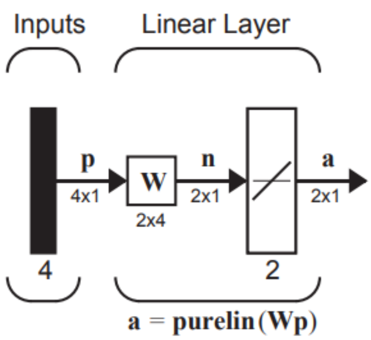
LICENCIATURA EM ENGENHARIA INFORMÁTICA
INTELIGÊNCIA COMPUTACIONAL
23/24

Aprendizagem Associativa e Generativa

Exercícios

Ex 01 – Aprendizagem de Hebb

Considere a seguinte rede linear:



Para o dataset de treino:

$$\left\{\mathbf{p}_1 = \begin{bmatrix}1\\-1\\1\\-1\end{bmatrix}, \mathbf{t}_1 = \begin{bmatrix}1\\-1\end{bmatrix}\right\} \quad \left\{\mathbf{p}_2 = \begin{bmatrix}1\\1\\-1\\-1\end{bmatrix}, \mathbf{t}_2 = \begin{bmatrix}1\\1\end{bmatrix}\right\}.$$

i) Determine os parâmetros pela regra de Hebb.
ii) Determine os parâmetros pela regra da pseudo-inversa.
iii) Teste i) para p1
iv) Teste ii) para p1

Resolução:

i)

$$\mathbf{P} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \\ 1 & -1 \\ -1 & -1 \end{bmatrix}, \qquad \mathbf{T} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}.$$

$$\mathbf{W}^h = \mathbf{T}\mathbf{P}^T = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & -2 \\ 0 & 2 & -2 & 0 \end{bmatrix}.$$

Teste da rede:

$$\mathbf{W}^h\mathbf{p}_1 = \begin{bmatrix} 2 & 0 & 0 & -2 \\ 0 & 2 & -2 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 4 \\ -4 \end{bmatrix} \neq \mathbf{t}_1$$

ii)

$$\mathbf{W} = \mathbf{T}\mathbf{P}^+$$

$$\mathbf{P}^+ = (\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T.$$

$$\mathbf{P}^+ = \left(\begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -1 & 1 \\ 1 & -1 \\ -1 & -1 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \end{bmatrix} = \left(\begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1}{4} & 0 \\ 0 & \frac{1}{4} \end{bmatrix} \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & -\frac{1}{4} & \frac{1}{4} & -\frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \end{bmatrix}$$

$$\mathbf{W}^p = \mathbf{T}\mathbf{P}^+ = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{4} & -\frac{1}{4} & \frac{1}{4} & -\frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & 0 & 0 & -\frac{1}{2} \\ 0 & \frac{1}{2} & -\frac{1}{2} & 0 \end{bmatrix}$$

Os padrões de treino são perpendiculares mas não normalizados.

Teste da rede:

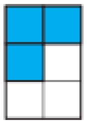
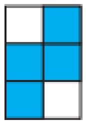
$$\mathbf{W}^p \mathbf{p}_1 = \begin{bmatrix} \frac{1}{2} & 0 & 0 & -\frac{1}{2} \\ 0 & \frac{1}{2} & -\frac{1}{2} & 0 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \mathbf{t}_1$$

OK dado que garante que minimiza:

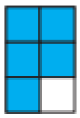
$$\sum_{q=1}^{2} \|\mathbf{t}_q - \mathbf{W}\mathbf{p}_q\|^2 ,$$

Ex 02

Considere os seguintes padrões:

$\mathbf{p}_1$ $\mathbf{p}_2$



$\mathbf{p}_t$

i) Verifique se são ortogonais entre si

$$\mathbf{p}_1 = \begin{bmatrix}1 & 1 & -1 & 1 & -1 & -1\end{bmatrix}^T \qquad \mathbf{p}_2 = \begin{bmatrix}-1 & 1 & 1 & 1 & 1 & -1\end{bmatrix}^T$$

$$\mathbf{p}_1^T\mathbf{p}_2 = \begin{bmatrix}1 & 1 & -1 & 1 & -1 & -1\end{bmatrix}\begin{bmatrix}-1\\1\\1\\1\\1\\-1\end{bmatrix} = 0$$

São perpendiculares embora não normalizados:

$$\mathbf{p}_1^T\mathbf{p}_1 = \mathbf{p}_2^T\mathbf{p}_2 = 6$$

ii) Desenhe uma rede auto-associativa. Determine os pesos pela regra de Hebb:

$$\mathbf{W} = \mathbf{T}\mathbf{P}^T,$$

$$\mathbf{P} = \mathbf{T} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \\ -1 & 1 \\ 1 & 1 \\ -1 & 1 \\ -1 & -1 \end{bmatrix}$$

$$\mathbf{W} = \mathbf{T}\mathbf{P}^T = \begin{bmatrix} 1 & -1 \\ 1 & 1 \\ -1 & 1 \\ 1 & 1 \\ -1 & 1 \\ -1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 & -1 & 1 & -1 & -1 \\ -1 & 1 & 1 & 1 & 1 & -1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & -2 & 0 & -2 & 0 \\ 0 & 2 & 0 & 2 & 0 & -2 \\ -2 & 0 & 2 & 0 & 2 & 0 \\ 0 & 2 & 0 & 2 & 0 & -2 \\ -2 & 0 & 2 & 0 & 2 & 0 \\ 0 & -2 & 0 & -2 & 0 & 2 \end{bmatrix}$$

iii) Calcule a resposta para o padrão de teste,

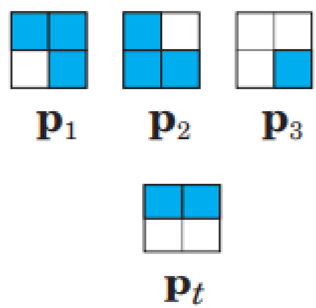
$$\mathbf{p}_t = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & -1 \end{bmatrix}^T$$

$$\mathbf{a} = \mathbf{hardlims}(\mathbf{W}\mathbf{p}_t) = \mathbf{hardlims}\left(\begin{bmatrix} 2 & 0 & -2 & 0 & -2 & 0 \\ 0 & 2 & 0 & 2 & 0 & -2 \\ -2 & 0 & 2 & 0 & 2 & 0 \\ 0 & 2 & 0 & 2 & 0 & -2 \\ -2 & 0 & 2 & 0 & 2 & 0 \\ 0 & -2 & 0 & -2 & 0 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \end{bmatrix} \right)$$

$$\mathbf{a} = \mathbf{hardlims}\left(\begin{bmatrix} -2 \\ 6 \\ 2 \\ 6 \\ 2 \\ -6 \end{bmatrix} \right) = \begin{bmatrix} -1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \end{bmatrix} = \mathbf{p}_2 \,.$$

Ex 03

Considere os padrões,

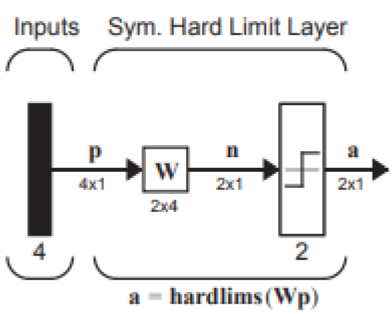


i) Calcule os pesos de uma rede perceptron para reconhecer os padrões p1,p2,p3

ii) Determine a resposta para pt

<u>Resolução:</u>

i)



$$\mathbf{p}_1 = \begin{bmatrix} 1 \\ -1 \\ 1 \\ 1 \end{bmatrix} \qquad \mathbf{p}_2 = \begin{bmatrix} 1 \\ 1 \\ -1 \\ 1 \end{bmatrix} \qquad \mathbf{p}_3 = \begin{bmatrix} -1 \\ -1 \\ -1 \\ 1 \end{bmatrix} \qquad \mathbf{p}_t = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix}$$

$$\mathbf{t}_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \qquad \mathbf{t}_2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \qquad \mathbf{t}_3 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

$$\mathbf{W} = \mathbf{T}\mathbf{P}^T = \begin{bmatrix} -1 & -1 & 1 \\ -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ -1 & -1 & -1 & 1 \end{bmatrix} = \begin{bmatrix} -3 & -1 & -1 & -1 \\ 1 & 3 & -1 & -1 \end{bmatrix}$$

ii)

$$\mathbf{a} = \mathbf{hardlims}(\mathbf{W}\mathbf{p}_t) = \mathbf{hardlims}\left(\begin{bmatrix} -3 & -1 & -1 & -1 \\ 1 & 3 & -1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix} \right)$$

$$= \mathbf{hardlims}\left(\begin{bmatrix} -2 \\ -2 \end{bmatrix} \right) = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \rightarrow \mathbf{p}_1 .$$

Ex4 – AutoEncoders

- Suponha que quer treinar um classificador para uma tarefa em que tem muitos dados de treino não etiquetados mas apenas algumas centenas de instâncias etiquetadas. De que forma os "autoencoders" podem ajudar na resolução desta tarefa?

  Para treinar um classificador com muitos dados de treino não rotulados (sem label) mas apenas algumas instâncias rotuladas, pode-se:
  - Começar por treinar um autoencoder profundo, no conjunto de dados completo (rotulado + não rotulado)
  - Em seguida, reutilizar a metade inferior do autoencoder (encoder) para o para o classificador (ou seja, reutilizar as camadas até à camada de codificação, inclusive) e treinar o classificador usando os dados rotulados. Se tiver muitos poucos dados

rotulados, deve-se ainda “congelar” as camadas reutilizadas ao treinar o classificador.

- O que entende por um modelo generativo?
  Um modelo generativo é um modelo capaz de gerar aleatoriamente resultados que se assemelham às instâncias de treino. Por exemplo, uma vez treinado com sucesso no conjunto de dados MNIST, um modelo generativo pode ser utilizado para gerar aleatoriamente imagens realistas de dígitos.

Ex5 - RNN

- Descreva algumas aplicações para as seguintes arquiteturas de modelos recorrentes RNN:

Sequência-para-sequência:

  - Previsão do tempo (ou qualquer outra ou qualquer outra série temporal
  - Tradução automática (utilizando uma arquitetura codificador-decodificador);
  - Legendagem de vídeo;
  - Geração de música (ou outra geração de sequência),

Sequência-para-vetor

  - Classificar amostras de música por género musical, analisar
  - Analisar o sentimento de um texto,
  - Prever a probabilidade que um utilizador vai querer ver um filme com base no seu histórico de visionamento.

Vetor-para-sequência

  - Legendagem de imagens;
  - Criação de uma lista de reprodução de música com base numa incorporação do artista atual;
  - Gerar uma melodia com base num conjunto de parâmetros;

- Identifique duas das principais dificuldades no treino de uma RNN.

As duas principais dificuldades no treino das RNNs são os gradientes instáveis (que explodem ou desaparecem (tendem para zero) e uma memória de curto prazo limitada. Estes problemas agravam-se no tratamento de sequências longas.

- Descreva a arquitetura de rede neuronal que poderá ser utilizada para classificar um vídeo.

Para classificar vídeos com base no seu conteúdo, uma arquitetura possível poderia basear-se na amostragem (por exemplo de um frame por segundo) e, em seguida, fornecer este frame a uma rede neuronal convolucional (por exemplo, um modelo Xception pré-treinado).

Deve-se congelar os parâmetros do modelo transferido, se o conjunto de dados for reduzido. De seguida alimentar uma RNN (sequência para vetor) com a sequência de saídas da CNN. Por fim, passar o vetor de saída por uma camada “softmax”, calculado.se assim as probabilidades dpara cada classe.

- Descreva sumariamente a arquitetura de uma célula LSTM.

A arquitetura de uma célula LSTM inclui um vetor de estado de curto prazo e um vetor de estado de vetor de estado de longo prazo. A cada passo de tempo, as entradas e o estado anterior de curto prazo são alimentados a uma célula RNN simples e a três portas:

- A porta de esquecimento decide o que remover do estado de longo prazo;

- A porta de entrada decide que parte da saída da célula RNN simples deve ser adicionada ao estado de longo prazo;

- A porta de saída decide qual a parte do estado de longo prazo que deve ser adicionada ao estado de longo prazo nesse momento.