

REDES NEURAIS ARTIFICIAIS: UMA ABORDAGEM MATEMÁTICA

Kauan Carlos Rigoletto Fernandes (PIC/Uem), Emerson Vitor Castelani (Orientador).
E-mail: evcastelani@uem.br.

Universidade Estadual de Maringá, Centro de Ciências Exatas, Maringá, PR.

Matemática/Matemática aplicada

Palavras-chave: otimização; aprendizagem de máquina; gradiente espectral

RESUMO

Este trabalho apresenta o estudo das redes neurais artificiais de uma camada, com foco nas propriedades matemáticas e nas técnicas de treinamento utilizadas. O trabalho explora os conceitos básicos das redes neurais, incluindo funções de ativação e métodos de otimização, como o gradiente descendente. O objetivo é fornecer uma compreensão da teoria e aplicação das redes neurais de uma camada em problemas de aprendizagem supervisionada.

INTRODUÇÃO

O avanço das tecnologias de inteligência artificial e aprendizado de máquina tem permitido a modelagem de sistemas complexos através de algoritmos capazes de identificar padrões em grandes volumes de dados, algo muito comum na atualidade. Entre as técnicas mais usadas no aprendizado supervisionado, as redes neurais artificiais se destacam por sua habilidade em modelar funções complexas com camadas de neurônios interconectados, sendo fundamentais em áreas como reconhecimento de imagem e processamento de linguagem natural (HAYKIN, 2001).

O estudo das redes neurais remonta à década de 1950, com a invenção do perceptron por Frank Rosenblatt (1958), um dos primeiros modelos de rede neural. Embora tenha sido inovador, o perceptron enfrentava limitações em problemas não linearmente separáveis, como a função XOR, superadas com o desenvolvimento das redes neurais profundas na década de 1980 (HOCHREITER; SCHMIDHUBER, 1997).

Este trabalho foca nas redes neurais de uma camada, arquitetura simples que oferece uma compreensão essencial dos métodos numéricos de treinamento e funções de ativação.

Embora limitadas em poder de modelagem, elas são eficazes para problemas linearmente separáveis. O objetivo é explorar os aspectos numéricos do treinamento dessas redes, com ênfase em técnicas de otimização como o gradiente descendente, essenciais para ajustar os parâmetros e minimizar a função de custo, especialmente em cenários com grandes volumes de dados e alta dimensionalidade (DAI et al., 2019).

MATERIAIS E MÉTODOS

Os modelos e os códigos foram implementados na linguagem Julia (BEYER et al., 2017). O modelo principal utilizado nos experimentos foi uma rede neural artificial de uma camada, com função de ativação logística, a qual se configura como um classificador linear. A seguir, apresenta-se uma ilustração do modelo considerando um banco de dados com três colunas, onde f representa a função de ativação e b o viés (do inglês bias).

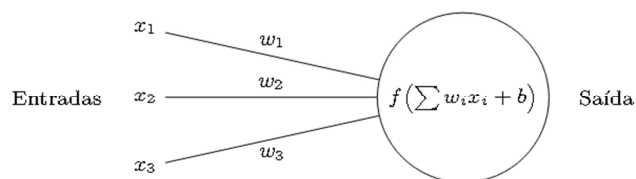


Figura 1 - Ilustração de rede neural simples com 1 camada

Esse modelo é capaz de ser treinado e realizar uma classificação linear no banco de dados. Para seu treinamento, a maioria dos métodos se baseia no algoritmo de gradiente descendente, que consiste em ajustar os parâmetros na direção oposta ao gradiente da função de custo que se deseja minimizar. Existem diferentes abordagens para esse ajuste, que variam desde a utilização de passos fixos — onde a atualização dos parâmetros é feita por um múltiplo constante do gradiente (tal múltiplo será chamado de tamanho do passo ou taxa de aprendizado) — até estratégias mais sofisticadas que buscam otimizar a taxa de aprendizado, de modo a minimizar a função de custo de maneira mais eficiente. Ao interpretar o viés como um peso w_0 cujo coeficiente de entrada é sempre $x_0 = 1$, é possível realizar uma atualização direta de todos os parâmetros da rede, utilizando a equação $w_{k+1} = w_k - \eta \nabla C(w_k)$, onde η é a taxa de aprendizado e $\nabla C(w)$ é o gradiente da função de custo. Durante esse processo iterativo, espera-se que o modelo convirja para um mínimo local da função de custo. Para avaliar o desempenho do modelo, é uma prática usual dividir o banco de dados em 80% para treinamento e 20% para teste, o que permite calcular a acurácia da rede neural ao medir a porcentagem de acerto no conjunto de teste depois que o modelo foi treinado.

RESULTADOS E DISCUSSÃO

Para expressar nossos resultados, utilizamos três variações do método do gradiente descendente: passo fixo — que será chamado Fixo —, passo com busca linear por seção áurea — que será chamado Cauchy — e passo espectral — que será chamado BB1.

Foram realizados na linguagem de programação Julia vários testes de treinamento de redes de uma camada, dentre os quais gostaríamos de elencar o treinamento de uma rede neural de uma camada com função de ativação sigmoide

(do inglês sigmoid) de acordo com a estrutura abaixo.
Os resultados apresentados têm como base:

- Problemas: 100 datasets linearmente separáveis de 500 linhas cada um, gerados aleatoriamente com suas dimensões variadas também aleatoriamente entre [2, 3, 5, 10, 15, 20, 25].
- Salvaguarda: Máximo de iterações igual a 300 mil (mas, caso atinja este valor sem atingir a tolerância do gradiente, consideramos que o problema não foi resolvido).
- Critério de parada: A norma do gradiente ficar menor que 10^{-4} .
- Ponto de partida: Em todos os problemas, inicializamos todos os parâmetros como zero.

Assim, obtemos os seguintes gráficos de perfis de desempenho abaixo.

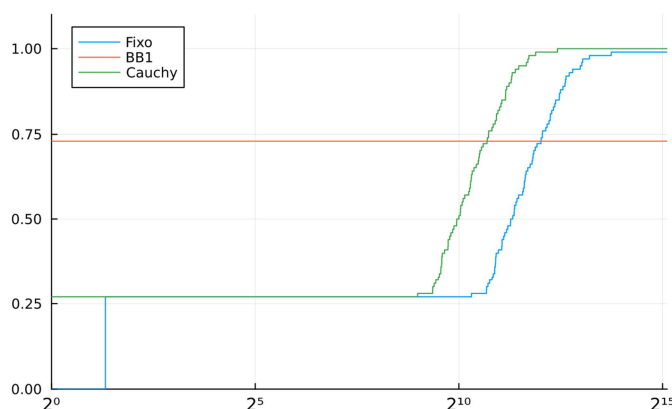


Figura 2 - Gráfico de quantidade de iterações

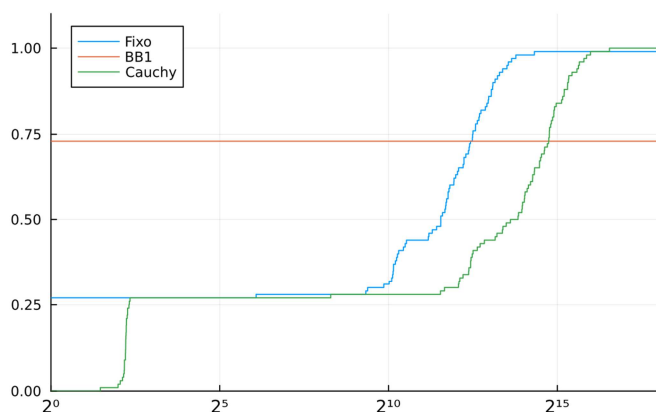


Figura 3 - Gráfico de tempo de execução

Portanto, de acordo com um dos testes feitos sob as circunstâncias descritas observamos que em 73 dos 100 problemas, o método BB1 apresentou convergência

significativamente mais rápida que os outros dois métodos, tanto em tempo de execução quanto em número de iterações. Contudo, em 23 dos problemas, tal método não convergiu dentro da tolerância exigida, atingindo o limite máximo de iterações. Note ainda que, ao reduzirmos o número máximo de iterações permitidas, algo que pode ocorrer na prática devido ao custo e dimensão do treinamento, os métodos Fixo e Cauchy podem não atingir uma resposta satisfatória. Em situações desse tipo, o método BB1 se mostra mais promissor.

CONCLUSÕES

Conclui-se que as redes neurais artificiais desempenham papéis importantes nas áreas de aprendizado de máquina e otimização. A análise e implementação de métodos numéricos são fundamentais para treinar modelos, destacando a importância da modelagem matemática e computacional no processo de otimização e aprendizado.

AGRADECIMENTOS

Agradeço à UEM, ao DMA, ao PET-Matemática da UEM e ao meu orientador, Prof. Dr. Emerson Vitor Castelani, por ter me aceitado orientar, pela paciência e incentivo.

REFERÊNCIAS

BEZANSON, Jeff et al. Julia: *A fresh approach to numerical computing*. SIAM review, v. 59, n. 1, p. 65-98, 2017.

DAI, Yu-Hong; HUANG, Yakui; LIU, Xin-Wei. A family of spectral gradient methods for optimization. *Computational Optimization and Applications*, New York, v. 74, n. 1, p. 43-65, jan. 2019. Disponível em: <https://doi.org/10.1007/s10589-019-00107-8>

HAYKIN, S. S. — *Redes Neurais: Princípios e Prática*. Porto Alegre: Bookman, 2001.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, v. 65, n. 6, p. 386-408, 1958.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural Computation*, v. 9, n. 8, p. 1735-1780, 1997.

RUSSELL, S.; NORVIG, P. — *Artificial Intelligence: A Modern Approach*, 3ª ed. Upper Saddle River, NJ: Prentice Hall, 2010.