

Regularização de redes ELM e RBF

João Pedro Miranda Marques - 2017050495

19 de junho de 2022

1 Introdução

A Regularização está relacionada a um dos dois fundamentais erros estatísticos em modelos científicos: overfitting e underfitting, como afirma McElreath em *Statistical Rethinking*[2].

Modelos com Underfitting possuem baixa capacidade de relacionar os dados de entrada e saída gerando assim uma elevada taxa de erro tanto nos dados de treinamento quanto de validação e teste.

Por outro lado o Overfitting, segundo Douglas M. Hawkins em *The Problem of Overfitting*[1], é o uso de modelos ou procedimentos que violam a parcimônia, ou seja, que incluem mais termos do que o necessário ou usam abordagens mais complicadas do que o necessário.[1] Dessa maneira o overfitting apresentam uma aproximação muito detalhada com relação aos dados de treinamento, absorvendo assim informações de ruído que não pertencem à função geradora do problema. Como resultado apresentam uma ótima acurácia com relação aos dados de treinamento, porém ao utilizar novos dados há prejuízos devido esse deslocamento da função geradora do problema.

Redes Neurais solucionam problemas de maneira multiobjetivo¹, ou seja, costumam ter não só uma solução ótima, mas um conjunto de soluções ótimas, denominado conjunto Pareto-Ótimas(PO). Esse conjunto PO é representado na Figura 1 pela região decrescente da curva em vermelho, nessa região há um trade-off entre erro de generalização e complexidade do modelo.

Vemos também que a curva em vermelho apresenta um vale. Nesse ponto encontramos a menor taxa de erro de generalização para o modelo utilizado. Vemos que a medida que aumenta a complexidade do modelo temos uma absorção do ruído de treinamento ocasionando o overfitting.

A regularização portanto consiste em métodos para reduzir os efeitos de overfitting e aproximar nossa solução do conjunto Pareto-Ótimo.

Este trabalho tem o objetivo de estudar os métodos de regularização implementados em redes ELM² e RBF³ e avaliar seu desempenho.

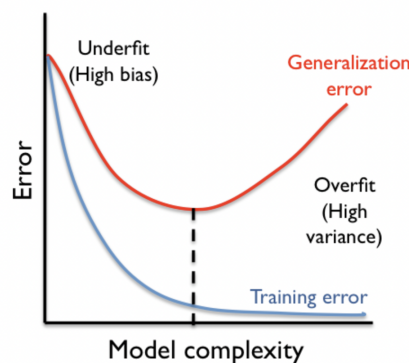


Figura 1: Conjunto de soluções para o problema.

2 Revisão de trabalhos correlacionados

O algoritmo de regularização é encontrado em diversas áreas de estudo, porém com nomes diferentes. Na estatística é conhecido como *Ridge regression*, em otimização de sistemas é *Regularização de Tikhonov* e em aprendizado de máquina como *weight decay*. O conceito de regularização abrange também outros algoritmos de otimização. Dentre eles o *Pruning*⁴, OBD⁵, OBS⁶ dentro outros.

Na tese de pós-graduação do autor Medeiros[3] há a proposta de um novo método de poda no qual ele nomeou de CAPE. Esse método é implementado em redes MLP⁷ e tem por objetivo reduzir o custo computacional da inversão da matriz Hessiana, cálculo esse muito utilizado em outros algoritmos. Em seguida o autor implementa o algoritmo em um caso real de identificação de falhas em motores de indução para avaliar e comparar a performance em relação ao algoritmo OBS.

3 Metodologia

Estudos ao livro *Redes neurais artificiais: teoria e aplicações* [4] O efeito do lambda é deslocar a solução do mínimo local.

¹Problemas multiobjetivo: Problemas com mais de um objetivo.

²ELM: Extreme Learning Machines.

³RBF: Radial Basis Function.

⁴Pruning (poda) algoritmo muito utilizado em Arvore de decisao.

⁵OBD: Optimal Brain Damage.

⁶OBS: Optimal Brain Surgeon.

⁷MLP: Multilayer Perceptron.

4 Resultados

O artigo *Neural Networks Regularization With Graph-Based Local Resampling*[AlexAssis] se refere ao tratamento dos dados utilizando o Grafo de Gabriel[GabrielGraph1] antes do treinamento e inferência de redes neurais baseadas em perceptron com projeções randômicas (ELM). Método esse chamado de *Graph-based Local Resampling of perceptron-like neural networks with random projections (RN-ELM)*. O algoritmo consiste em tratar os dados geometricamente para o treinamento da rede neural. Problemas de classificação multi-objetivo⁸, quando aplicado esse método, possuem boa eficiência devido a complexidade da rede treinada ser reduzida quando os dados de entrada são tratados.

O artigo *Large Margin Gaussian Mixture Classifier With a Gabriel Graph Geometric Representation of Data Set Structure*[LuizBambirra] explicita o uso de Grafo de Gabriel para modelagem dos padrões de entrada nos problemas de classificação. O modelo é capaz de solucionar problemas de redes neurais que tem por objetivos encontrar uma solução ótima para maior capacidade de generalização e menor complexidade do algoritmo. Sendo assim problemas classificados como multiobjetivos. Problemas assim costumam ter não só uma solução ótima, mas um conjunto de soluções ótimas, denominado conjunto Pareto-Ótimas (PO). O algoritmo consiste nas etapas de construção geométrica do grafo sobre as amostras, remoção dos ruídos e, em seguida, detecção de bordas. Por fim, define-se a região de separação entre os conjuntos.

O artigo *Enhancing Performance of Gabriel Graph-Based Classifiers by a Hardware Co-Processor for Embedded System Applications*[HardwareGabrielGraph] tem uma abordagem similar ao objetivo desse trabalho. Nele foi analisada a performance de um classificador utilizando Grafo de Gabriel implementado em Hardware Embarcado. A aplicação foi implementada em um FPGA⁹, onde se é capaz de reorganizar blocos lógicos de processamento de forma a obter um Hardware dedicado para aquela tarefa. Como a implementação de Redes Neurais requer elevado cálculo matricial e paralelo, a implementação

em Hardware dedicado para tal modelo apresenta boa performance quando se comparado à implementação em software de alto nível. A linguagem de descrição de Hardware utilizada para programação do FPGA foi VHDL.¹⁰

Esse artigo se difere com projeto desejado no que se diz ao Hardware escolhido. No artigo foi-se utilizado um hardware capaz de processar a rede de forma mais performática. No projeto proposto nesse trabalho o hardware utilizado será de Arquitetura ARM¹¹ com baixa capacidade de processamento. Por esse motivo, o objetivo não é ter um resultado melhor do que os implementados em computadores pessoais mas sim em obter um resultado aceitável para a utilização em uma aplicação distribuída de baixo custo.

Por fim, foi estudado diversos projetos utilizando o microcontrolador ESP32 e algoritmos de predição utilizando Redes Neurais Artificiais. Portanto, creio ser um projeto realizável.

5 Discussões

Estudos ao livro *Redes neurais artificiais: teoria e aplicações* [4] foram realizados de forma a gerar maior compreensão sobre o assunto. A princípio, foi estudado o primeiro modelo matemático de neurônios artificiais, o neurônio MCP¹². Em seguida, implementados na linguagem python o modelo de neurônio de aproximação linear Adaline¹³ e de um modelo de Adaline com uma função de ativação, tornando um neurônio classificador com melhores resultados quando comparado ao MCP, o neurônio de classificação Perceptron Simples.

Entender o funcionamento do modelo computacional de um neurônio permite também melhor compreensão dos estudos de redes de neurônios. Para a aplicação a ser desenvolvida a ideia inicial é de detecção facial por meio de reconhecimento de imagens. Portanto, estudos sobre redes neurais convolucionais (CNN)¹⁴ estão sendo feitos por se tratar de um modelo de rede utilizado para tratamento de imagens.

6 Conclusões

Com o objetivo de concretizar estudos nas áreas de Redes Neurais Artificiais e em Arquitetura e Organização de Computadores esse projeto final de curso tem por finalidade implementar uma rede neural em um módulo microcontrolador ESP32 de arquitetura ARM.

⁸Problemas multiobjetivo: Problemas com mais de um objetivo.

⁹FPGA: Field-programmable gate array.

¹⁰VHDL: VHSIC Hardware Description Language..

¹¹ARM is a family of reduced instruction set computer (RISC) instruction set architectures for computer processors.

¹²MCP: McCulloch-Pitts Neuron Mankind's First Mathematical Model Of A Biological Neuron..

¹³Adaline: Adaptive Linear Neuron or later Adaptive Linear Element.

¹⁴CNN: Convolutional Neural Networks.

Referências

- [1] Douglas M. Hawkins. “The Problem of Overfitting”. Em: *Journal of Chemical Information and Computer Sciences* 44.1 (2004). PMID: 14741005, pp. 1–12. DOI: [10.1021/ci0342472](https://doi.org/10.1021/ci0342472). eprint: <https://doi.org/10.1021/ci0342472>. URL: <https://doi.org/10.1021/ci0342472>.
- [2] R. McElreath. *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*. Chapman & Hall/CRC Texts in Statistical Science. CRC Press, 2016. ISBN: 9781482253467. URL: <https://books.google.com.br/books?id=mDo0CwAAQBAJ>.
- [3] Cláudio Marques de Sá Medeiros. “A contribution to the problem of selection of neural models using the beginning of maximum correlation of the errors.” Em: *Tese (Doutorado em Engenharia de Teleinformática)-Centro de Tecnologia, Universidade Federal do Ceará, Fortaleza* (2008). URL: <http://www.repositorio.ufc.br/handle/riufc/16117>.
- [4] A. de Pádua Braga, A.C.P. de Leon Ferreira Carvalho e T.B. Ludermit. *Redes neurais artificiais: teoria e aplicações*. Livros Técnicos e Científicos, 2000. ISBN: 9788521612186. URL: <https://books.google.com.br/books?id=cUgEaAEACAAJ>.