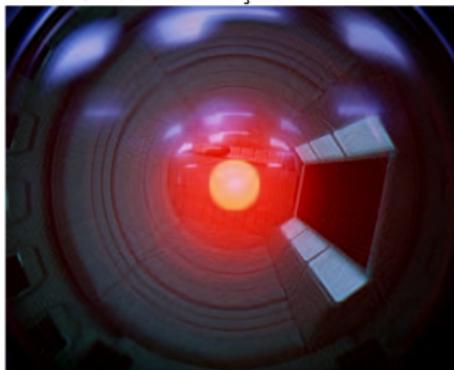
500 componentes principais  
Erro de reconstrução = 8.40e-22



100 componentes principais  
Erro de reconstrução = 5.69e+01



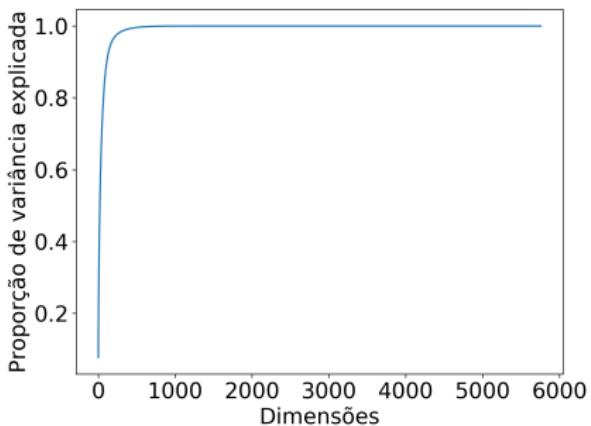
50 componentes principais  
Erro de reconstrução = 2.14e+02



25 componentes principais  
Erro de reconstrução = 5.38e+02



# Exemplo de compactação de imagem com PCA



Número de linhas: 1176

Número de colunas: 1920

Número de canais: 3

# Exemplo de compactação de imagem com PCA

500 componentes principais  
Erro de reconstrução =  $1.09e+03$



100 componentes principais  
Erro de reconstrução =  $3.15e+04$



50 componentes principais  
Erro de reconstrução =  $7.42e+04$

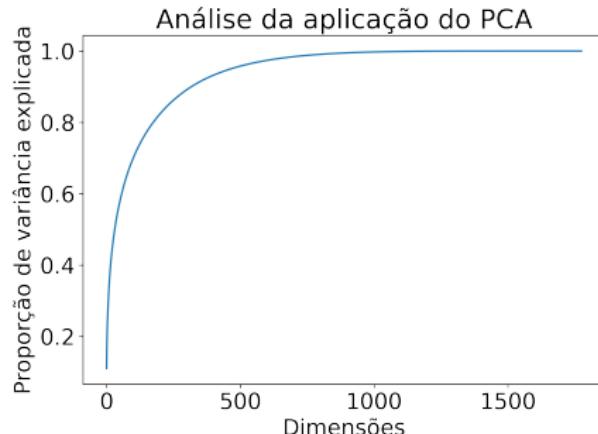


25 componentes principais  
Erro de reconstrução =  $1.22e+05$

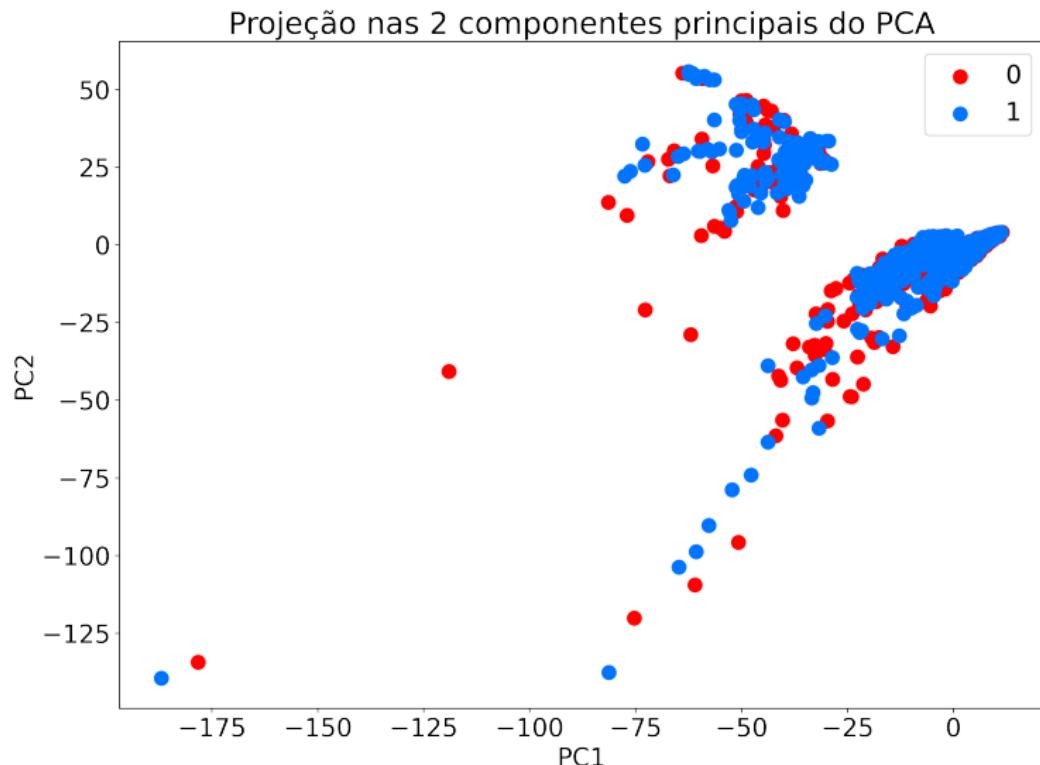


# Exemplo de pré-processamento com PCA

- **Dados:** Bioresponse - 3751 padrões, 2 classes, 30% para teste.
- **Modelo:** MLP ( $N_H = 64$ , tanh).
  - Atributos: 1776 - Erro no treinamento: 0%; Erro no teste: 24.02%
  - Atributos (projeção via PCA, 0.95 de variância explicada): 466
    - Erro no treinamento: 0.11%; Erro no teste: 23.75%
  - Atributos (projeção via PCA, 0.9 de variância explicada): 318
    - Erro no treinamento: 0.61%; Erro no teste: 22.95%
  - Atributos (projeção via PCA, 0.8 de variância explicada): 176
    - Erro no treinamento: 2.85%; Erro no teste: 22.95%



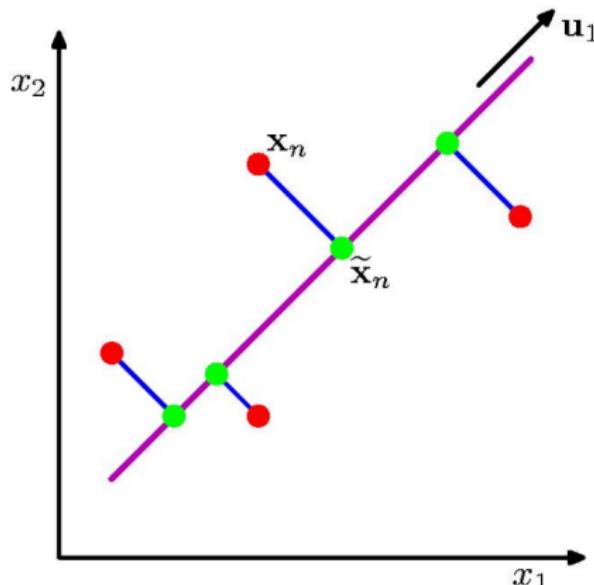
# Visualização de dados multidimensionais



Biorespose - 1776 dimensões projetadas em 2.

# Análise de Componentes Principais

- Duas abordagens para derivar o algoritmo PCA:
  - **Maximização da variância projetada:** espalhamento dos pontos projetados (verdes).
  - **Minimização do erro de reconstrução:** distância entre pontos originais (vermelhos) e a reconstrução.



# Análise de Componentes Principais

## Algoritmo PCA (visão alternativa)

- ① Encontrar os  $M$  vetores que compõem a matriz  $\mathbf{P} \in \mathbb{R}^{M \times D}$  e que solucionem o problema abaixo:

$$\underset{\mathbf{P}}{\text{Minimize}} \sum_{i=1}^N \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2, \quad \text{em que } \hat{\mathbf{x}}_i = \boldsymbol{\mu} + \mathbf{P}^\top \mathbf{P}(\mathbf{x}_i - \boldsymbol{\mu}).$$

- ② **Solução:**  $M$  autovetores  $\mathbf{p}_m|_{m=1}^M$  da matriz de covariância  $\Sigma$  correspondentes aos  $M$  maiores autovalores  $\lambda_m|_{m=1}^M$ .

- A demonstração está em Barber (2012), pg. 324, em “Deriving the optimal linear reconstruction”.

## PCA para dados de alta dimensão

- Para cenários em que  $\mathbf{X} \in \mathbb{R}^{N \times D}$ ,  $D \gg N, D \gg 1$ , a matriz de covariância  $\Sigma$  pode ser grande demais:

$$\Sigma = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^\top = \frac{1}{N-1} \bar{\mathbf{X}}^\top \bar{\mathbf{X}} \in \mathbb{R}^{D \times D}.$$

- No entanto, o número de autovalores não-nulos é limitado pelo menor valor entre  $N$  e  $D$ .
- Assim, a decomposição espectral  $\mathbf{U} \in \mathbb{R}^{D \times N}$ ,  $\Lambda \in \mathbb{R}^{N \times N}$  será:

$$\frac{1}{N-1} \bar{\mathbf{X}}^\top \bar{\mathbf{X}} \mathbf{U} = \mathbf{U} \Lambda,$$

$$\frac{1}{N-1} \bar{\mathbf{X}} \bar{\mathbf{X}}^\top \bar{\mathbf{X}} \mathbf{U} = \bar{\mathbf{X}} \mathbf{U} \Lambda,$$

$$\frac{1}{N-1} \bar{\mathbf{X}} \bar{\mathbf{X}}^\top \tilde{\mathbf{U}} = \tilde{\mathbf{U}} \Lambda.$$

- Sendo  $\tilde{\mathbf{U}} = \bar{\mathbf{X}} \mathbf{U} \in \mathbb{R}^{N \times N}$  os autovetores de  $\bar{\mathbf{X}} \bar{\mathbf{X}}^\top \in \mathbb{R}^{N \times N}$ :

$$\mathbf{U} = (N-1)^{-1/2} \mathbf{X}^\top \tilde{\mathbf{U}} \Lambda^{-1/2}, \quad (\text{Bishop, pg. 570}).$$

# Agenda

## ① Algoritmo PCA probabilístico

Solução analítica

Solução via algoritmo EM

## ② Análise fatorial

## ③ Tópicos adicionais

## ④ Referências

# Modelos de variáveis latentes contínuas

- Modelos de misturas possuem variáveis latentes (não observadas) discretas.
- Algumas situações exigem modelar variáveis latentes por distribuições contínuas.
- **Motivação:** Observações multidimensionais semelhantes podem estar próximas em um espaço projetado (um *manifold*) de dimensão (muito) inferior.

# Modelos de variáveis latentes contínuas

- Definimos uma **priori**  $p(z)$  para a variável latente contínua.
- Definimos uma **verossimilhança**  $p(x|z)$  relacionando a variável latente e a observação.
- Expressões de interesse:
  - **Verossimilhança marginal:**

$$p(x) = \int p(x|z)p(z)dz.$$

→ **Posteriori:**

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)}.$$

- **Visão gerativa:** Variáveis latentes são amostradas a partir de uma priori, que por sua vez geram “observações” via uma verossimilhança.
- **Análise de Componentes Principais (PCA):** Modelo **linear** em que a priori e a verossimilhança são **Gaussianas**.

# PCA probabilístico

- Seja a observação  $x \in \mathbb{R}^D$  associada à variável latente  $z \in \mathbb{R}^L$ .
- Definimos uma **priori Gaussiana** para o espaço latente:

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

- Para a observação, escolhemos uma **verossimilhança Gaussiana** e uma relação **linear** com  $z$ :

$$p(x|z) = \mathcal{N}(x | \mathbf{W}z + \boldsymbol{\mu}, \sigma^2 \mathbf{I}), \quad \mathbf{W} \in \mathbb{R}^{(D \times L)}, \quad \boldsymbol{\mu} \in \mathbb{R}^D, \quad \sigma^2 > 0.$$

→ Os parâmetros fixos de  $p(z)$  são suficientemente gerais, pois podemos modificar os parâmetros de  $p(x|z)$  livremente.

- Note que podemos **fatorar**  $p(x|z)$  pelas dimensões de  $x$ :

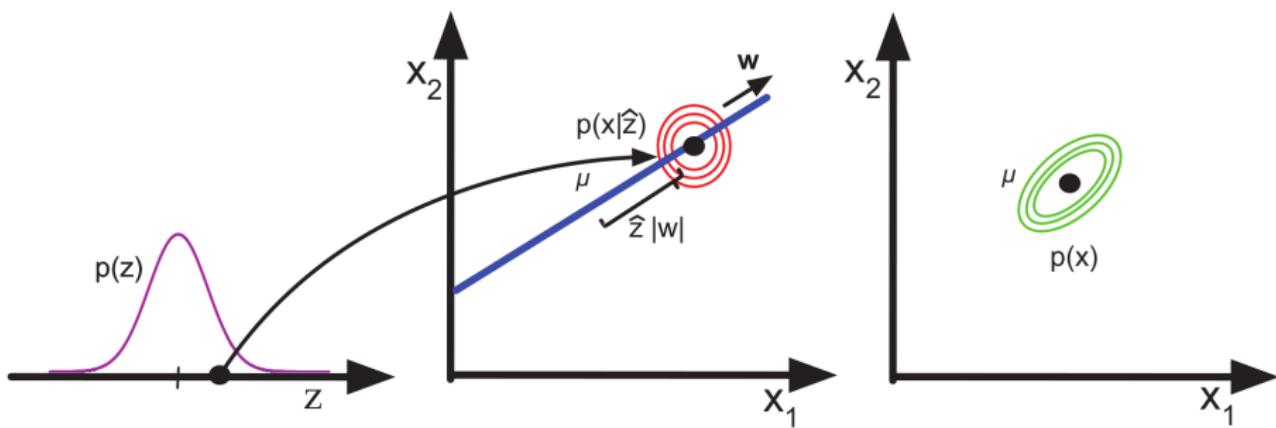
$$p(x|z) = \prod_{d=1}^D \mathcal{N}(x_d | \mathbf{w}_d^\top z + \mu_d, \sigma^2),$$

em que  $\mathbf{w}_d$  é a  $d$ -ésima linha de  $\mathbf{W}$ .

# PCA probabilístico

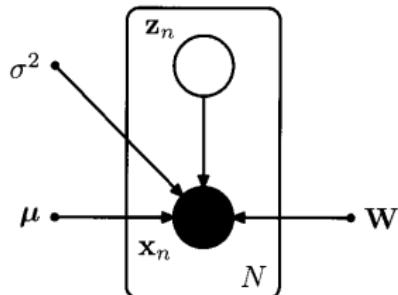
- Alternativamente, podemos imaginar uma amostra  $\hat{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  usada para obter  $x$  a partir de uma transformação linear e um **ruído de observação**  $\epsilon$ :

$$x = W\hat{z} + \mu + \epsilon, \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}).$$



- Note que  $p(x|z)$  é uma Gaussiana esférica, enquanto a marginal  $p(x)$  pode ser uma Gaussiana não-esférica.

# PCA probabilístico - Modelo gráfico probabilístico



$$\begin{aligned} p(x, z | \mathbf{W}, \boldsymbol{\mu}, \sigma^2) &= p(x|z, \mathbf{W}, \boldsymbol{\mu}, \sigma^2)p(z) \\ &= \mathcal{N}(x | \mathbf{W}z + \boldsymbol{\mu}, \sigma^2 \mathbf{I})\mathcal{N}(z | \mathbf{0}, \mathbf{I}). \end{aligned}$$

- Círculos indicam **variáveis aleatórias**.
  - Nós preenchidos são **observados**.
  - Nós em branco não são observados (são **latentes**).
- Variáveis **determinísticas** são representados por pontos.
- Setas direcionadas indicam **dependência**.
- A moldura (*plate*) indica a **repetição** dos nós que cerca.

# PCA probabilístico

- A **verossimilhança marginal** da observação é dada por:

$$\begin{aligned} p(\mathbf{x}) &= \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z} \\ &= \int \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2 \mathbf{I}) \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I}) d\mathbf{z} \\ &= \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}). \end{aligned}$$

- No entanto, a escolha dos parâmetros do modelo **não é única**.
- Suponha que façamos  $\tilde{\mathbf{W}} = \mathbf{W}\mathbf{R}$  em que  $\mathbf{R} \in \mathbb{R}^{(L \times L)}$  é uma matriz **ortogonal** (ou seja,  $\mathbf{R}\mathbf{R}^\top = \mathbf{I}$ ):

$$\tilde{\mathbf{W}} \tilde{\mathbf{W}}^\top = \mathbf{W}\mathbf{R}(\mathbf{W}\mathbf{R})^\top = \mathbf{W}\mathbf{R}\mathbf{R}^\top \mathbf{W}^\top = \mathbf{W}\mathbf{W}^\top.$$

- Qualquer matriz de rotação ortogonal  $\mathbf{R}$  não altera o modelo.

# PCA probabilístico

- Buscamos estimações de **máxima verossimilhança** para os parâmetros  $\mathbf{W}, \boldsymbol{\mu}, \sigma^2$ .
- Considerando  $N$  observações organizadas na matriz  $\mathbf{X} \in \mathbb{R}^{(N \times D)}$ :

$$\begin{aligned}\log p(\mathbf{X} | \mathbf{W}, \boldsymbol{\mu}, \sigma^2) &= \log \prod_{i=1}^N p(\mathbf{x}_i | \mathbf{W}, \boldsymbol{\mu}, \sigma^2) \\ &= \sum_{i=1}^N \log \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}) \\ &= -\frac{ND}{2} \log(2\pi) - \frac{N}{2} \log |\Sigma| - \frac{1}{2} \sum_{i=1}^N (\mathbf{x}_i - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}),\end{aligned}$$

em que  $\Sigma = \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}$ .

# PCA probabilístico

- Precisamos computar a inversa  $\Sigma^{-1} = (\mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I})^{-1}$   $D$ -dimensional, mas podemos usar a **identidade de Woodbury**:

$$(\mathbf{A} + \mathbf{U}\mathbf{C}\mathbf{V})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{U} (\mathbf{C}^{-1} + \mathbf{V}\mathbf{A}^{-1} \mathbf{U})^{-1} \mathbf{V}\mathbf{A}^{-1}.$$

- Assim:

$$\Sigma^{-1} = (\mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I})^{-1} = \frac{1}{\sigma^2} \mathbf{I} - \frac{1}{\sigma^2} \mathbf{W} (\mathbf{W}^\top \mathbf{W} + \sigma^2 \mathbf{I})^{-1} \mathbf{W}^\top,$$

em que precisamos inverter a matriz  $\mathbf{M} = \mathbf{W}^\top \mathbf{W} + \sigma^2 \mathbf{I}$ , que é  $L$ -dimensional.

- Como usualmente temos  $L < D$ , damos preferência à última expressão.

# PCA probabilístico

- As estimativas de máxima verossimilhança serão dadas por:

$$\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i, \quad \hat{\mathbf{W}} = \mathbf{V} (\Lambda - \hat{\sigma}^2 \mathbf{I})^{\frac{1}{2}} \mathbf{R}, \quad \hat{\sigma}^2 = \frac{1}{D-L} \sum_{j=L+1}^D \lambda_j,$$

- $\mathbf{V} \in \mathbb{R}^{(D \times L)}$  é formada pelos  $L$  autovetores da matriz  $\mathbf{S} = \frac{1}{N-1} \sum_i (\mathbf{x}_i - \boldsymbol{\mu})^\top (\mathbf{x}_i - \boldsymbol{\mu})$  correspondentes aos  $L$  maiores autovalores;
- $\Lambda$  é a matriz diagonal formada pelos  $L$  maiores autovalores da matriz  $\mathbf{S}$ ;
- A estimativa da variância  $\hat{\sigma}^2$  é dada pela média dos autovalores da matriz  $\mathbf{S}$  não presentes na diagonal de  $\Lambda$ ;
- $\mathbf{R} \in \mathbb{R}^{(L \times L)}$  é uma matriz ortogonal qualquer (podemos escolher  $\mathbf{R} = \mathbf{I}$ ).
- Para  $\sigma^2 \rightarrow 0$  temos  $\hat{\mathbf{W}} \rightarrow \mathbf{V} \Lambda^{\frac{1}{2}}$ , proporcional ao PCA tradicional.

# PCA probabilístico

- A **posteriori** de  $z_i$  também é **analítica**:

$$\begin{aligned} p(\mathbf{z}_i | \mathbf{x}_i) &= \frac{p(\mathbf{x}_i | \mathbf{z}_i)p(\mathbf{z}_i)}{p(\mathbf{x}_i)} \\ &\propto \mathcal{N}(\mathbf{x}_i | \hat{\mathbf{W}}\mathbf{z}_i + \hat{\boldsymbol{\mu}}, \hat{\sigma}^2 \mathbf{I}) \mathcal{N}(\mathbf{z}_i | \mathbf{0}, \mathbf{I}) \\ &= \mathcal{N}(\mathbf{z}_i | \hat{\mathbf{M}}^{-1} \hat{\mathbf{W}}^\top (\mathbf{x}_i - \hat{\boldsymbol{\mu}}), \hat{\sigma}^2 \hat{\mathbf{M}}^{-1}). \end{aligned}$$

em que usamos a propriedade de condicionamento da Gaussiana e fizemos  $\hat{\mathbf{M}} = \hat{\mathbf{W}}^\top \hat{\mathbf{W}} + \hat{\sigma}^2 \mathbf{I}$ .

- Para  $\sigma^2 \rightarrow 0$  temos  $\hat{\mathbf{W}} \rightarrow \mathbf{V}$ ,  $\hat{\mathbf{M}} \rightarrow \mathbf{I}$ ,  $\hat{\mathbf{z}}_i \rightarrow \mathbf{V}^\top (\mathbf{x}_i - \hat{\boldsymbol{\mu}})$  e  $\hat{\mathbf{x}}_i \rightarrow \mathbf{V}\hat{\mathbf{z}}_i + \hat{\boldsymbol{\mu}}$ , como no PCA tradicional.

# PCA probabilístico

## Algoritmo PCA probabilístico

- ① Calcule a matriz de covariância  $S$  a partir dos dados  $\mathbf{x}_i|_{i=1}^N$ :

$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i, \quad S = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_i - \hat{\mu})(\mathbf{x}_i - \hat{\mu})^\top.$$

- ② Construa  $V \in \mathbb{R}^{D \times L}$  com os  $L$  autovetores de  $S$  correspondentes aos maiores autovalores  $\lambda_j|_{j=1}^L$  e  $\Lambda = \text{diag}(\lambda_j|_{j=1}^L)$ .

- ③ Calcule os demais parâmetros do modelo:

$$\hat{W} = V(\Lambda - \hat{\sigma}^2 I)^{\frac{1}{2}}, \quad \hat{\sigma}^2 = \frac{1}{D-L} \sum_{j=L+1}^D \lambda_j, \quad \hat{M} = \hat{W}^\top \hat{W} + \hat{\sigma}^2 I,$$

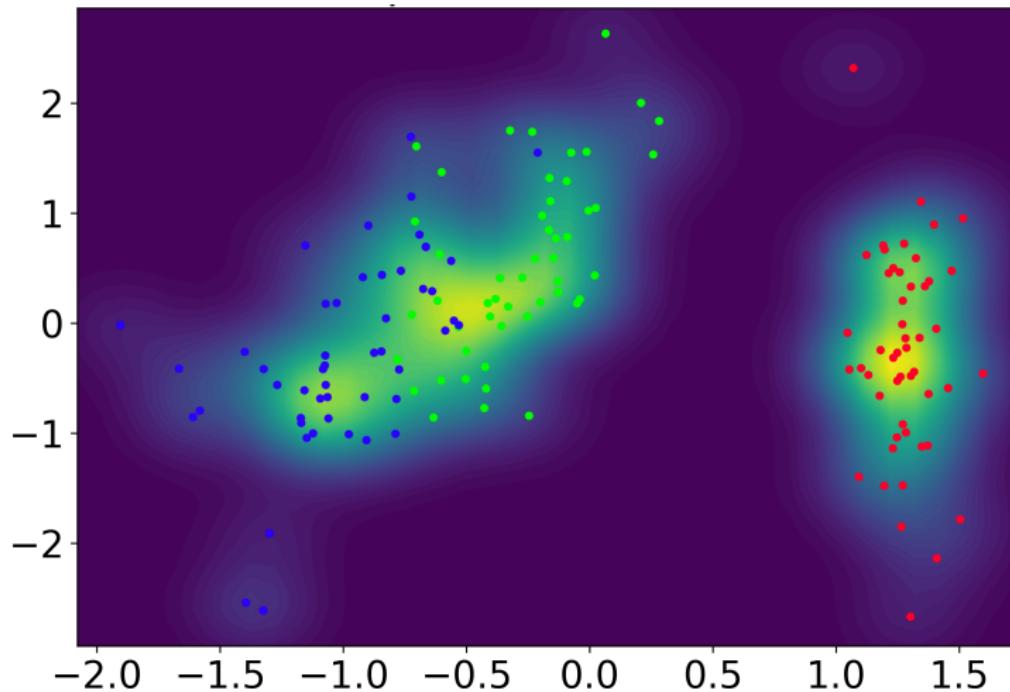
- **Projeção linear probabilística dos dados:**

$$p(z_i|\mathbf{x}_i) = \mathcal{N}(z_i|\hat{M}^{-1} \hat{W}^\top (\mathbf{x}_i - \hat{\mu}), \hat{\sigma}^2 \hat{M}^{-1}).$$

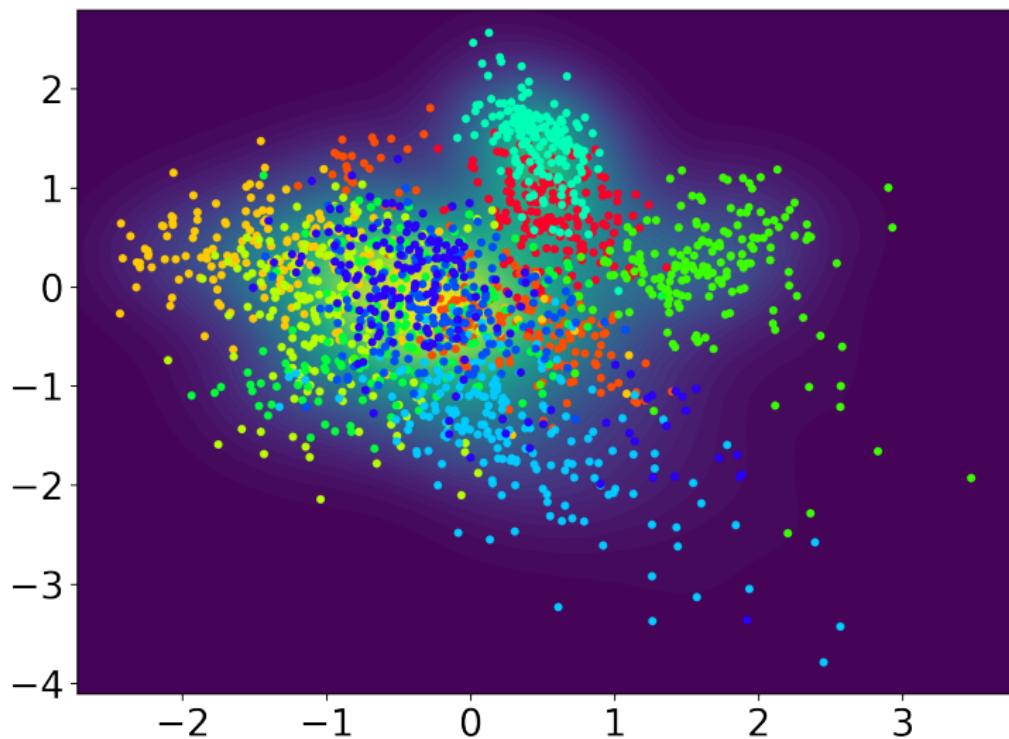
- **Reconstrução probabilística das projeções:**

$$p(\mathbf{x}_i|z_i) = \mathcal{N}(\mathbf{x}_i|\hat{W} z_i + \hat{\mu}, \hat{\sigma}^2 I).$$

# PCA probabilístico - Projeção 2D - Íris

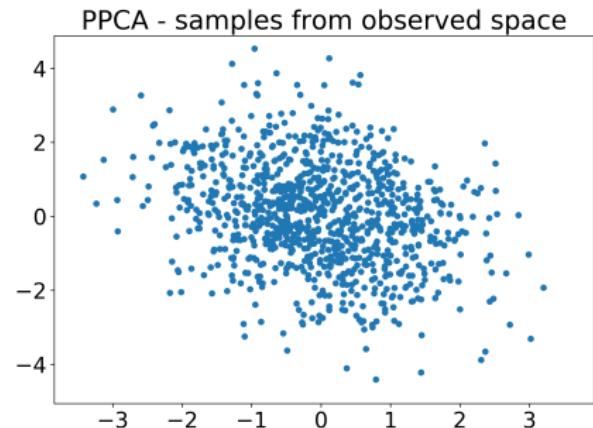
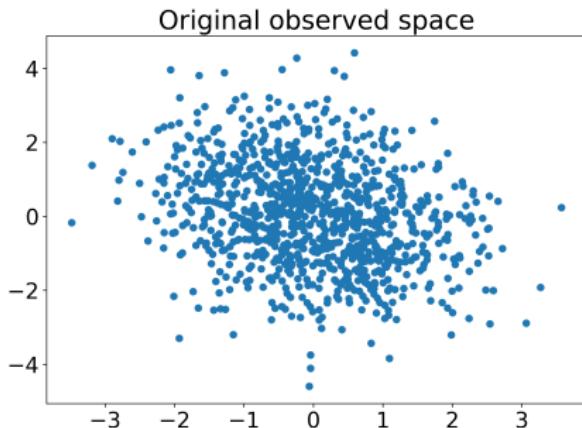


# PCA probabilístico - Projeção 2D - Digits



# PCA probabilístico - gerando amostras

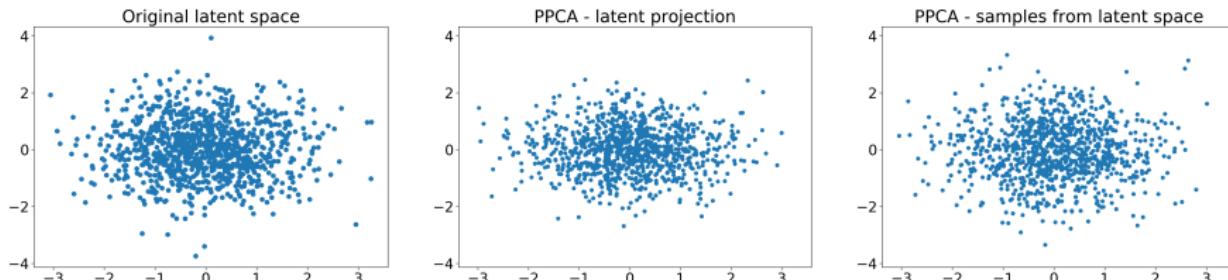
$D = 2, L = 1$



$$W = \begin{bmatrix} -0.40941532 \\ 0.95788668 \end{bmatrix}, \quad \hat{W} = \begin{bmatrix} -0.42237625 \\ 0.95572549 \end{bmatrix}$$

# PCA probabilístico - espaço latente

$D = 5, L = 2$



$$W = \begin{bmatrix} -0.40941532 & 0.95788668 \\ -1.03887743 & -1.11146061 \\ 3.93156115 & 2.78681167 \\ 0.18581575 & 0.56349231 \\ 1.53804514 & 2.49286947 \end{bmatrix}, \quad \hat{W} = \begin{bmatrix} -0.3359679 & -0.97197374 \\ 1.5385019 & 0.09994756 \\ -4.85505746 & 0.58785244 \\ -0.5047775 & -0.30446013 \\ -2.78102841 & -0.79828427 \end{bmatrix}$$

Problema de identificabilidade por causa de simetrias nos dados!

A estimativa pode ser rotacionada em um ângulo  $\theta$  pela matriz  $R$ :

$$R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

# PCA probabilístico - espaço latente

$D = 5, L = 2$

- Se tivermos  $\mathbf{W}$ , podemos aproximar a matriz de rotação  $\mathbf{R}$ :

$$\mathbf{W} = \hat{\mathbf{W}} \mathbf{R},$$

$$\hat{\mathbf{R}} = (\hat{\mathbf{W}}^\top \hat{\mathbf{W}})^{-1} \hat{\mathbf{W}}^\top \mathbf{W},$$

$$\hat{\mathbf{W}}_{\text{rotated}} = \hat{\mathbf{W}} \hat{\mathbf{R}}.$$

- Para o exemplo anterior:

$$\mathbf{W} = \begin{bmatrix} -0.40941532 & 0.95788668 \\ -1.03887743 & -1.11146061 \\ 3.93156115 & 2.78681167 \\ 0.18581575 & 0.56349231 \\ 1.53804514 & 2.49286947 \end{bmatrix}, \hat{\mathbf{W}} = \begin{bmatrix} -0.3359679 & -0.97197374 \\ 1.5385019 & 0.09994756 \\ -4.85505746 & 0.58785244 \\ -0.5047775 & -0.30446013 \\ -2.78102841 & -0.79828427 \end{bmatrix}$$

$$\hat{\mathbf{R}} = \begin{bmatrix} -0.73206802 & -0.66917947 \\ 0.65061785 & -0.77107569 \end{bmatrix}, \hat{\mathbf{W}}_{\text{rotated}} = \begin{bmatrix} -0.3864321 & 0.97428815 \\ -1.06126038 & -1.10660102 \\ 3.93669961 & 2.79562605 \\ 0.17144427 & 0.57254854 \\ 1.51652397 & 2.47654471 \end{bmatrix}$$

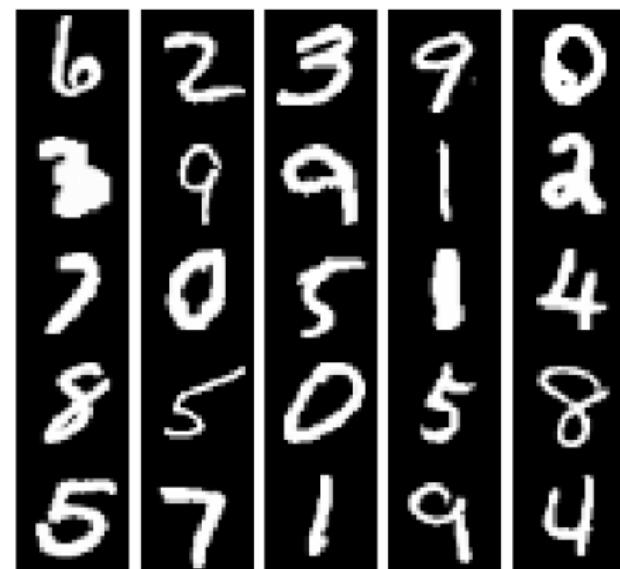
Somente útil para confirmar que estimamos os eixos adequadamente.

# PCA probabilístico - MNIST ( $D = 784$ )

Original samples from class 9



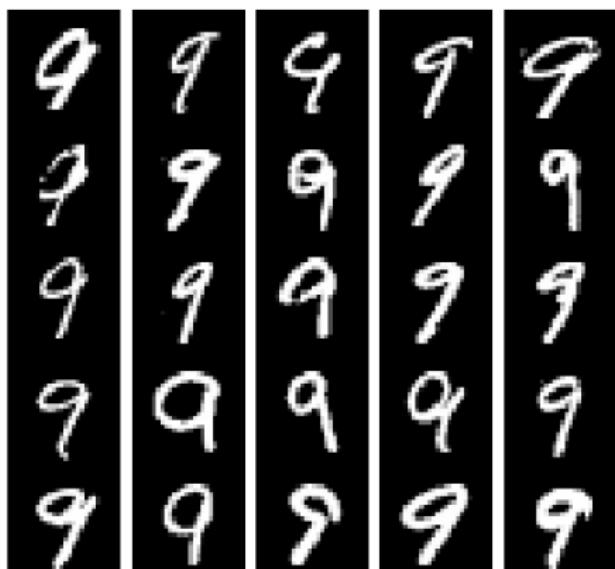
Original samples from all classes



- **Reconstrução:** Obter um modelo generativo capaz de codificar e decodificar padrões.
- **Geração:** Obter um modelo generativo e amostrar de um espaço latente para gerar novas “observações” (sem adição de ruído).

# PPCA - Reconstrução MNIST9 ( $D = 784, L = 2$ )

Original samples from class 9

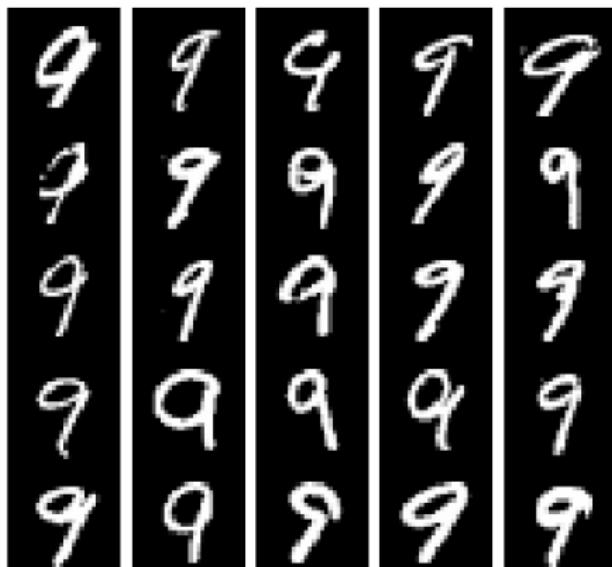


PPCA - Reconstructions

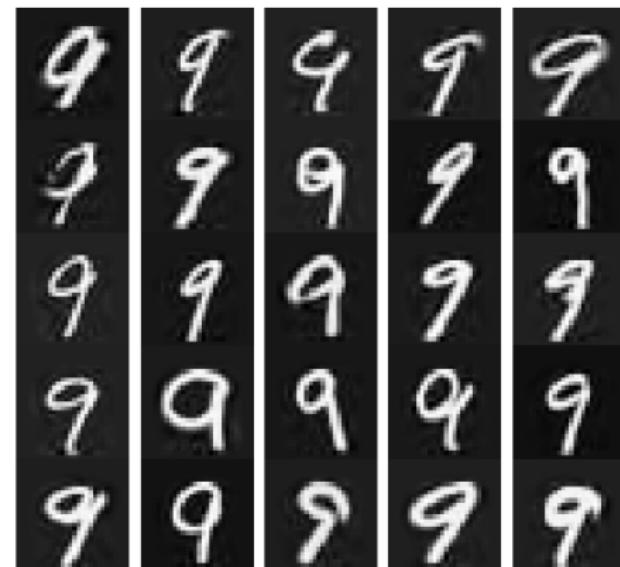


# PPCA - Reconstrução MNIST9 ( $D = 784$ , $L = 128$ )

Original samples from class 9

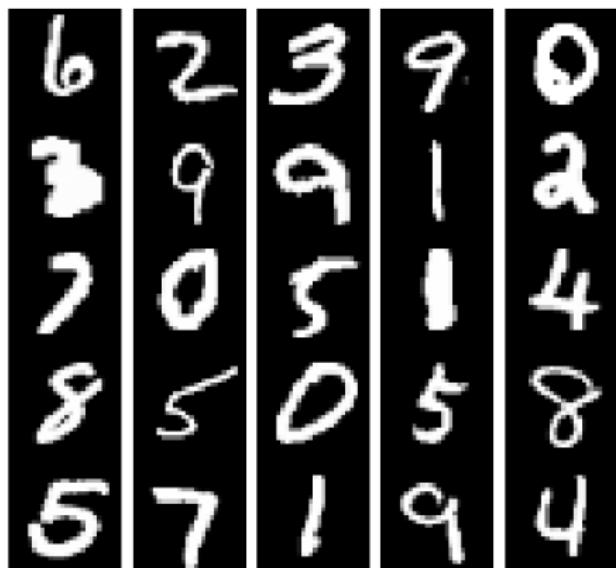


PPCA - Reconstructions

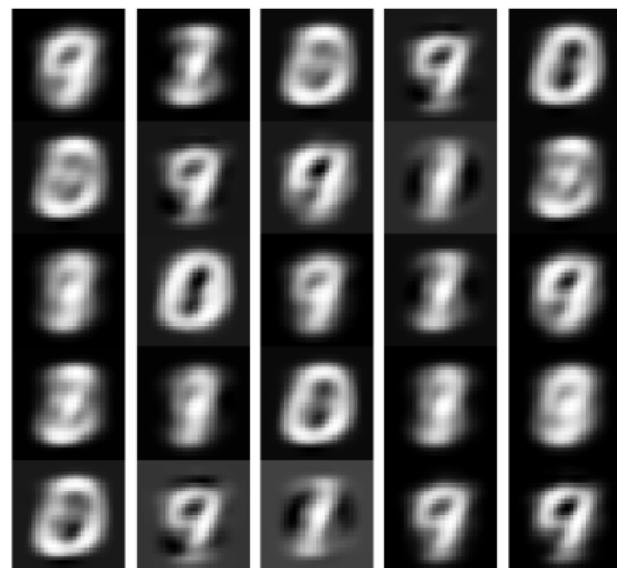


# PPCA - Reconstrução MNIST ( $D = 784, L = 2$ )

Original samples from all classes

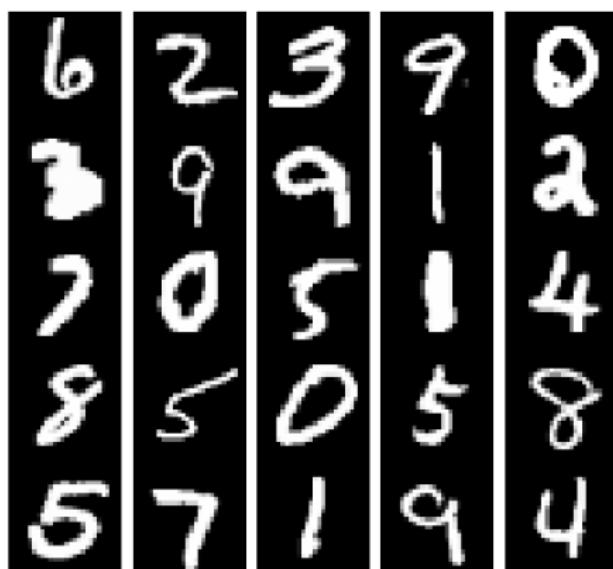


PPCA - Reconstructions

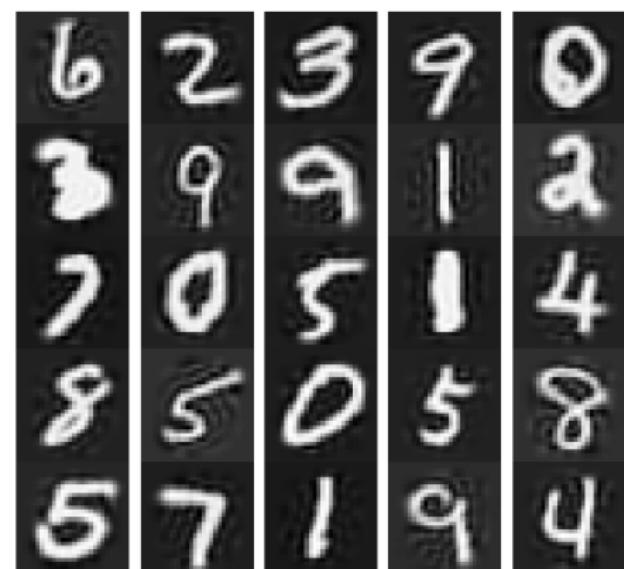


# PPCA - Reconstrução MNIST ( $D = 784$ , $L = 128$ )

Original samples from all classes



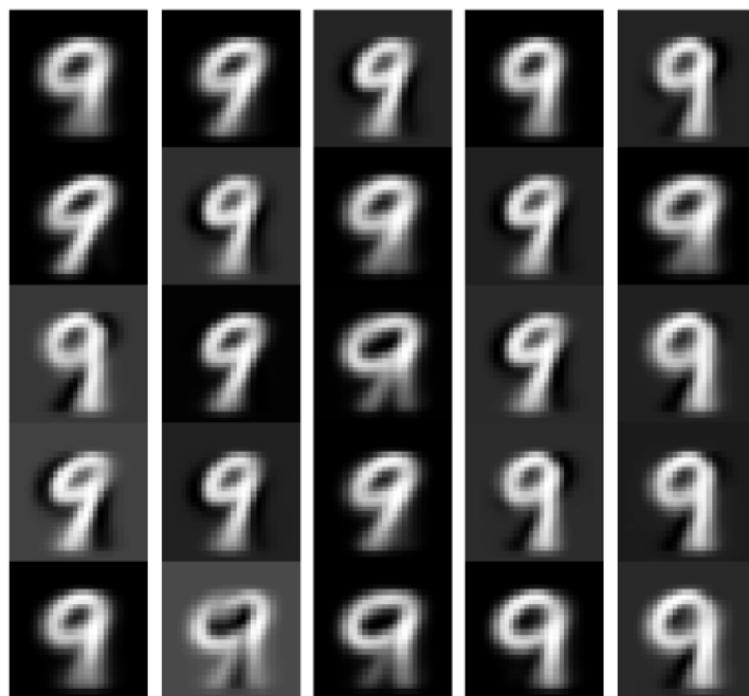
PPCA - Reconstructions



- Maior dimensão latente resulta em melhores reconstruções.

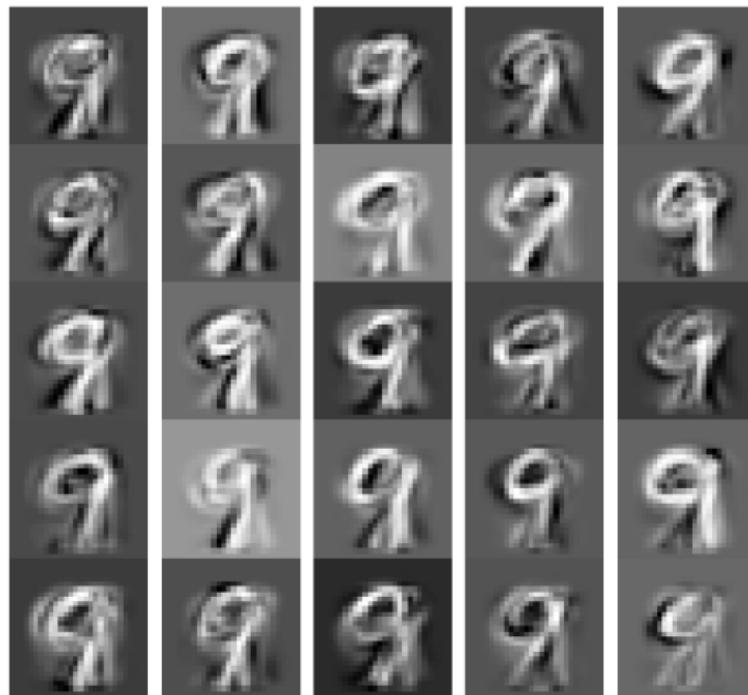
# PPCA - Geração MNIST9 ( $D = 784, L = 2$ )

PPCA - samples from observed space



# PPCA - Geração MNIST9 ( $D = 784, L = 128$ )

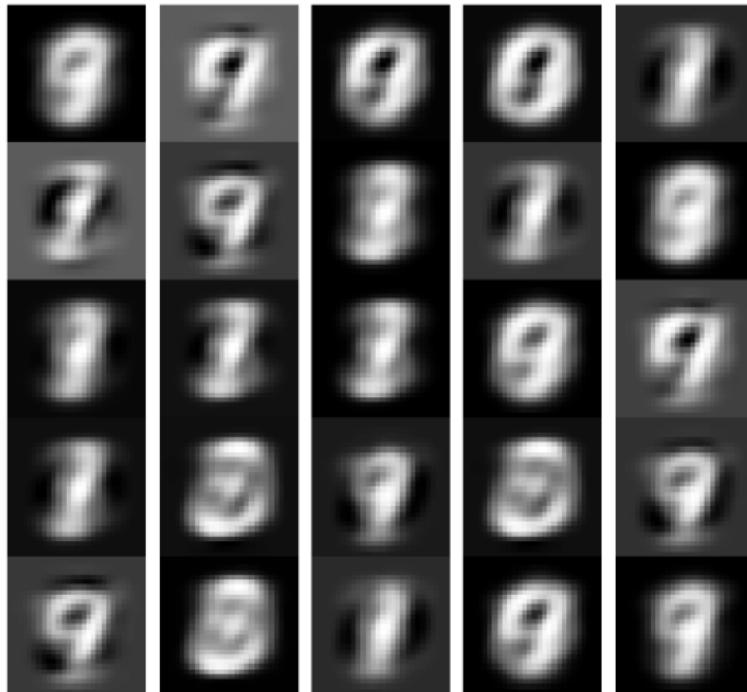
PPCA - samples from observed space



- Maior dimensão latente não implica em melhores amostras!

# PPCA - Geração MNIST ( $D = 784, L = 2$ )

PPCA - samples from observed space



- Ainda é um modelo **generativo linear!**

# PCA probabilístico via algoritmo EM

- Apesar da solução analítica para o PPCA, podemos usar o **algoritmo EM**, que possui as seguintes vantagens:
  - É mais rápido para valores altos de  $N, D$ , pois evita o cálculo de autovetores e autovalores;
  - Pode ser usado para stream de dados;
  - Pode lidar com dados faltantes;
  - Pode ser estendido para modelos mais complexos (como misturas de PPCA).
- O algoritmo converge para a solução ótima, mas não há a garantia que a matriz  $W$  será ortogonal ou possua autovetores em ordem decrescente de autovalores.
- Ambas as características acima podem ser obtidas via pós-processamento.

# PCA probabilístico via algoritmo EM

- Começamos escrevendo a **verossimilhança dos dados completos**:

$$\begin{aligned}\log p(\mathbf{X}, \mathbf{Z} | \boldsymbol{\mu}, \mathbf{W}, \sigma^2) &= \sum_{i=1}^N [\log p(\mathbf{x}_i | \underline{\mathbf{z}}_i) + \log p(\underline{\mathbf{z}}_i)] \\ &= \sum_{i=1}^N [\log \mathcal{N}(\mathbf{x}_i | \mathbf{W}\mathbf{z}_i + \boldsymbol{\mu}, \sigma^2 \mathbf{I}) + \log \mathcal{N}(\mathbf{z}_i | \mathbf{0}, \mathbf{I})] \\ &= - \sum_{i=1}^N \left\{ \frac{D}{2} \log(2\pi\sigma^2) + \frac{L}{2} \log(2\pi) + \frac{1}{2} \text{Tr}(\underline{\mathbf{z}}_i \underline{\mathbf{z}}_i^\top) \right. \\ &\quad \left. + \frac{1}{2\sigma^2} \|\mathbf{x}_i - \boldsymbol{\mu}\|^2 - \frac{1}{\sigma^2} \underline{\mathbf{z}}_i^\top \mathbf{W}^\top (\mathbf{x}_i - \boldsymbol{\mu}) + \frac{1}{2\sigma^2} \text{Tr}(\underline{\mathbf{z}}_i \underline{\mathbf{z}}_i^\top \mathbf{W}^\top \mathbf{W}) \right\}.\end{aligned}$$

# PCA probabilístico via algoritmo EM

- Para o **passo E**, tomamos a esperança em relação a  $\mathbf{z}_i$ :

$$\begin{aligned} & \mathbb{E}[\log p(\mathbf{X}, \mathbf{Z} | \boldsymbol{\mu}, \mathbf{W}, \sigma^2)] \\ &= - \sum_{i=1}^N \left\{ \frac{D}{2} \log(2\pi\sigma^2) + \frac{L}{2} \log(2\pi) + \frac{1}{2} \text{Tr}(\mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top]) \right. \\ & \quad \left. + \frac{1}{2\sigma^2} \|\mathbf{x}_i - \boldsymbol{\mu}\|^2 - \frac{1}{\sigma^2} \mathbb{E}[\mathbf{z}_i]^\top \mathbf{W}^\top (\mathbf{x}_i - \boldsymbol{\mu}) + \frac{1}{2\sigma^2} \text{Tr}(\mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top] \mathbf{W}^\top \mathbf{W}) \right\}. \end{aligned}$$

- As quantidades  $\mathbb{E}[\mathbf{z}_i]$  e  $\mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top]$  podem ser computadas a partir dos parâmetros estimados até a iteração atual:

$$\mathbf{M} = \mathbf{W}^\top \mathbf{W} + \sigma^2 \mathbf{I},$$

$$p(\mathbf{z}_i | \mathbf{x}_i) = \mathcal{N}(\mathbf{z}_i | \mathbf{M}^{-1} \mathbf{W}^\top (\mathbf{x}_i - \boldsymbol{\mu}), \sigma^2 \mathbf{M}^{-1}),$$

$$\mathbb{E}[\mathbf{z}_i] = \mathbf{M}^{-1} \mathbf{W}^\top (\mathbf{x}_i - \boldsymbol{\mu}),$$

$$\mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top] = \text{cov}[\mathbf{z}_i] + \mathbb{E}[\mathbf{z}_i] \mathbb{E}[\mathbf{z}_i]^\top = \sigma^2 \mathbf{M}^{-1} + \mathbb{E}[\mathbf{z}_i] \mathbb{E}[\mathbf{z}_i]^\top,$$

em que omitiu-se o índice  $(t-1)$  dos parâmetros  $\mathbf{W}, \boldsymbol{\mu}, \sigma^2$ .

# PCA probabilístico via algoritmo EM

- No **passo M** fazemos novas estimativas para os parâmetros a partir da otimização da expressão obtida no **passo E** (calculando o gradiente e igualando a zero):

$$\begin{aligned}\boldsymbol{\mu} &= \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i, \\ \mathbf{W} &= \left[ \sum_{i=1}^N (\mathbf{x}_i - \boldsymbol{\mu}) \mathbb{E}[\mathbf{z}_i]^\top \right] \left[ \sum_{i=1}^N \mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top] \right]^{-1}, \\ \sigma^2 &= \frac{1}{ND} \sum_{i=1}^N \left\{ \|\mathbf{x}_i - \boldsymbol{\mu}\|^2 - 2\mathbb{E}[\mathbf{z}_i]^\top \mathbf{W}^\top (\mathbf{x}_i - \boldsymbol{\mu}) \right. \\ &\quad \left. + \text{Tr}(\mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top] \mathbf{W}^\top \mathbf{W}) \right\},\end{aligned}$$

em que omitiu-se o índice  $t$  dos parâmetros atualizados.

- Note que o parâmetro  $\boldsymbol{\mu}$  não depende das variáveis latentes, sendo constante ao longo das iterações.

# PCA probabilístico via algoritmo EM

## PPCA via algoritmo EM

- ① Inicialize os parâmetros  $\mu = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i$ ,  $\mathbf{W}^{(0)}$ ,  $(\sigma^2)^{(0)}$ .  
Escalar
- ② Repita até convergir:

- ① Passo E (índices das iterações omitidos):

$$\begin{aligned}\mathbb{E}[\mathbf{z}_i] &= \mathbf{M}^{-1} \mathbf{W}^\top (\mathbf{x}_i - \mu), \quad \mathbf{M} = \mathbf{W}^\top \mathbf{W} + \sigma^2 \mathbf{I}, \\ \mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top] &= \sigma^2 \mathbf{M}^{-1} + \mathbb{E}[\mathbf{z}_i] \mathbb{E}[\mathbf{z}_i]^\top.\end{aligned}$$

- ② Passo M (índices das iterações omitidos):

$$\begin{aligned}\mathbf{W} &= \left[ \sum_{i=1}^N (\mathbf{x}_i - \mu) \mathbb{E}[\mathbf{z}_i]^\top \right] \left[ \sum_{i=1}^N \mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top] \right]^{-1}, \\ \sigma^2 &= \frac{1}{ND} \sum_{i=1}^N \left\{ \|\mathbf{x}_i - \mu\|^2 - 2\mathbb{E}[\mathbf{z}_i]^\top \mathbf{W}^\top (\mathbf{x}_i - \mu) + \text{Tr}(\mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top] \mathbf{W}^\top \mathbf{W}) \right\}.\end{aligned}$$

- Projeção linear probabilística dos dados:

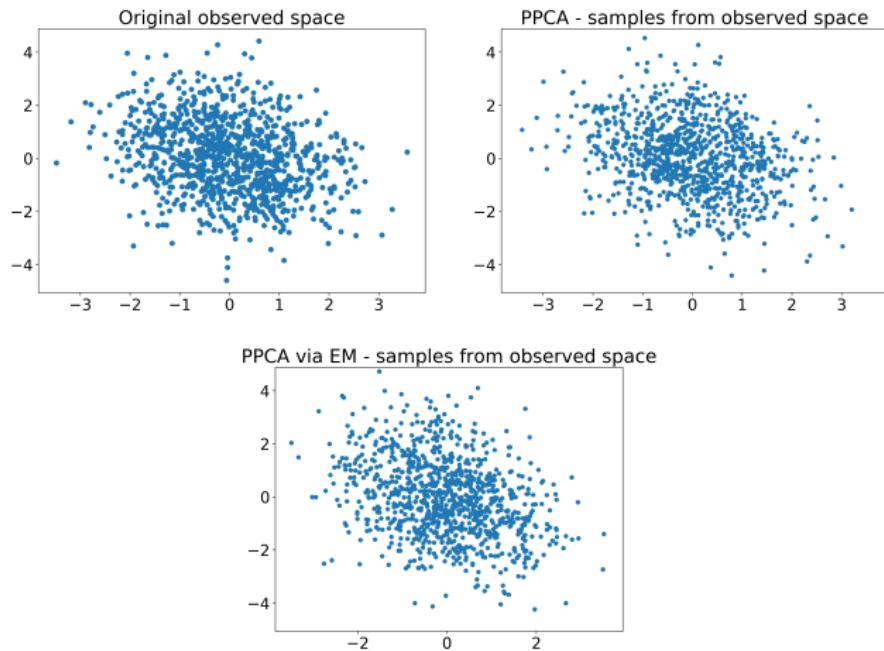
$$p(\mathbf{z}_i | \mathbf{x}_i) = \mathcal{N}(\mathbf{z}_i | \hat{\mathbf{M}}^{-1} \hat{\mathbf{W}}^\top (\mathbf{x}_i - \hat{\mu}), \hat{\sigma}^2 \hat{\mathbf{M}}^{-1}).$$

- Reconstrução probabilística das projeções:

$$p(\mathbf{x}_i | \mathbf{z}_i) = \mathcal{N}(\mathbf{x}_i | \hat{\mathbf{W}} \mathbf{z}_i + \hat{\mu}, \hat{\sigma}^2 \mathbf{I}).$$

# PCA probabilístico via algoritmo EM

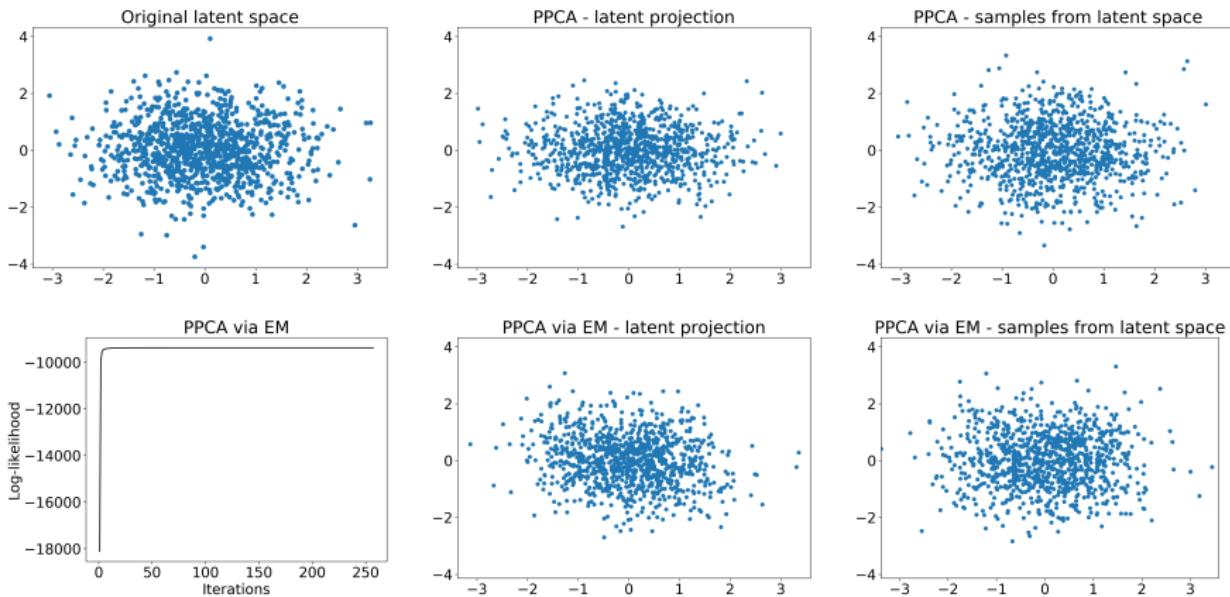
$D = 2, L = 1$



$$W = \begin{bmatrix} -0.40941532 \\ 0.95788668 \end{bmatrix}, \quad \hat{W} = \begin{bmatrix} -0.42237625 \\ 0.95572549 \end{bmatrix}, \quad \hat{W}_{\text{EM}} = \begin{bmatrix} -0.422165 \\ 0.95524748 \end{bmatrix}$$

# PCA probabilístico via algoritmo EM

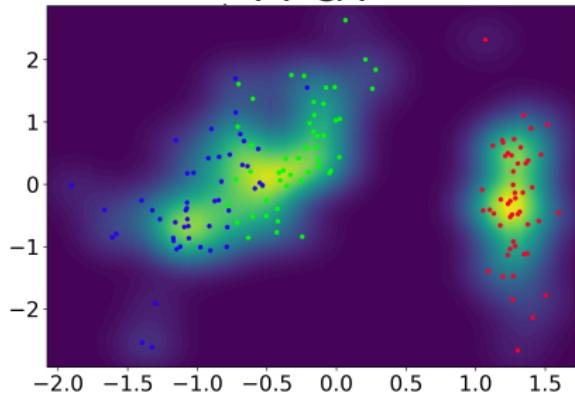
$D = 5, L = 2$



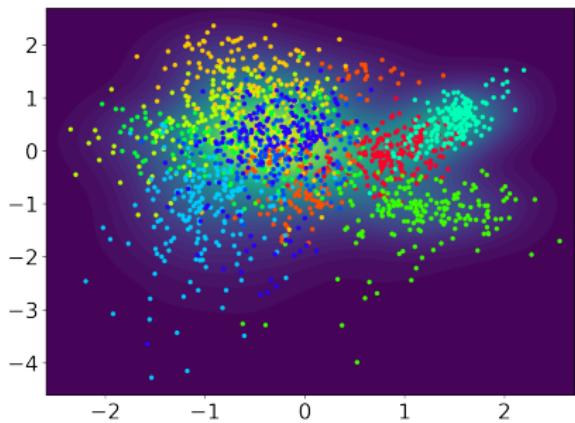
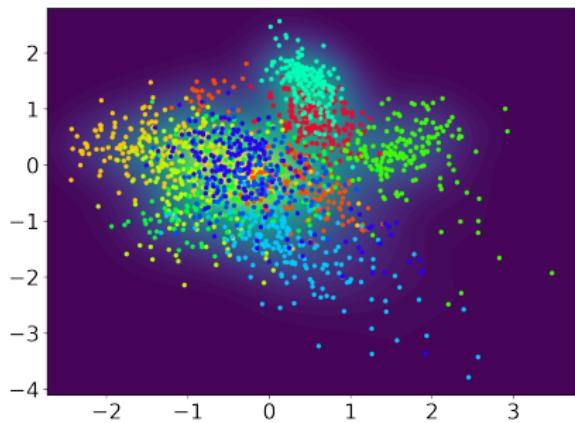
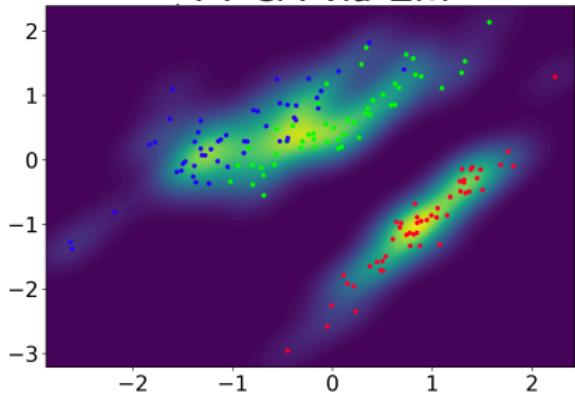
**Importante:** A estimação via algoritmo EM não garante uma matriz de projeção ortogonal, com eixos ortonormais!

# PCA probabilístico via algoritmo EM - Iris e Digits

PPCA



PPCA via EM



# Agenda

## ① Algoritmo PCA probabilístico

Solução analítica

Solução via algoritmo EM

## ② Análise fatorial

## ③ Tópicos adicionais

## ④ Referências

# Análise fatorial

- O método de análise fatorial (*factor analysis*) corresponde ao PPCA com variâncias para cada dimensão da variável observada.
- A covariância da verossimilhança dos dados passa a ser uma **matriz diagonal  $\Psi$**  ( $D \times D$ ), em vez de uma **matriz isotrópica  $\sigma^2 I$** :

$$p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x} | \mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \boldsymbol{\Psi}).$$

- As colunas da matriz  $\mathbf{W}$  são chamadas de “**cargas fatoriais**” (*factor loadings*).
- Os valores na diagonal de  $\boldsymbol{\Psi}$  são chamados de “**especificidades**” (*uniquenesses*).
- Não há uma solução analítica para todos os parâmetros, mas podemos estimá-los via **algoritmo EM**, semelhante ao PPCA.

# Análise factorial

## Análise factorial via algoritmo EM

- ① Inicialize os parâmetros  $\mu = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i$ ,  $\mathbf{W}^{(0)}$ ,  $\Psi^{(0)}$ .
- ② Repita até convergir:

- ① Passo E (índices das iterações omitidos):

$$\begin{aligned}\mathbb{E}[\mathbf{z}_i] &= \mathbf{G} \mathbf{W}^\top \Psi^{-1} (\mathbf{x}_i - \mu), \quad \mathbf{G} = (\mathbf{W}^\top \Psi^{-1} \mathbf{W} + \mathbf{I})^{-1}, \\ \mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top] &= \mathbf{G} + \mathbb{E}[\mathbf{z}_i] \mathbb{E}[\mathbf{z}_i]^\top.\end{aligned}$$

- ② Passo M (índices das iterações omitidos):

$$\mathbf{W} = \left[ \sum_{i=1}^N (\mathbf{x}_i - \mu) \mathbb{E}[\mathbf{z}_i]^\top \right] \left[ \sum_{i=1}^N \mathbb{E}[\mathbf{z}_i \mathbf{z}_i^\top] \right]^{-1},$$

$$S = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_i - \mu) (\mathbf{x}_i - \mu)^\top, \quad \Psi = \text{diag} \left[ S - \mathbf{W} \frac{1}{N} \sum_{i=1}^N \mathbb{E}[\mathbf{z}_i] (\mathbf{x}_i - \mu)^\top \right].$$

- **Projeção linear probabilística dos dados:**

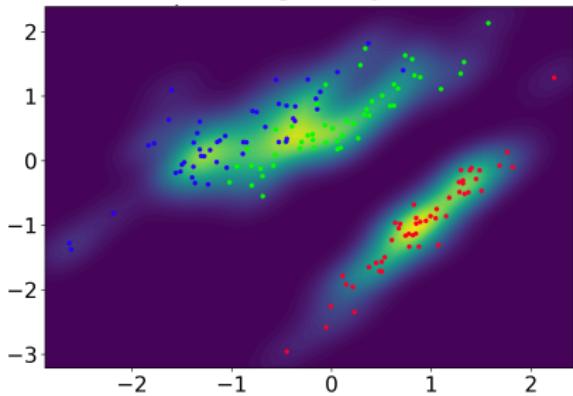
$$p(\mathbf{z}_i | \mathbf{x}_i) = \mathcal{N}(\mathbf{z}_i | \hat{\mathbf{G}} \hat{\mathbf{W}}^\top \hat{\Psi}^{-1} (\mathbf{x}_i - \hat{\mu}), \hat{\mathbf{G}}).$$

- **Reconstrução probabilística das projeções:**

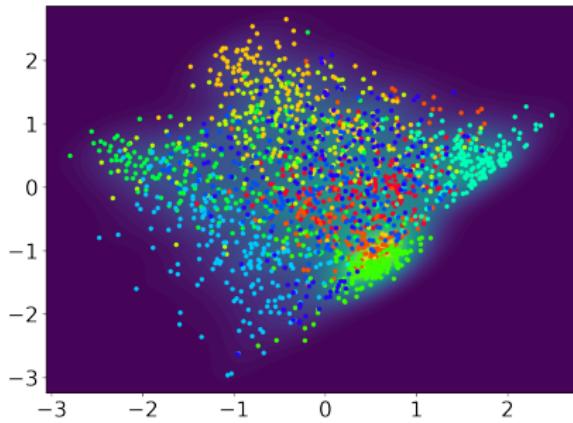
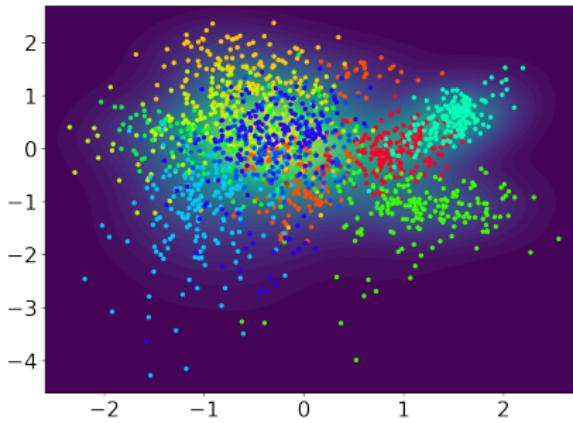
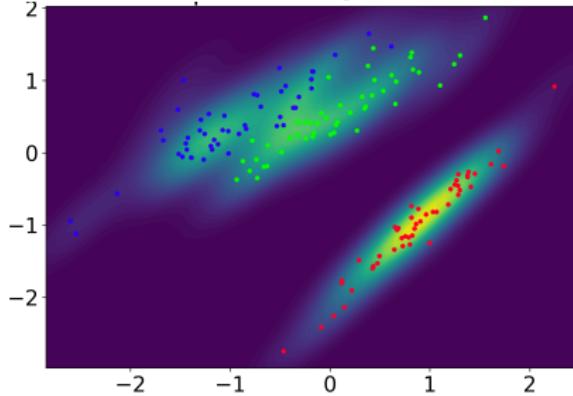
$$p(\mathbf{x}_i | \mathbf{z}_i) = \mathcal{N}(\mathbf{x}_i | \hat{\mathbf{W}} \mathbf{z}_i + \hat{\mu}, \hat{\Psi}).$$

# Análise fatorial via algoritmo EM - Iris e Digits

PPCA via EM

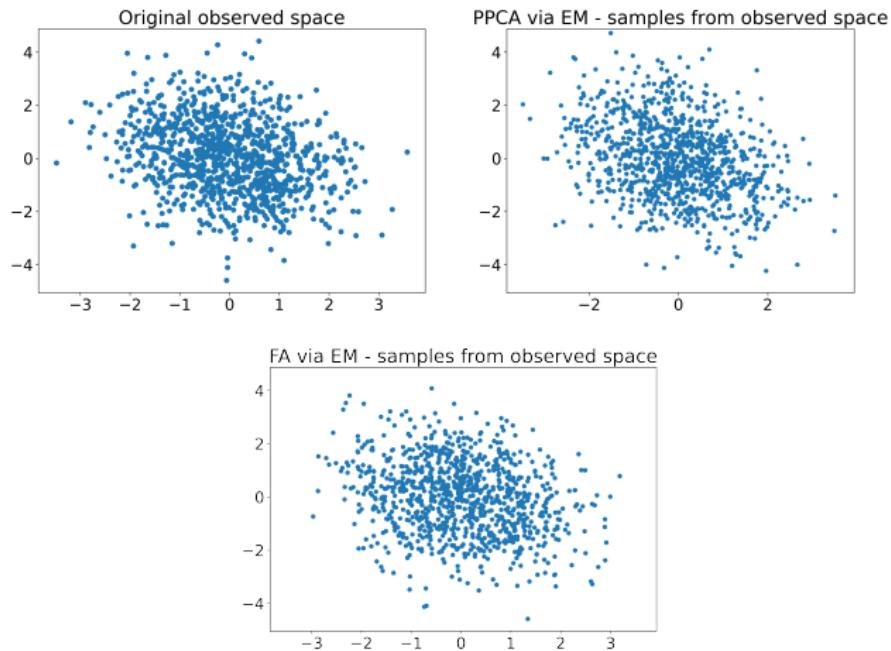


FA via EM



# Análise fatorial via algoritmo EM

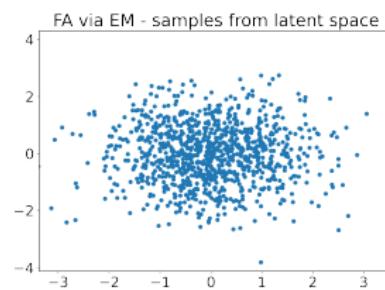
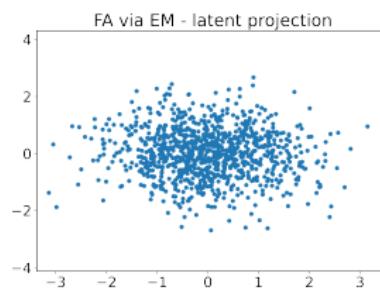
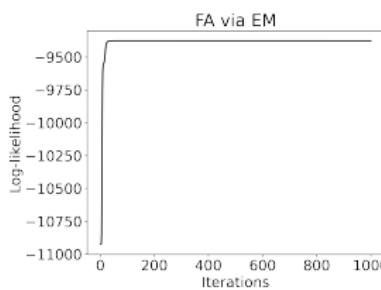
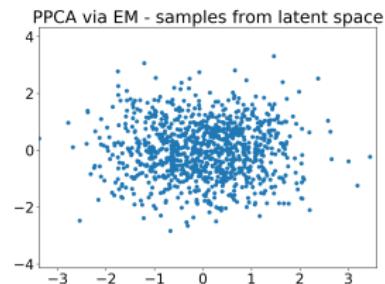
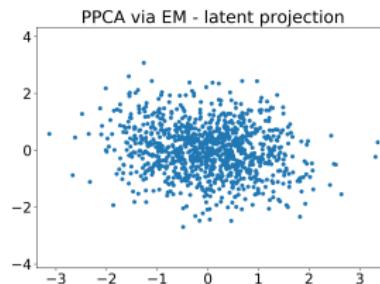
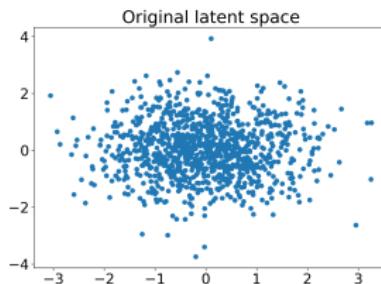
$D = 2, L = 1$



$$W = \begin{bmatrix} -0.40941532 \\ 0.95788668 \end{bmatrix}, \quad \hat{W}_{\text{PPCA}} = \begin{bmatrix} -0.422165 \\ 0.95524748 \end{bmatrix}, \quad \hat{W}_{\text{FA}} = \begin{bmatrix} 0.56282318 \\ -0.71379428 \end{bmatrix}$$

# Análise factorial via algoritmo EM

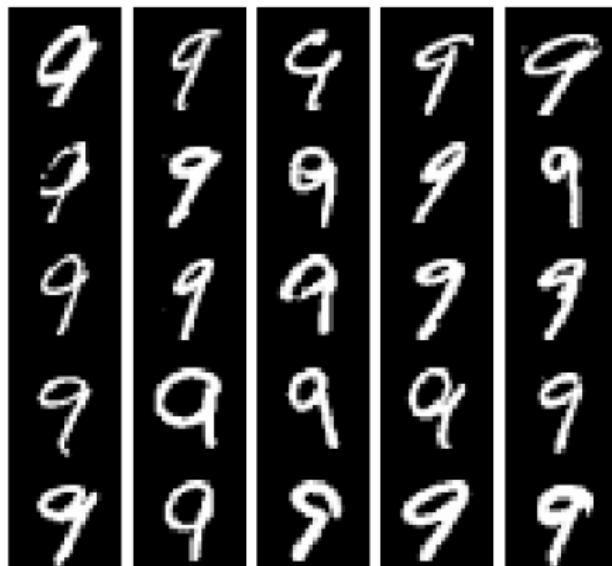
$$D = 5, L = 2$$



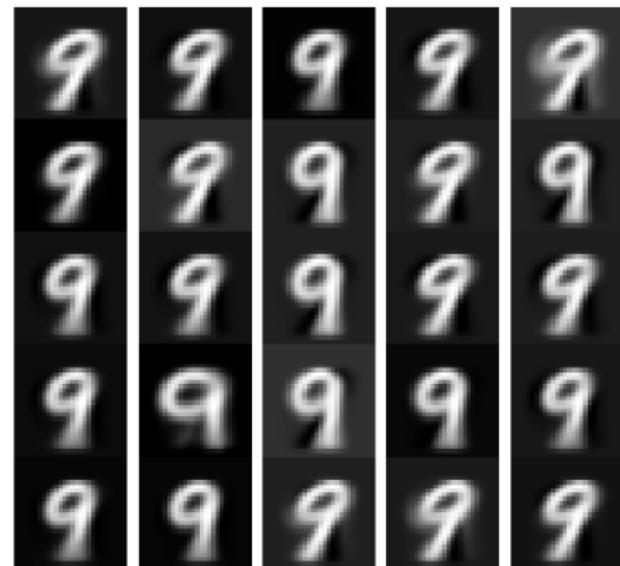
**Importante:** Lembre que o modelo FA difere do PPCA por permitir variâncias independentes para os ruídos das dimensões observadas.

# FA - Reconstrução MNIST9 ( $D = 784, L = 2$ )

Original samples from class 9

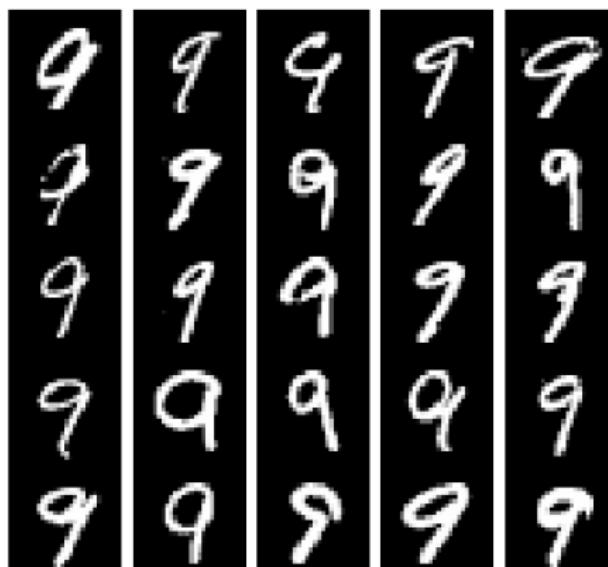


FA with EM - Reconstructions

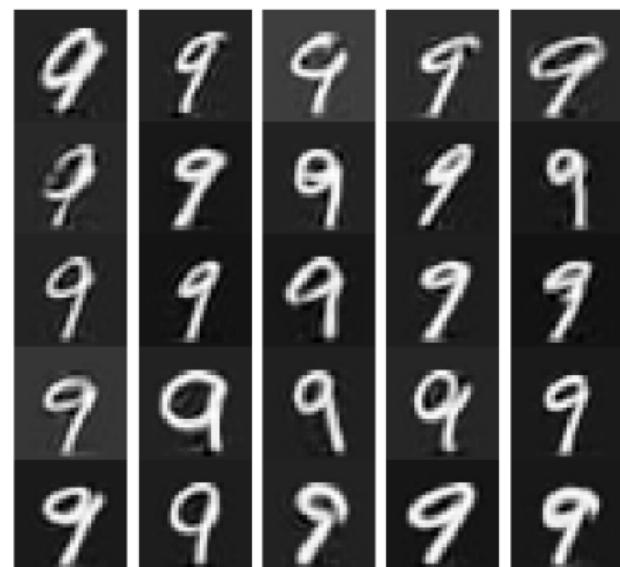


# FA - Reconstrução MNIST9 ( $D = 784$ , $L = 128$ )

Original samples from class 9



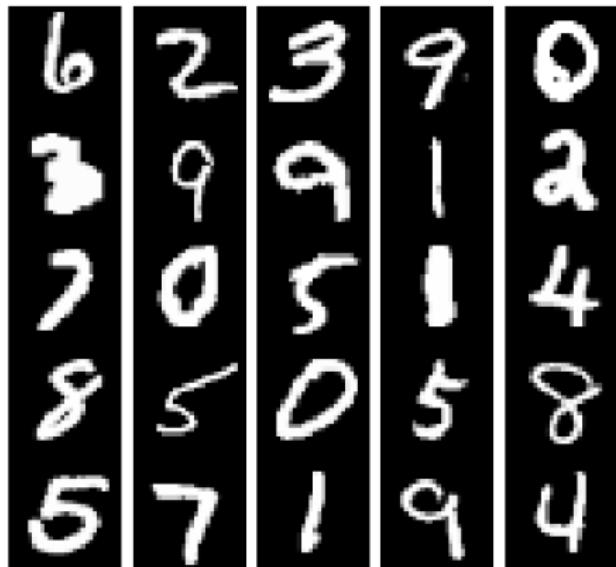
FA with EM - Reconstructions



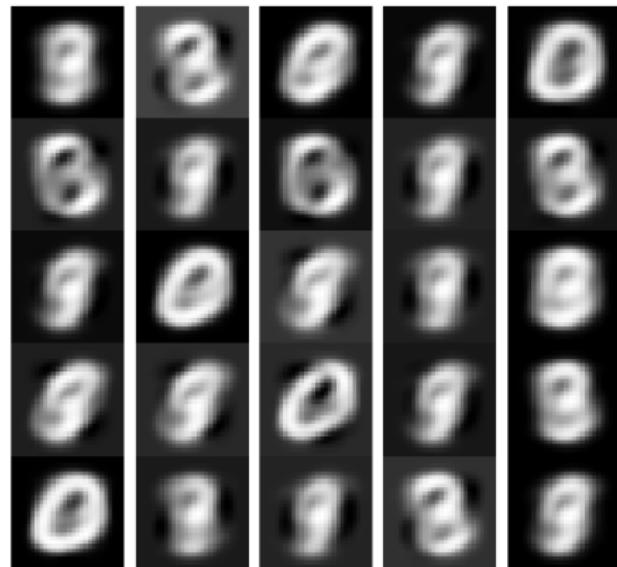
- Maior dimensão latente resulta em melhores reconstruções.

# FA - Reconstrução MNIST ( $D = 784, L = 2$ )

Original samples from all classes

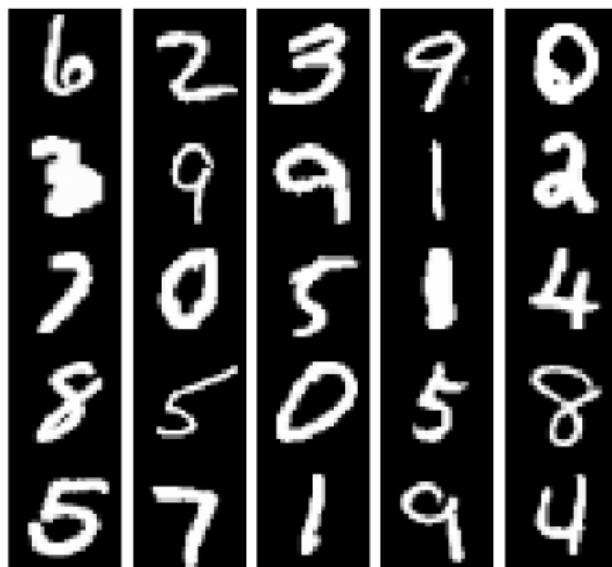


FA with EM - Reconstructions



# FA - Reconstrução MNIST ( $D = 784$ , $L = 128$ )

Original samples from all classes



FA with EM - Reconstructions



- Maior dimensão latente resulta em melhores reconstruções.

# PPCA e FA - Geração MNIST9 $D = 784, L = 2$

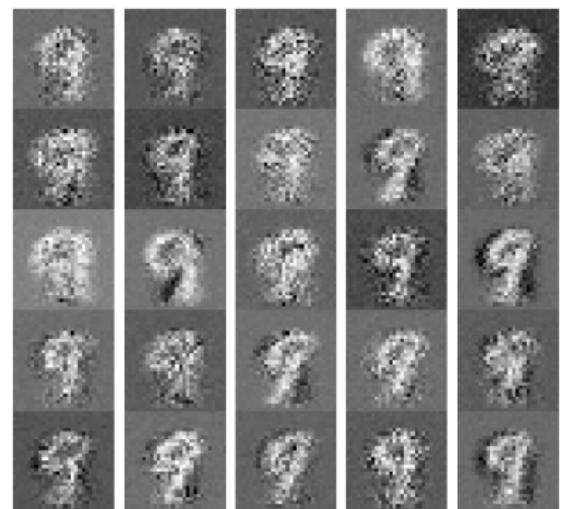
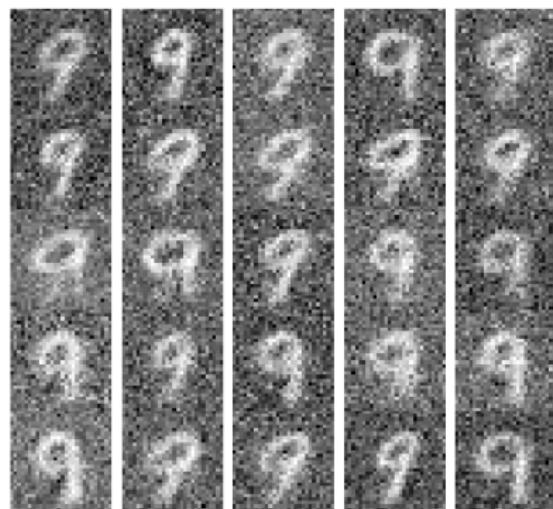
PPCA with EM - samples from observed space    FA with EM - samples from observed space



# PPCA e FA - Geração MNIST9 (com ruído)

$D = 784, L = 2$

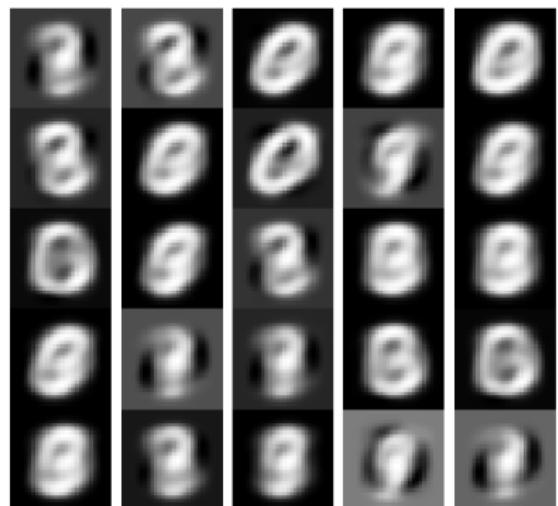
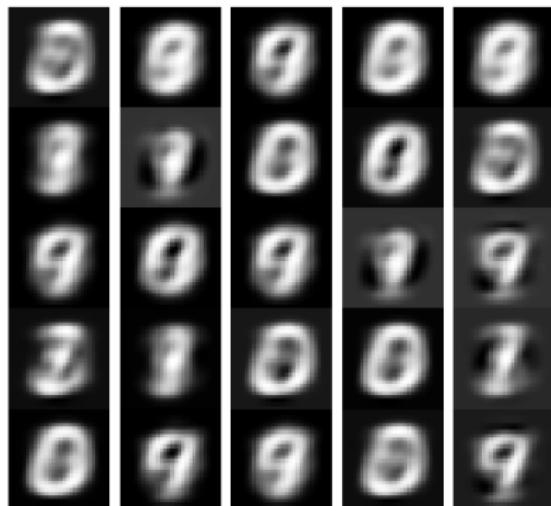
PPCA with EM - samples from observed space    FA with EM - samples from observed space



- O modelo FA é capaz de determinar variâncias de ruído diferentes para cada dimensão.

# PPCA e FA - Geração MNIST $D = 784, L = 2$

PPCA with EM - samples from observed space    FA with EM - samples from observed space

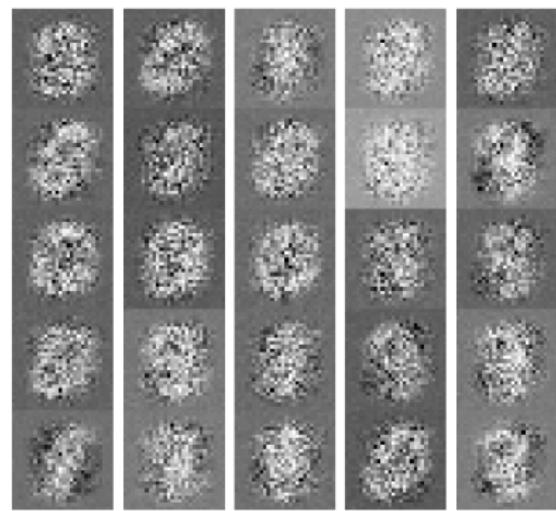
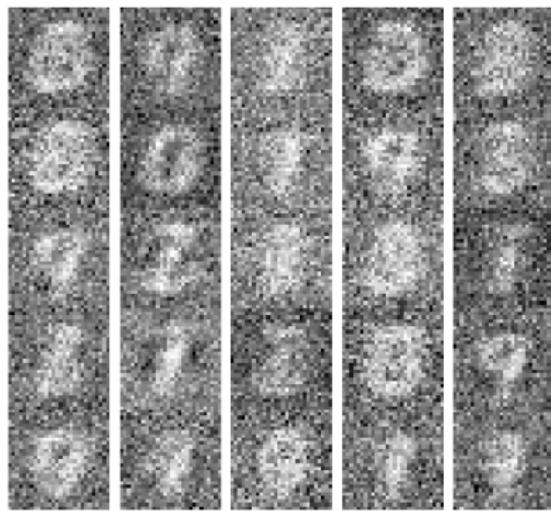


- Ainda é um modelo **generativo linear!**

# PPCA e FA - Geração MNIST (com ruído)

$D = 784, L = 2$

PPCA with EM - samples from observed space    FA with EM - samples from observed space



- Ainda é um modelo **generativo linear!**

# Agenda

## ① Algoritmo PCA probabilístico

Solução analítica

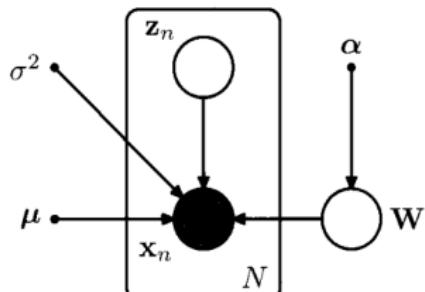
Solução via algoritmo EM

## ② Análise fatorial

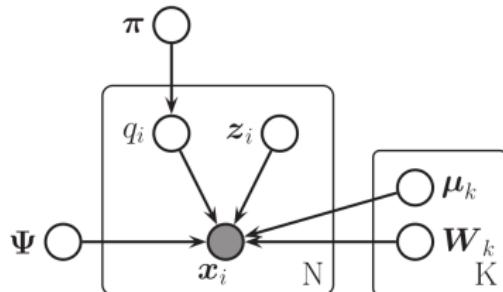
## ③ Tópicos adicionais

## ④ Referências

# Tópicos adicionais



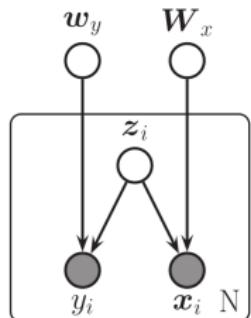
PCA Bayesiano



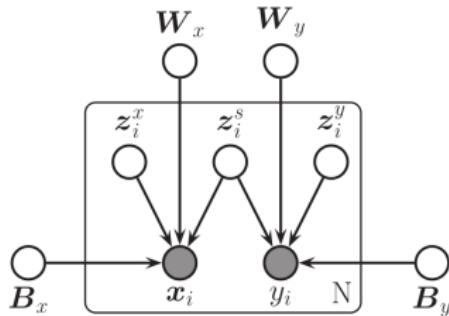
Mistura de modelos FA

- Modelo PCA para dados categóricos.
- Modelos de misturas de PCA.
- Modelos de misturas de componentes de análise fatorial.
- Modelo PCA Bayesiano.
  - Marginalização dos parâmetros da transformação linear.
- Modelo Kernel PCA.
  - O truque do kernel permite a construção de variantes não-lineares.

# Tópicos adicionais



PCA supervisionado

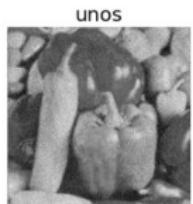
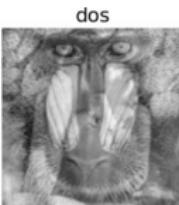
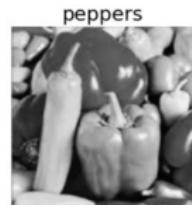


Canonical Correlation Analysis

- Modelo PCA supervisionado:

$$\begin{aligned} p(\mathbf{z}) &= \mathcal{N}(\mathbf{0}, \mathbf{I}), \\ p(y|\mathbf{z}) &= \mathcal{N}(\mathbf{w}_y^\top \mathbf{z} + \mu_y, \sigma_y^2), \\ p(\mathbf{x}|\mathbf{z}) &= \mathcal{N}(\mathbf{W}_x \mathbf{z} + \boldsymbol{\mu}_x, \sigma_x^2 \mathbf{I}). \end{aligned}$$

# Tópicos adicionais



<https://github.com/vishwajeet97/Cocktail-Party-Problem>

- Análise de Componentes Independentes (ICA).
  - Priori latente fatorada e não-Gaussianas (deixa de ser invariante à rotação):  $p(z) = \prod_j p(z_j)$ .
  - “Cocktail party problem”.

# Agenda

## ① Algoritmo PCA probabilístico

Solução analítica

Solução via algoritmo EM

## ② Análise fatorial

## ③ Tópicos adicionais

## ④ Referências

# Referências bibliográficas

- Cap. 12 - MURPHY, Kevin P. **Machine learning: a probabilistic perspective**, 2012.
- Cap. 12 - BISHOP, Christopher M. **Pattern recognition and machine learning**, 2006.
- Caps. 15 e 21 - BARBER, D. **Bayesian reasoning and machine learning**, 2012.