

- 
- 
- 
- 
- 

# Análise de Componentes Principais

(Principal Component Analysis - PCA)

Aluizio Fausto Ribeiro Araújo  
Universidade Federal de Pernambuco  
Centro de Informática



# Conteúdo

- Introdução
- Revisão Matemática
  - Revisão Estatística
  - Revisão de Álgebra Matricial
- Análise de Componentes Principais
  - Método da Covariância
    - Exemplo Ilustrativo
- PCA e Aprendizagem Auto-Organizada
  - Algoritmo Hebbiano Generalizado – AHG
  - Extração Adaptativa de Componentes Principais (*Adaptive Principal Components Extraction*) – APEX
- Kernel PCA
- Referências

# Introdução

- Análise de Componentes Principais é um método de identificar padrões nos dados, para expressá-los de modo a salientar as similaridades e diferenças existentes entre eles.
- Além de se encontrar os padrões nos dados, PCA pode ser usado para comprimir ou reduzir sua dimensionalidade.
- Portanto, PCA pode ser entendido como um método estatístico para seleção de características e redução de dimensionalidade:
  - Seleção de características envolve um processo no qual um espaço de dados é transformado em um espaço de características.
  - A transformação diminui o número de características tomadas, reduzindo dimensionalidade do conjunto de dados mas preservando a maior parte da informação intrínseca a ele.

# Introdução

- Razões para reduzir dimensionalidade:
  - A maioria das técnicas de aprendizagem de máquinas perdem desempenho para dados de alta dimensionalidade:
    - Maldição da dimensionalidade;
    - Métricas de distância perdem capacidade discriminante;
  - A dimensão relevante pode ser muito menor;
  - Permite visualização: Projeta dados em 2D ou 3D;
  - Possibilita compressão de dados;
  - Reduz tempo de processamento e número de parâmetros;
  - Viabiliza remoção de ruído.

# Revisão de Estatística e Álgebra

- A seguir são expostos os conceitos básicos de estatística e álgebra necessários à compreensão do processo de Análise de Componentes Principais.
  - Conceitos de estatística:
    - Média, desvio padrão, variância, covariância, matriz de covariância.
  - Conceitos de álgebra matricial:
    - Autovalores e autovetores.

# Conceitos de Estatística: Média

- Seja um conjunto  $X$  de dados retirados aleatoriamente de um conjunto de dados qualquer. Nela,  $X_1$  denota o primeiro elemento do conjunto,  $X_i$  o  $i$ -ésimo, e assim por diante até  $X_n$  que representa o último elemento do conjunto contendo  $n$  elementos.
- A *média* do conjunto de dados  $X$  pode ser calculada por:

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} \quad (1)$$

- onde  $\bar{X}$  indica o valor médio do conjunto de dados  $X$ .

# Conceitos de Estatística: Desvio Padrão

- O **desvio padrão** de um conjunto de dados define o quão espalhado em torno da média esse conjunto está. Calcula-se o desvio padrão através da equação abaixo.

$$s = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)}} \quad (2)$$

- onde  $s$  é o símbolo utilizado comumente para representar o desvio padrão de uma amostra.

# Conceitos de Estatística: Variância

- **Variância** é outra medida do espalhamento do conjunto de dados. O cálculo da variância é apresentado a seguir:

$$s^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)} \quad (3)$$

- $s^2$  denota a variância de um conjunto de dados. A equação (3) pode ser reescrita de forma mais clara:

$$\text{var}(X) = \frac{\sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})}{(n-1)} \quad (4)$$

onde  $\text{var}(X)$  caracteriza a variância do conjunto  $X$ .



# Conceitos de Estatística: Variância

- O desvio padrão e a variância operam apenas sobre uma dimensão.
  - Em uma amostra com mais de uma dimensão seria necessário calculá-los para cada dimensão independentemente das outras.
- Contudo, usualmente é interessante ter uma medida de como os dados em cada dimensão variam em função da média e como estas variações, em dimensões distintas, se relacionam entre si.
  - A *covariância* é responsável por essa medida e é sempre calculada entre duas dimensões.

# Conceitos de Estatística: Covariância

- A **covariância** entre duas variáveis aleatórias reais  $X$  e  $Y$ , com valores esperados  $E(X) = \bar{X}$ ,  $E(Y) = \bar{Y}$  é definida como uma medida de como as duas variáveis se modificam conjuntamente:

$$\text{cov}(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)} \quad (5)$$

- A covariância entre uma dimensão e ela mesma é a variância.
- O sinal da covariância é importante pois
  - Se positivo, indica que ambas as dimensões crescem ou decrescem juntas;
  - Se negativo, indica que se uma dimensão cresce a outra decresce;
  - Quando a covariância é zero, indica que as dimensões são descorrelacionadas (não necessariamente independentes).

# Conceitos de Estatística: Covariância

- A expressão geralmente usada para cálculo da covariância é:  
 $\text{cov}(X, Y) = E(XY) - E(X) \cdot E(Y) = E(XY) - \bar{X} \cdot \bar{Y}$
- Se  $X$  e  $Y$  são independentes, então a sua covariância é zero.
- Se  $X$  e  $Y$  são variáveis aleatórias de valor real e  $a, b, c$  e  $d$  constantes, então:

$$\text{cov}(X, Y) = \text{cov}(Y, X); \text{cov}(aX + b, cY + d) = ac \text{cov}(Y, X)$$

$$\text{cov}\left(\sum_i X_i, \sum_j Y_j\right) = \sum_i \sum_j \text{cov}(X_i, Y_j)$$

# Conceitos de Estatística: Matriz de Covariância

- **Matriz de covariância** é uma matriz simétrica que apresenta a covariância entre  $n$  variáveis.

$$M_{\text{cov}} = E[(\mathbf{X} - \bar{\mathbf{X}})(\mathbf{X} - \bar{\mathbf{X}})^T] \therefore$$
$$M_{\text{cov}} = \begin{bmatrix} E[(X_1 - \bar{X}_1)(X_1 - \bar{X}_1)] & E[(X_1 - \bar{X}_1)(X_2 - \bar{X}_2)] & \dots & E[(X_1 - \bar{X}_1)(X_n - \bar{X}_n)] \\ E[(X_2 - \bar{X}_2)(X_1 - \bar{X}_1)] & E[(X_2 - \bar{X}_2)(X_2 - \bar{X}_2)] & \dots & E[(X_2 - \bar{X}_2)(X_n - \bar{X}_n)] \\ \vdots & \vdots & \vdots & \vdots \\ E[(X_n - \bar{X}_n)(X_1 - \bar{X}_1)] & E[(X_n - \bar{X}_n)(X_2 - \bar{X}_2)] & \dots & E[(X_n - \bar{X}_n)(X_n - \bar{X}_n)] \end{bmatrix} \quad (6)$$

- onde,  $M_{\text{cov}}$  é a matriz com  $n$  linhas e  $n$  colunas.
- Assim, a célula da linha 2 e coluna 3, apresenta a covariância entre o 2º e 3º elementos.

# Conceitos de Álgebra Matricial

- Seja  $T : V \rightarrow V$  um operador linear. Um vetor  $\mathbf{v} \in V$ ,  $\mathbf{v} \neq \mathbf{0}_V$ , é dito ser **autovetor**, vetor próprio ou vetor característico do operador  $T$ , se existir  $\lambda \in R$  tal que  $T(\mathbf{v}) = \lambda \cdot \mathbf{v}$ .
- A escalar  $\lambda$  é chamada de **autovalor**, valor próprio ou valor característico do operador linear  $T$  associado ao autovetor  $\mathbf{v}$ .
- Por exemplo:  $T : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ ,  $\begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow \begin{bmatrix} x + y + z \\ 2y + z \\ 2y + 3z \end{bmatrix}$ , então  
 $T(1,1,2) = \begin{bmatrix} 4 \\ 4 \\ 8 \end{bmatrix} = 4 \cdot \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} = \lambda \mathbf{v}$ , autovetor  $\mathbf{v}$  associado a um autovalor  $\lambda$ .

# Conceitos de Álgebra Matricial

- Autovetores de uma transformação são vetores cuja direção não é alterada por essa transformação.
- Seja  $\mathbf{A}$  uma matriz que representa uma transformação linear entre dois vetores.
- Se existir um vetor  $\mathbf{v} \in R^n \neq \mathbf{0}_v$  tal que

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{I}\mathbf{v}, \quad \text{onde } \mathbf{v} = [v_1 \ v_2 \ \dots v_n]^T, \quad \mathbf{I}: \text{matriz diagonal unitária}$$

então  $\lambda_i$  é uma grandeza escalar chamada de autovalor (*eigenvalue*) de  $\mathbf{A}$  cujo autovetor (*eigenvector*) associado é  $\mathbf{v}_i$ .

# Conceitos de Álgebra Matricial

- Propriedades dos autovetores:
  - Só há autovetores para matrizes quadradas;
    - Nem toda matriz quadrada os apresenta.
  - Uma matriz  $n \times n$  tem no máximo  $n$  autovalores;
  - Se um autovetor for multiplicado por uma escalar, ele ainda será um autovetor relacionado ao mesmo autovalor;

$$2 \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 6 \\ 4 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 6 \\ 4 \end{pmatrix} = \begin{pmatrix} 24 \\ 16 \end{pmatrix} = 4 \times \begin{pmatrix} 6 \\ 4 \end{pmatrix}$$

- Autovetores são linearmente independentes entre si formando base de um espaço. Autovetores de matrizes simétricas são ortogonais.

# Conceitos de Álgebra Matricial

- Essa última característica torna os autovetores de uma matriz interessante para representá-la em seu espaço, isto é, os autovetores são usados como os eixos do espaço de  $\mathbf{A}$ .
- Nesse caso para facilitar os cálculos usa-se autovetores unitários (magnitude igual a 1) correspondentes aos autovalores.
- Por fim, dá-se o nome de *autoespaço* (*eigenspace*) aos autovetores que estão relacionados ao mesmo autovalor. O vetor nulo faz parte desse espaço, apesar de não ser um autovetor.



# Conceitos de Álgebra Matricial

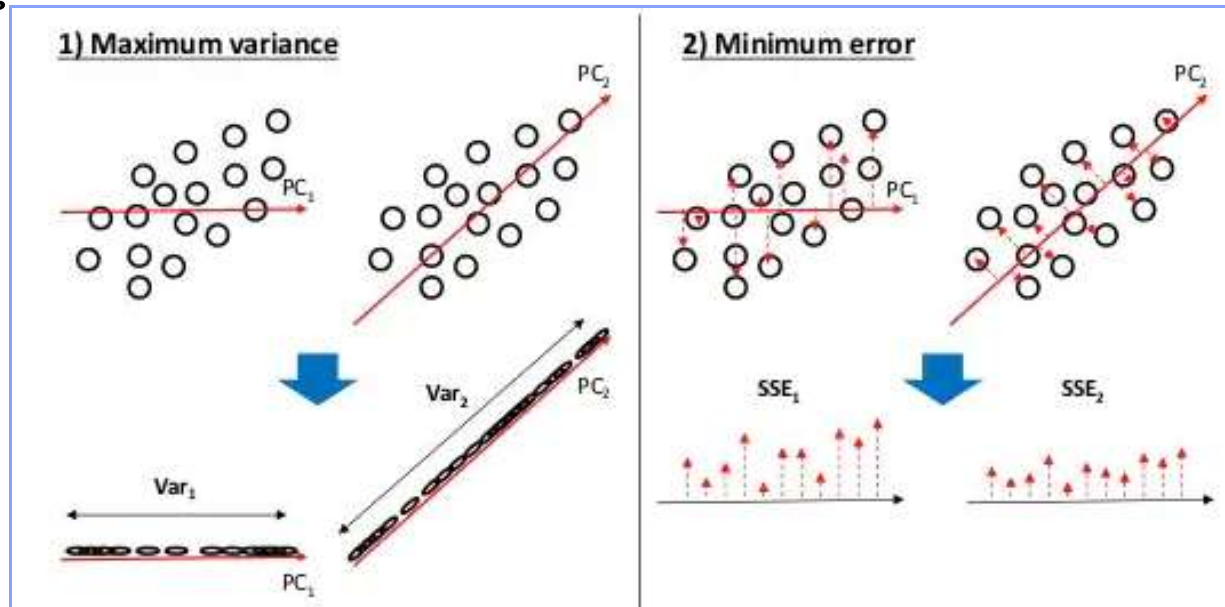
- Existem vários processos usados para encontrar os autovalores e autovetores de uma matriz quadrada:
  - Resolução direta do sistema linear;
  - Emprego do polinômio característico;
  - Diagonalização de matriz ou “*eigendecomposition*”;
  - Algoritmos iterativos.
- Se  $\mathbf{A}$  for a matriz canônica que representa o operador linear  $T$ :
  - Cálculo de autovalores de  $\mathbf{A}$ :  $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$ ;
  - Cálculo de autovetores de  $\mathbf{A}$ : para cada  $\lambda$ , são as soluções da equação  $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$  ou  $(\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = 0$ .

# Análise de Componentes Principais - PCA

- PCA é uma transformação linear dos dados para um novo sistema de coordenadas, onde a maior variância de todas as projeções dos dados, será posicionada como primeira coordenada (chamada componente principal), a segunda maior variância será a segunda coordenada e assim por diante.
  - PCA é a transformação linear ótima que visa determinar o subespaço que tem a maior variância.
- No PCA é empregada a Transformada (discreta) de Karhunen-Loève - KLT ou Transformada de Hotelling:
  - Toma uma coleção de dados de entrada e cria uma base ortogonal (a base KLT) para estes dados.

# Análise de Componentes Principais - PCA

- Princípio do PCA:
  - Maximização da variância dos dados projetados sobre uma dada dimensão;
  - Minimização do erro médio quadrado entre os dados e suas projeções.



# Análise de Componentes Principais - PCA

- Existem diferentes métodos para se realizar PCA:
  - Estatísticos e Algébricos
    - Método da Covariância
    - Método da Correlação
  - Mapas Auto-organizáveis
    - Algoritmo Hebbiano Generalizado – AHG
    - Adaptive Principal Components Extraction – APEX
  - Kernel PCA

# PCA- Método de Covariância

- Descrição:
  - O objetivo desse método é transformar um conjunto de dados  $\mathbf{x}$  de dimensão qualquer em um conjunto alternativo de dados  $\mathbf{y}$  de dimensão menor que o anterior.
  - Portanto, procura-se a matriz  $\mathbf{Y}$ , onde  $\mathbf{Y}$  é a transformada de Karhunen-Loève (KLT) da matriz  $\mathbf{X}$ :

$$\mathbf{Y} = \text{KLT}\{\mathbf{X}\}$$

# PCA- Método de Covariância

## Organização dos dados

- Suponha que se tem um conjunto de dados correspondendo a um conjunto de observações de  $M$  variáveis.
    - E que se queria reduzir os dados para  $L$  variáveis,  $L < M$ .
  - Suponha, também, que os dados estejam arrumados em um conjunto de  $N$  vetores de dados  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ .
    - Cada vetor representa uma única observação de  $M$  variáveis.
1. Escreva  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$  como vetores dispostos em colunas, cada um com  $M$  linhas.
  2. Arrume os vetores dispostos em colunas em uma única matriz  $\mathbf{X}$  de dimensão  $M \times N$ .

# PCA- Método de Covariância

## Cálculo da média empírica

3. Ache a média empírica para cada componente da matriz  $\mathbf{X}$ , para  $m = 1, \dots, M$ .
4. Coloque as médias calculadas em um vetor de médias  $\mathbf{u}$  de dimensão  $M \times 1$ .

$$u_m = \frac{\sum_{n=1}^N x_{mn}}{N}$$

# PCA- Método de Covariância

## Cálculo dos desvios a partir da média

5. Subtraia o vetor de médias  $\mathbf{u}$  de cada coluna da matriz  $\mathbf{X}$ .
6. Armazene os dados com a média subtraída na matriz  $\mathbf{B}$ ,  $M \times N$ .

$$\mathbf{B} = \mathbf{X} - \mathbf{u} \cdot \mathbf{h}^T$$

onde  $\mathbf{h}$  é um vetor  $N \times 1$ , formado apenas de 1s:

$$\mathbf{h} = [1 \ 1 \ 1 \ \dots \ 1 \ 1]^T$$



# PCA- Método de Covariância

## Determinação da matriz de covariância

7. Ache a matriz de covariância empírica  $\mathbf{C}$ ,  $M \times M$ , a partir do produto externo da matriz  $\mathbf{B}$  com ela mesma:

$$\mathbf{C} = E[\mathbf{B} \otimes \mathbf{B}] = E[\mathbf{B}\mathbf{B}^T] = \frac{1}{N-1} \mathbf{B}\mathbf{B}^T$$

onde

$E$  é o valor esperado;

$\otimes$  é o produto externo;

$^T$  denota uma matriz transposta.

# PCA- Método de Covariância

## Determinação dos autovetores e autovalores da matriz de covariância

8. Compute a matriz **D** de autovalores e a matriz **V** de autovetores a partir da matriz de covariância **C**:

$$\mathbf{C} \cdot \mathbf{V} = \mathbf{D} \cdot \mathbf{V}$$

9. A matriz **D** será uma matriz diagonal  $M \times M$ , onde

$$D_{pq} = \begin{cases} \lambda_m, & p = q = m; \\ 0, & p \neq q \end{cases}$$

onde  $\lambda_m$  é o *m-ésimo* autovalor da matriz de covariância **C**.

# PCA- Método de Covariância

## Determinação dos autovetores e autovalores da matriz de covariância

10. A matriz  $V$ , de dimensão  $M \times M$ , contém  $M$  vetores dispostos em colunas, cada um com dimensão  $M$ , que representam os  $M$  autovetores da matriz de covariância  $C$ .
11. Os autovalores e autovetores estão ordenados e pareados. O *m-ésimo* autovalor corresponde ao *m-ésimo* autovetor.

# PCA- Método de Covariância

## Reordenação dos autovetores e autovalores

12. Arrume as colunas da matriz de autovetores  $\mathbf{V}$  e a matriz de autovalores  $\mathbf{D}$  em ordem decrescente de autovalor.
13. Mantenha a mesma relação entre as colunas das duas matrizes.

# PCA- Método de Covariância

## Cálculo do índice de energia acumulada para cada autovetor

14. Os autovalores representam a distribuição de energia do conjunto original de dados em relação a cada autovetor, componentes da base dos dados.
15. O índice de energia acumulada  $g$  para cada  $m$ -ésimo autovetor é a soma do índice de energia de todos os autovetores de 1 até  $m$ :

$$g_m = \sum_{q=1}^m D_{pq}, \quad \text{para } p = q \text{ e } m = 1, \dots, M.$$

# PCA- Método de Covariância

**Seleção de subconjunto dos autovetores como vetores base**

16. Salve as primeiras  $L$  colunas de  $V$  como a matriz,  $M \times L$ ,  $W$ :

$$W_{pq} = V_{pq}, \quad \text{para } p = 1, \dots, M; \quad q = 1, \dots, L; \quad \text{e } 1 \leq L \leq M.$$

17. Use o vetor  $g$  como um guia de escolha do valor apropriado para  $L$ . A idéia é escolher um valor para  $L$  tão pequeno quanto possível, desde que se tenha um valor percentual alto para  $g$ .

Por exemplo,  $g[m = L] \geq 90\%$

# PCA- Método de Covariância

## Conversão dos dados originais para z-scores

18. Crie um vetor do desvio padrão empírico,  $M \times 1$ ,  $\mathbf{s}$  a partir da raiz quadrada de cada elemento ao longo da diagonal principal da matriz de covariância.

$$\mathbf{s} = \left[ \sqrt{C_{pq}} \right] \quad p = q = m = 1, \dots, M$$

19. Calcule a matriz,  $M \times N$ ,  $z$ -scores:

$$\mathbf{Z} = \frac{\mathbf{B}}{\mathbf{s} \cdot \mathbf{h}}$$

Divide-se elemento por elemento.

# PCA- Método de Covariância

## Projeção dos z-scores na nova base

20. Os vetores projetados são as colunas da matriz

$$\mathbf{Y} = \mathbf{W}^T \cdot \mathbf{Z} = \text{KLT}\{\mathbf{X}\}$$

21. A coluna da matriz  $\mathbf{Y}$  representam a transformação de Karhunen-Loeve (KLT) dos vetores de dados das colunas da matriz  $\mathbf{X}$ .

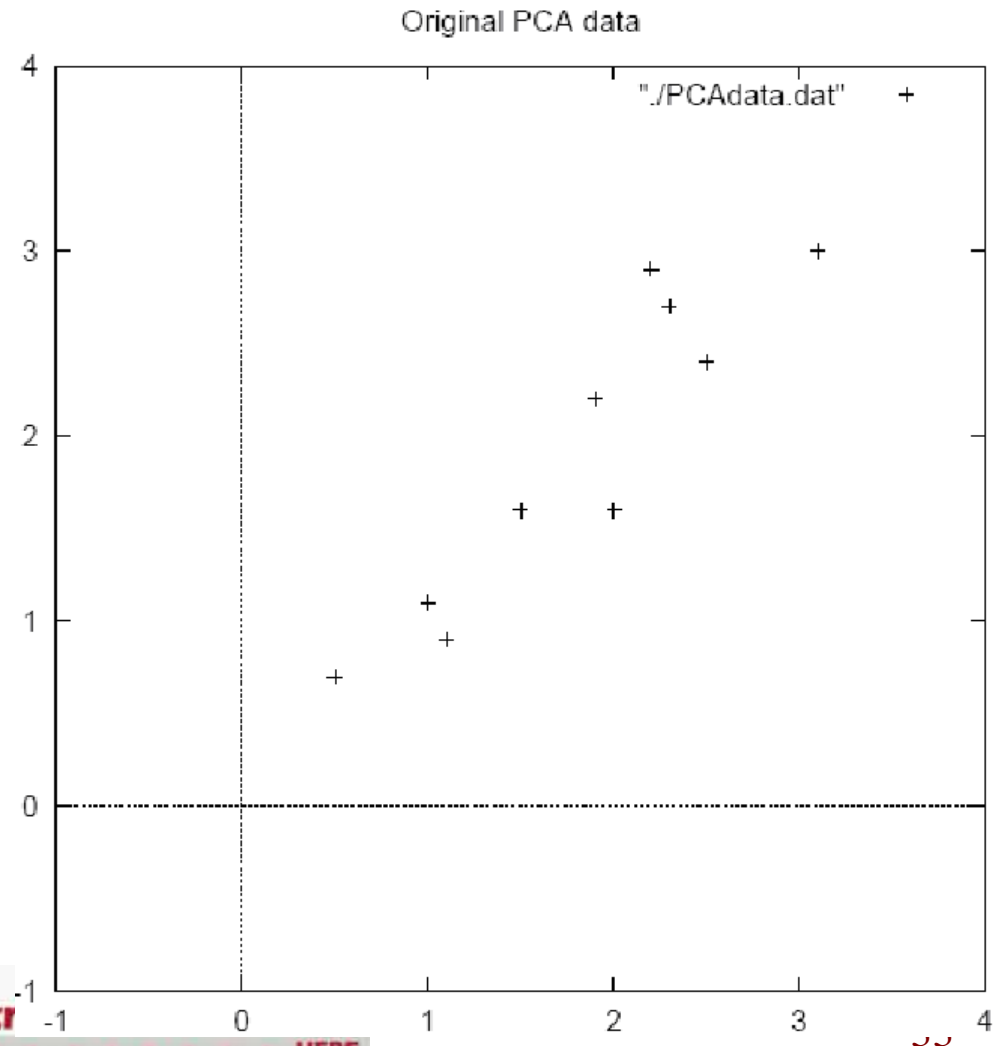


# PCA- Método de Covariância

## Exemplo

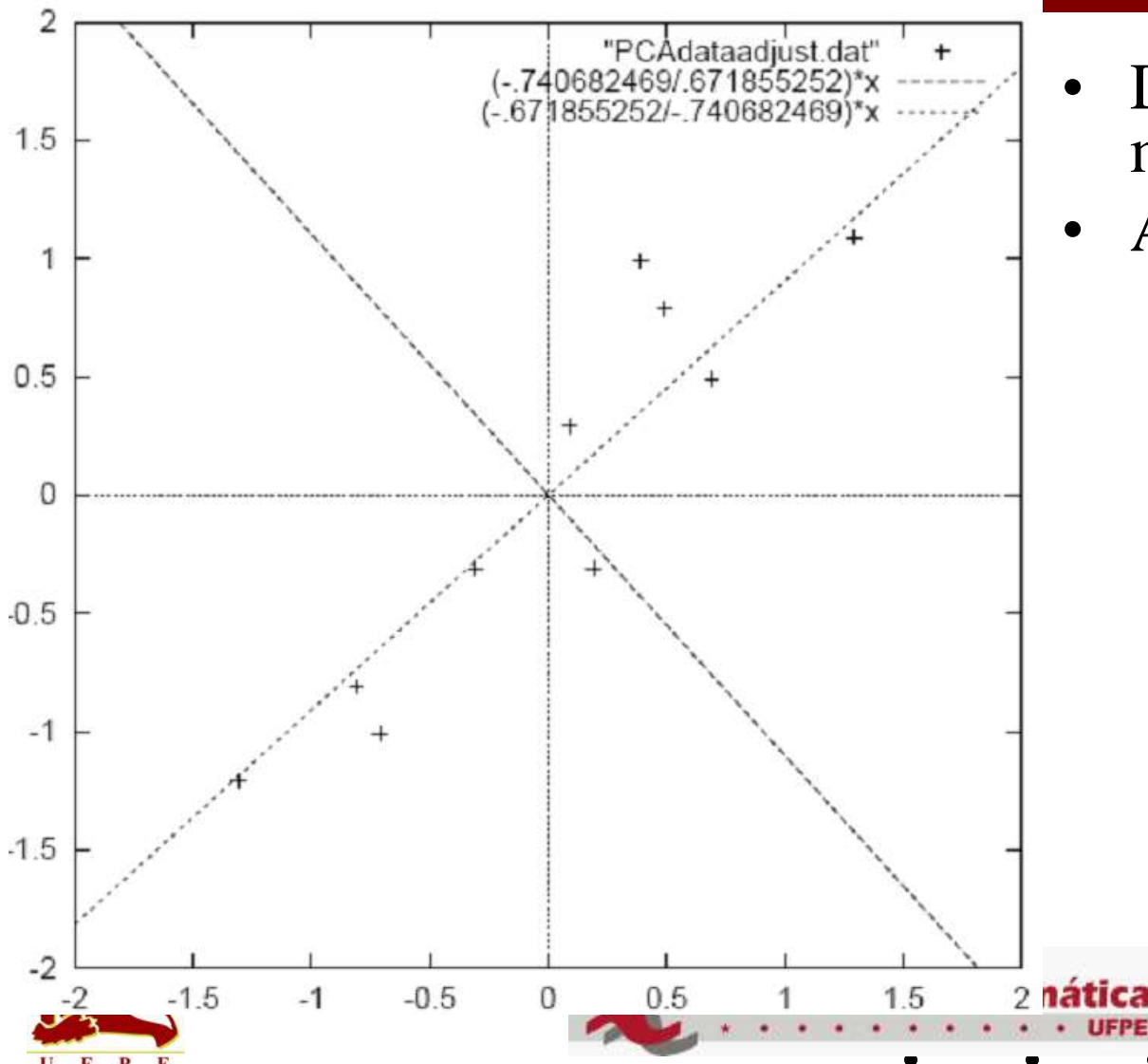
Dados em 2 dimensões:

$x$	$y$
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9



# PCA- Método de Covariância

## Exemplo

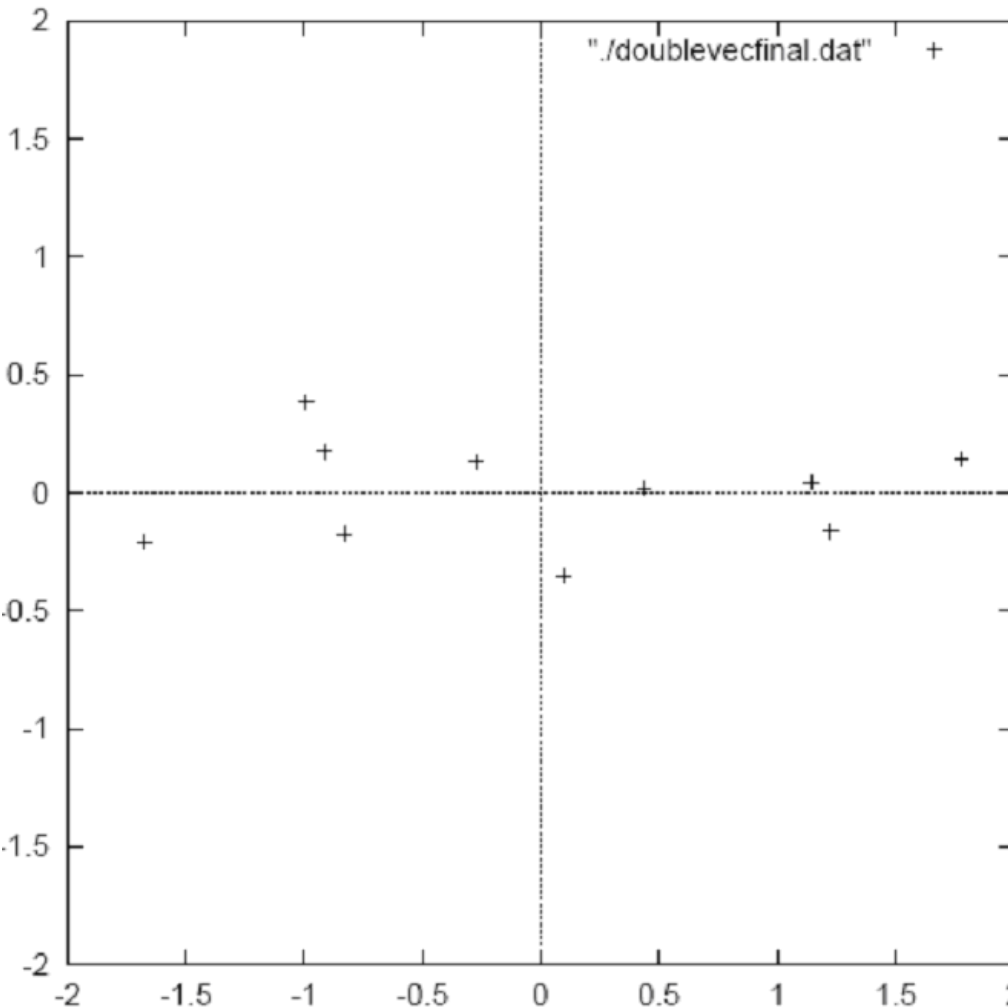


- Dados após remoção da média estão plotados.
- Autovetores computados:
  - Diagonais na imagem.

# PCA- Método de Covariância

## Exemplo

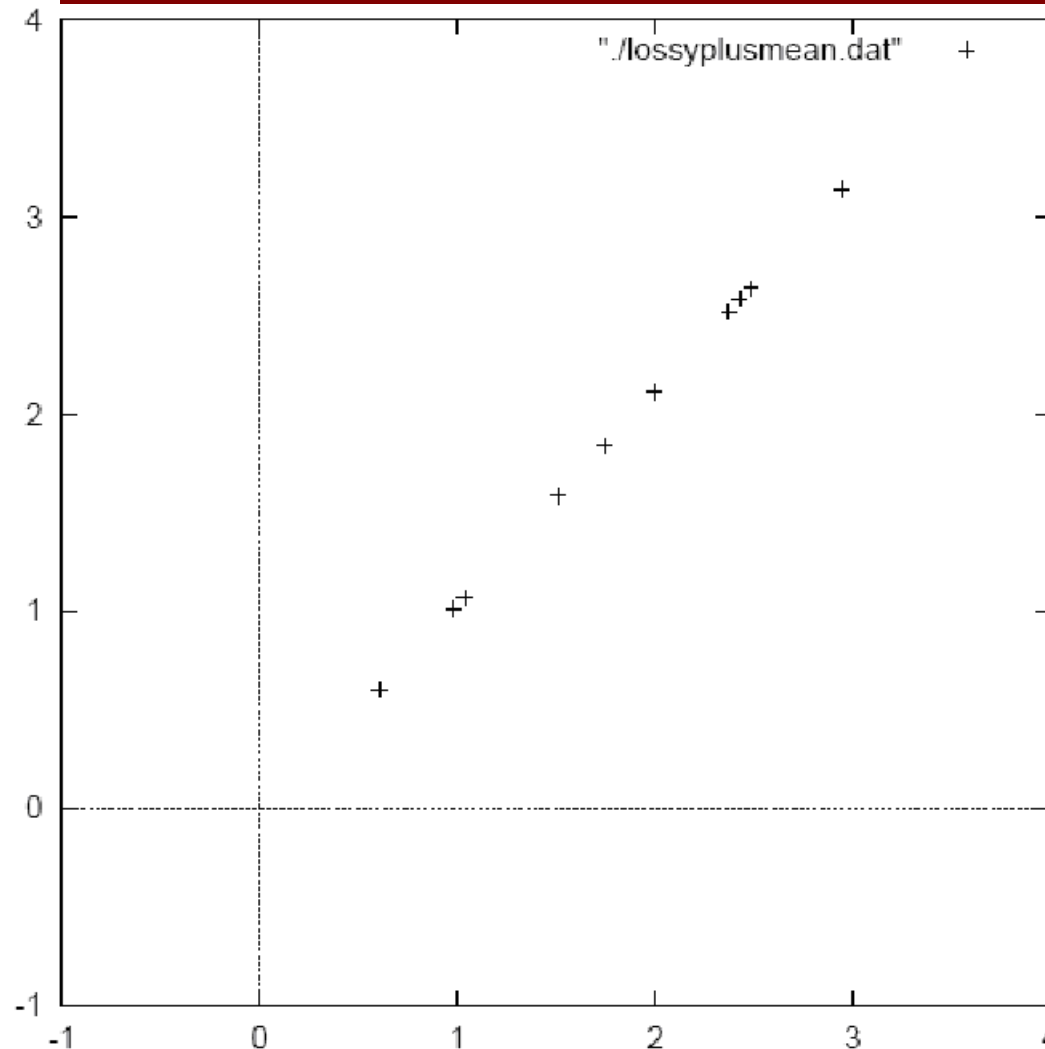
Data transformed with 2 eigenvectors



- Dados transformados para as bases dos autovetores calculados.

# PCA- Método de Covariância

## Exemplo



- Dados após cálculo do PCA usando um dos dois autovetores transformados de volta para a base original.
- Houve redução de dimensionalidade dos dados para apenas 1 dimensão.

# PCA

## Aprendizagem Auto-organizável

- Auto-organização de um sistema ocorre sem pressão explícita ou imposição do mundo exterior ao sistema,
  - As relações estruturais importantes são internas ao sistema, decorrem de interações entre seus componentes independentemente de sua natureza física.
- A organização pode se modificar no tempo ou no espaço, tendo comportamento estável ou um transitório,
  - A auto-organização busca regras gerais de crescimento e evolução de estruturas de sistemas, sua forma e predição de organização futura devido a mudanças em seus componentes;
  - Resultados em um sistema devem ser aplicáveis a outros com características similares de rede.

# PCA

## Aprendizagem Auto-organizável

- Quatro princípios são fundamentais para a auto-organização, segundo von der Malsburg (Haykin, 2001):
  1. Modificações nos pesos sinápticos tendem a se auto-amplificar;
  2. A limitação de recursos leva as sinapses a competirem entre si, causando fortalecimento de umas em detrimento de outras;
  3. As modificações em pesos sinápticos tendem a cooperar entre si;
  4. Os padrões de entrada apresentam informações redundantes que são assimiladas pela rede neural na forma de conhecimento.

# PCA

## Aprendizagem Auto-organizável

- A auto-organização de uma rede neural acontece em dois níveis que interagem entre si:
  - *Atividade*: Padrões de atividade são gerados pela rede em resposta a sinais de entrada;
  - *Conectividade*: Pesos sinápticos da rede são alterados em resposta aos padrões de atividades;
- Mapas auto-organizáveis (SOMs) podem se comportar analogamente ao método estatístico PCA,
  - SOMs tendem a encontrar o subespaço PCA ao invés das componentes principais exatas;

# PCA

## Aprendizagem Auto-organizável

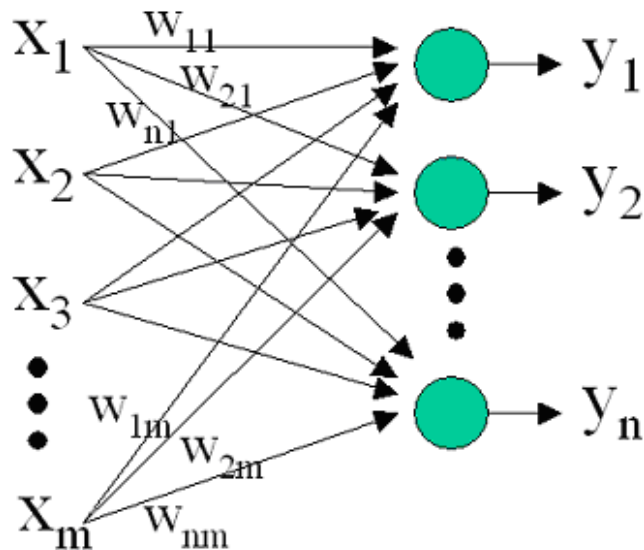
- Sobre o emprego de SOMs para realizar PCA:
  - Processos são não estacionários;
  - Existe correspondência entre o comportamento de redes neurais auto-organizadas e o método estatístico do PCA;
  - Uma única unidade linear de processamento com uma regra de adaptação do tipo Hebbiano para seus pesos sinápticos pode se adaptar para extrair as componentes de maneira incremental através do ajustes dos pesos.



# PCA

## Algoritmo Hebbiano Generalizado - GHA

- Solução para o PCA, empregando redes neurais, que pertence à classe dos algoritmos de estimação das componentes principais;
- Seja uma rede alimentada para frente e com aprendizado Hebbiano, possuindo os seguintes parâmetros estruturais:



1. Cada nodo na camada de saída da rede é linear;
2. A rede tem  $m$  entradas e  $n$  saídas, sendo ambas pré-especificadas. A rede apresenta menos saídas que entradas,  $n < m$ .

# PCA

## Algoritmo Hebbiano Generalizado - GHA

- Os nodos de saída devem determinar as componentes principais se consideradas as duas hipóteses mencionadas e um tipo particular de aprendizagem hebbiana.
- O aprendizado consiste na adaptação do conjunto de pesos sinápticos,  $\{w_{ji}\}$ , propagando os valores de entrada até os nodos da camada de saída, onde  $i=1, 2, \dots, m$  e  $j=1, 2, \dots, n$ .
- A função de saída,  $y_j(k)$ , de cada nodo de saída,  $j$ , no tempo,  $k$ , produz uma resposta a partir dos valores de entrada  $\{x_i(k)/i=1, 2, \dots, m\}$ , e caracteriza-se por:

$$y_j(k) = \sum_{i=1}^m w_{ji}(k) x_i(k) \quad , \quad j = 1, 2, \dots, n$$

# PCA

## Algoritmo Hebbiano Generalizado - GHA

- O peso sináptico  $w_{ji}$  é adaptado via regra generalizada de aprendizagem hebbiana, conforme equação abaixo:

$$\Delta w_{ji}(k) = \eta \left[ y_j(k)x_i(k) - y_j(k) \sum_{l=1}^j w_{li}(k)y_l(k) \right], \quad \begin{matrix} i = 1, 2, \dots, m \\ j = 1, 2, \dots, n \end{matrix}$$

- Esta modificação aplicada ao peso sináptico  $w_{ji}(k)$  no tempo  $k$ , tem  $\eta$  como sua taxa de aprendizagem.

- Para melhor entendimento do GHA:

$$\therefore \Delta w_{ji}(k) = \eta y_j(k) \left[ x_i(k) - \sum_{l=1}^{j-1} (w_{li}(k)y_l(k)) - w_{ji}(k)y_j(k) \right]$$

$$\therefore \Delta w_{ji}(k) = \eta y_j(k) [x'_i(k) - w_{ji}(k)y_j(k)]$$

# PCA

## Algoritmo Hebbiano Generalizado - GHA

O termo  $y_j(k)x'_i(k)$  corresponde à auto-amplificação decorrente do princípio 1 da auto-organização.

O termo negativo,  $-w_{ji}(k)y_j(k)$ , é responsável pela estabilização decorrente do princípio 2, competição entre sinapses dos neurônios.

- Existem duas formas de alimentação que atuam no neurônio:
  - Alimentação positiva para auto-amplificação, i.e., causa crescimento de peso sináptico devido à entrada externa;
  - Alimentação negativa devido ao termo  $-y_j(k)$  que controla o crescimento, i.e., causa estabilização do peso sináptico.

# PCA

## Algoritmo Hebbiano Generalizado - GHA

- Cálculo do vetor  $\mathbf{x}'(k)$  para a atualização dos pesos:

$$\Delta w_{ji}(k) = \eta y_j(k) [x'_i(k) - w_{ji}(k) y_j(k)], \text{ logo tem-se que:}$$

$$j = 1, \quad \mathbf{x}'(k) = \mathbf{x}(k)$$

$$j = 2, \quad \mathbf{x}'(k) = \mathbf{x}(k) - \mathbf{w}_1(k) \mathbf{y}_1(k)$$

$$j = 3, \quad \mathbf{x}'(k) = \mathbf{x}(k) - \mathbf{w}_1(k) \mathbf{y}_1(k) - \mathbf{w}_2(k) \mathbf{y}_2(k)$$

...

$$j = i, \quad \mathbf{x}'(k) = \mathbf{x}(k) - \mathbf{w}_1(k) \mathbf{y}_1(k) - \mathbf{w}_2(k) \mathbf{y}_2(k) - \dots - \mathbf{w}_{i-1}(k) \mathbf{y}_{i-1}(k)$$

# PCA

## GHA - Algoritmo

- Os cálculos do GHA estão em 3 passos dados a seguir:
  - Passo 1: Inicialize aleatoriamente os pesos sinápticos da rede,  $w_{ji}$ , com valores pequenos, e valor positivo pequeno à taxa de aprendizagem  $\eta$ .
  - Passo 2: Para  $k=1$ ,  $i=1, 2, \dots, m$  e  $j=1, 2, \dots, n$ , calcule os valores de saída de cada nodo,  $y_j(k)$ , e os valores de ajuste dos pesos,  $\Delta w_{ji}(k)$ , usando as equações anteriores.
  - Passo 3: Incremente  $k$  em 1, vá para o passo 2 e continue até que os pesos sinápticos,  $w_{ji}$ , alcancem seus valores de equilíbrio. Para  $k$  grande, o peso sináptico  $w_{ji}$  do neurônio  $j$  converge para a  $i$ -ésima componente do autovetor associado com o  $j$ -ésimo autovalor da matriz de correlação do vetor de entrada  $\mathbf{x}(t)$ .
- Os pesos dos nodos são as componentes de cada autovetor associado a seu autovalor relacionado com a saída  $y_j$  do nodo.

# PCA

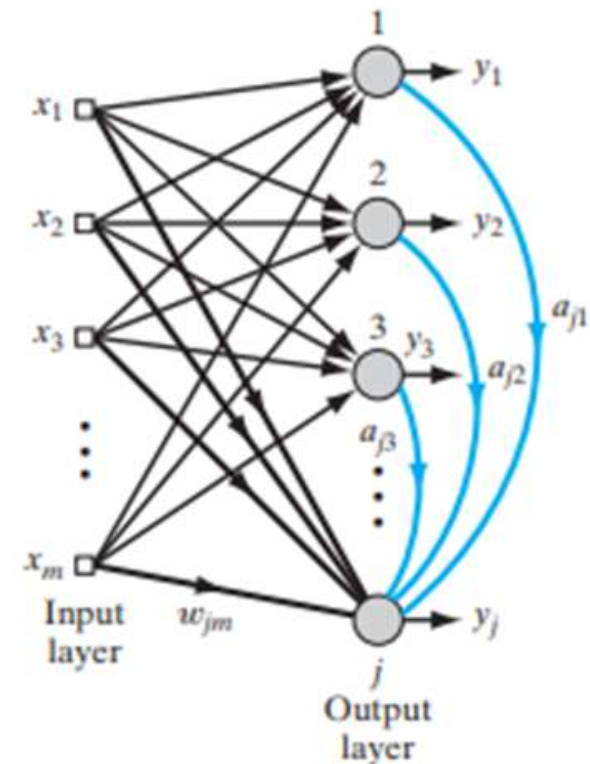
## Extração de Componentes Principais Adaptativa - APEX

- O Algoritmo Extração de Componentes Principais Adaptativa (Adaptive Principal Components Extraction - APEX) emprega rede neural com com alimentação para frente e realimentação. APEX pertence à classe dos algoritmos de descorrelação dos métodos de PCA.
- Este é um algoritmo iterativo que determina o  $j$ -ésimo componente principal com base nos  $j-1$  componentes anteriores;

# PCA

## Extração de Componentes Principais Adaptativa - APEX

- Sua estrutura é definida como:
  - Há  $m$  nodos de entrada que propagam o sinal;
  - Todos  $j$  nodos na camada de saída são lineares;
  - A rede tem conexões excitatórias entre cada nodo de entrada para todos os nodos de saída, com  $m > j$ . Estas conexões usam aprendizagem Hebbiana e realizam auto-amplificação. As conexões excitatórias são denotadas por  $w(k)$ ;
  - A rede tem conexões laterais de realimentação. Assim, o nodo  $j$  recebe conexões vindas dos nodos 1 até  $j-1$ . Estas conexões são inibitórias e usam aprendizagem anti-Hebbiana. As conexões inibitórias são denotadas por  $a(k)$ .

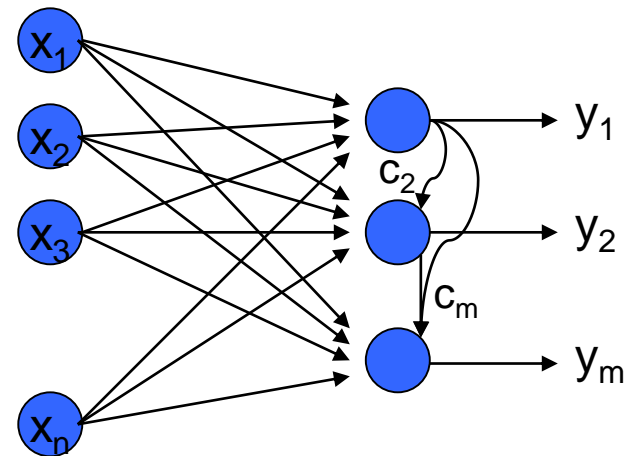




# PCA

## Extração de Componentes Principais Adaptativa - APEX

- Considerando a arquitetura



$$\mathbf{y} = \mathbf{W}\mathbf{x} - \mathbf{C}\mathbf{y},$$

onde

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ c_{21} & 0 & 0 & \dots & 0 \\ c_{31} & c_{32} & 0 & \dots & 0 \\ \vdots & & \ddots & & \vdots \\ c_{m-1,1} & \dots & c_{m-1,m-2} & 0 & 0 \\ c_{m,1} & \dots & \dots & c_{m,m-1} & 0 \end{bmatrix}$$

# PCA

## Extração de Componentes Principais Adaptativa - APEX

- Algoritmo APEX:
  - Passo 1: Inicialize os pesos  $w_{ji}$ , e  $a_{jl}$  com valores pequenos, e atribua valor positivo pequeno à taxa de aprendizado  $\eta$ .
  - Passo 2: Para  $j=1$ ,  $k=1, 2, \dots$  e  $i=1, 2, \dots, m$ , calcule as saídas de cada nodo,  $y_1(k)$ , e os ajustes dos pesos,  $\Delta w_{1i}(k)$ :

$$y_1(k) = \mathbf{w}_1^T(k) \mathbf{x}(k)$$

$$\Delta \mathbf{w}_1(k+1) = \eta [y_1(k) \mathbf{x}(k) - y_1^2(k) \mathbf{w}_1(k)]$$

onde para valores altos de  $k$ , o vetor de pesos  $\mathbf{w}_1$  tende para o autovetor  $\mathbf{v}_1$  associado ao maior autovalor  $\lambda_1$ .

# PCA

## Extração de Componentes Principais Adaptativa - APEX

- Algoritmo APEX (continuação):
  - Passo 3: Para  $j=2, k=1, 2, \dots$  e  $i=1, 2, \dots, m$ , calcule:
$$\mathbf{y}_{j-1}(k) = [y_1(k) \ y_2(k) \ \dots \ y_{j-1}(k)]$$
$$y_j(k) = \mathbf{w}_j^T(k)\mathbf{x}(k) + \mathbf{a}_j^T(k)\mathbf{y}_{j-1}(k)$$
$$\Delta \mathbf{w}_j(k+1) = \eta [y_j(k)\mathbf{x}(k) - y_j^2(k)\mathbf{w}_j(k)]$$
$$\Delta \mathbf{a}_j(k+1) = \eta [y_j(k)\mathbf{y}_{j-1}(k) - y_j^2(k)\mathbf{a}_j(k)]$$
  - Passo 4: Incremente  $j$  de 1 e vá para o passo 3. Repita o processo até atingir o número pré-determinado de componentes principais. Para  $k$  grande, os vetores de pesos excitatórios tendem ao autovetores e os inibitórias tendem a zero.

# PCA

## Kernel PCA

- PCA linear:
  - Projeta ortogonalmente as amostras em um espaço linear de dimensão mais baixa;
  - Obtém a máxima variância dos dados projetados;
  - Apresenta limitações decorrentes da linearidade;
- KPCA é um tipo de PCA não-linear que se caracteriza por:
  - Levar pontos no espaço de entrada para o espaço de características através de uma transformação não-linear  $\phi(\mathbf{x})$ , função de kernel;
  - Analisar ponto no espaço de características com PCA linear;

# PCA

## Kernel PCA

- Funções de kernel:
  - Mapeamento de  $x$  ao espaço de características (produto interno)

$$k(x, x') = \phi(x)^T \phi(x')$$

$$x = x_1, x_2, \dots, x_n$$

$$\phi(x) = y_1, y_2, \dots, y_m$$

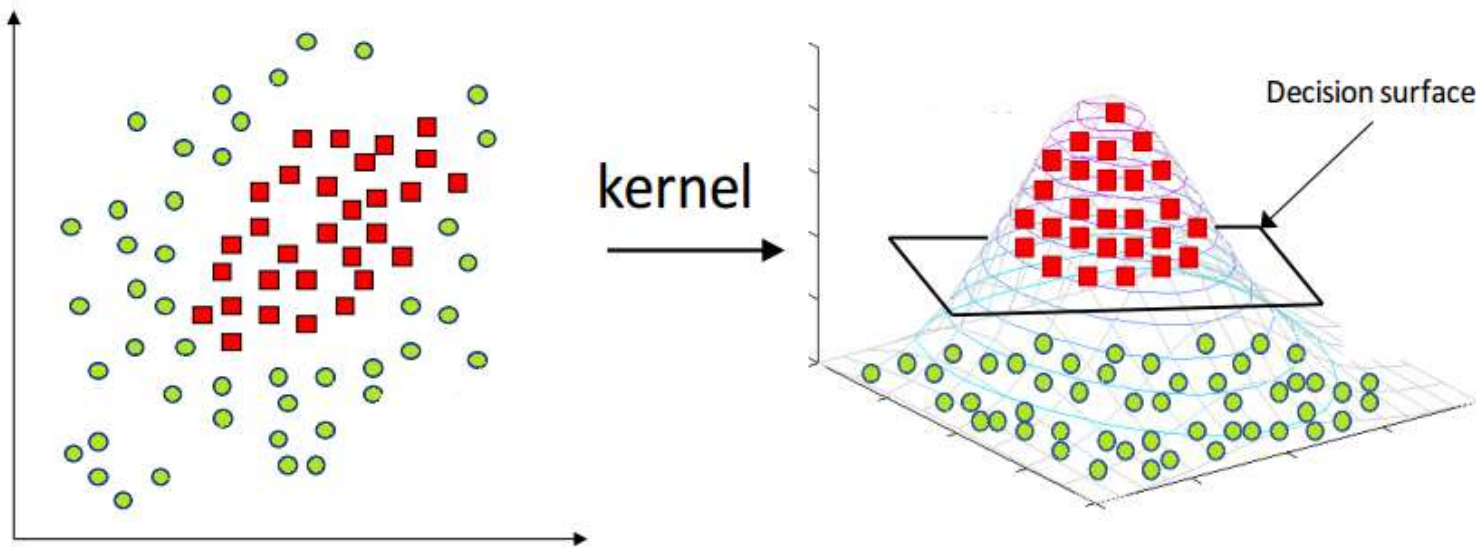
$$k(x, x') = y_1 y'_1 + y_2 y'_2 + \dots + y_m y'_m$$

- Dimensões: espaço de características  $M \geq N$  espaço de entrada;

# PCA

## Kernel PCA

- Funções de kernel:



- Seja um conjunto de dados  $\{\mathbf{x}_1, \dots, \mathbf{x}_L\} \in \mathbb{R}^N$ , KPCA consiste em um PCA que transporta os dados para um outro espaço de dimensão maior empregando uma função kernel.

# PCA

## Kernel PCA

- Algoritmo KPCA:
  - Escolha uma função de kernel;
  - Construa a matriz normalizada para o conjunto de padrões de treinamento  $\{\mathbf{x}_i\}_{i=1\dots M}$ ,  $\mathbf{x} \in \mathcal{R}^N$ ,
    - Matriz de kernel ( $N \times N$ ) definida por  $\mathbf{K} = \{K(\mathbf{x}_i, \mathbf{x}_j)\}$ , onde  $K(\mathbf{x}_i, \mathbf{x}_j) = \phi^T(\mathbf{x}_i) \phi(\mathbf{x}_j)$ ;
  - Determine os autovalores e autovetores do sistema:  $\mathbf{K}\mathbf{v} = \lambda \mathbf{v}$ ; onde  $\lambda$  é um autovalor da matriz  $\mathbf{K}$  e  $\mathbf{v}$  é seu autovetor associado;
  - Normalize os autovetores:  $\mathbf{v}_k^T \mathbf{v}_k = 1/\lambda_d$ ,  $k=1,2,\dots,d$ ; onde  $\lambda_d$  é o menor autovalor diferente de zero da matriz  $\mathbf{K}$ , colocando os autovalores em ordem decrescente;

# PCA

## Kernel PCA

- Algoritmo KPCA:
  - Para a extração dos componentes principais de qualquer ponto de teste  $\mathbf{x}$ , calcule as projeções:  $\mathbf{v}_j = \sum_{i=1}^N v_{ji} \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j)$ ,  $j=1, \dots, d$ ;  
onde  $v_{ji}$  é o  $i$ -ésimo elemento do autovetor  $\mathbf{v}_j$ .



# PCA

## Exemplo: Compressão de Imagens

- Imagem de 372 x 492 pixels é dividida em janelas de 12x12 pixels, gerando vetor 144D:



# PCA

## Exemplo: Compressão de Imagens

144D:



66D:



16D:



6D:



# Referências

- Du, K.-L. & Swamy M. N. S. (2019). *Neural Networks and Statistical Learning*. Springer, 2nd edition.
- Haykin, S. (2009). *Neural Networks and Learning Machines*. Third Edition. Pearson.
- Gonzalez, R.C., e Woods, R.E. (2000). *Processamento de Imagens Digitais*, Editora Edgard Blücher, 2000.
- Smith, L. I. (2002). *A Tutorial on Principal Component Analysis*, [http://www.cs.otago.ac.nz/cosc453/student\\_tutorials/principal\\_components.pdf](http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf).
- Lateef , Z. (2019). All You Need To Know About Principal Component Analysis (PCA). <https://www.edureka.co/blog/principal-component-analysis/>