Biblioteca em Python para extração e análise de parâmetros a partir de sinais eletrocardiográficos

ANA BEATRIZ S. N. R. OLIVEIRA (IFPB, Campus João Pessoa), JOSÉ RAIMUNDO BARBOSA (IFPB, Campus João Pessoa), RAFAEL DUARTE DE SOUSA (IFPB, Campus João Pessoa), CARLOS DANILO M. REGIS (IFPB, Campus João Pessoa)

E-mails: beatriz.nogueira@academico.ifpb.edu.br, jsraimundob@gmail.com, rafaelds57@gmail.com, danilo.regis@ifpb.edu.br.
Área de conhecimento:(Tabela CNPq): Engenharias IV

Palavras-Chave: eletrocardiograma; vetocardiograma; reconstrução do espaço de fases; python; biblioteca.

1 Introdução

De acordo com a Organização Mundial da Saúde, as doenças cardiovasculares representam a maior causa de mortes no mundo. Em escala nacional o cenário não é diferente, segundo o Ministério da Saúde esse tipo de doença é responsável por quase um terço dos óbitos no Brasil (SAUDE, 2019). Muitas dessas mortes poderiam ser postergadas e até evitadas, para que isso aconteça é fundamental o aumento das possibilidades de diagnóstico e tratamento, bem como da disponibilidade de serviços para a prevenção.

O Eletrocardiograma (ECG) é uma das formas mais tradicionais de diagnóstico para doenças cardiovasculares. No ECG é avaliada a atividade elétrica do coração através de eletrodos conectados ao corpo do paciente para registrar as variações dos potenciais elétricos do coração. Assim, o cardiologista interpreta as ondas (derivações) obtidas no exame, a fim de identificar a ocorrência de uma suposta enfermidade. Outro método utilizado é o Vetocardiograma (VCG), que é um registro tridimensional da atividade elétrica do coração. Para obter-se o VCG, um dos métodos mais utilizados é o da Regressão Linear de Kors, que por meio de uma combinação linear das derivações de um ECG é possível gerar o VCG (KORS et al., 1990).

Trabalhos anteriores que tratam sobre a aplicação de redes neurais para detecção de patologias cardíacas utilizando o método de reconstrução do espaço de fases e extraindo parâmetros que são relacionados a caoticidade do sistema, obtiveram resultados positivos (ROOPAEI et al., 2010).

Tendo isso em vista, esse trabalho descreve brevemente a criação de uma biblioteca em Python com funções para a leitura de ECGs, geração de VCGs, reconstrução do espaço de fases, visualização, armazenamento e o cálculo de parâmetros associados a esses sinais.

2 Materiais e Métodos

2.1 Aquisição dos Arquivos

As bases de dados que são utilizadas em centros de pesquisa cardíaca normalmente são padronizadas, facilitando seu processamento. Porém as diferentes bases possuem diferentes formas de organização dos dados e armazenamento dos sinais. Os sinais de ECG consistem em uma matriz contendo os valores das derivações em certos instantes de tempo, de forma que cada derivação corresponde a uma série temporal. Para o presente trabalho foi utilizada a base do Instituto de Metrologia da Alemanha PTB-XL (WAGNER et al., 2020).

2.2 Classes da Biblioteca

Após a definição dos dados a serem utilizados, foram definidas as classes que formam a biblioteca. Na seções abaixo são apresentadas as classes, seus atributos e métodos desenvolvidos.

2.2.1 Classe GPDSHelper

A classe GPDSHelper possui funções importantes para o funcionamento da biblioteca. Ela é responsável pela leitura dos arquivos que contém os sinais de ECG através do método *read_file*, a qual recebe como argumentos o caminho e o tipo do arquivo de ECG. Essa classe também é responsável por reconstruir o VCG a partir das 12 derivações do ECG utilizando o método *ecg to vcg*. Por último, a classe também converte um objeto ECG em um arquivo .CSV pelo método *ecg_to_csv*.

2.2.2 Classe *ECG*

Na classe ECG é possível agrupar os atributos do ECG, assim como as operações que serão feitas nesse sinal. Os atributos dessa classe são: as 12 derivações de um ECG, o nome do arquivo de origem do sinal, a frequência de amostragem e as 12 derivações após algum processo de filtragem, caso seja necessário. Seus métodos são a filtragem das 12 derivações e a exibição delas ou de apenas uma derivação de interesse, definida pelo usuário.

2.2.3 Classe *VCG*

A classe VCG possui métodos que exibem os sinais de VCG em três dimensões ou em seus planos ortogonais a partir dos seus atributos, sendo eles: as três derivações reconstruídas, o método de reconstrução do espaço de fases, a frequência de amostragem do sinal de ECG utilizado na reconstrução e o nome do arquivo. Para que isso seja possível foram desenvolvidos os métodos *plot3D* que mostra uma representação tridimensional do VCG onde a análise pode ser feita em qualquer perspectiva, e *plot_planes* que mostra uma representação dos planos ortogonais do VCG.

Pode ser reconstruída também a representação do espaço de fases de uma derivação de VCG. A reconstrução pode ocorrrer utilizando um valor de τ definido pelo usuário ou um valor de τ ótimo, sendo ele o primeiro mínimo local da curva de informação mútua (FRASER; SWINNEY, 1986). Com o valor de τ, é obtida uma versão do sinal com esse valor de atraso, gerando um gráfico, que pode ser apenas exibido ou salvo para a aplicação das métricas na avaliação do espaço de fases.

2.2.4 Classe *Metrics*

Nessa classe foram definidos os atributos *image1*, *image2*, que representam as imagens binarizadas dos espaços de fases, *blocksize*, que contém o tamanho do bloco e dois atributos contendo os caminhos para as imagens originais, *imgPath1* e *imgPath2*. Dentros os métodos implementados está o *convert2binary*, que binariza a imagem do espaço de fases por meio da sua conversão para escala de cinza e binarização pelo método de Otsu, onde os pixels que não recebem a trajetória do espaço de fases recebem o valor de 255 correspondente a cor branca, e os pixels que recebem a trajetória do espaço de fases recebem o valor de 0. Após isso são aplicados os métodos para o cálculo das métricas descritas por Roopaei (ROOPAEI et al., 2010), os quais correspondem na classe aos métodos *count_method*, *dif_method*, *sim_method* e *pond_method*.

3 Resultados e Discussão

Dessa maneira, os arquivos de sinais descritos em Wagner et al. (2020), foram organizados de forma matricial, de modo que contenha as 12 derivações e seu cabeçalho (frequência de amostragem do sinal, sua quantidade de amostras, o número de derivações, as unidades de medida para cada derivação, o ordenamento das 12 derivações e os comentários acerca do sinal).

Após isso, utilizando funções da classe *GPDSHelper* foi realizada a leitura dos arquivos e também a conversão e reconstrução do VCG a partir das 12 derivações do sinal de ECG. Com funções da classe *VCG* foi

possível obter a representação do espaço de fases gerado e em seguida, utilizando a classe *Metrics*, extrair seus parâmetros.

A Figura 1 demonstra o uso das classes mencionadas anteriormente e suas funções, onde foi utilizado um arquivo com os sinais e, a partir dele, foi possível observar os valores dos parâmetros obtidos.

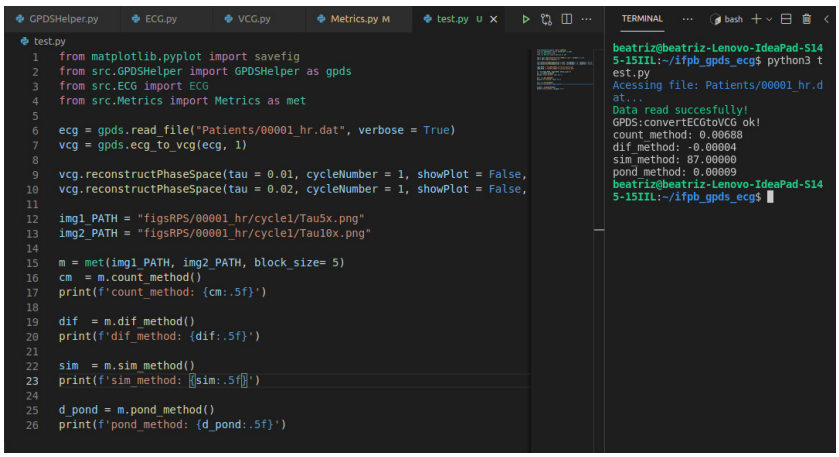

Figura 1: Exemplo de código utilizando as funções da biblioteca.

4 Considerações Finais

O presente trabalho teve como objetivo a criação de uma biblioteca em Python com funções que realizasse a leitura de ECGs, geração de VCG, reconstrução do espaço de fases, visualização, armazenamento e cálculo de parâmetros. Sendo assim, após a análise dos resultados, pode-se concluir que os objetivos foram alcançados, de modo que é possível a utilização da biblioteca proposta em trabalhos diversos, como por exemplo a aplicação de redes neurais associada a técnica de reconstrução do espaço de fases para identificar possíveis patologias cardíacas.

Agradecimentos

Agradecemos ao Instituto Federal da Paraíba (IFPB) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pela estrutura e apoio necessário para a realização da pesquisa.

Referências

FRASER, A. M.; SWINNEY, H. L. Independent coordinates for strange attractors from mutual information. *Phys. Rev. A*, American Physical Society, v. 33, p. 1134–1140, Feb 1986. Disponível em: <https://link.aps.org/doi/10.1103/PhysRevA.33.1134>.

KORS, J. A. et al. Reconstruction of the frank vectorcardiogram from standard electrocardiographic leads: diagnostic comparison of different methods. *European Heart Journal*, v. 11, n. 12, p. 1083 – 1092, 1990.

ROOPAEI, M. et al. Chaotic based reconstructed phase space features for detecting ventricular fibrillation. *Biomedical Signal Processing and Control*, Elsevier, v. 5, n. 4, p. 318 – 327, 2010.

SAUDE, M. D. *Saúde Brasil 2018. Uma análise da situação de saúde e das doenças e agravos crônicos: desafios e perspectivas.* [S.l.], 2019. Disponível em: <https://bvsms.saude.gov.br/bvs/publicacoes/saude_brasil_2018_analise_situacao_saude_doencas_agravos_cronicos_desafios_perspectivas.pdf>. Acesso em: 19 Julho. 2021.

WAGNER, P. et al. Ptb-xl, a large publicly available electrocardiography dataset. *Scientific Data*, Nature Publishing Group, v. 7, n. 1, p. 1–15, 2020.