

UNIVERSIDADE DE BRASÍLIA

DISCIPLINA: TÓPICOS EM ENGENHARIA - PROCESSAMENTO DE SINAIS BIOMÉDICOS

TURMA: A

SEMESTRE: 2021/02

GRUPO: 9

DATAS DE REALIZAÇÃO: 28/01/2022

Membros:

Luís Eduardo Ribeiro Guerra 190091924

Guilherme Carvalho de Olivera Martins 110119801

Tiago Rodrigues dos Santos 150022689

Atividade 1

Gráficos Paciente Idoso

Código: f2o05m

Intervalo: 5 minutos

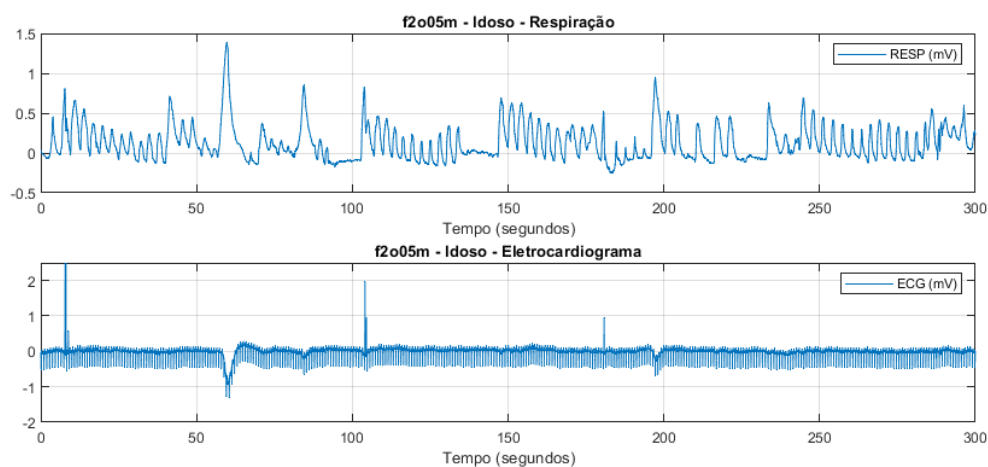
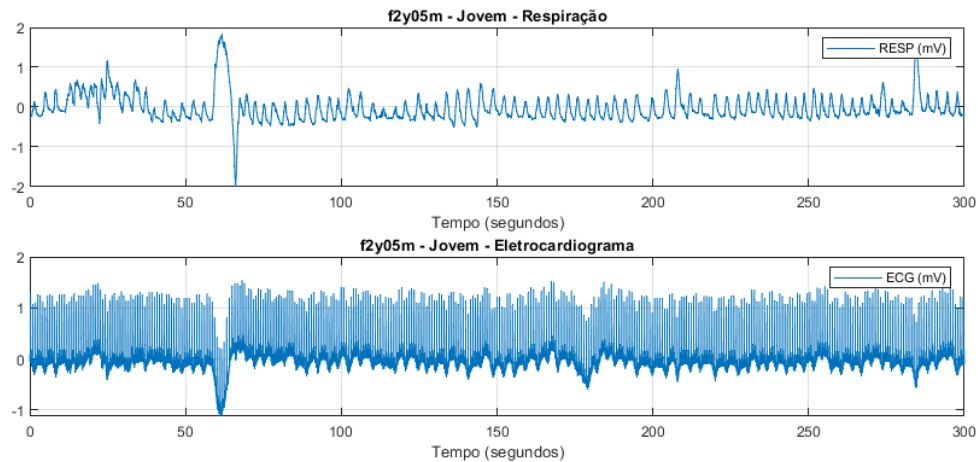


Gráfico Paciente Jovem

Código: f2y05m

Intervalo: 5 minutos



Artigo 1

Artigo: Shipra Saraswat, Geetika Srivastava, Sachida Nand Shukla.

Malignant Ventricular Ectopy Classification using Wavelet Transformation and Probabilistic Neural Network Classifier, 2016.

Link:

<https://sciresol.s3.us-east-2.amazonaws.com/IJST/Articles/2016/Issue-40/Article2.pdf>

A) Comentário do autor sobre a base de dados: Foi utilizado dados de pacientes saudáveis da bases de dados fantasia e foi utilizado outras bases de dados para pacientes não saudáveis.

B) Como foi utilizado: Os dados foram utilizados para treinar e testar um método para avaliar eletrocardiogramas não saudáveis utilizando rede neural probabilística utilizando a transformada wavelet discreta. O método atingiu acurácia de 97,22% enquanto outro método chamado de Cardioverter Defibrillators atingiu somente 60,3% de acurácia.

Artigo 2

Artigo: Horia Mihail Teodorescu. A COLLECTIVE BIOLOGICAL PROCESSING ALGORITHM FOR ECG SIGNALS, 2011.

Link:

https://cs.harvard.edu/malan/publications/HSA_Biosignal_paper_v3BB.pdf

A) Comentário do autor sobre a base de dados: Os dados do banco de dados fantasia foram utilizados para testar a filtragem HS de ruídos por eles serem considerados com ruídos de um cenário real, ao mesmo tempo foi utilizado dados com ruído artificial introduzido para testar a capacidade de filtragem.

B) Como foi utilizado: Os dados foram utilizados para a testagem de algoritmos de filtragem ruídos, como simulação do cenário real.

Código fonte da atividade:

<https://github.com/luisrguerra/processamentosinaisbiomedicos2022unb>

Código utilizado na atividade (atividade1.m):

```
plotATM("f2y05m", "Jovem",5); %f2y05m paciente escolhido
pause(1.3); %Pausa para mostrar que existe duas janelas uma em cima da outra
plotATM("f2o05m", "Idoso",5); %f2o05m paciente escolhido

function plotATM(Arquivo,Title,tempoMinutos)

% usage: plotATM('RECORDm')
%
% This function reads a pair of files (RECORDm.mat and RECORDm.info) generated
% by 'wfdb2mat' from a PhysioBank record, baseline-corrects and scales the time
% series contained in the .mat file, and plots them. The baseline-corrected
% and scaled time series are the rows of matrix 'val', and each
% column contains simultaneous samples of each time series.
%
% 'wfdb2mat' is part of the open-source WFDB Software Package available at
% http://physionet.org/physiotools/wfdb.shtml
% If you have installed a working copy of 'wfdb2mat', run a shell command
% such as
% wfdb2mat -r 100s -f 0 -t 10 >100sm.info
% to create a pair of files ('100sm.mat', '100sm.info') that can be read
% by this function.
%
```

```
% The files needed by this function can also be produced by the
% PhysioBank ATM, at
% http://physionet.org/cgi-bin/ATM
%
```

```
% plotATM.m      O. Abdala      16 March 2009
%                James Hislop   27 January 2014   version 1.1
```

```
Name = strcat("dados\", Arquivo); %Adiciona o endereço do diretórios dos dados dos
cardiogramas
```

```
infoName = strcat(Name, '.info');
```

```
matName = strcat(Name, '.mat');
```

```
Octave = exist('OCTAVE_VERSION');
```

```
load(matName);
```

```
fid = fopen(infoName, 'rt');
```

```
fgetl(fid);
```

```
fgetl(fid);
```

```
fgetl(fid);
```

```
[freqint] = sscanf(fgetl(fid), 'Sampling frequency: %f Hz Sampling interval: %f sec');
```

```
interval = freqint(2);
```

```
fgetl(fid);
```

```
if(Octave)
```

```
    for i = 1:size(val, 1)
```

```
        R = strsplit(fgetl(fid), char(9));
```

```
        signal{i} = R{2};
```

```
        gain(i) = str2num(R{3});
```

```
        base(i) = str2num(R{4});
```

```
        units{i} = R{5};
```

```
    end
```

```
else
```

```
    for i = 1:size(val, 1)
```

```
        [row(i), signal(i), gain(i), base(i),
```

```
units(i)]=strread(fgetl(fid),'%d%s%f%f%s','delimiter','\t');
```

```
    end
```

```
end
```

```
fclose(fid);
```

```
val(val== -32768) = NaN;
```

```
for i = 1:size(val, 1)
```

```
    val(i, :) = (val(i, :) - base(i)) / gain(i);
```

```
end
```

```
x = (1:size(val, 2)) * interval;
```

```

% Ajuste na posição e tamanho da janela
janela = figure('name',Name);
posicao_x = 400;
posicao_y = 300;
altura = 400;
comprimento = 1000;
janela.Position = [posicao_x posicao_y comprimento altura];

%Calculo para o intervalo de tempo
frequencia = 250;
tempo_x = tempoMinutos*60*frequencia;
%disp(tempo_x);

subplot(2,1,1);% Subplotar em cima
plot(x(1:tempo_x)', val(1, 1:tempo_x)); %Plotar aplicando o intervalo de tempo
title( strcat(Arquivo," - ",Title, ' - Respiração') );
legend( strcat(signal{1}, ' (', units{1}, ')') ); %Legenda da linha com unidade de medida
xlabel('Tempo (segundos)');
grid on; %Ativar linha no gráfico

subplot(2,1,2);% Subplotar em baixo
plot(x(1:tempo_x)', val(2, 1:tempo_x));%Plotar aplicando o intervalo de tempo
title( strcat(Arquivo," - ",Title,' - Eletrocardiograma') );
legend( strcat(signal{2}, ' (', units{2}, ')') ); %Legenda da linha com unidade de medida
xlabel('Tempo (segundos)');
grid on; %Ativar linha no gráfico

end % Fim da função plotATM()

```