

Lista 1 - Redes Neurais Artificiais

Felipe Bartelt de Assis Pessoa - 2016026841

1.1

Dada uma RNA do tipo MCP com função de ativação limiar e $\mathbf{x} = [-5 \ 7 \ 1]^T$ e $\mathbf{w} = [3 \ 7 \ b]$, terá a saída \hat{y} dada pelo produto interno canônico de forma que $\hat{y} = f(u)$, onde $u = \mathbf{x}^T \cdot \mathbf{w} = 34 + b$. Assim, assumindo-se

$$f(u) = \begin{cases} 1, & u \geq 0 \\ 0, & c.c \end{cases}$$

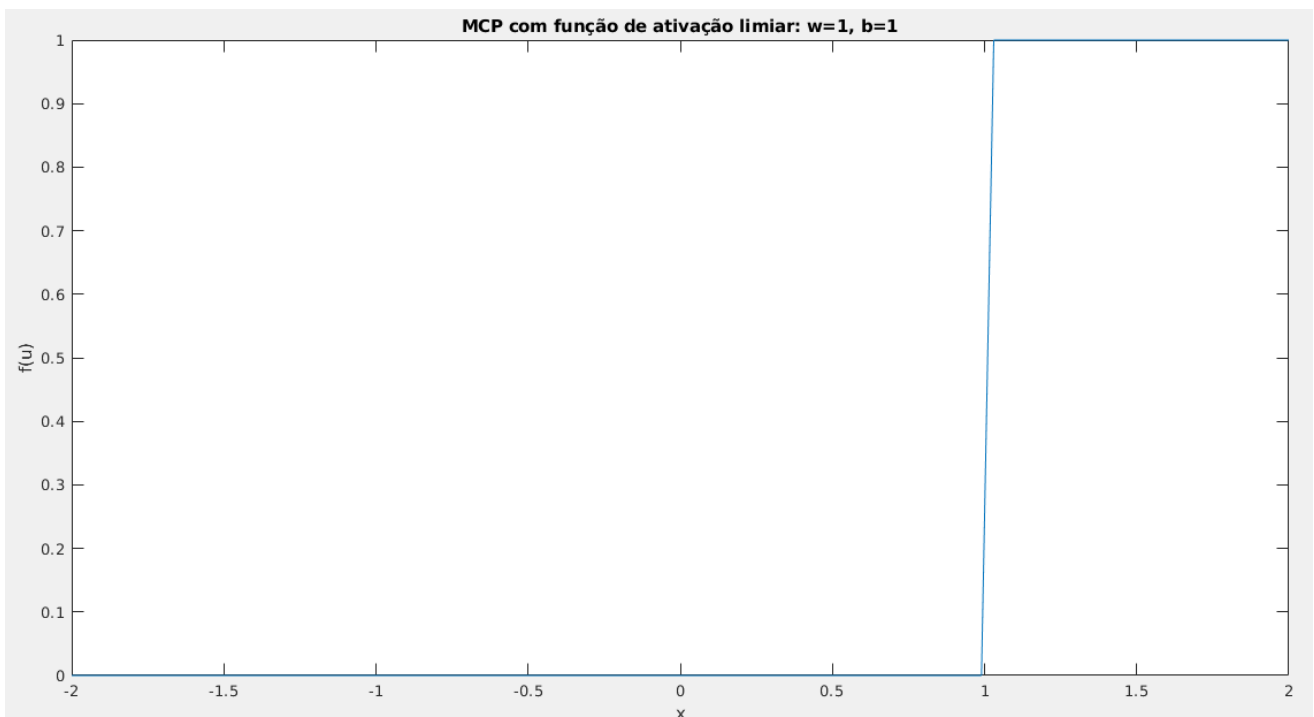
Tem-se que os valores de b para que a saída seja 0 ou 1 são dados por:

- Para $\hat{y} = 1$, segue que $u = 34 + b \geq 0 \therefore b \geq -34$
- Para $\hat{y} = 0$, segue que $u = 34 + b < 0 \therefore b < -34$

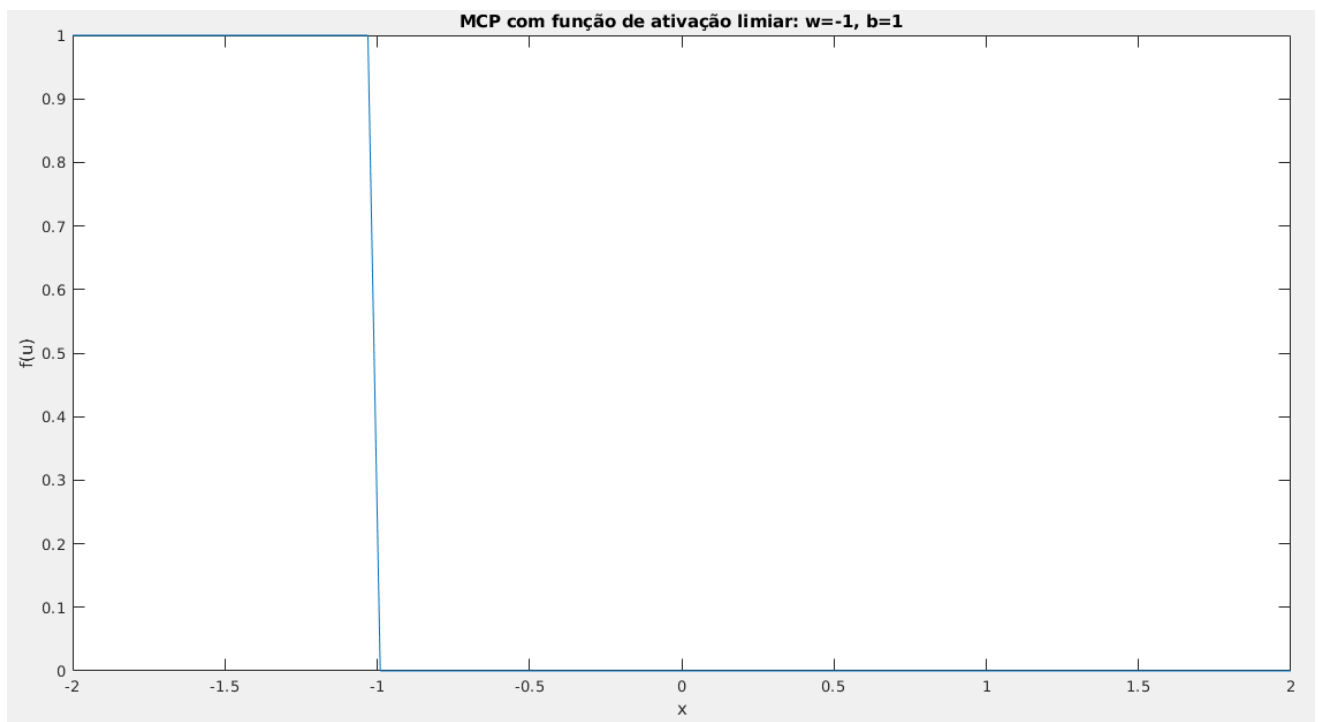
1.2

Para um neurônio MCP com função de ativação limiar, esboçou-se via matlab:

- $w = 1, b = 1$

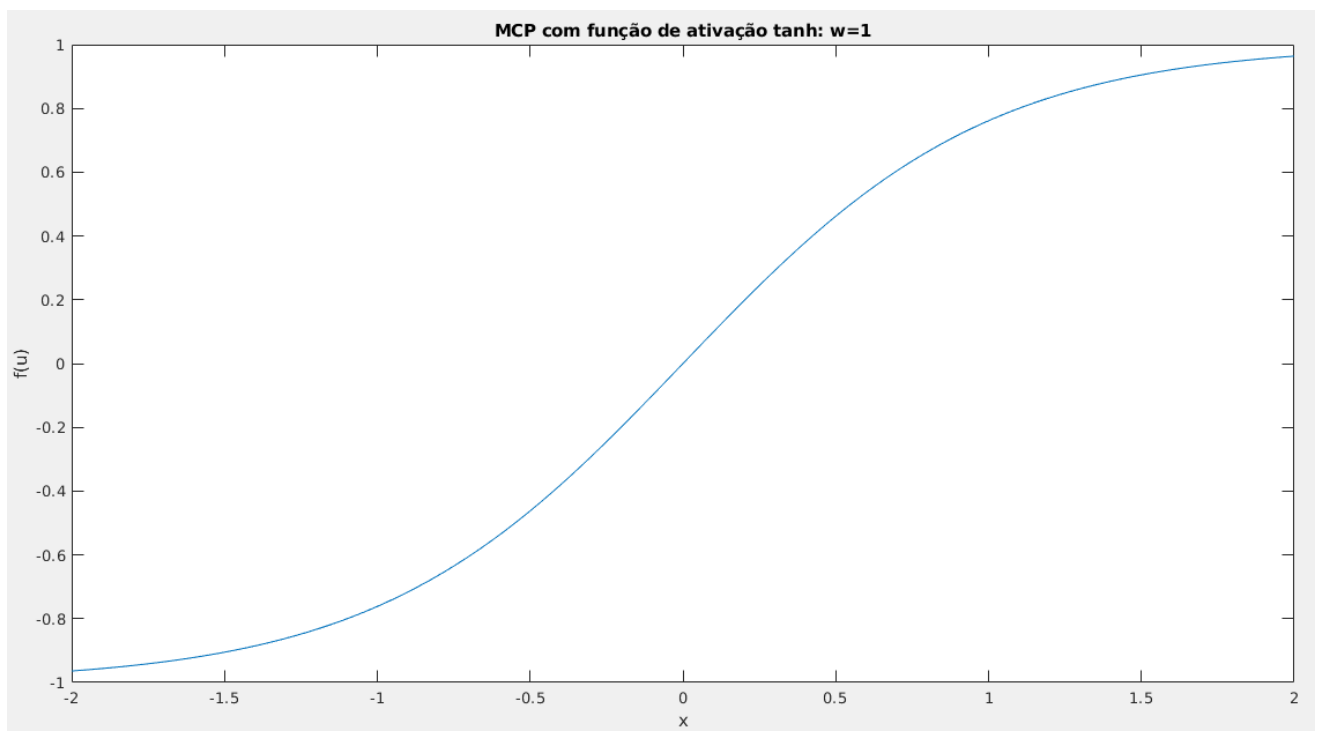


- $w = -1, b = 1$

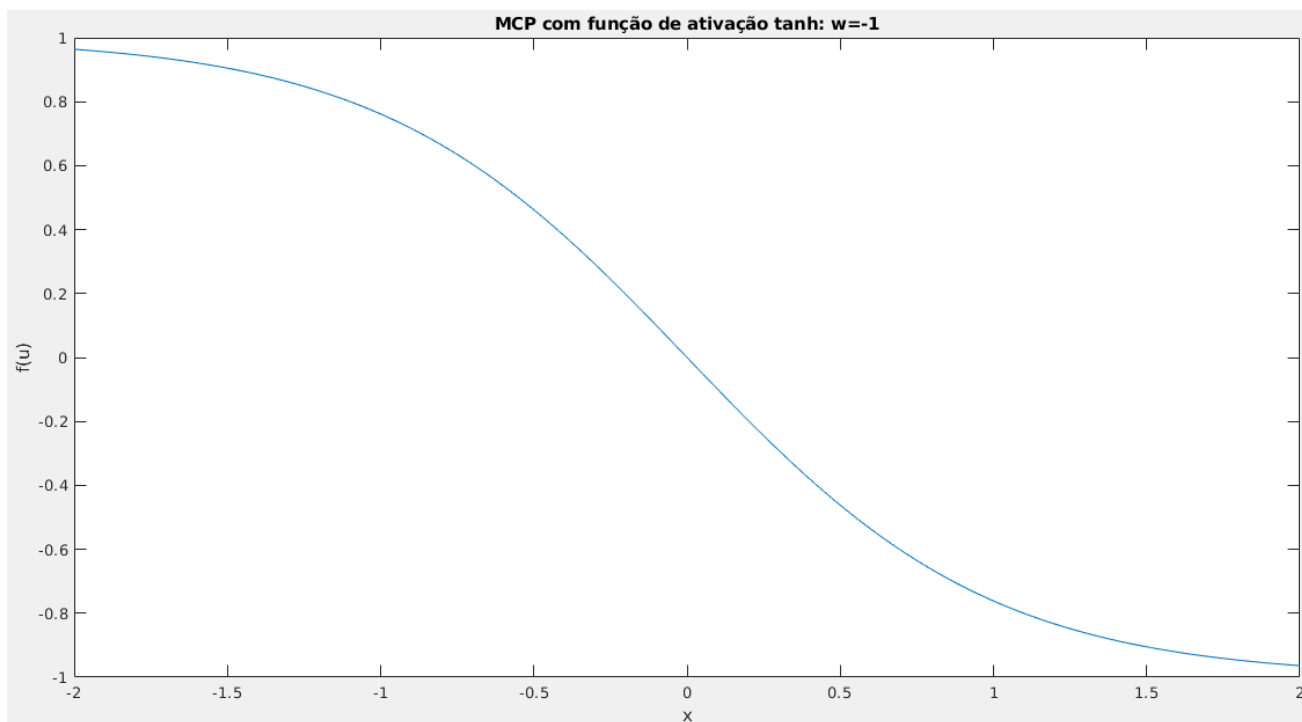


Para um neurônio MCP com função de ativação tangente hiperbólica:

- $w = 1$



- $w = -1$



2

Tomou-se como base: R.A. Teixeira, A.P. Braga, R.H.C. Takahashi, and R. Rezende. **Improving generalization of mlps with multi-objective optimization.**

O artigo busca estudar uma nova forma de treinamento de redes neurais supervisionadas. Uma das grandes dificuldades para a modelagem de redes neurais é o ajuste do algoritmo de forma a evitar *underfitting* e *overfitting*, além de garantir certa precisão para casos genéricos. Para atender a esses critérios, muitas vezes o espaço de amostras disponível é dividido entre conjunto de treino, validação e teste, de forma a tentar encontrar o melhor grau de polinômio para o problema, assim como os melhores parâmetros para o modelo funcionar para casos gerais. Para o artigo, isso se traduz em minimizar o erro quadrático, o que seria melhorar a generalização do modelo; e otimizar a norma dos vetores de peso, que pode ser interpretado como garantir uma complexidade, grau de polinômio, ideal para o modelo.

A abordagem de otimização multi-objetivo consiste, justamente, em encontrar o ponto ótimo para esses dois critérios supracitados. Dessa forma, o primeiro passo abordado é encontrar um conjunto pareto-ótimo, isso é, uma superfície n -dimensional formada por condições pareto-ótimas. Essa superfície, pela abordagem escolhida é um cone centrado na solução utópica, formado pela diferença entre os vetores de solução ótima para os objetivos individuais e a solução utópica. Constrói-se então, com base nesse cone, um vetor v_k por meio de uma combinação convexa, que permite formar uma região com um único mínimo global, tornando o problema, antes multi-objetivo, em um problema mono-objetivo. Para a solução desse problema mono-objetivo é utilizado o "algoritmo elipsoide", que é uma espécie de atalho para problemas de dilatação espacial, permitindo uma menor complexidade do algoritmo ao restringir o espaço de soluções ótimas a um volume menor. Esse método, porém, como o próprio Naum Z. Shor aponta, para problemas de dimensão maior que 2 é necessário procedimentos deveras conturbados e, aparentemente, o método se torna dispensável a partir da dimensão 4.

Após enunciar o método proposto, o artigo busca compará-lo com *backpropagation* e SVM, para problema de classificação e regressão. Assim, o algoritmo se mostrou superior ao *backpropagation*, uma vez que este claramente sofreu de *overfitting* em ambos os testes. Já com relação a SVM, o novo algoritmo se mostrou equiparável, obtendo respostas muito próximas. De certa forma, tanto SVM quanto o método proposto se baseiam em dilatação espacial e é possível que, mesmo selecionando outra função de similaridade, para outros tipos de problema, o novo método pudesse ter melhor desempenho que o SVM. É possível que o método proposto seja excelente para algum tipo de situação que se desconhece, uma vez que, pelo mínimo conhecimento que se tem, imagina-se que todo conjunto pareto-ótimo seja de custo computacional alto. Especula-se, sem autoridade, que essa situação seja conjunta de um problema de dimensão muito maior que o número de exemplos de treino ou para o caso oposto, onde a dimensão do problema é pequena, mas existem muitos exemplos, uma vez

que, para esses casos, é mais recomendado se utilizar SVM sem kernel, algo que poderia permitir a soberania do algoritmo proposto dentro dessas situações.