

1. Explique com suas palavras o que é um neurônio artificial e o seu funcionamento traçando uma analogia com o neurônio biológico.

Neurônios artificiais são modelos computacionais baseados em neurônios biológicos. A idéia por trás dos neurônios artificiais é que eles possam refletir, de maneira relativamente simples, o funcionamento de um neurônio biológico, a partir de uma analogia com seus três elementos principais: dendrito, corpo celular e axônio. Assim como um neurônio biológico recebe estímulos elétricos através de seus dendritos, um artificial recebe suas entradas, que são ponderadas, agregadas e processadas, da mesma forma como são processados os estímulos elétricos no corpo celular. O processamento das entradas de um neurônio artificial gera um potencial de ativação, que uma vez verificado pela função de ativação, produz uma saída que pode vir a sensibilizar camadas de neurônios seguintes. O mesmo comportamento é verificado nos neurônios biológicos, que podem ter seus estímulos elétricos propagados por seus axônios de acordo com a ação de neurotransmissores.

2. Explique o objetivo do uso do Bias.

O bias é um fator de limitação de ativação do neurônio, atuando diretamente na saída do mesmo, visto que influencia no valor combinado das entradas, determinando o valor final do disparo do neurônio.

3. Descreva o objetivo do uso da função de ativação de um neurônio.

O propósito da função de ativação de um neurônio artificial é delimitar a saída do mesmo, de forma a controlar o estímulo enviado para a próxima camada, ou para a saída da rede. Este comportamento da função de ativação reflete o papel dos neurotransmissores em neurônios biológicos.

4. Faça a implementação do algoritmo de treinamento do neurônio perceptron. Realize o treinamento do neurônio para a solução do problema do operador lógico OU partindo da inicialização do pesos abaixo:

w1: 0.3192

w2: 0.3129

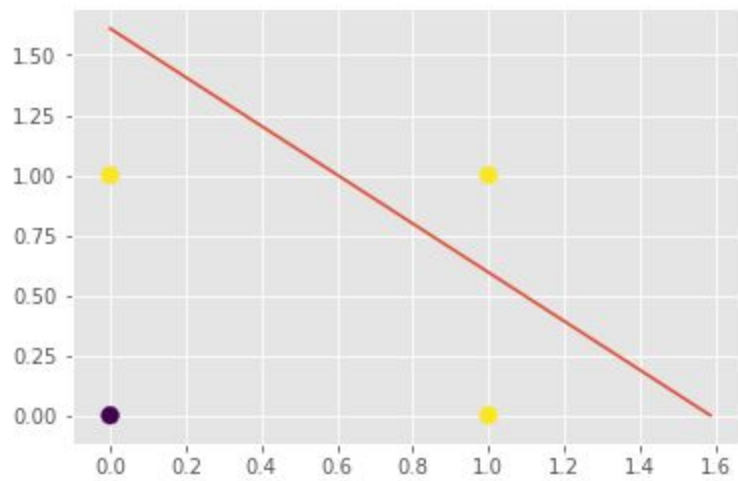
bias: -0.8649

a) Em quantos passos o treinamento convergiu?

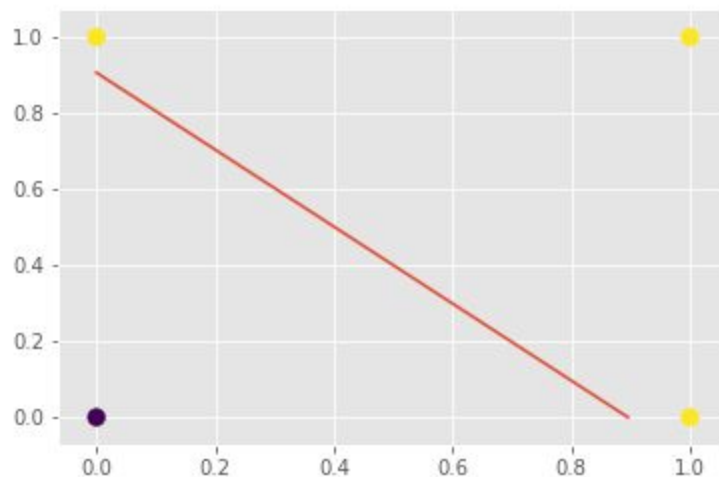
Utilizando taxa de aprendizagem 0.1, o treinamento convergiu em 3 épocas.

b) Exiba algumas figuras (pelo menos 2) mostrando o hiperplano de separação do problema nas diferentes épocas do treinamento.

Ao final da época 1:



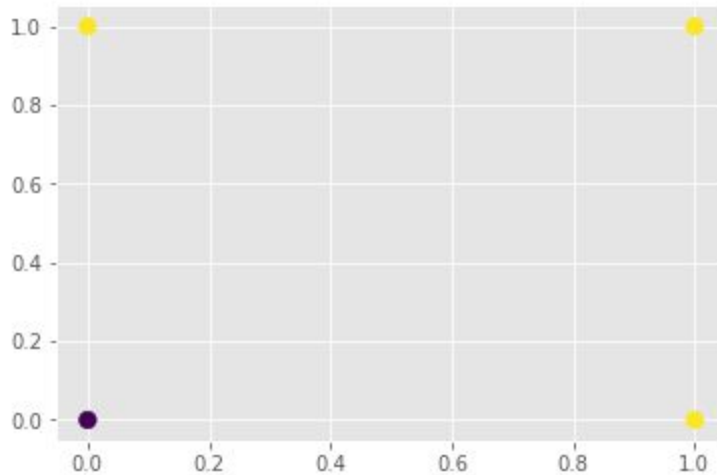
Ao final da época 3:



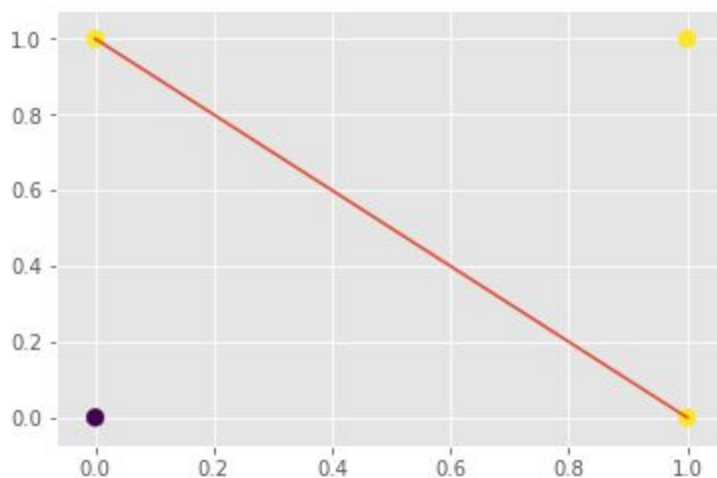
c) Defina outro conjunto de valores de inicialização dos pesos e repita os itens a) e b)

Utilizando zero como os valor inicial dos pesos, e 0.1 como taxa de aprendizagem, o treinamento convergiu em 4 épocas.

Ao final da época 1:



Ao final da época 4:



5. Qual a diferença entre o algoritmo de treinamento do neurônio perceptron e o Adaline?

A diferença entre o algoritmo de treinamento do perceptron e do Adaline é a regra de aprendizado que cada um utiliza.

O perceptron utiliza a regra de Hebb, que diz que os pesos sinápticos e limiares da rede serão atualizados proporcionalmente em relação aos seus sinais de entrada, de forma que tais atualizações podem ser excitatórias (incrementando os pesos e limiares) ou inibitórias (decrementando os pesos e limiares). Os ajustes serão excitatórios quando as saídas do perceptron condizem com os valores desejados, e serão inibitórios quando as saídas forem diferentes dos valores esperados. No perceptron, estes ajustes acontecem até uma época de treinamento em que todas as saídas produzidas sejam correspondentes às saídas esperadas.

Já o Adaline utiliza a regra Delta como regra de aprendizado, que se diferencia da regra de Hebb por ter uma forma diferente de atualização dos pesos, assim como outra condição de parada. Na regra Delta os pesos são atualizados considerando não uma saída de uma função de ativação degrau ou degrau bipolar, mas uma função identidade, em que a saída da função de ativação é o próprio potencial de ativação. Desta forma, os pesos são atualizados considerando a diferença entre o sinal desejado e o valor do potencial de ativação. A regra Delta ainda determina como condição de parada para o algoritmo um erro médio quadrático calculado para todas as entradas, após uma época de atualização dos pesos, até que a diferença entre o erro médio da época atual para o erro médio da época anterior seja igual ou inferior a uma precisão estabelecida.

6. Independente dos valores iniciais assumidos para o vetor de pesos do Adaline, uma mesma configuração final para w (final) será sempre obtida após a sua convergência. Essa afirmação é falsa ou verdadeira? Justifique.

Para o Adaline esta afirmação é verdadeira, pois o uso da regra Delta no treinamento do neurônio indica que ele só irá parar quando uma configuração final, ótima, for atingida. A regra Delta implementa o método gradiente descendente na função de erro quadrático, de forma que usar uma variação do erro quadrático médio, como condição de parada no treinamento, possibilita achar um erro mínimo tolerado, e, conseqüentemente, uma mesma configuração final dos pesos sinápticos que correspondam a este erro mínimo.

7. Faça a implementação do algoritmo de treinamento do neurônio Adaline. Realize o treinamento do neurônio para a solução do problema do operador lógico OU partindo da inicialização dos pesos abaixo:

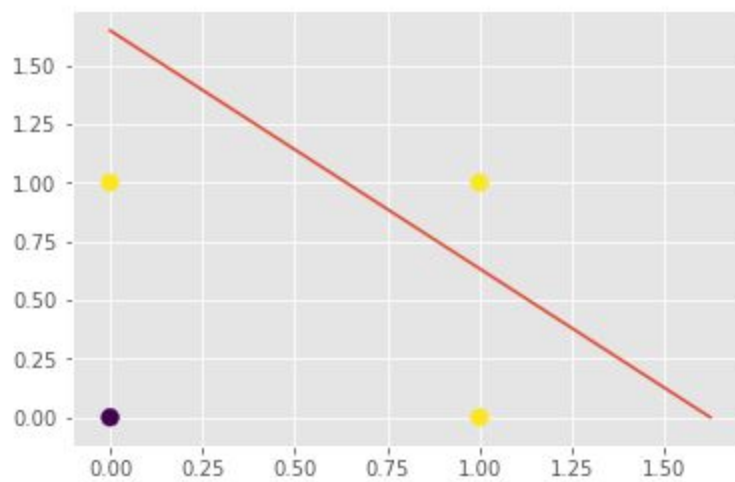
**w_1 : 0.3192
 w_2 : 0.3129
bias: -0.8649**

a) Em quantos passos o treinamento convergiu?

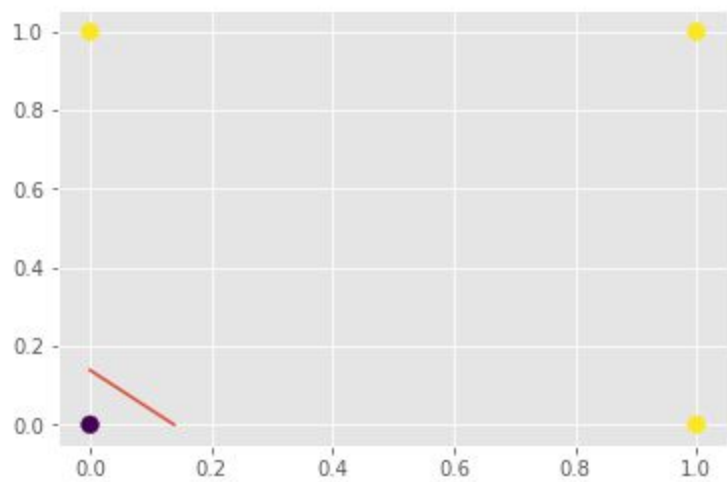
Utilizando taxa de aprendizagem 0.1 e precisão 0.000001, o treinamento convergiu em 1188 épocas.

b) Exiba algumas figuras (pelo menos 2) mostrando o hiperplano de separação do problema nas diferentes épocas do treinamento.

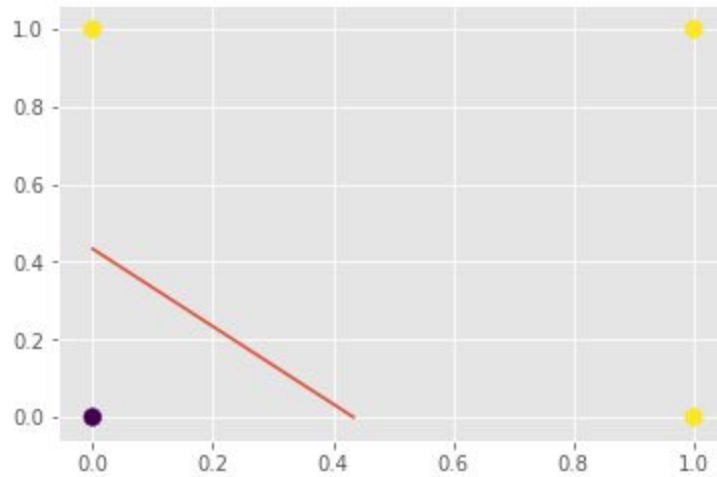
Ao final da época 1:



Ao final da época 50:



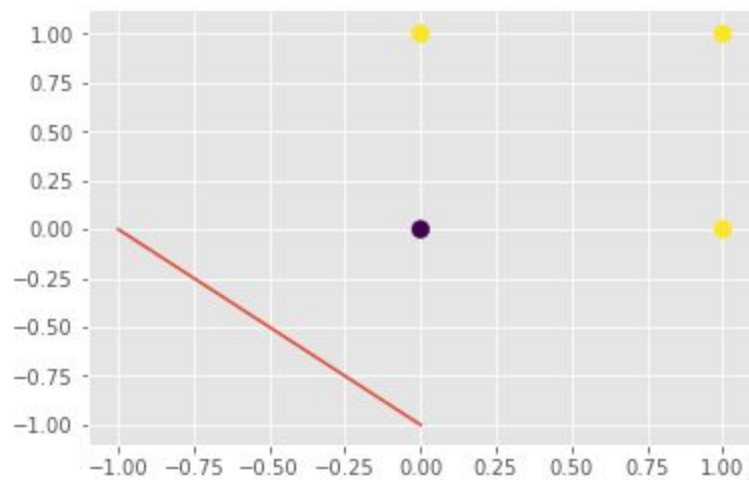
Ao final da época 1188:



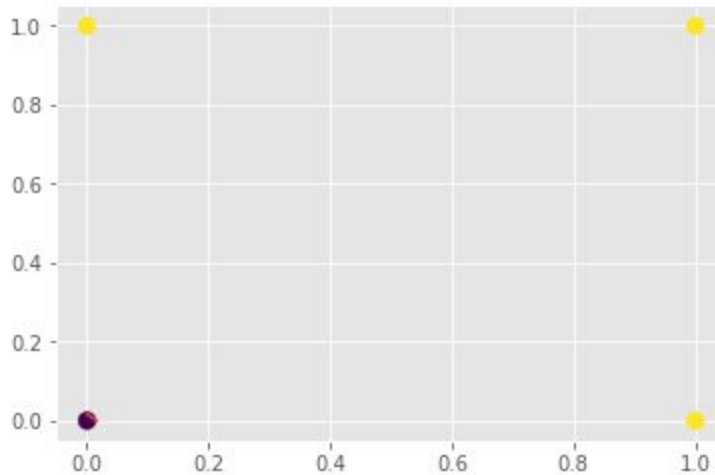
c) Defina outro conjunto de valores de inicialização dos pesos e repita os itens a) e b)

Utilizando zero como o valor inicial dos pesos, 0.1 como taxa de aprendizagem e 0.000001 como precisão, o treinamento convergiu em 1206 épocas.

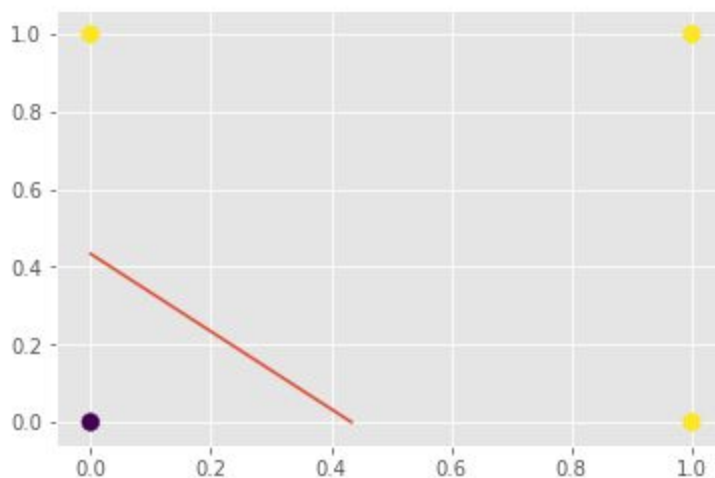
Ao final da época 1:



Ao final da época 50:



Ao final da época 1206:



8. Qual o objetivo e importância da taxa de aprendizado no processo de treinamento de um neurônio artificial?

A taxa de aprendizagem de uma rede neural determina o ritmo em que ela irá convergir para uma solução. Valores muito grandes para uma taxa de aprendizagem podem fazer com que não haja convergência, visto que ela traria instabilidade à atualização dos pesos, fazendo com que variem muito enquanto são ajustados, entre valores muito grandes ou muito pequenos, sem conseguir chegar em valores intermediários adequados. Por este motivo, convém usar valores menores, entre 0 e 1, para que a atualização dos pesos seja mais controlada, possibilitando a convergência. Vale notar que valores muito pequenos para a taxa de aprendizagem também podem dificultar a convergência, visto que a atualização dos pesos seria muito pequena, e o aprendizado da rede, muito lento.