

# Lista 3 de IPN0007 - Redes Neurais na Engenharia Nuclear

Aluno: Luís Felipe de Melo  
Número USP: 9297961

## Exercício 1

A base do algoritmo Hebbiano de Aprendizado, assim como a base de todos os outros algoritmos de aprendizado é a repetição de exemplos de treinamento.

## Exercício 2

O tipo de memória que mais se assemelha com os métodos usados nas RNA é a memória reflexiva, já que ela é relacionada a pensamento e raciocínio, portanto, ao aprendizado.

## Exercício 3

Podemos fazer análises relacionadas a:

**Familiaridade** o quanto de semelhança um novo padrão tem em relação a algum já visto.

**Análise de componente principal** enquanto o acima é em relação a uma unidade, este se refere a uma série.

**Agrupamento de padrões** dada uma entrada, a rede dá como saída a categoria com a qual a entrada se parece.

**Codificação** a rede pode fornecer uma versão codificada da entrada.

**Mapeamento de características** se as unidades da saída possuem um arranjo geometricamente fixo, a rede pode fazer um mapeamento topográfico das entradas de modo a acionar unidades de saída vizinhas para entradas parecidas.

## Exercício 4

Em função da regra básica existem dois tipos de algoritmo de aprendizado:

**Regras de Correção de Erro** com esta regra, a atualização dos pesos é função do erro das saídas.

**Regras de Gradiente Descendente** com esta regra, a atualização segue o gradiente negativo de uma função custo para os pesos.

## Exercício 5

Um ADALINE possui saída contínua e linear, suas funções de transferência são diferenciáveis, ele tenta minimizar uma função de custo (soma dos erros quadráticos) e utiliza uma regra de gradiente descendente.

## Exercício 6

Um processo de aprendizado passo-a-passo utiliza uma regra de correção de erro. Sua aplicação ocorre após a exibição de cada exemplo. Isso quer dizer que após cada exemplo fornecido para o treinamento, ele verifica o erro e altera os pesos para que a próxima rodada de treinamento forneça um erro menor.

## Exercício 7

Um processo de aprendizado de batelada utiliza uma regra de correção de erro. Sua aplicação ocorre após a exibição de todo o conjunto de dados, ou seja, os pesos são atualizados menos vezes que o aprendizado passo-a-passo.

## Exercício 8

O objetivo desse tipo de algoritmo é fazer a atualização necessárias, no pesos, na direção da solução ótima. Tal condição é alcançada usando o gradiente da função custo.

## Exercício 9

Quando os exemplos (animais) são alterados, a saída não se altera muito. Isso se deve ao fato de que as características já estão predefinidas. Por exemplo, tirei a baleia dos exemplos e coloquei a cobra. Coloquei a baleia na hora do teste e o valor resultante estava bem perto do que era antes. Quanto mais se aumenta a taxa de aprendizado, mais a saída cresce, porém, ainda é possível entender os resultados. Quando ela chega em 0,009, nem todos os valores são concretos. Algo parecido ocorre com os pesos iniciais.

## Exercício 10

Quando se aumenta a taxa de aprendizado já para 0,2, a RNA não consegue aprender corretamente. As outras funções precisam de taxas de aprendizado menores do que 0,1 para que a saída seja correta.