ICMC-USP

Lista de Exercícios 5 - Redes RBF
SCC-5809 - Redes Neurais

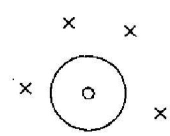
2o. Semestre de 2011 - Prof. João Luís

1. Identifique os pontos fortes e fracos do teorema de Cover:

   > Um problema de classificação de padrões complexos que "cai", de forma não-linear, num espaço de alta dimensão é mais provável ser linearmente separável do que em um espaço de baixa dimensão, desde que o espaço não seja densamente povoado.

2. O exemplo da figura 1 mostra uma dicotomia esfericamente separável. Assume-se que os quatro pontos de dados fora da superfície de separação situam-se num círculo e que o único ponto de dados dentro situa-se no centro da superfície de separação. Investigue como esta amostra de pontos de dados é transformada não-linearmente, usando
   (a) a multiquádrica $\varphi(x) = (x^2 + 1)^{1/2}$.
   (b) a multiquádrica inversa $\varphi(x) = \frac{1}{(x^2+1)^{1/2}}$.

Figure 1: Dicotomia esfericamente separável [1].

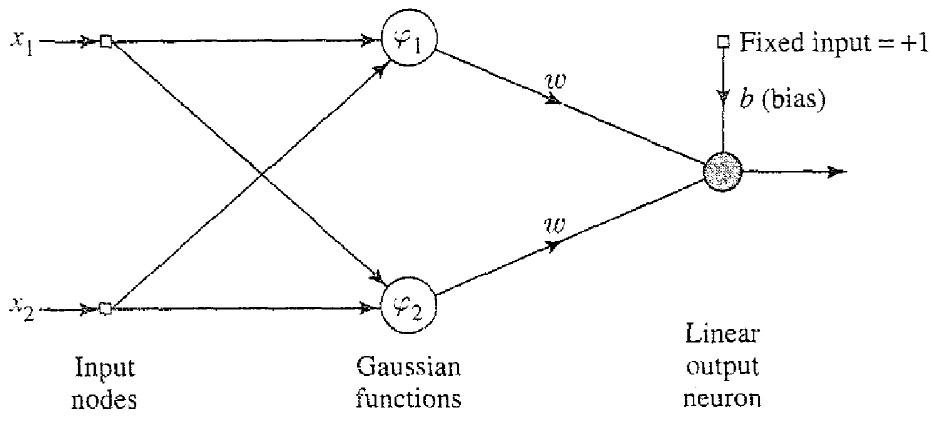


3. O conjunto de valores para o vetor de pesos **w** (equação 1) da rede RBF da figura 2 apresenta uma solução possível para o problema do XOR. Investigue um outro conjunto de valores para o vetor de pesos **w** para resolver esse problema.

$$\mathbf{w} = \begin{bmatrix} -2.5018 \\ -2.5018 \\ +2.8404 \end{bmatrix} \tag{1}$$

4. Na figura 2 apresenta-se uma solução ao problema do XOR usando uma rede RBF com duas unidades escondidas. Considere uma solução exata do problema do XOR usando uma rede RBF com quatro unidades escondidas, com cada centro da função de base radial sendo determinado por cada porção dos dados de entrada. Os quatro padrões de entrada possíveis são definidos por (0,0), (0,1), (1,1) e (1,0), que representam os cantos de um quadrado ordenados ciclicamente.

Figure 2: Rede Neural para resolução do problema do XOR [1].



(a) Construa a matriz de interpolação Φ para a rede RBF resultante. Depois, compute a matriz inversa Φ^{-1}.

(b) Calcule os pesos lineares da camada de saída da rede.

5. O relacionamento entrada-saída de uma rede RBF baseada em funções gaussianas é definido por

$$y(i) = \sum_{j=1}^{K} w_j(n) exp(-\frac{1}{2\sigma^2(n)} \parallel \mathbf{x}(i) - \mathbf{t}_j(n) \parallel^2), \quad i = 1, ..., n \tag{2}$$

onde $\mathbf{t}_j(n)$ é o ponto centro da j-ésima unidade gaussiana, a largura $\sigma(n)$ é comum a todas as K unidades e $w_j(n)$ é o peso linear atribuído à saída da j-ésima unidade; todos os parâmetros são medidos no tempo n. A função custo usada para treinar a rede é

$$\xi = \frac{1}{2} \sum_{i=1}^{n} e^2(i) \tag{3}$$

onde $e(i) = d(i) - y(i)$. ξ é uma função convexa dos pesos lineares na camada de saída, mas não-convexa com respeito aos centros e à largura das unidades gaussianas.

(a) Avalie as derivadas parciais da função custo com respeito a cada um dos parâmetros da rede $w_j(n)$, $\mathbf{t}_j(n)$ e $\sigma(n)$ para todo i.

(b) Use os gradientes obtidos na parte (a) para expressar as fórmulas atualizadas para todos os parâmetros da rede, assumindo os parâmetros taxa de aprendizado η_w, $\eta_{\mathbf{t}}$ e η_σ para os parâmetros ajustáveis da rede, respectivamente.

(c) O vetor gradiente ${}^{\partial\xi}/_{\mathbf{t}_j(n)}$ tem um efeito nos dados de entrada que é similar ao *clustering*. Justificar esta afirmação.

References

[1] S. Haykin, *Neural networks - a comprehensive foundation*, 2nd. ed. Prentice Hall, 1999.