UNIVERSIDADE FEDERAL DO CEARÁ

DEPARTAMENTO DE COMPUTAÇÃO

Aprendizagem de Máquina

César Lincoln Cavalcante Mattos

2024

Agenda

1. Redução de dimensionalidade
2. Análise de Componentes Principais
3. Tópicos adicionais
4. Referências

Redução de dimensionalidade

- Muitos problemas de aprendizagem apresentam uma **grande quantidade de atributos**.
  - $\rightarrow$ Reconhecimento de imagens;
  - $\rightarrow$ Classificação de textos;
  - $\rightarrow$ Identificação de padrões em dados clínicos/biológicos.

Redução de dimensionalidade

- Muitos problemas de aprendizagem apresentam uma **grande quantidade de atributos**.
  - → Reconhecimento de imagens;
  - → Classificação de textos;
  - → Identificação de padrões em dados clínicos/biológicos.
- **Problemas**: Muitos atributos podem resultar em dificuldades na aprendizagem:
  - → Dificuldade nos algoritmos de otimização (mais grave);
  - → *Overfitting* (mais grave);
  - → Aumento do custo computacional;
  - → Maior custo de armazenamento.

Redução de dimensionalidade

- Muitos problemas de aprendizagem apresentam uma **grande quantidade de atributos**.
  - → Reconhecimento de imagens;
  - → Classificação de textos;
  - → Identificação de padrões em dados clínicos/biológicos.
- **Problemas**: Muitos atributos podem resultar em dificuldades na aprendizagem:
  - → Dificuldade nos algoritmos de otimização (mais grave);
  - → *Overfitting* (mais grave);
  - → Aumento do custo computacional;
  - → Maior custo de armazenamento.
- **Ideia**: **Selecionar** ou **combinar** atributos.

Redução de dimensionalidade

Seleção de atributos

- Visa encontrar um subconjunto de atributos que melhore o desempenho de um algoritmo.
- **Abordagem independente do modelo** (*filter*):
  - $\rightarrow$ Define um critério e o usa para selecionar bons atributos a partir dos dados.

Redução de dimensionalidade

Seleção de atributos

- Visa encontrar um subconjunto de atributos que melhore o desempenho de um algoritmo.
- **Abordagem independente do modelo** (*filter*):
  - $\rightarrow$ Define um critério e o usa para selecionar bons atributos a partir dos dados.
- **Abordagem associada a um modelo** (*wrapper*):
  - $\rightarrow$ Escolhe bons subconjuntos de atributos a partir de seu resultado em um modelo específico.

Redução de dimensionalidade

Seleção de atributos

- Visa encontrar um subconjunto de atributos que melhore o desempenho de um algoritmo.
- **Abordagem independente do modelo** (*filter*):
  - → Define um critério e o usa para selecionar bons atributos a partir dos dados.
- **Abordagem associada a um modelo** (*wrapper*):
  - → Escolhe bons subconjuntos de atributos a partir de seu resultado em um modelo específico.
- **Abordagem embutida em um modelo** (*embedded*):
  - → O algoritmo de aprendizagem do modelo realiza o treinamento e a seleção de atributos relevantes simultaneamente.
  - → **Exemplos**: certas árvores de decisão, LASSO (*least absolute shrinkage and selection operator*), *autoencoders*, etc.

Seleção de atributos - abordagem *filter*

Fisher *score*

- Bons atributos em uma tarefa de classificação devem possuir
  - $\rightarrow$ valores semelhantes nos padrões de uma mesma classe;
  - $\rightarrow$ valores diferentes nos padrões de classes diferentes.

Seleção de atributos - abordagem *filter*

Fisher *score*

- Bons atributos em uma tarefa de classificação devem possuir
  - $\rightarrow$ valores semelhantes nos padrões de uma mesma classe;
  - $\rightarrow$ valores diferentes nos padrões de classes diferentes.
- Sejam D atributos e K classes, calcule S_d para cada atributo d:

$$S_d = \frac{\sum_{k=1}^{K} N_k \left(\mu_{dk} - \mu_d\right)^2}{\sum_{k=1}^{K} N_k \sigma_{dk}^2}, \quad d \in \{1, \cdots, D\}.$$

  - $\rightarrow$ μ_d: média do d-ésimo atributo;
  - $\rightarrow$ μ_{dk}: média do d-ésimo atributo na k-ésima classe;
  - $\rightarrow$ σ_{dk}^2: variância do d-ésimo atributo na k-ésima classe;
  - $\rightarrow$ N_k: número de exemplos da classe k.
- **Vantagens**: Fácil de usar e independente do modelo escolhido.
- **Problema**: Desconsidera possíveis redundâncias entre atributos.

Ilustração de seleção de atributos via Fisher *score*

Seleção de atributos - abordagem *wrapper*

- Atributos são selecionados para um modelo específico.
- Métricas de qualidade são calculadas via validações cruzadas.

Seleção de atributos - abordagem *wrapper*

- Atributos são selecionados para um modelo específico.
- Métricas de qualidade são calculadas via validações cruzadas.
- **Problema**: Tarefa de otimização combinatória com solução ótima potencialmente difícil de ser obtida.
- **Ideia**: Seguir estratégias gulosas (*greedy*).
  - → **Greedy forward selection**: Atributos são testados e adicionados ao subconjunto selecionado.
  - → **Greedy backward selection**: Atributos são testados e removidos do subconjunto selecionado.

Seleção de atributos - abordagem *wrapper*

- Atributos são selecionados para um modelo específico.
- Métricas de qualidade são calculadas via validações cruzadas.
- **Problema**: Tarefa de otimização combinatória com solução ótima potencialmente difícil de ser obtida.
- **Ideia**: Seguir estratégias gulosas (*greedy*).
  - → **Greedy forward selection**: Atributos são testados e adicionados ao subconjunto selecionado.
  - → **Greedy backward selection**: Atributos são testados e removidos do subconjunto selecionado.
- **Ideia**: Usar algoritmos de otimização global estocástica:
  - → **Algoritmos metaheurísticos**: Algoritmos genéticos, *Particle Swarm Optimization* (PSO), etc.
  - → **Otimização Bayesiana**.

Agenda

1. Redução de dimensionalidade
2. Análise de Componentes Principais
3. Tópicos adicionais
4. Referências

Combinação de atributos

- **Problema**: Selecionar subconjuntos de atributos pode resultar em perda de informação dos dados.

Combinação de atributos

- **Problema**: Selecionar subconjuntos de atributos pode resultar em perda de informação dos dados.
- **Ideia**: Combinar os atributos originais em vetores de menor dimensão.

Combinação de atributos

- **Problema**: Selecionar subconjuntos de atributos pode resultar em perda de informação dos dados.
- **Ideia**: Combinar os atributos originais em vetores de menor dimensão.
- **Problemas**: Como combinar? Linear ou não linearmente? Qual critério de combinação?

Análise de Componentes Principais

Principal Component Analysis (PCA)

- Método **não-supervisionado** de geração de novos atributos a partir da **combinação linear dos atributos** originais.
- Também chamado de **Transformada de Karhunen-Loève**.
- Pode ser usado para redução de dimensionalidade escolhendo-se um subconjunto dos atributos gerados.
- **Considerações**:
  - $\rightarrow$ Os dados são contínuos;
  - $\rightarrow$ A informação está na maneira como os atributos variam.

Ilustração do conceito de componentes principais

Análise de Componentes Principais

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

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

  em que $\boldsymbol{P} \in \mathbb{R}^{M \times D}$ é a **matriz de projeção**.
- **Objetivo**: Maximizar a variância dos dados projetados.

Análise de Componentes Principais

- Seja $\boldsymbol{p}_m \in \mathbb{R}^D$ a m-ésima linha da matriz de projeção $\boldsymbol{P}$.

Análise de Componentes Principais

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

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

Análise de Componentes Principais

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

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

- Trabalhando a expressão anterior, temos:

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

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

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

Análise de Componentes Principais

- Desejamos maximizar a variância projetada $\sigma_1^2 = \boldsymbol{p}_1^\top \boldsymbol{\Sigma} \boldsymbol{p}_1$.
- Para isso, limitamos a norma do vetor de projeção: $\boldsymbol{p}_1^\top \boldsymbol{p}_1 = 1$.

Análise de Componentes Principais

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

$$\begin{aligned} \underset{\boldsymbol{p}_1}{\text{Maximize}} \ & \boldsymbol{p}_1^\top \boldsymbol{\Sigma} \boldsymbol{p}_1, \\ \text{s.a.} \ & \boldsymbol{p}_1^\top \boldsymbol{p}_1 = 1. \end{aligned}$$

Análise de Componentes Principais

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

$$\begin{aligned} \underset{\boldsymbol{p}_1}{\text{Maximize}}\ & \boldsymbol{p}_1^\top \boldsymbol{\Sigma} \boldsymbol{p}_1, \\ \text{s.a.}\ & \boldsymbol{p}_1^\top \boldsymbol{p}_1 = 1. \end{aligned}$$

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

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

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

$$\begin{aligned} 2\boldsymbol{\Sigma} \boldsymbol{p}_1 - 2\lambda_1 \boldsymbol{p}_1 &= 0 \\ \boldsymbol{\Sigma} \boldsymbol{p}_1 &= \lambda_1 \boldsymbol{p}_1. \end{aligned}$$

Análise de Componentes Principais

- A condição $\boldsymbol{\Sigma}\boldsymbol{p}_1 = \lambda_1 \boldsymbol{p}_1$ garante que $\boldsymbol{p}_1$ é **autovetor** da matriz $\boldsymbol{\Sigma}$ dos dados originais, com **autovalor** correspondente λ_1.

Análise de Componentes Principais

- A condição $\boldsymbol{\Sigma}\boldsymbol{p}_1 = \lambda_1 \boldsymbol{p}_1$ garante que $\boldsymbol{p}_1$ é **autovetor** da matriz $\boldsymbol{\Sigma}$ dos dados originais, com **autovalor** correspondente λ_1.
- A variância projetada σ_1^2 será agora dada por:

$$\sigma_1^2 = \boldsymbol{p}_1^\top \boldsymbol{\Sigma} \boldsymbol{p}_1 = \lambda_1 \boldsymbol{p}_1^\top \boldsymbol{p}_1 = \lambda_1.$$

Análise de Componentes Principais

- A condição $\boldsymbol{\Sigma}\boldsymbol{p}_1 = \lambda_1 \boldsymbol{p}_1$ garante que $\boldsymbol{p}_1$ é **autovetor** da matriz $\boldsymbol{\Sigma}$ dos dados originais, com **autovalor** correspondente λ_1.
- A variância projetada σ_1^2 será agora dada por:

$$\sigma_1^2 = \boldsymbol{p}_1^\top \boldsymbol{\Sigma} \boldsymbol{p}_1 = \lambda_1 \boldsymbol{p}_1^\top \boldsymbol{p}_1 = \lambda_1.$$

- Logo, σ_1^2 será maximizada escolhendo-se $\boldsymbol{p}_1$ igual ao **autovetor correspondente ao maior autovalor**.
- Os demais vetores de projeção correspondem aos demais autovetores da matriz $\boldsymbol{\Sigma}$.

Análise de Componentes Principais

Algoritmo de estimação do PCA

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

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

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

- **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

Algoritmo SVD (*Singular Value Decomposition*)

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

$$\boldsymbol{M} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$$

  - $\rightarrow$ $\boldsymbol{S} \in \mathbb{R}^{A \times B}$ reúne em sua "diagonal" os valores singulares de $\boldsymbol{M}$.
  - $\rightarrow$ $\boldsymbol{U} \in \mathbb{R}^{A \times A}$ é ortogonal.
  - $\rightarrow$ $\boldsymbol{V}^\top \in \mathbb{R}^{B \times B}$ é ortogonal.
- Para uma matriz $\boldsymbol{M}$ quadrada:
  - $\rightarrow$ $\boldsymbol{S}$ é diagonal com elementos iguais aos autovalores de $\boldsymbol{M}$.
  - $\rightarrow$ $\boldsymbol{U} = \boldsymbol{V}$ possui os autovetores de $\boldsymbol{M}$ nas colunas.
- Usualmente usado para computar a matriz de projeção do PCA:

$$\boldsymbol{P} = \boldsymbol{U}^\top.$$

Análise de Componentes Principais

- **Matriz de projeção**: $\boldsymbol{P} \in \mathbb{R}^{M \times D}$.
- **Projeção linear dos dados**: $\boldsymbol{z}_i = \boldsymbol{P}(\boldsymbol{x}_i - \boldsymbol{\mu})$.
- **Reconstrução das projeções**: $\hat{\boldsymbol{x}}_i = \boldsymbol{\mu} + \boldsymbol{P}^\top \boldsymbol{z}_i$.
- 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 = 2.07e-22

100 componentes principais
Erro de reconstrução = 7.01e+01

50 componentes principais
Erro de reconstrução = 2.59e+02

25 componentes principais
Erro de reconstrução = 6.39e+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 = 1.14e+03

100 componentes principais
Erro de reconstrução = 3.22e+04

50 componentes principais
Erro de reconstrução = 7.56e+04

25 componentes principais
Erro de reconstrução = 1.23e+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

Bioresponse - 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 de estimação do PCA (visão alternativa)

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

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

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

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

Agenda

1. Redução de dimensionalidade
2. Análise de Componentes Principais
3. Tópicos adicionais
4. Referências

Tópicos adicionais

- Existem diversas variantes e extensões do algoritmo PCA clássico para abordagens probabilísticas e/ou não lineares:
  - → *Probabilistic* PCA (PPCA);
  - → Bayesian PCA;
  - → *Factorial Analysis*;
  - → Mistura de PCAs;
  - → Kernel PCA;
  - → PCA multinomial (mPCA);
  - → PCA supervisionado;
  - → *Gaussian Process Latent Variable Model* (GPLVM).
- Modelos lineares e não-lineares de variáveis latentes contínuas.

Agenda

1. Redução de dimensionalidade
2. Análise de Componentes Principais
3. Tópicos adicionais
4. Referências

Referências bibliográficas

- **Cap. 12** - MURPHY, Kevin P. **Machine learning: a probabilistic perspective**, 2012.
- **Cap. 10** - DEISENROTH, M. *et al.* **Mathematics for machine learning**. 2019.
- **Cap. 12** - BISHOP, C. **Pattern recognition and machine learning**, 2006.