ECG Signal Denoising

Anuvarshini SP
(CB.EN.U4AIE19011)
CSE-AI
Amrita Vishwa Vidyapeetam

P Rama Sailaja
(CB.EN.U4AIE19045)
CSE-AI
Amrita Vishwa Vidyapeetham

Chapter 1

Introduction

1.1 Electrocardiogram

An electrocardiogram (ECG) is a simple test that can be used to check a heart's rhythm and electrical activity.It can be used to investigate symptoms of a possible heart problem, such as chest pain, palpitations, dizziness and shortness of breath.Cardiac diseases have a wide variety of patterns, which change drastically with respect to the individual and are the leading cause of death globally.ECG test is the most reliable procedure for the identification of these changes in the diseases. The distinct features of the cardiovascular diseases are captured in the P-wave, which represents the depolarisation of atria, QRS complex, which represents the depolarisation of the ventricles, and T-wave, which represents repolarisation of ventricles, components of the ECG signal. ECG can capture all abnormalities that can take place in the cardiac activities. It does so by recording the electrical impulses but it is susceptible to noises.The electrical impulses leave the trace of the abnormalities in its ECG structure. The classification of this type of ECG can be difficult. So, for the proper diagnosis of ECG such types of noises should be removed. ECG signal denoising is a major pre-processing step which attenuates the noises and accentuates the typical wave in an ECG signal.

Chapter 2

Objective Theory

2.1 Least Square Weighted Regularisation

$$\min_{x} \| Dx \|_2^2$$

while this part of the equation makes sure x is smooth so doing only that contains no signal information.
Adding both will make sure that the output signal is similar to input signal and at the same time smoothing it by removing the noise.

The smoothness of a signal can be measured by a second-order differential matrix. The smoothing effect is captured in the Second Order Differential Matrix because the signal has to be smoothed in order to remove the variations in it due to the noise. If we go for higher order Matrix, the smoothing effect gets higher and higher and loses its original signal information. The smoothing effect of third order is analyzed in the result part.]The objective of this project is to perform - ecg signal denoising based on least square weighted regularization approach. The idea is to obtain a signal similar to the noisy one, but smoother.

$$\min_{x} \left(\| \mathrm{y} - \mathrm{x} \|_2^2 + \lambda \| Dx \|_2^2 \right)$$

where y is the noisy signal and x is the output signal that is smoothened and λ is the control parameter. The effect of λ on smoothening the signal is analyzed in the result part.

$$\min_x \| y - x \|_2^2$$

This part of the equation makes sure output signal x is similar to input signal y so doing only that, x might contain some noise,

$$\min_x \| Dx \|_2^2$$

while this part of the equation makes sure x is smooth so doing only that contains no signal information.
Adding both will make sure that the output signal is similar to input signal and at the same time smoothing it by removing the noise.

The smoothness of a signal can be measured by a second-order differential matrix. The smoothing effect is captured in the Second Order Differential Matrix because the signal has to be smoothed in order to remove the variations in it due to the noise. If we go for higher order Matrix, the smoothing effect gets higher and higher and loses its original signal information. The smoothing effect of third order is analyzed in the result part.

Chapter 3

Methodology

3.1 Arriving to the required equations

Suppose there is a vector v of length N, the second order differential equation would be

$$v(i+1) - 2 * v(i) + v(i-1)$$

where i represents the index of the vector and ranges from [1:N-1] if the starting index is considered to be 0. Thus, the second-order differential matrix would be-

$$D = \begin{bmatrix} 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & & \ddots & \\ & & & 1 & -2 & 1 \end{bmatrix}$$

where 1,-2,1 are the coefficients and the dimension of the matrix D is of size (N-2,N).

Now formulating

$$\min_x \left(\| y - x \|_2^2 + \lambda \| Dx \|_2^2 \right)$$
$$(y-x)^t (y-x) + \lambda (Dx)^t (Dx)$$
$$y^t y - y^t x - x^t y + x^t x + \lambda x^t D^t Dx$$

Here the question is to minimize the noise of the signal which means their derivative should be equal to 0.

$$\frac{\partial(x,y)}{\partial x} = y^t y - y^t x - x^t y + x^t x + \lambda x^t D^t D x = 0$$

$$= 0 - y^t x - x^t y + x^t x + \lambda x^t D^T D x$$

since we know $x^t y = y^t x$ [a]

=> derivative of $y^t x = y,$ [b]

and derivative of $x^t x \Rightarrow 2x$ [c]

$$\Rightarrow -2y + 2x + \lambda x^t D^T D x = 0$$

[d] derivative of $\lambda x^t D^T D x$ will be $M = D^T D,\ x^t(M)x = \lambda 2Mx$

$$\Rightarrow -2y + 2x + 2\lambda D^T D x = 0$$

$$\Rightarrow -y + x + \lambda D^T D x = 0$$

$$\Rightarrow -y + x(I + \lambda D^T D) = 0$$

Minimising according to $x \Rightarrow x(I + \lambda D^T D) = y$

$$\min \| y - x \|_2^2 + \lambda \| Dx \|_2^2 \quad \Longrightarrow \quad x = (I + \lambda D^T D)^{-1} y$$

[a] If two vectors are column vectors then $A^T B = B^T A$.

[b] Suppose we have two column vectors A and X with

$$A=\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \text{ and } X=\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$A^T X = \begin{bmatrix} a_1 & a_2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \Rightarrow a_1 x_1 + a_2 x_2$$

$$\text{now } A^T X \Rightarrow \frac{\partial(A^T X)}{\partial X} = \begin{bmatrix} \frac{\partial(A^T X)}{\partial x_1} \\ \frac{\partial(A^T X)}{\partial x_2} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = A$$

$$\therefore A^T X = A$$

[c] $X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ thus $X^T X = x_1^2 + x_2^2$

$$\frac{\partial X^T X}{\partial X} = \begin{bmatrix} \frac{\partial(X^T X)}{\partial x_1} \\ \frac{\partial(X^T X)}{\partial x_2} \end{bmatrix} = 2\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 2X$$

[d] Suppose we have vector X, $X=\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

and M is positive symmetric definite matrix which means its eigen values are positive and its transpose is same as the original matrix

$$X^T MX = \begin{bmatrix} x_1 & x_2 \end{bmatrix}\begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$X^T MX = 3x_1^2 + 4x_1 x_2 + x_2^2$$

$$\frac{\partial(X^T MX)}{\partial X} = \begin{bmatrix} \frac{\partial(X^T MX)}{\partial x_1} \\ \frac{\partial(X^T MX)}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 6x_1 + 4x_2 \\ 4x_1 + 2x_2 \end{bmatrix} = \begin{bmatrix} 6 & 4 \\ 4 & 2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\Rightarrow 2\begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 2MX$$

$$\therefore X^T MX = 2MX$$

Chapter 4

Code

4.1 Using matrices of various ordered differential matrices

The code given in MATLAB is converted to Python

```
import math
import numpy as np
import matplotlib.pyplot as plt

signal=np.loadtxt("ecg.txt")
y1=signal[:,0]

#normalising to bring the values to a same range, here from [-1,0]
y2=((y1-max(y1)))/(max(y1)-min(y1))
#adding noise to the signal
noisy_signal=y2+ np.random.randn(len(y2))*0.01
noisy_signal.shape
#USING SECOND ORDER DIFFERENTIAL MATRIX
def leastSquareDenoising(noisy_signal):
    """Generates the denoised signal from the noisy signal
    Args:
        noisy signal => signal with noise added
    """
    N=len(noisy_signal)
    d_mat=np.eye(N-2,N) #Identity matrix of size (N-2,N)
    for i in range(0,N-2):
        d_mat[i,i+1]=-2 #1,-2,1 #SECOND ORDER DIFFERENTIAL MATRIX
        d_mat[i,i+2]=1
    I=np.eye(N,N) #identity matrix
    for lam in [1,10,100,1000,10000,100000,500000]:
        denoised=I+lam*(np.dot(np.transpose(d_mat),d_mat))
```

```
        #(I+lam*(d_mat^T*d_mat))*signal, denoised=NxN+NxN=NxN
        denoise_inv=np.linalg.inv(denoised)  #inverse
        denoised_signal=np.dot(denoise_inv, noisy_signal) #(NxN)*(Nx1)=Nx1
        plt.plot(denoised_signal)
        plt.title(f"Denoised Signal when Lamda = {lam} ")
        plt.show()
    return None
plt.plot(noisy_signal)
plt.title("Signal with Noise")
plt.show()
leastSquareDenoising(noisy_signal)

#USING FIRST ORDER DIFFERENTIAL MATRIX
def leastSquareDenoising1O(noisy_signal):
    N=len(noisy_signal)
    d_mat=np.eye(N-1,N)
    for i in range(0,N-1):
        d_mat[i,i+1]=-1 #1,-1
    I=np.eye(N,N) #identity matrix
    for lam in [1,10,100,1000,10000,100000]:
        denoised=I+lam*(np.dot(np.transpose(d_mat),d_mat))
        #inverse(I+lam*(d_mat^T*d_mat))*signal
        #denoised=NxN+NxN=NxN
        denoise_inv=np.linalg.inv(denoised) #inverse
        #(NxN)*(Nx1)=Nx1
        denoised_signal=np.dot(denoise_inv, noisy_signal)
        plt.plot(denoised_signal)
        plt.title(f"Denoised Signal when Lamda = {lam}")
        plt.show()
    return None
plt.plot(noisy_signal)
plt.title("Signal with Noise")
plt.show()
leastSquareDenoising1O(noisy_signal)


#USING THIRD ORDER DIFFERENTIAL MATRIX
def leastSquareDenoising3O(noisy_signal):
    N=len(noisy_signal)
    d_mat=np.eye(N-3,N)
    for i in range(0,N-3):
        d_mat[i,i+1]=-3 #1,-3,3,-1
        d_mat[i,i+2]=3
        d_mat[i,i+3]=-1
    I=np.eye(N,N) #identity matrix
    for lam in [1,10,100,1000,10000,100000,500000]:
```

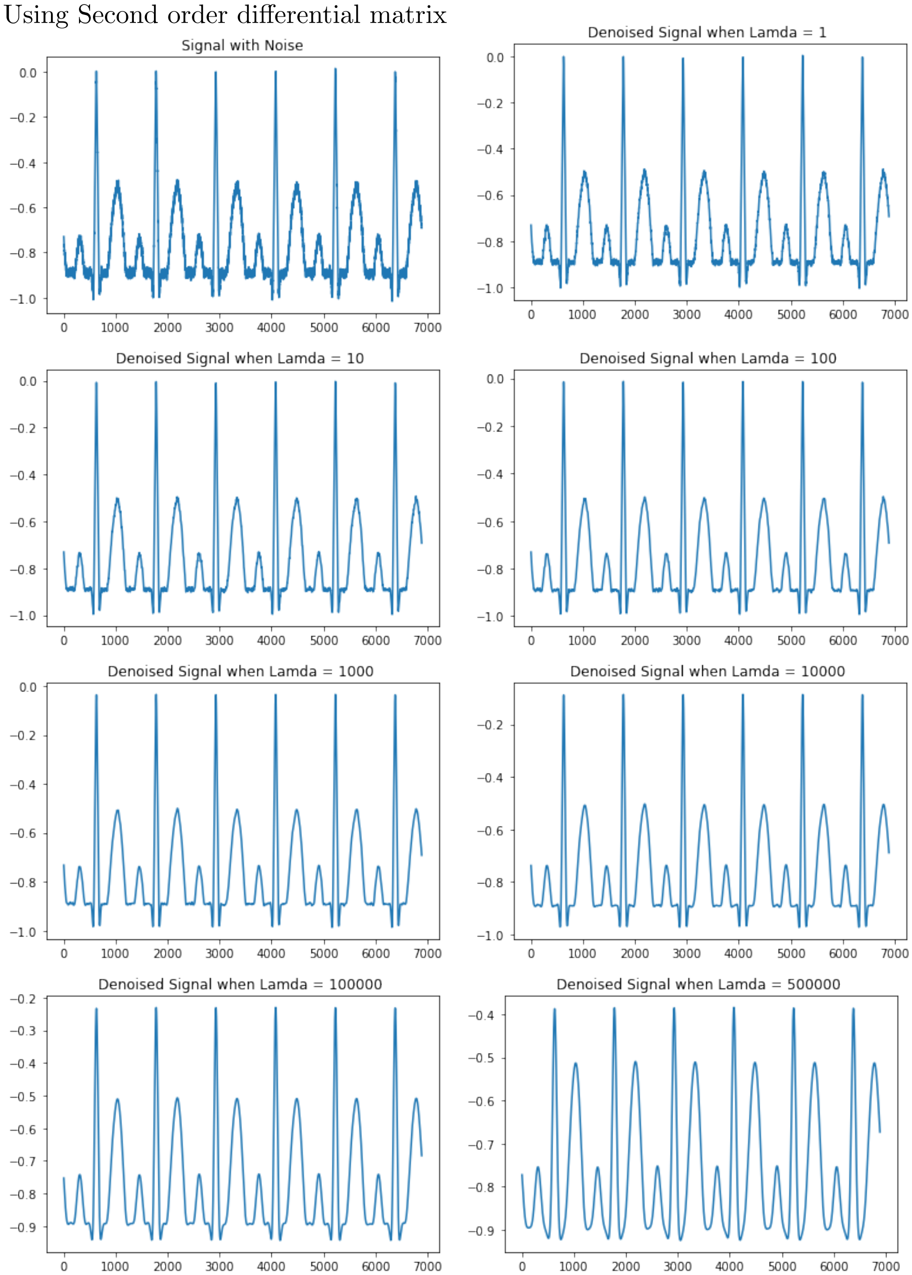
```
        denoised=I+lam*(np.dot(np.transpose(d_mat),d_mat))

        #inverse(I+lam*(d_mat^T*d_mat))*signal
        #denoised=NxN+NxN=NxN
        denoise_inv=np.linalg.inv(denoised)  #inverse
        #(NxN)*(Nx1)=Nx1
        denoised_signal=np.dot(denoise_inv, noisy_signal)
        plt.plot(denoised_signal)
        plt.title(f"Denoised Signal when Lamda = {lam}")
        plt.show()
    return None
plt.plot(noisy_signal)
plt.title("Signal with Noise")
plt.show()
leastSquareDenoising3O(noisy_signal)
```

Chapter 5

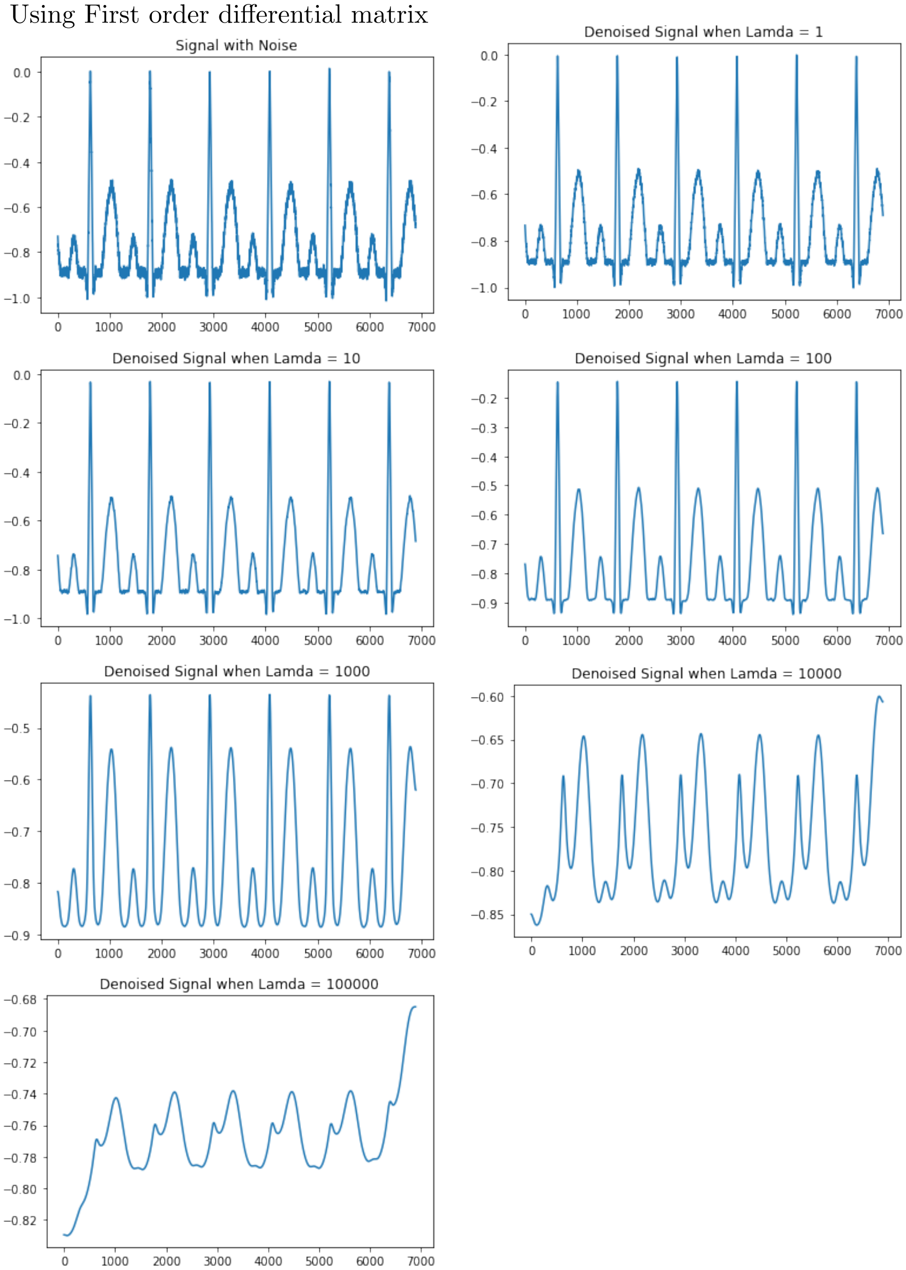
Output

5.1 Observation of signal behaviour at various values of Lambda

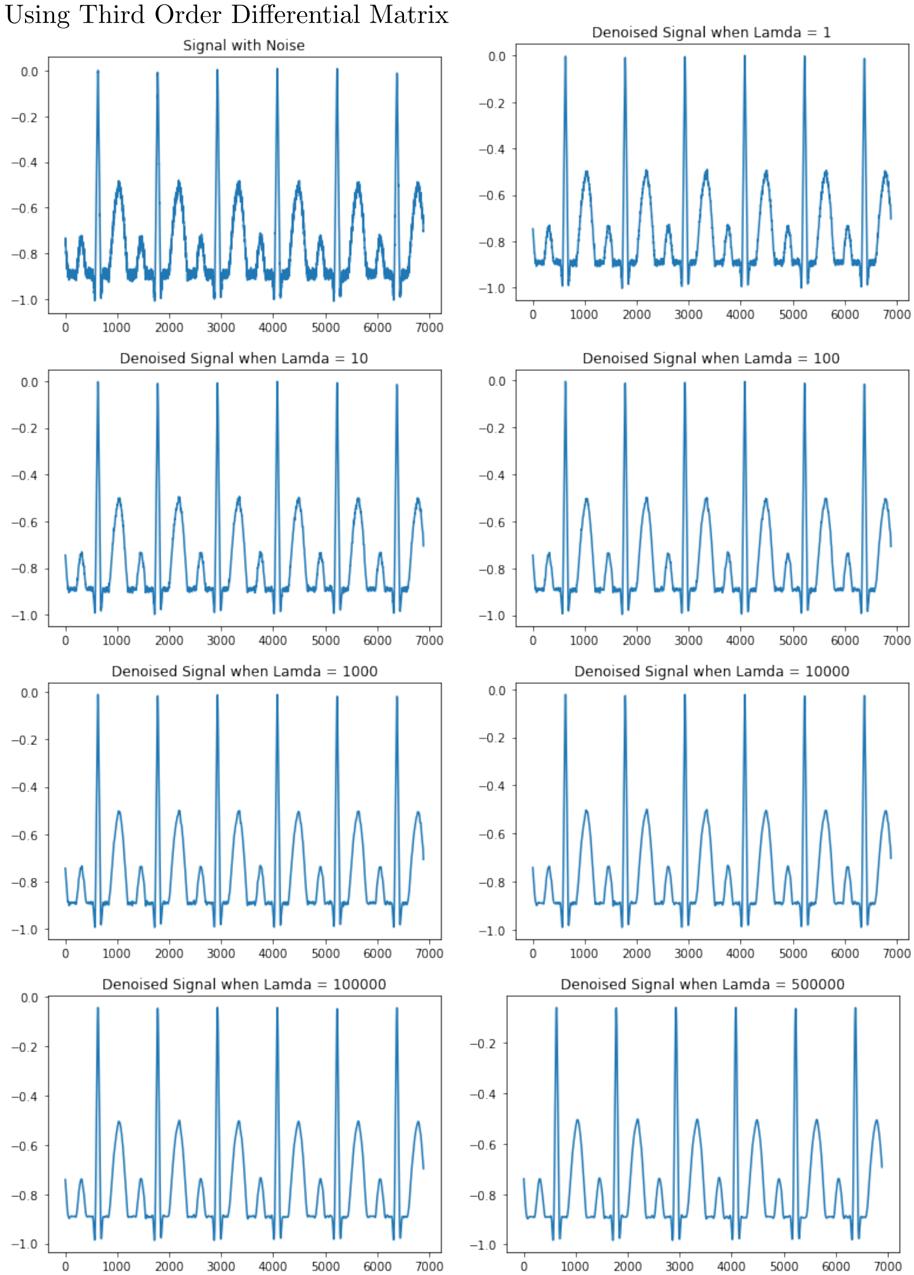
Using Second order differential matrix



Using First order differential matrix



Using Third Order Differential Matrix



Chapter 6

Conclusion

Denoising the signal using the second order differential matrix produced an output without much loss of information with respect to the input signal while removing the noise but as the lambda value increased it couldn't save the information while smoothing the signal.]From the outputs that have been displayed earlier, in the case of third or higher order differential matrices, the output signal is smoothed without the information being lost even at higher values of lambda.

Denoising the signal using the second order differential matrix produced an output without much loss of information with respect to the input signal while removing the noise but as the lambda value increased it couldn't save the information while smoothing the signal.

Bibliography

[1] https://eeweb.engineering.nyu.edu/iselesni/lecture_notes/least_squares/least_squares_SP.pdf

[2] https://www.researchgate.net/publication/350931213