

Classificação de Eletrocardiograma utilizando Redes Neurais

Prabhát Kumar de Oliveira¹; Adhvan Novais Furtado²; Roberto José da Silva Badaró³; Erick Giovanni Sperandio Nascimento⁴

1 Bolsista; Implantação de infraestrutura de pesquisa em simulação e modelagem computacional no estado da Bahia utilizando processamento de alto desempenho - FAPESB ; bartkoliveira@gmail.com

2 Doutorando em Modelagem Computacional e Tecnologias Industriais; Centro Universitário SENAI CIMATEC; Salvador-BA; adhvan@fieb.org.br

3 Doutorado; Centro Universitário SENAI CIMATEC; Salvador-BA; badaro@fieb.org.br

4 Doutorado; Centro Universitário SENAI CIMATEC; Salvador-BA; erick.sperandio@fieb.org.br

RESUMO

O eletrocardiograma (ECG) é uma ferramenta médica não invasiva que exhibe o ritmo e o status do coração. Através da análise dos sinais captados pelo ECG é possível identificar alterações associadas a patologias cardíacas. Uma solução computacional que realize a detecção de alterações nestes sinais é um passo importante para a aceleração da emissão de laudos, podendo classificar casos que demandem atenção urgente e aumentar a confiabilidade do diagnóstico, em especial em locais onde um médico especialista não possa ser encontrado rapidamente. Este trabalho possui o objetivo de realizar um estudo sobre redes neurais para resolver o problema de classificação (normal e anormal). São abordadas redes *Multi Layer Perceptron* (MLP) e *Convolutional Neural Networks* (CNN) para verificar a precisão do diagnóstico baseado em uma base de dados conhecida. É apresentado um resultado preliminar obtido através destes estudos, demonstrando o quanto promissor pode ser abordar esse tipo de problema com o auxílio de redes neurais artificiais.

PALAVRAS-CHAVE: eletrocardiograma, inteligência artificial, classificação

1. INTRODUÇÃO

O suporte computacional ao diagnóstico de patologias a partir de séries temporais de sinais fisiológicos como o ECG é um campo de pesquisa bastante explorado ¹. Dentre as muitas técnicas de processamento de sinais, as redes neurais em geral e, em especial, as CNNs e as redes neurais recorrentes (RNN) têm sido recentemente utilizadas com sucesso para classificação e predição de séries temporais ^{2,3,4}.

O objeto deste relatório é demonstrar os primeiros esforços para a utilização de redes neurais para a classificação de exames de eletrocardiograma apresentando a implementação de dois tipos de redes neurais. O código foi desenvolvido utilizando a linguagem Python e bibliotecas gratuitas. As redes neurais foram treinadas para classificação de exames como normais ou anormais.

2. METODOLOGIA

As redes neurais são modelos computacionais inspirados na estrutura neuronal do cérebro. Em um modelo simplificado do cérebro, estruturas computacionais básicas, neurônios, são conectados entre si através de uma rede de comunicação, através da qual o cérebro é capaz de realizar processamentos complexos. Redes neurais artificiais exploram este paradigma para gerar estruturas computacionais. Comumente, a estrutura de uma rede neural artificial é definida como um grafo cujos nós são os neurônios e cada aresta dirigida interliga a saída de algum neurônio à entrada de outro neurônio.

Para o desenvolvimento dos experimentos foram realizadas implementações de duas redes neurais na linguagem Python. Uma rede neural MLP com uma camada oculta (*hidden layer*) e um rede neural CNN com uma dimensão. O objetivo de ambas é classificar os registros de entrada, contendo uma amostra de uma derivação de um ECG, entre normal ou anormal. Como resultado, já que codificamos normais e anormais como 1 e 0, respectivamente, os algoritmos recuperam a probabilidade de um ECG normal.

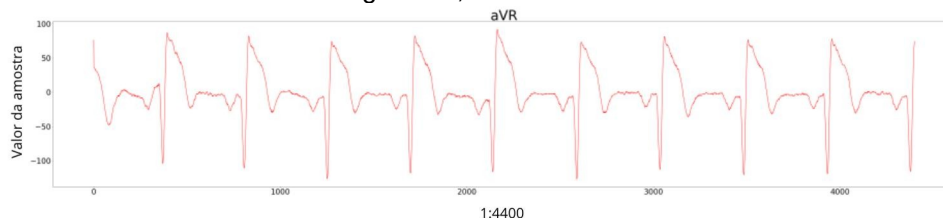
Os códigos utilizam a biblioteca Keras ⁵, amplamente difundida para acelerar o desenvolvimento de aplicações de aprendizagem de máquina. O *framework* de treinamento Keras tem três estágios principais, o primeiro para especificar a arquitetura, o segundo para compilar o modelo, e um estágio para ajustar o modelo

determinando o número de épocas, o tamanho do lote e a divisão da validação.

Preparação da base de dados

Como dados de entrada foram utilizados exames cedidos por uma empresa de diagnóstico à distância. Os dados foram fornecidos em dois formatos, laudos de exames com o diagnóstico descrito por um médico especialista em um arquivo PDF e dados brutos do paciente, incluindo os sinais de 12 derivações medidas por um ECG, em um arquivo WXML. Após anonimizados, os dados foram pré-processados e convertidos para um formato CSV (*dataset*). Este formato tornou mais prática sua manipulação.

Figura 1 - Representação gráfica de 1 derivação de um paciente do *dataset*. O gráfico é representado pelo valor de cada amostra registrada, em um total de 4400 amostras.



O *dataset* utilizado neste trabalho representa a compilação de 404 arquivos de exames. Cada exame está associado a somente um paciente e possui sinais de 12 derivações medidos a uma taxa de 600 amostras por segundo. Cada derivação possui uma série com 4400 amostras. Os dados foram segmentados de tal forma que cada registro estivesse associado à medição de uma derivação e a um diagnóstico. Isso significa que exames com somente um diagnóstico e doze derivações geraram doze entradas no arquivo final. Exames com múltiplos diagnósticos geraram múltiplos registros associados a cada derivação.

Com o pré-processamento foi gerado um arquivo com 16.869 registros, cada registro é composto por 4401 colunas, a primeira sendo a classificação (normal ou anormal) e as 4400 seguintes, os valores registrados das amostras do ECG para a derivação em questão. Ao final os registros foram embaralhados aleatoriamente para evitar algum viés (*bias*) imposto pelas rotinas de pré-processamento. Uma vez estruturada a massa de dados de entrada, foram definidos os atributos iniciais das redes neurais. Foram estabelecidos os conjuntos de treinamento/validação e de teste como 80% e 20% da amostra, respectivamente.

3. RESULTADOS E DISCUSSÃO

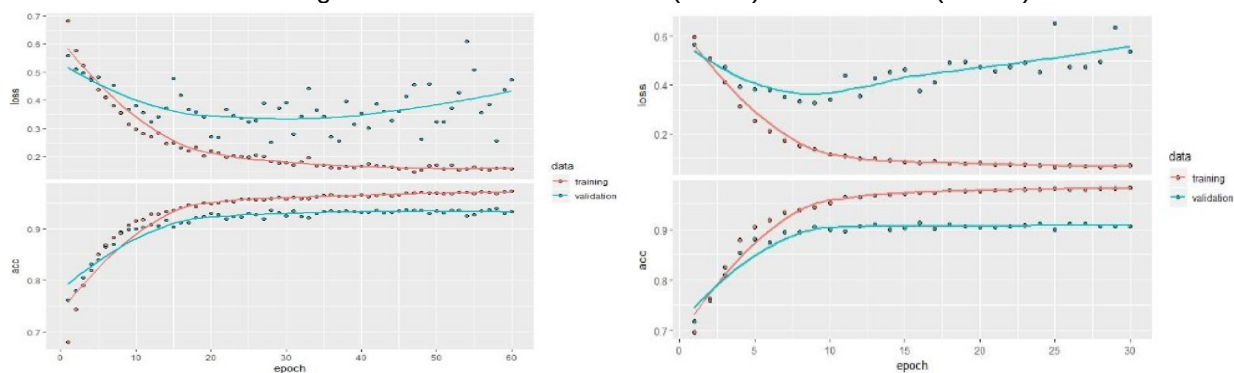
A rede neural MLP foi configurada com 3 camadas, utilizando os seguintes parâmetros: 80 neurônios na camada de entrada, 40 na camada intermediária e 2 na camada de saída. As duas primeiras camadas possuem "Relu" como função de ativação, e a última utiliza a função de ativação "softmax".

Entre as camadas 1 e 2, e 2 e 3, existem taxas de dropout, de 0.4 e 0.3, respectivamente, o *dropout* é uma técnica de regularização utilizada para reduzir o sobreajuste (*overfitting*) da rede neural. *Overfitting* é um modelo estatístico que indica que a rede aprendeu demais sobre os dados, de forma a decorá-los. Já a função de ativação serve para o reajuste dos pesos durante o processo de retropropagação de erro (*backpropagation*). A função *relu* é uma das mais utilizadas em aprendizado profundo⁶. A função de ativação softmax é utilizada para o cômputo da probabilidade das classes através da função de distribuição, sendo nesse caso, o resultado, uma probabilidade de pertencer à classe 1 (normal) ou 0 (anormal).

O modelo foi compilado utilizando a função '*categorical_crossentropy*' que mapeia os valores de uma ou mais variáveis em um intervalo de ponto flutuante. Ela foi utilizada pois mede o desempenho de um modelo cuja a saída está em um intervalo [0, 1]. A função de perda aumenta à medida que o valor medido diverge do real.

Na rede MLP, a função de *loss* obtida foi de 0.4669437 e a acurácia de 0.9269015, já para a rede CNN a acurácia obtida foi de 0,905 e o *loss* em 0,599. A acurácia é o quanto a rede neural acerta em relação aos dados reais, já a função de *loss* é a perda que a rede tem durante o treinamento. Em ambos os casos, nota-se que os valores de treinamento se divergem dos valores de validação, demonstrando que a rede sofreu *overfitting*.

Figura 2 - A esquerda pode ser visto os gráficos de loss da rede MLP (acima) e gráfico de acurácia (abaixo), e a direita os gráficos de loss da rede CNN (acima) e de acurácia (abaixo).



4. CONSIDERAÇÕES FINAIS

O objetivo deste trabalho foi demonstrar um método de classificação automática de exames de ECG utilizando redes neurais. A melhor acurácia obtida, aproximadamente, foi de 0.927 e a *loss* 0.467. É um resultado promissor, mas ainda está longe de um algoritmo ótimo. Existem muitas frentes de atuação para melhorar os resultados obtidos neste trabalho, como testes com redes RNN, LSTM, entre outras, assim como melhorar a preparação dos dados, trazendo mais elementos do domínio do problema. Desta forma, atributos relevantes que influenciam o comportamento fisiológico do coração, como frequência cardíaca, idade e sexo do paciente deverão ser inseridos nos modelos.

Neste trabalho foi tratado de forma equivalente todas as derivações de todos os pacientes. Modificar a modelagem agrupando todas as derivações de um paciente em uma mesma entrada pode melhorar os resultados, uma vez que todos os sinais representam medidas em posições diferentes, mas no mesmo período de tempo.

Um ponto fundamental para o treinamento de redes neurais, em especial as redes neurais profundas é a existência de uma grande base de dados para treinar o modelo. O modelo utilizado tinha apenas 404 exames. Acessar uma base maior ou utilizar *data augmentation* é mandatório para melhorar o desempenho do algoritmo. A técnica de *data augmentation* consiste em modificar características do *dataset* original, como aplicar rotação, translação, inversão, escalonamento, entre outros, de forma a criar novas características, ampliando o tamanho do *dataset*.

Agradecimentos

Ao Centro de Supercomputação para Inovação Industrial do SENAI CIMATEC e à FAPESB.

5. REFERÊNCIAS

- ¹ Schläpfer, Jürg; Wellens, Hein J, Computer-Interpreted Electrocardiograms: Benefits and Limitations. Journal of the American College of Cardiology, ISSN: 1558-3597, Vol: 70, Issue: 9, Page: 1183-1192, 2017
- ² Weems A, Harding M and Choi A 2016 Classification of the ECG Signal Using Artificial Neural Network Proceedings of the 3rd International Conference on Intelligent Technologies and Engineering Systems (ICITES2014) p 545–555
- ³ J. T. Connor, D. Martin, and L. E. Atlas. Recurrent neural networks and robust time series prediction. IEEE Transactions on Neural Networks, 5(2):240–254, 1994
- ⁴ Hüsken, Michael and Peter Stagge. "Recurrent neural networks for time series classification." Neurocomputing 50 (2003): 223-235
- ⁵ François Chollet at al. Keras, <https://keras.io>, 2015
- ⁶ Ramachandran, Prajit; Barret, Zoph; Quoc, V. Le "Searching for Activation Functions". arXiv:1710.05941 2017