Relatório sobre a atividade “12 - Prática: Redes Neurais (II)”

Lucas Gabriel Arenhardt

1. Introdução

Nesse módulo são apresentadas alguns tipos de regressão, sendo elas a linear, a polinomial e a múltipla. Também são apresentados diversos métodos utilizados para a criação de modelos em aprendizado de máquina. Todos os conceitos aprendidos nas aulas estão apresentados a seguir:

2. Conceitos

2.1. Neurônio Artificial

É um componente fundamental de uma rede neural artificial. Ele é uma unidade computacional que recebe um ou mais sinais de entrada, processa-as e gera uma saída. Cada entrada possui um peso relacionado.

2.2. Função de Soma

Esta função soma todas as entradas ponderadas (multiplicadas por seus respectivos pesos).

2.3. Função de Ativação

O resultado da função de soma é passado para a função de ativação, que introduz não-linearidade ao modelo. Alguns exemplos são: sigmóide, tangente hiperbólica e ReLU.

2.4. Camadas

As camadas são grupos de neurônios organizados de maneira sequencial ou hierárquica, e cada camada realiza uma transformação específica nos dados de entrada. As camadas principais em uma rede neural são:

2.4.1. Camada de Entrada

Esta é a primeira camada de uma rede neural. Ela recebe os dados brutos.

2.4.2. Camadas Ocultas

Estas são as camadas intermediárias entre a camada de entrada e a de saída. Cada camada oculta consiste em múltiplos neurônios que recebem as entradas da camada anterior, processam essas informações e passam os resultados adiante.

2.4.3. Camada de Saída

Esta é a última camada de uma rede neural. Ela produz a saída final da rede, após o processamento feito pelas camadas ocultas.

2.5. Erro

O erro é fundamental para entender a eficácia de uma rede neural artificial. O seu cálculo é baseado em comparar o valor real esperado com o valor obtido na saída da rede neural e isso é realizado através da Função de Perda (Loss Function) ou Função de Custo (Cost Function).

2.6. Backpropagation

É o processo de ajustar os pesos da rede neural para minimizar o erro. Este processo envolve o cálculo do gradiente, que indica como ajustar da melhor forma os valores de peso.

Diferentes algoritmos de otimização podem ser usados para ajustar os pesos, por exemplo:

2.6.1. Stochastic Gradient Descent (SGD)

Calcula o erro para cada registro e atualiza os pesos utilizando uma amostra de treinamento por vez.

2.6.2. Batch Gradient Descent (BGD)

Calcula o erro para todos os registros e atualiza os pesos.

2.6.3. Mini-Batch Gradient Descent

Atualiza os pesos utilizando um conjunto de amostras escolhido, não necessitando todos os registros.

2.7. Bias

O bias é uma entrada adicional de valor “1” e sua função é ajustar a função de ativação. Isso torna a rede neural mais flexível, de forma a modelar corretamente os dados (especialmente quando as entradas são todas zero).

2.8. Underfitting

Ocorre quando o modelo é muito simples para capturar a complexidade dos dados. Nesse caso, o modelo tende a apresentar resultados ruins já na fase de treinamento.

2.9. Overfitting

Ocorre quando o modelo é muito complexo e se ajusta demais aos dados de treinamento. Isso resulta em um ótimo desempenho no treinamento, mas um péssimo desempenho nos testes.

3. Conclusão

Este módulo foi fundamental para compreender melhor como funciona e como é construída uma rede neural. As aulas foram muito bem elaboradas e a exemplificação com gráficos, imagens e código foi muito bem construída. Todos os conceitos ficaram muito claros durante a explicação.