

Advanced Signal Processing Coursework

Xuan Wing, Choo

(01365303)

Date: November 6, 2021

Contents

1 Random Signals and Stochastic Processes	2
1.1 Statistical Estimation	2
1.2 Stochastic processes	5
1.3 Estimation of probability distributions	7
2 Linear Stochastic Modelling	9
2.1 Autocorrelation Functions	9
2.2 Cross-correlation Function	10
2.3 Autoregressive modelling	11
2.4 Cramer-Rao Lower Bound do this	14
2.5 Real world signals: ECG from iAmp experiment	17
3 Spectral Estimation and Modelling	19
3.1 Averaged Periodogram Estimates	20
3.2 Spectrum of autoregressive processes	21
3.3 Least squares estimation of AR coefficients	23
3.4 Spectrogram for time-frequency analysis	26
3.5 Respiratory sinus arrhythmia from RR-Intervals	28
4 Optimal filtering - fixed and adaptive	29
4.1 Wiener Filter	29
4.2 Least mean squares algorithm	30
4.3 Gear shifting	31
4.4 Identification of AR processes	31
4.5 Speech recognition	32
4.6 Sign algorithms	34
5 MLE for signal frequency	34
5.1	34
5.2	35
5.3	35
5.4	36
5.5	36

1 Random Signals and Stochastic Processes

1.1 Statistical Estimation

1.1.1 Mean Estimation

Since $X \sim \mathcal{U}(0, 1)$, the probability distribution of \mathbf{x} , $f(x)$ is uniform and equal to 1. The theoretical mean $\mathbb{E}(x)$ can be calculated as follows:

$$\begin{aligned}\mathbb{E}\{x\} &= \int_{-\infty}^{\infty} xf(x)dx = 0.5 \\ \int_0^1 xdx &= \frac{x^2}{2} \Big|_0^1 = 0.5 \\ m &= 0.5\end{aligned}$$

A few runs of the matlab 'mean' function gives an approximate value of $\hat{m} = 0.5 \pm 0.02$ which indicates a max error of 4% which is fairly accurate. The error would decrease with the number of samples due to the central limit theorem.

1.1.2 Standard Deviation

The random vector \mathbf{x} from above is reused to estimate the Standard deviation. Theoretical std. dev. is calculated from the formula:

$$\begin{aligned}\sigma &= \sqrt{\mathbb{E}\{X - \mathbb{E}\{X\}\}^2} = \sqrt{\mathbb{E}\{X^2\} - \mathbb{E}\{X\}^2} \\ &= \sqrt{\int_0^1 x^2 f(x)dx - m^2} = \sqrt{\frac{1}{12}} \approx 0.2887\end{aligned}$$

Std command in matlab gives $\hat{\sigma} = 0.2887 \pm 0.003$ which has a bias of approx 1%. This estimation can also be improved by increasing the number of samples. The sample σ in Matlab has Bessel's correction which removes the bias in the estimation for variance.

1.1.3 Bias of sample mean and std. dev.

Two figures are made by plotting the ensemble mean and std dev against their theoretical values. Both figures show no bias as they sit around the theoretical values, and accuracy can be improved by increasing the number of ensembles.

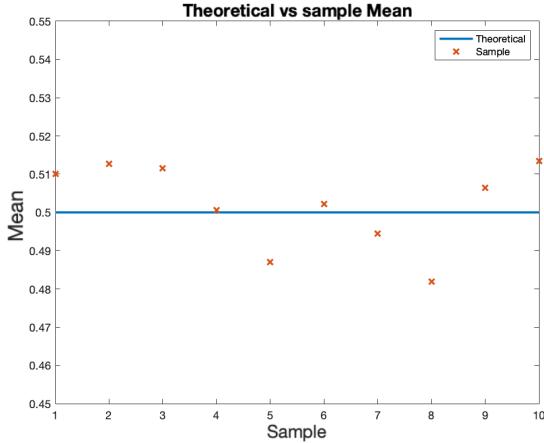


Figure 1: Mean

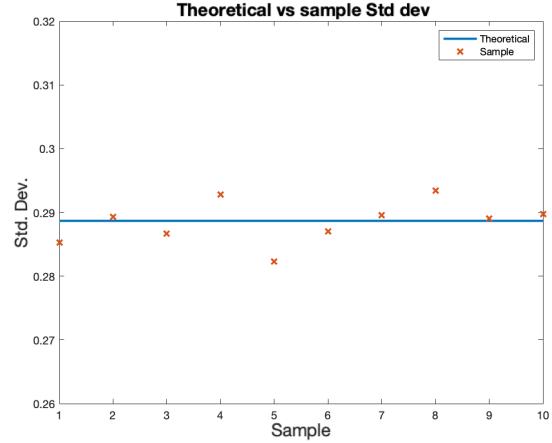


Figure 2: Standard Deviation

1.1.4 Histograms

The histogram function in matlab is used to estimate the PDF of samples. The 2 graphs directly at the bottom are made with 1000 samples. The graph with the smaller amount of bins more closely replicates the theoretical PDF (red line) as compared to the one on the right. This is due to the mean of a larger amount of samples used to make the bin average.

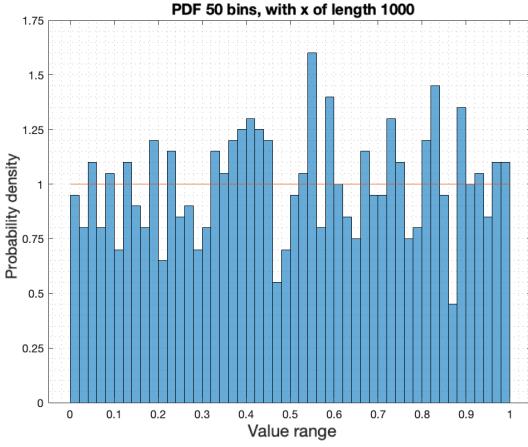


Figure 3: 50 bins, length 1000

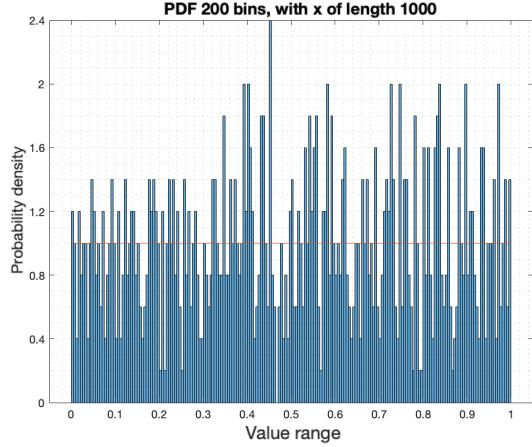


Figure 4: 200 bins, length 1000

The bottom 2 graphs show histograms made from 100000 samples. The PDF is so much better represented here as much more samples are used. Increasing bin size gives a better estimation to the theoretical PDF.

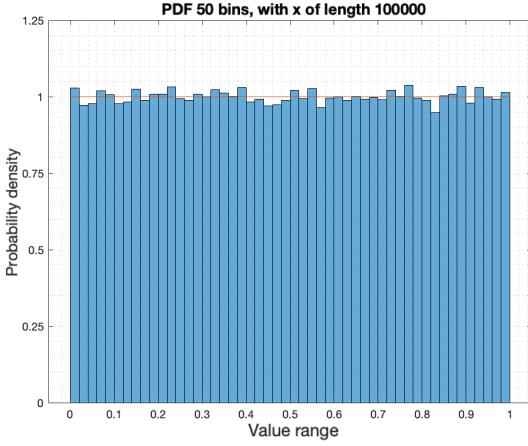


Figure 5: 50 bins, length 100000

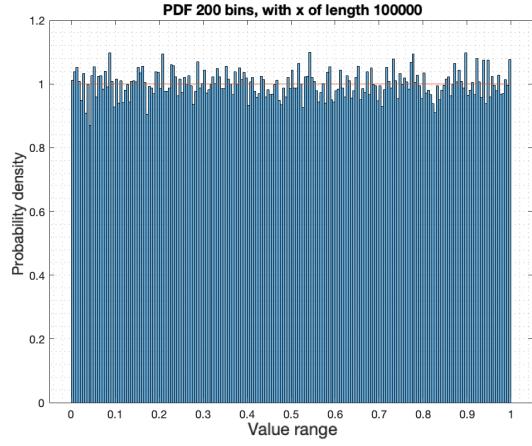


Figure 6: 200 bins, length 100000

1.1.5 Gaussian Random Variables

The Gaussian PDF is represented by the formula below, $\sigma = std.div$ and $\mu = mean$

$$f(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp -\frac{(x - \mu)^2}{2\sigma^2}$$

After integration, the expected value is 0 for the mean. By applying the formula in 1.1.2 we get 1 for the standard deviation.

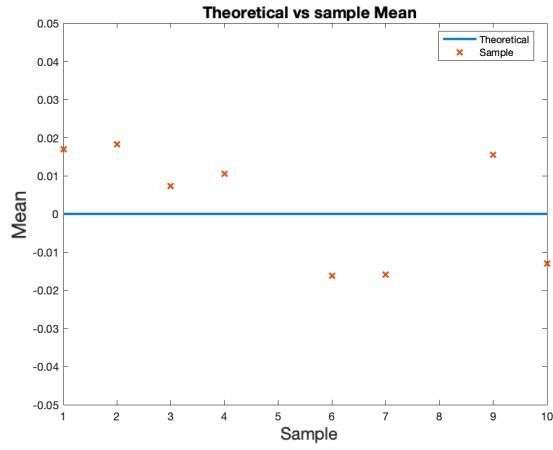


Figure 7: Mean

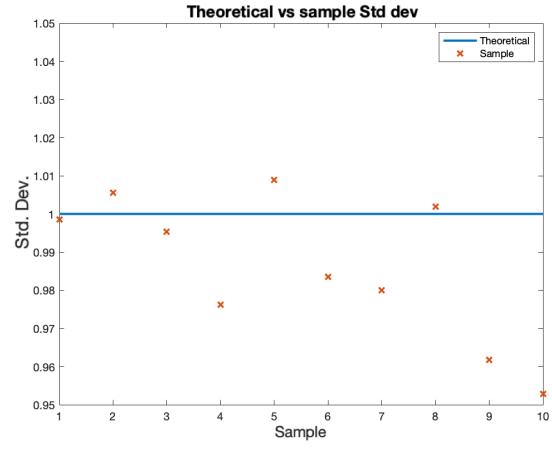


Figure 8: Standard deviation

A similar 10 ensembles 1000 samples each vector \mathbf{x} was generated with matlab randn, the graph above was generated. Both the sample mean and sample std dev hover around their theoretical values, thus there is no visible bias.

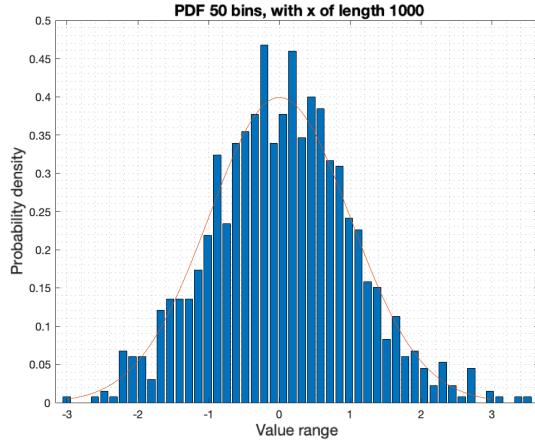


Figure 9: 50 bins, length 1000

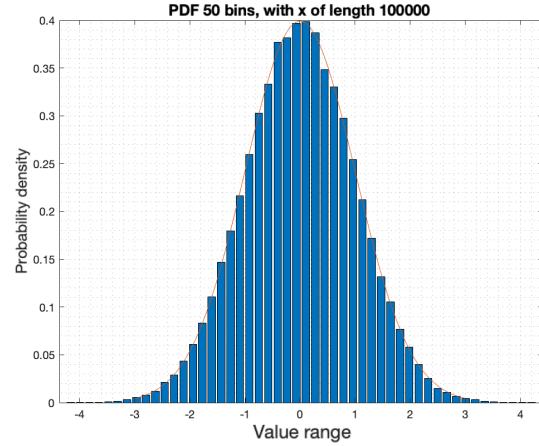


Figure 10: 50 bins, length 100000

The larger the sample length, the better the PDF is modelled closely to the actual gaussian. This is because the more the available data, the better the average calculation.

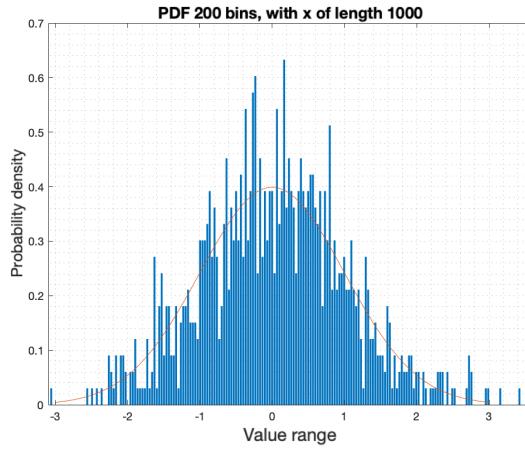


Figure 11: 200 bins, length 1000

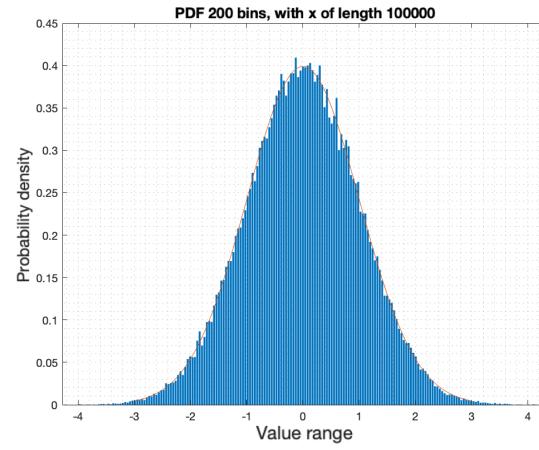


Figure 12: 200 bins, length 100000

As the size of the bin decreases, the less data is used to model the bin height, thus it fluctuates more around the actual (red line).

1.2 Stochastic processes

1.2.1 Ensemble mean and SD

An ensemble of 100 members and each of length 100 is generated.

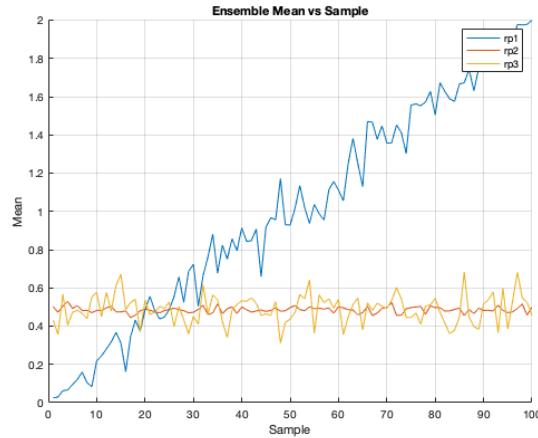


Figure 13: Ensemble Mean

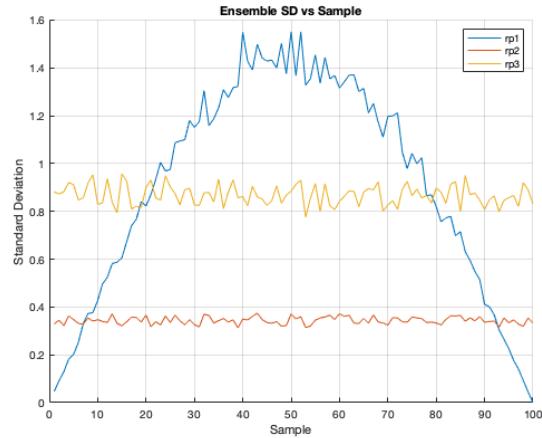


Figure 14: Ensemble Standard Deviation

Stationarity is defined as $m_{x_n} = \mathbf{E}\{x_n\} = m_x$, where m_x is the mean of an overall ensemble and m_{x_n} is the mean of the individual data points.

RP2 and RP3 are stationary as the average of each data points lie around the mean of 0.5, which satisfies the equation above. Thus, they are also time invariant.

RP1 has a mean which steadily increases with the sample, this does not satisfy the equation as the mean is not constant. It is not stationary and not time invar.

1.2.2 Ergodicity

Defined as $\langle x[n] \rangle = \mathbf{E}\{x_n\} = m_x$, the time average of each specific whole process is the same as the mean of many similar processes.

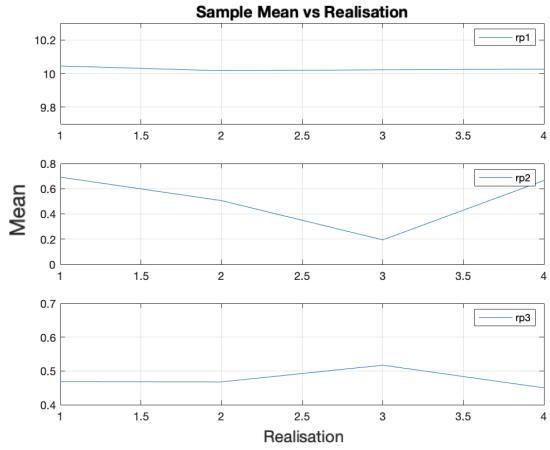


Figure 15: Sample Mean

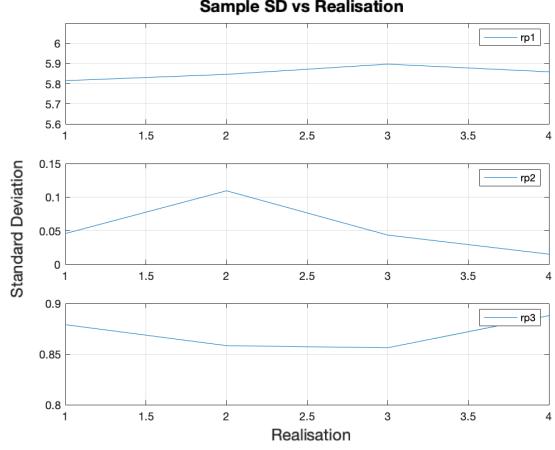


Figure 16: Sample Standard Deviation

RP3 has a mean which does not fluctuate much and is close to the sample mean, this makes it ergodic. RP2 has mean fluctuations of ± 0.2 which is large as compared to the other 2, it might not be ergodic. RP1 has a steady mean, but since it isn't stationary as described above, it cannot be ergodic.

1.2.3 Mathematical description

M is number of processes, while N is the samples in each process. U is a random variable with uniform distribution between 0 and 1.

1. RP1

$$x[n] = (U - 0.5) * 5\sin\left(\frac{n\pi}{N}\right) + 0.02n \quad (1)$$

- Theoretical mean

$$\begin{aligned} \mathbf{E}\{x[n]\} &= \mathbf{E}\{(U - 0.5) * 5\sin\left(\frac{n\pi}{N}\right)\} + \mathbf{E}\{0.02n\} \\ &= 5\sin\left(\frac{n\pi}{N}\right)\mathbf{E}\{0.5 - 0.5\} + \mathbf{E}\{0.02n\} \\ &= 0.02n \end{aligned}$$

Mean varies with n, so not stationary.

- Theoretical Variance

$$\begin{aligned} \text{Var}(x[n]) &= \mathbf{E}\{(x[n] - m_x)^2\} \\ &= \mathbf{E}\{(5(U - 0.5)\sin\left(\frac{n\pi}{N}\right) + 0.02n - 0.02n)^2\} \\ &= 25\sin^2\left(\frac{n\pi}{N}\right)\mathbf{E}\{(U - 0.5)^2\} \\ &= 25\sin^2\left(\frac{n\pi}{N}\right)(\mathbf{E}\{U^2\} - \mathbf{E}\{U\} + 0.25) \\ &= 25\sin^2\left(\frac{n\pi}{N}\right)\left(\frac{1}{3} - 0.5 + 0.25\right) \\ &= \frac{25}{12}\sin^2\left(\frac{n\pi}{N}\right) \end{aligned}$$

- Standard deviation

$$SD = \frac{5}{\sqrt{12}}\sin\left(\frac{n\pi}{N}\right)$$

SD also varies with n, both equation sits well with the graph in Fig:1.2.1 as the values change with n

2. RP2

$$x[n] = (U - 0.5)U + U \quad (2)$$

- Theoretical Mean

$$\begin{aligned} \mathbf{E}\{x[n]\} &= \mathbf{E}\{U - 0.5\}\mathbf{E}\{U\} + \mathbf{E}\{U\} \\ &= (0.5 - 0.5)(0.5) + 0.5 \\ &= 0.5 \end{aligned}$$

- Theoretical Variance

$$\begin{aligned} Var(x[n]) &= \mathbf{E}\{x^2[n]\} - m_x^2 \\ &= \mathbf{E}\{(U - 0.5)^2U^2 + U^2 + 2(U - 0.5)U^2\} - 0.5^2 \\ &= \mathbf{E}\{U^2\}(\mathbf{E}\{(U - 0.5)^2\} + 1 + 2\mathbf{E}\{U - 0.5\}) - 0.5^2 \\ &= \frac{1}{3}\left(\frac{1}{12} + 1\right) - 0.5^2 \\ &= \frac{1}{9} \end{aligned}$$

- Standard Deviation

$$SD = \frac{1}{3}$$

Both graphs in Fig:1.2.1 represent the theoretical values of mean and SD accurately. Since it does not change with n, it is stationary

3. RP3

$$x[n] = (U - 0.5) * 3 + 0.5 \quad (3)$$

- Theoretical Mean

$$\begin{aligned} \mathbf{E}\{x[n]\} &= 3\mathbf{E}\{U - 0.5\} + 0.5 \\ &= 3(0.5 - 0.5) + 0.5 \\ &= 0.5 \end{aligned}$$

- Theoretical Variance

$$\begin{aligned} Var(x[n]) &= \mathbf{E}\{(x[n] - m_x)^2\} \\ &= \mathbf{E}\{(U - 0.5) * 3 + 0.5 - 0.5\}^2 \\ &= 9\mathbf{E}\{(U - 0.5)^2\} \\ &= \frac{9}{12} \end{aligned}$$

- Standard Deviation

$$SD = \frac{3}{\sqrt{12}}$$

Both equations represent the mean and SD from graphs, and does not vary with n. So it is stationary

1.3 Estimation of probability distributions

1.3.1 1 & 2

The Fig:17 shows the estimate of a gaussian PDF, blue line is calculated while bars are estimates. Graph 1 on the right Fig:18 is the estimates for RP3 which is both stationary and ergodic. As data length N increases, the bars become more uniform and tend towards the red calculated PDF.

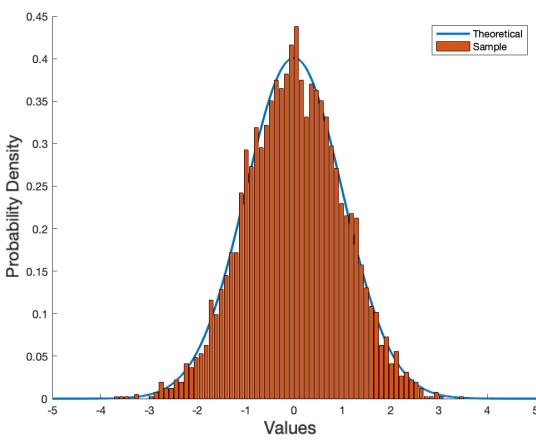


Figure 17: Gaussian PDF Estimate

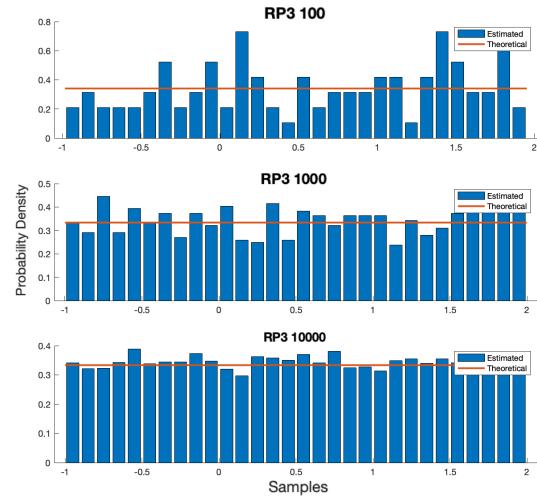


Figure 18: PDF of RP functions

1.3.2 PDF of nonstationary process

The pdf of a nonstationary process would keep changing each time the programme is run as its parameters keep changing. My function probably would not work on those.

For a sample who's mean changes from 0 to 1, the mean on N=500 would be 0.5. If we assume this process has a uniform distribution between [-0.5,0.5], and slide its probability dist. along the x axis, the area between [0,1] would have the most overlap. Thus its pdf would look like a triangle.

2 Linear Stochastic Modelling

2.1 Autocorrelation Functions

2.1.1 xcorr of WGN

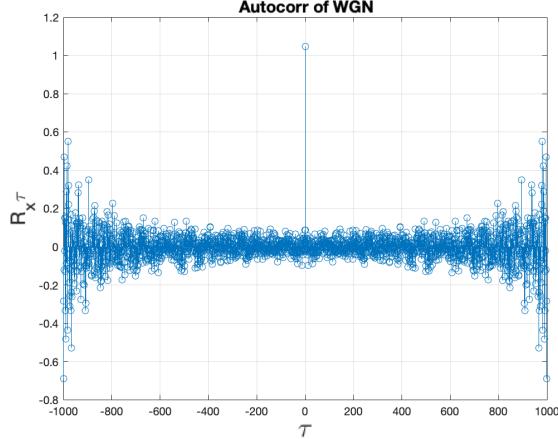


Figure 19: WGN Autocorrelation

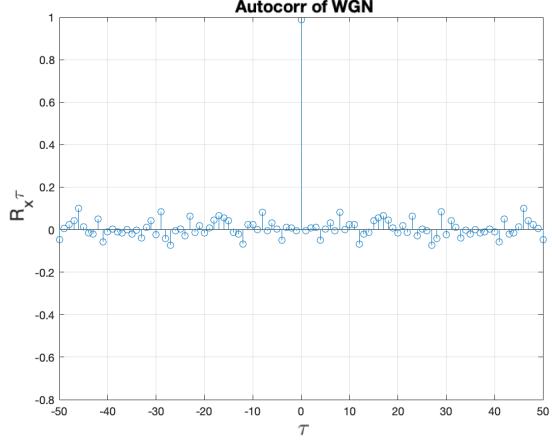


Figure 20: Zoomed in

Correlation functions show the correlation between a signal and a time lagged version of itself. Uncorrelated samples in a signal would produce a correlation where at $\tau = 0$, $R_x(\tau) = \text{variance}$, while $R_x(\tau) = 0$ elsewhere. It is also always symmetrical about 0

2.1.2 Zoomed xcorr

The graph to the right shows the zoomed version of the graph on the left. The lag within ± 50 has its values closer to 0 than larger lag values which have very large correlation values. Theoretically, the correlation is supposed to approach 0 as lag increases as signals are random and not correlated.

2.1.3 Large autocorrelation lag

Correlation seems to increase at large lag values, because smaller amounts of data are used for the calculation as lag reaches the end of data (signal length does not extend to infinity), thus being more unreliable.

$$\begin{aligned} \text{var}\{\hat{R}_X(\tau)\} &= \frac{1}{(N - |\tau|)^2} \sum_{n=0}^{N-|\tau|-1} \text{var}\{x[n]x[n + \tau]\} \\ &= \frac{N - |\tau| - 1}{(N - |\tau|)^2} (\sigma_x^2)^2 \approx \frac{(\sigma_x^2)^2}{N - |\tau|} \end{aligned}$$

As we can see from the variance of the estimator (equation above), as the lag τ reaches N the denominator becomes smaller. Thus, the fraction (estimator variance) would increase making large lags less reliable. A good threshold for lags would be $10 \log_{10}(N)$, so in this case it would be wise to include up to the 30th lag.

2.1.4 Moving average

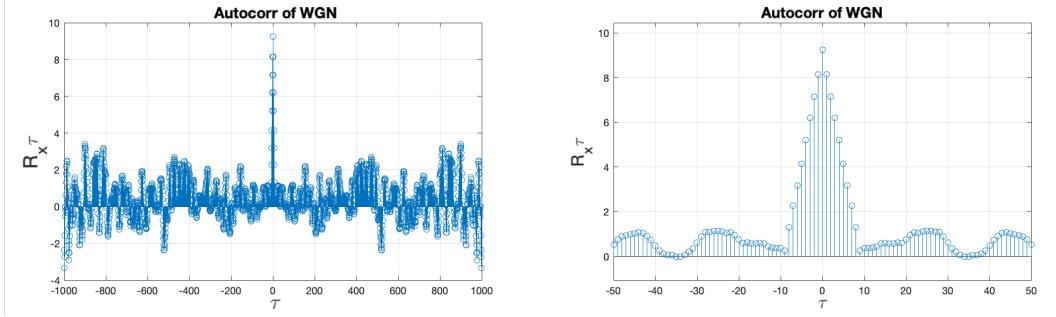


Figure 21: Autocorrelation of filtered WGN

The difference equation of the filter is $y[n] = x[n] + x[n-1] + x[n-2] \dots + x[n-q]$, where $y[n] =$ filtered signal and $x[n] =$ WGN. It was found that the general equation for MA filtered autocovariance, $Cov[y_t, y_{t+\tau}] = \sigma_x^2 \sum_{j=0}^{q-\tau} \lambda_j \lambda_{j+\tau}$, where $q =$ filter order, $\lambda =$ MA coefficients, $\tau =$ lag, $\sigma^2 =$ variance of white noise. When $\tau > q$, $Cov[y_t, y_{t+\tau}] = 0$. This is shown when $\tau=0$ approx 9, $\tau=1$ approx 8, and so on.

The autocorrelation at a specific lag is defined as $R_x(\tau) = \frac{\gamma(\tau)}{\gamma(0)}$. An example at $R_x(0) = \frac{9}{9} = 1$ and $R_x(1) = \frac{8}{9}$. It seems that coefficients of the graph above have not been normalised, this unnormalised version is also known as the autocovariance. Filter order dictates the amount of 'strong' coefficients. In this case there are 9 which coincides with filter order. Not always able to calculate mean with just the ACF as it does not carry any information about the mean.

2.1.5 R_y

$$R_Y(\tau) = R_X(\tau) * R_h(\tau)$$

Since X is an uncorrelated process, $R_X(\tau) = \sigma_x^2 \delta(\tau)$. $R_h(\tau)$ represents the autocorrelation of the filter. Convoluting both together gives $R_Y(\tau)$ which is the ACF of the filter multiplied by the variance of the uncorrelated process.

2.2 Cross-correlation Function

2.2.1 CCF of x and y

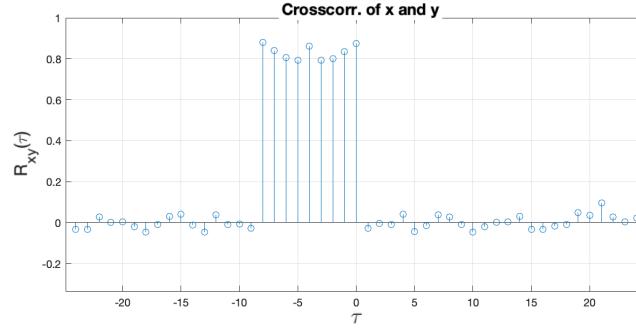


Figure 22: X & Y cross correlation

The graph shows high correlation between time lag 0 through -8. This is expected as filtered values at a present time is the average of 8 points before it due to MA filter.

R_{xy} is the convolution of filter impulse response and ACF of x . Since X_t is an uncorrelated process (let's assume Gaussian for this case), $R_x(\tau)$ would be its variance 1, a direct delta. Then, $R_{XY}(\tau) = h(\tau)*\delta = h(\tau)$, Cross corr. would just be the impulse response of filter multiplied by uncorr. process variance.

2.2.2 System Identification

Relating to the question above, if we put a gaussian white noise through an unknown system the plot of the cross correlation would reveal the impulse response of a system. Thus, we can obtain its coefficients and magnitude of the transfer function of the system.

2.3 Autoregressive modelling

2.3.1 Autoregressive Stability

Random coefficients $a_1 \in [-2.5, 2.5]$ and $a_2 \in [-1.5, 1.5]$ were generated and fed into the AR(2) process $x[n] = a_1x[n - 1] + a_2x[n - 2] + w[n]$, where $w[n]$ is normally distributed between 0 and 1.

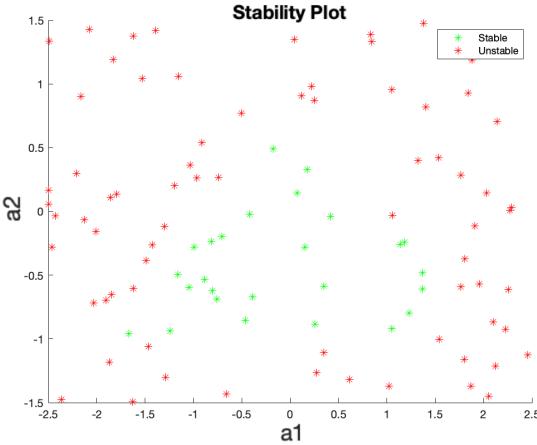


Figure 23: Stability Plot

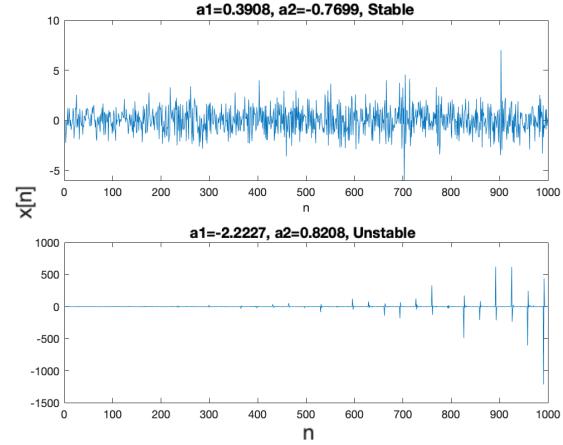


Figure 24: Stable & Unstable AR model

Graph on the left shows the colour coded stable and unstable coefficients, more data points would clearly show the stability triangle with green inside. Top graph on the right shows a stable AR model while the one on the bottom is unstable (diverging/exploding) because of its coefficients.

For a system to be inherently stable, the roots of the characteristic equation i.e. $z^2 - a_1z - a_2 = 0$ (z transform of AR equation above) must lie inside the unit circle.

$$\frac{a_1 + \sqrt{a_1^2 + 4a_2}}{2} < 1 \text{ and } \frac{a_1 - \sqrt{a_1^2 + 4a_2}}{2} > -1$$

Solving for both a_1 and a_2 and we get 2 inequalities: $a_1 + a_2 < 1$ and $a_2 - a_1 < 1$

Let's say the characteristic equation has 2 real roots, $(z - \lambda_1)(z - \lambda_2) = z^2 - (\lambda_1 + \lambda_2)z + \lambda_1\lambda_2 = 0$, stationarity requires both $|\lambda|$ to be less than 1. This makes $|\lambda_1\lambda_2| = |a_2| < 1$.

The final 3 equations below make up the stability triangle for AR(2) processes.

- $a_1 + a_2 < 1$
- $a_2 - a_1 < 1$
- $-1 < a_2 < 1$

2.3.2 Sunspot ACF

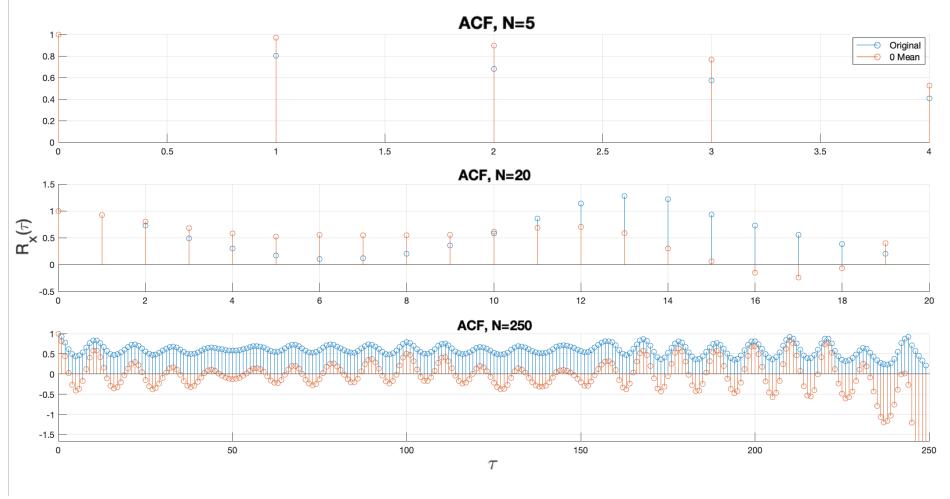


Figure 25: Sunspot ACF

ACF of dataset without zero mean has been 'pushed' upwards like bias, the sunspot ACF should look sinusoidal because of its repeating nature. By normalising the dataset, we can more easily observe this sinusoidal trend which follows the sunspots period.

This trend can only be seen above $N=20$ datapoints as 5 datapoints is too little, the ACF of $N=250$ at large lags is also quite unreliable because less data is used to calculate it, as seem above. So a good practise is to remove the mean/trend from the original data before calculating the ACF.

2.3.3 Yule-Walker PCF

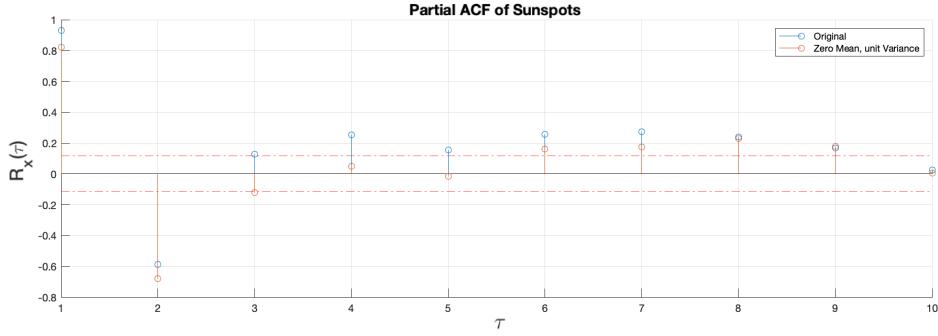


Figure 26: Sunspot PCF

Partial Autocorrelation plot shows the order of the autoregressive model better than the ACF plot. The normalised data has its value below the confidence interval while the original data has almost all its value above the confidence interval.

The threshold was found with $\pm 1.96/\sqrt{N}$ where N was the length of data (288), threshold was found to be ± 0.115 . We can see data after the 2nd lag lies close to the threshold, thus we can ignore those values. This sunspot model can be represented with AR(2).

2.3.4 Model Order Selection



Figure 27: Model Order vs Loss

Model order selection is used to select the best order for our model. Too high orders would increase overfitting and complexity while low orders decrease the models accuracy. We need to find the balance so as to not overfit the data and still being able to predict the models trend. The loss function (calculated by multiplying aryule's estimated variance by data length, N) does not drastically improve at high orders, it is perfectly fine to model it at a lower order.

In this case order AR(2) would be sufficient, order 9 works too but its more complex and the error does not decrease much, so there is no point. MDL, AIC, and AICc are mathematical methods to evaluate model fit from original.

2.3.5 Sunspot Predictions

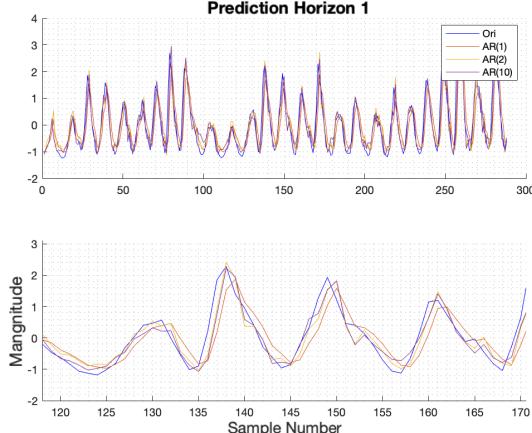


Figure 28: Prediction Horizon 1

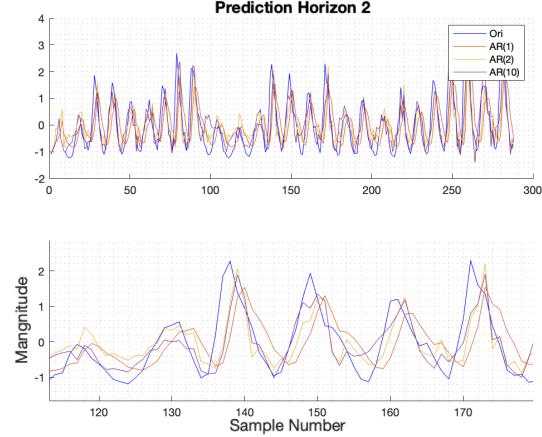


Figure 29: Prediction Horizon 2

Predicting a day ahead Fig:28 was the most accurate for all AR models, as they all produced similar results. In prediction horizon 2 Fig:29 AR(1) had a larger lag than AR(2 and 10), as the former has the least number of coefficients to manipulate the regressing data. AR(10 and 2) seems to have predicted the model quite well though AR(2) has stronger peaks while AR(10) has stronger troughs.

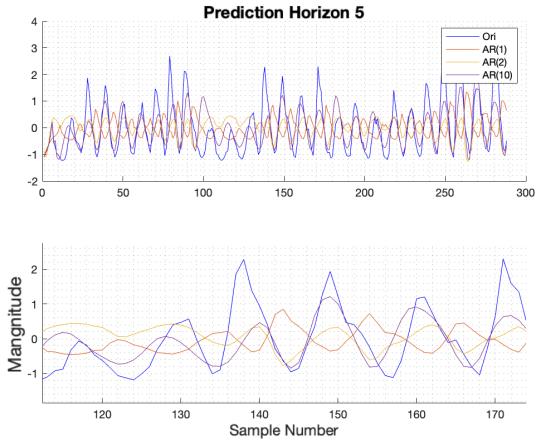


Figure 30: Prediction Horizon 5

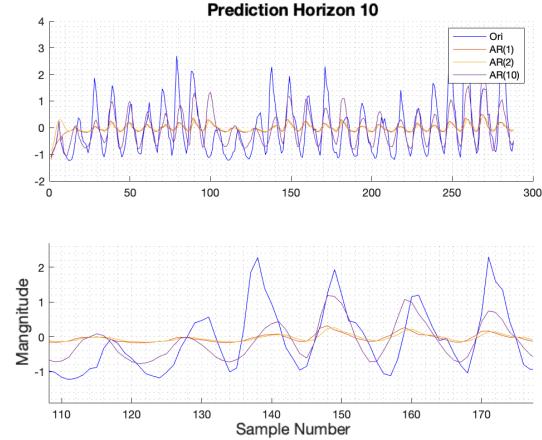


Figure 31: Prediction Horizon 10

Prediction of AR(1) and AR(2) was particularly bad in both prediction horizons 5 and 10, and all models had lower amplitudes. In Fig:30 AR(1) was totally out of sync and predicted peaks where in fact there was a trough, AR(10) seems to predict large fluctuations with accuracy while AR(2) only showed the small general trend.

in Fig:31 AR(10) was the best model out of the 3, and had an error on sample 95. The amplitude of AR(1 and 2) was much smaller than prediction horizon 5. We can see that AR(2) model order is adequate for a small prediction horizon, but in order to predict further into the future eg 5 yrs, we need a large model order to get reliable data.

2.4 Cramer-Rao Lower Bound do this

2.4.1 Autoregressive Model Order, NASDAQ

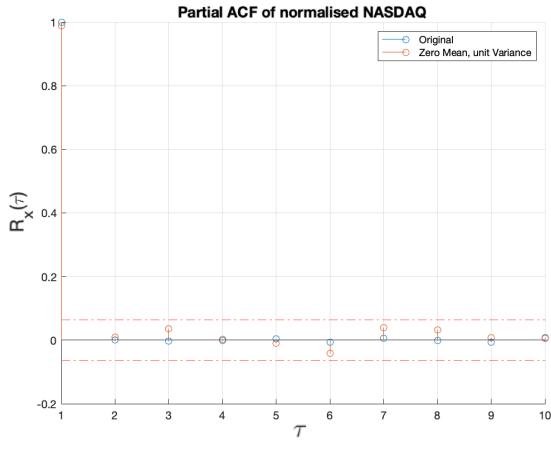


Figure 32: PCF NASDAQ

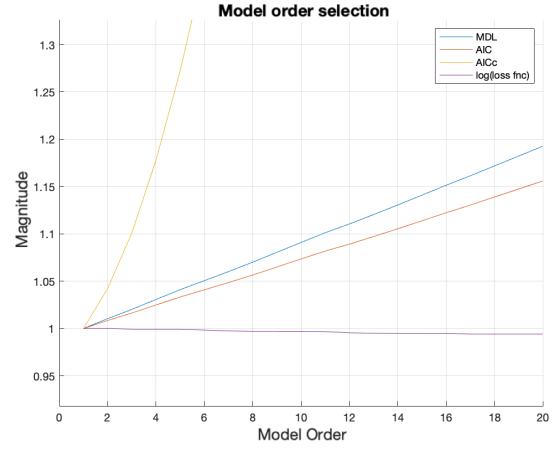


Figure 33: NASDAQ model order

Data detrended with detrend and normalised with zscore, based of Fig 33 it can be seen that order 1 is the best order as the magnitude of the error increases after 1 nor does the loss function decrease significantly.

2.4.2 Fisher information matrix

From equation 18 in the coursework, for P=1, we can deduce that:

$$\ln[\hat{P}_x(f; \boldsymbol{\theta})] = \ln[\hat{\sigma}^2] - \ln[1 - \hat{a}_1 e^{-j2\pi f}] - \ln[1 - \hat{a}_1 e^{j2\pi f}] \quad (4)$$

Since $\boldsymbol{\theta} = [a_1, \sigma^2]$ and we want $[\mathbf{I}(\boldsymbol{\theta})]_{22}$, we only consider σ^2 :

$$\frac{\partial \ln[\hat{P}_x(f; \boldsymbol{\theta})]}{\partial \sigma^2} = \frac{1}{\sigma^2} \quad (5)$$

Using equation 19 from the coursework:

$$\begin{aligned} [\mathbf{I}(\boldsymbol{\theta})]_{22} &= \frac{N}{2} \int_{-0.5}^{0.5} \left(\frac{\partial \ln[\hat{P}_x(f; \boldsymbol{\theta})]}{\partial \sigma^2} \right)^2 df \\ &= \frac{N}{2\sigma^4} \int_{0.5}^{0.5} 1 df = \frac{N}{2\sigma^4} \end{aligned} \quad (6)$$

Taking into account the other components of the fisher information matrix given in the coursework, the final 2x2 matrix becomes:

$$\mathbf{I}(\boldsymbol{\theta}) = \begin{bmatrix} \frac{Nr_{xx}(0)}{\sigma^2} & 0 \\ 0 & \frac{N}{2\sigma^4} \end{bmatrix} \quad (7)$$

2.4.3 Variance

The lower bound of the variance is the inverse of the fisher information matrix.

For σ^2 :

$$\begin{aligned} var(\theta_2) &\geq [\mathbf{I}^{-1}(\boldsymbol{\theta})]_{22} \\ var(\hat{\sigma}^2) &\geq \frac{1}{\frac{N}{2\sigma^4}} \\ \therefore var(\hat{\sigma}^2) &\geq \frac{2\sigma^4}{N} \end{aligned} \quad (8)$$

for a_1 :

$$\begin{aligned} var(\hat{a}_1) &\geq [\mathbf{I}^{-1}(\boldsymbol{\theta})]_{11} \\ var(\hat{a}_1) &\geq \frac{\sigma^2}{Nr_{xx}(0)} = \frac{1}{N} \frac{\mathbf{E}[x^2[n]] - \mathbf{E}^2[x[n]]}{\mathbf{E}[x^2[n]]} = \frac{1}{N} \left(1 - \frac{\mathbf{E}^2[x[n]]}{\mathbf{E}[x^2[n]]} \right) \\ \therefore var(\hat{a}_1) &\geq \frac{1}{N} (1 - a_1^2) \end{aligned} \quad (9)$$

In Fig 34 and Fig 35, the CRLB increases as noise variance increases, and decreases as the number of data points increases because of the increased statistics for estimation.

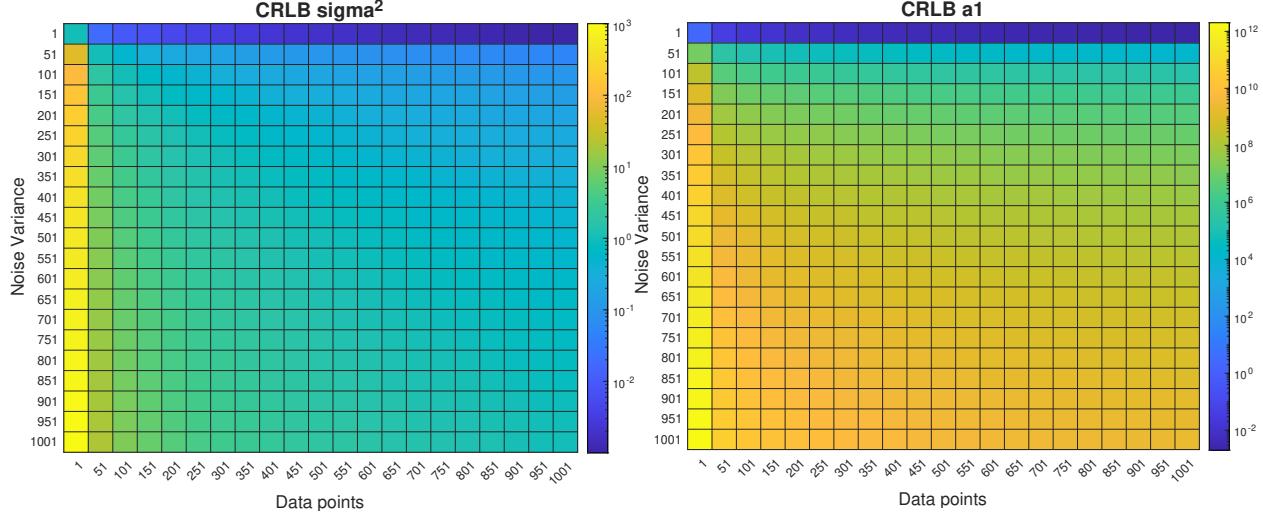


Figure 34: CRLB of σ^2

Figure 35: CRLB of a_1

Variance of a_1 can be found from eqn 9, the non normalised version of nasdaq was used for the calculations.
 $a_1 = \sqrt{\frac{\mathbf{E}^2[x[n]]}{\mathbf{E}[x^2[n]]}} = \sqrt{\frac{4.31e6}{4.344e6}} = 0.97$. a_1 is very close to 1, so $(1 - a_1^2)$ would approach 0. This means the poles would head towards the unit circle making the system unstable if a was outside the unit circle.

2.4.4 Bounds in terms of $\mathbf{A}(f)$

When $P=1$, the power spectrum equation becomes:

$$\hat{P}_x(f; \boldsymbol{\theta}) = \frac{\hat{\sigma}^2}{|1 - \hat{a}_1 e^{-j2\pi f}|^2} \quad (10)$$

$$A(f) = 1 - a_1 e^{-j2\pi f} \quad (11)$$

For a_1 :

$$\begin{aligned} \frac{\partial \hat{P}_x(f; \boldsymbol{\theta})}{\partial a_1} &= \hat{\sigma}^2 \frac{\partial}{\partial a_1} \left(\frac{1}{1 - \hat{a}_1 e^{-j2\pi f}} \frac{1}{1 - \hat{a}_1 e^{j2\pi f}} \right) \\ &= \hat{\sigma}^2 \left(\frac{e^{-j2\pi f}}{A(f)^2} \frac{1}{A(-f)} + \frac{1}{A(f)} \frac{e^{j2\pi f}}{A(-f)^2} \right) \\ &= \hat{\sigma}^2 \left(\frac{1 - A(f)}{\hat{a}_1 A(f)^2 A(-f)} + \frac{1 - A(-f)}{\hat{a}_1 A(-f)^2 A(f)} \right) \\ &= \frac{\hat{\sigma}^2}{\hat{a}_1} \left(\frac{A(f) + A(-f) - 2|A(f)|^2}{|A(f)|^4} \right) = C \end{aligned} \quad (12)$$

For σ^2 :

$$\begin{aligned} \frac{\partial \hat{P}_x(f; \boldsymbol{\theta})}{\partial \sigma^2} &= \frac{\partial}{\partial \sigma^2} \left(\frac{\hat{\sigma}^2}{|1 - \hat{a}_1 e^{-j2\pi f}|^2} \right) \\ &= \frac{1}{|1 - \hat{a}_1 e^{-j2\pi f}|^2} \\ &= \frac{1}{|A(f)|^2} = D \end{aligned} \quad (13)$$

Substituting eqn 12 and eqn 13 into eqn 22 in the coursework:

$$\begin{aligned}
var(\hat{P}_x(f; \theta)) &\geq [C \quad D] \begin{bmatrix} \frac{Nr_{xx}(\mathbf{0})}{\sigma^2} & 0 \\ 0 & \frac{N}{2\sigma^4} \end{bmatrix}^{-1} \begin{bmatrix} C \\ D \end{bmatrix} = \frac{C^2\sigma^2}{Nr_{xx}(\mathbf{0})} + \frac{2D^2\sigma^4}{N} \\
\therefore var(\hat{P}_x(f; \theta)) &\geq \frac{(A(f) + A(-f) - 2|A(f)|^2)^2\sigma^6}{\hat{a}_1^2|A(f)|^8Nr_{xx}(\mathbf{0})} + \frac{2\sigma^4}{|A(f)|^4N}
\end{aligned} \tag{14}$$

2.5 Real world signals: ECG from iAmp experiment

2.5.1 PDE of original and estimated HR

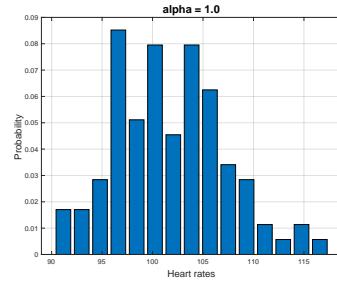
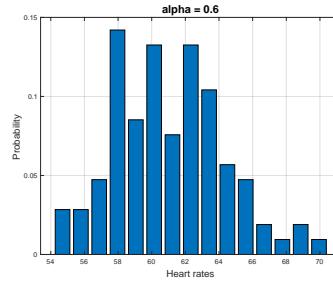
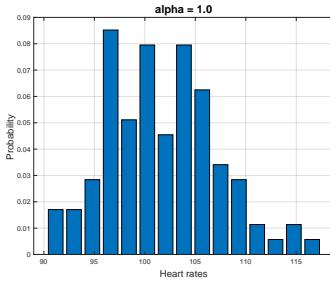


Figure 36: Original

Figure 37: averaged, alpha=0.6

Figure 38: averaged, alpha=1

2.5.2 How does alpha affect PDE

Fig37 and Fig38 are averaged on every 10 samples. We can easily see that the variance of the original is much larger than the variance of the averaged samples. This is linked by the formula $\sigma_{\hat{H}}^2 = \frac{\sigma_H^2}{L}$ with $L = 10$.

The mean of the signal would be shifted proportionally to the size of alpha, as we can see in Fig35 and Fig36 where it has been shifted to the left by 40 in alpha = 0.6, useful to convert biased signals to unbias. In section 1.2.3 (RP3 variance), alpha would be squared whilst calculating the variance, so an alpha less than 1 would decrease the variance.

2.5.3 RRI Autocorrelation

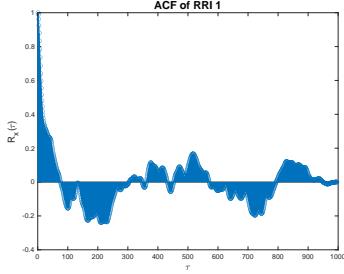


Figure 39: ACF RRI 1

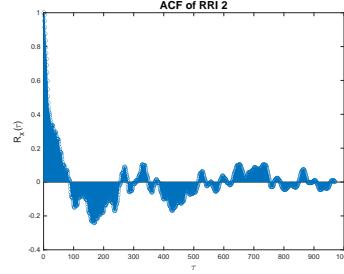


Figure 40: ACF RRI 2

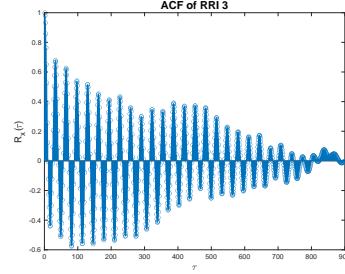


Figure 41: ACF RRI 2

An MA process has an ACF with a sharp cutoff depending on its model order, while an AR process has a slowly decaying ACF or a damped sine wave. We can see in Fig37 - 39 that the ACF decays to 0 as the lag increases, it also does not have a sharp cutoff like MA processes. Thus, we can conclude that the RRI should be modelled as an autoregressive process.

2.5.4 Optimal AR model order

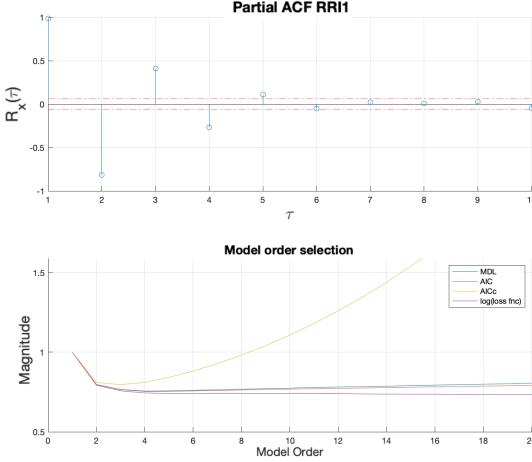


Figure 42: RRI1 Model order & PCF

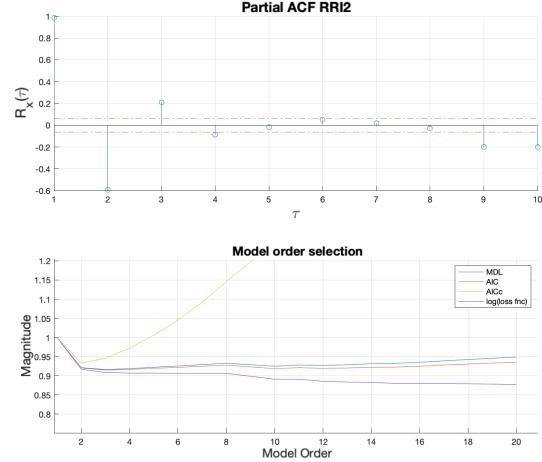


Figure 43: RRI2 Model order & PCF

RRI 1 is represented by Fig42, the PACF has about 4 plots above the threshold while AICc shows an optimum model order of 3. The MDL and AIC does not show much improvement from order 3 to 4, so it is best to model RRI 1 as an AR(3) process instead of higher to save complexity.

RRI 2 is shown in Fig43, AICc has a minimum at 2 while MDL and AIC shows a minimum at 3. The loss function also shows a small improvement from model 2 to 3. This can be modelled as AR(3) by also taking into account the number of points of the PCF above the threshold.

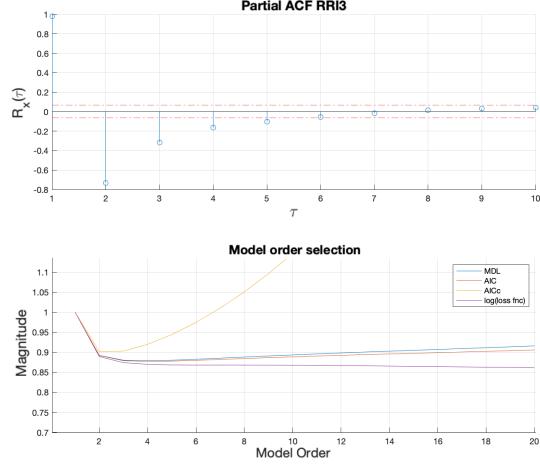


Figure 44: RRI3 Model order & PCF

Fig44 represents RRI3, the minimum points for MDL and AIC is around 4 while AICc is 3. There does not seem to be a huge improvement from 3 to 4, so to save on computational cost we can model this as AR(3).

3 Spectral Estimation and Modelling

The theoretical PSD of WGN is supposed to be an equal magnitude of 1 at all frequencies. We can see from Fig 45 through Fig 47 that the periodogram repeats itself at 0.5 normalised frequency or π . This is because the `fft` was used to compute the PSD, and we know the Fourier transform is symmetric.

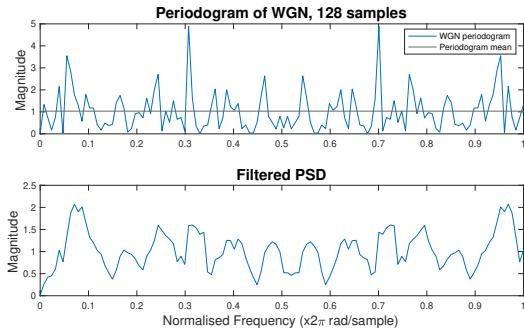


Figure 45: PSD of 128 sample WGN

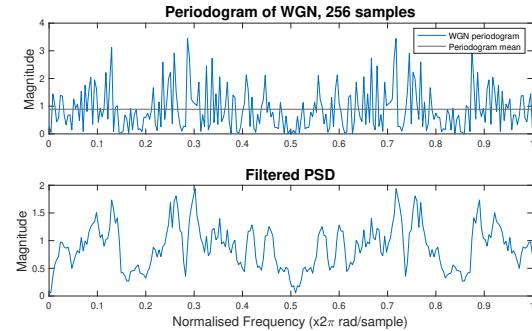


Figure 46: PSD of 256 sample WGN

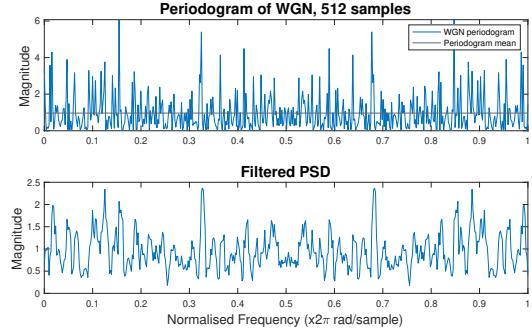


Figure 47: PSD of 512 sample WGN

Segment length	Unfiltered		Filtered
	Mean	Variance	Variance
128	0.8628	1.2042	0.3596
256	0.9383	0.7959	0.1978
512	0.9187	0.8208	0.2463

Table 1: Statistics of filtered and unfiltered PSD

As we can see from Table 3 increasing the segment length does not cause the PSD to tend to theoretical where $\text{mean} = 1$. There is also no such correlation in the variance as signals are uncorrelated. Variance decreases in general by filtering the PSD, but is uncorrelated with the increase in segment length.

There would be 2 different periodograms if we were to use the Wiener–Khintchine theorem. This is because the ACF can be calculated as biased or unbiased, the unbiased ACF has large values at large τ thus the PSD would have a high variance. The biased ACF uses bartlett window which gives lower weights at large τ , thus a WGN ACF would look more like a direct delta @ $\tau = 0$, giving a better estimate of the PSD.

3.1 Averaged Periodogram Estimates

3.1.1 FIR filter

Filtering the PSD in Fig 45 - 47 (bottom) returns a PSD with decreased variance (Table 3). Unfortunately, this includes a bias into the new filtered data and frequency resolution would also decrease.

3.1.2 WGN Segments

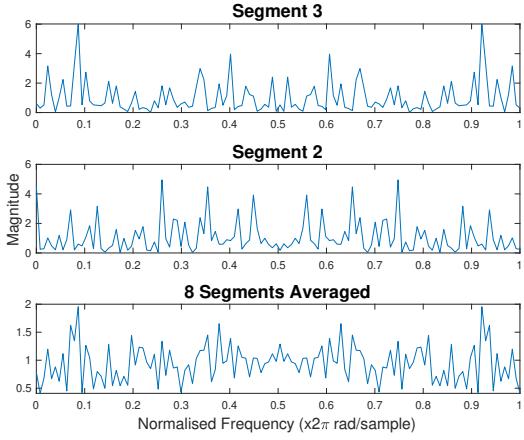


Figure 48: PSD of segments and their average

The segments all had random variance and PSD peaks, the mean hovered around 1 for all the segments. Samples can be seen above in Fig 48 (top and middle)

Segments	1	2	3	4	5	6	7	8
Variance	1.1727	1.5254	1.5904	0.8813	0.4608	1.0795	1.1937	0.7021

Table 2: Variance of Segments

3.1.3 Averaged Periodogram

The averaged of all the 8 segments are in Fig 48 (bottom). The variance of the averaged decreased by a factor of $\frac{1}{M}$, where M is the number of segments. $\sigma^2 = 0.1097$. More segments can decrease the variance, but this would decrease the length of each segment which would in turn decrease the PSD frequency resolution. Welch's method can be used to overlap signals to offset the reduced signal length.

3.2 Spectrum of autoregressive processes

3.2.1 Theoretical PSD

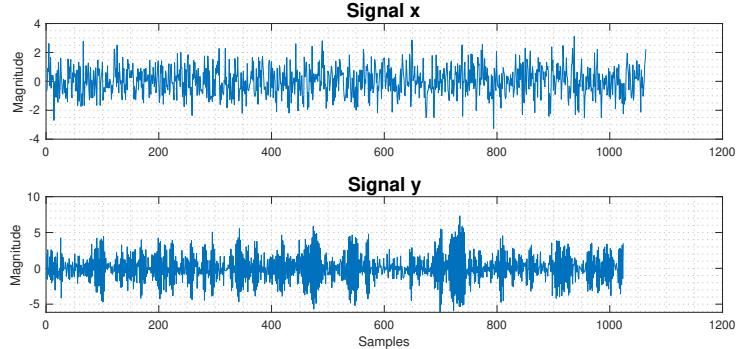


Figure 49: Signal x and y, time domain

The cut-off frequency of the filter can be taken from Fig 50 (blue line) as around 0.4 Hz normalised frequency. This can be further strengthened by referring to Fig ?? where the low frequency components has been filtered and high frequency components left alone.

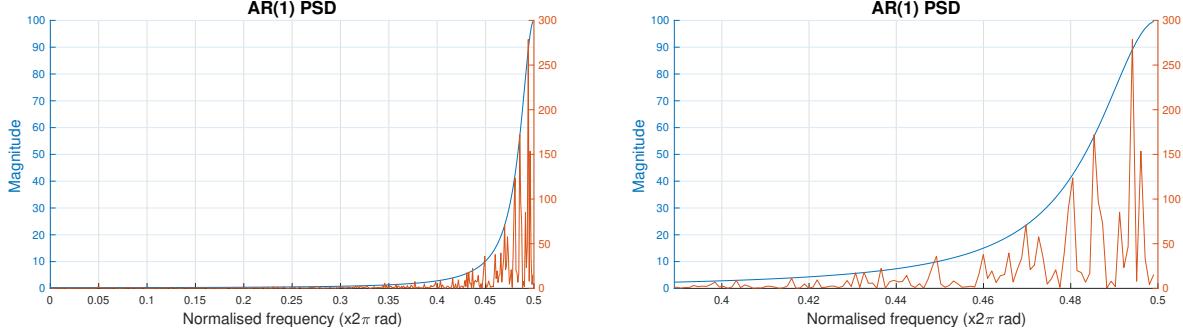


Figure 50: PSD of AR(1) process (theoretical and estimate)

3.2.2 Periodogram of y

The exact PSD, Fig 50 blue line is more smooth because it was made with the transfer function of the filter which is an analytical solution. PGM (red line) is made by taking the absolute magnitude of the fourier transform of the signal which has discrete points.

3.2.3 Rectangular windowing

The PSD of a finite signal is calculated by splitting the signal into segments and convolving the FT of the segments by the window of your choice. The FT of a rectangular window is a sinc function. In our case, the low frequency components have been attenuated, so the convolution at those points would be close to 0. This would be the opposite for high frequencies where the components have been preserved.

3.2.4 Model based PSD

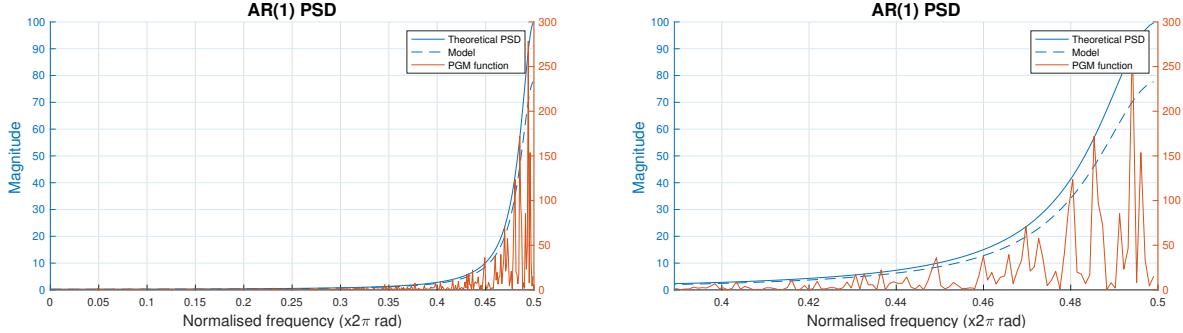


Figure 51: Model PSD of AR(1) process

Based of Fig 51, the model based estimate is very similar to the exact periodogram. The PGM function uses the fourier transform to estimate the PSD, it only takes into account present values of signals and has no information about the next prediction in the signal. Model based estimated on the other hand use the autocorrelation function which is able to predict future signals. Based of this, the model based estimate is expected to fare better in predicting the PSD of the signal. \hat{a}_1 was 0.8912 while \hat{a}_X^2 was 1.0031, used in the model estimate, was very similar to the transfer function of the filter which was 0.9 and variance 1. This estimate is not affected by periodogram windowings.

3.2.5 PSD of sunspot

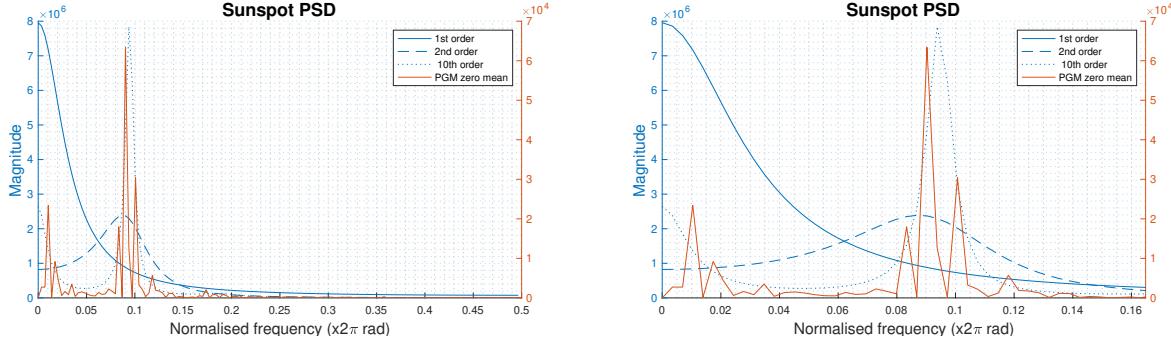


Figure 52: Sunspot PSD estimates

AR(1) model gave a result which was different than the others, this order was clearly not representative of the PSD. The AR(10) model gave a really sharp peak at 0.1 normalised frequency, and really low magnitude elsewhere. This meant that the sunspot was really predictable with a regular occurrence, which isn't the case, overmodelling. The second model order has a wide band at around 0-0.1 normalised frequency, this meant that there are various smaller frequency components which made up the sunspot series. This would be a good estimate as sunspots are not regular and have a probability of varying time between sunspots.

3.3 Least squares estimation of AR coefficients

3.3.1 General form of cost function

The cost function is the sum of error squared:

$$J(\theta) = \sum_{n=0}^{N-1} e^2[n] = \sum_{n=0}^{N-1} (x[n] - s[n])^2 \quad (15)$$

Eqn 15 can also be written as:

$$J(\theta) = (\mathbf{x} - \mathbf{H}\theta)^T (\mathbf{x} - \mathbf{H}\theta) \quad (16)$$

Where $\mathbf{s} = \mathbf{H}\theta$ and $\mathbf{H}, \theta, \mathbf{x}$ are all column vectors with length matching sample size. Since $e^2[n]$ is not a vector, we know that the form above is true as a vector transpose multiplied by itself would return a scalar. Based of eqn 33 of the coursework, the cost function which incorporates autocorrelation coefficients is:

$$J = \sum_{k=1}^M \left[\hat{r}_{xx}[k] - \sum_{i=1}^p a_i \hat{r}_{xx}[k-i] \right]^2 \quad (17)$$

We can see that the form of eqn 17 matches eqn 15, thus implying that:

$$\mathbf{x} = \begin{bmatrix} \hat{r}_{xx}[0] \\ \hat{r}_{xx}[1] \\ \hat{r}_{xx}[2] \\ \vdots \\ \hat{r}_{xx}[M] \end{bmatrix} \quad \boldsymbol{\theta} = \begin{bmatrix} a[0] \\ a[1] \\ a[2] \\ \vdots \\ a[M] \end{bmatrix} \quad \mathbf{H} = \begin{bmatrix} \hat{r}_{xx}[0] & \hat{r}_{xx}[1] & \dots & \hat{r}_{xx}[p] \\ \hat{r}_{xx}[1] & \hat{r}_{xx}[0] & \dots & \hat{r}_{xx}[p-1] \\ \vdots & \vdots & \vdots & \vdots \\ \hat{r}_{xx}[p] & \hat{r}_{xx}[p-1] & \dots & \hat{r}_{xx}[0] \end{bmatrix} \quad (18)$$

We can use the orthogonality condition that $(\mathbf{x} - \mathbf{s}) \perp \mathbf{H} = 0$ and $(\mathbf{x} - \mathbf{s})^T \perp \mathbf{H} = 0$.

$$\begin{aligned}
(\mathbf{x} - \mathbf{H}\boldsymbol{\theta})^T \mathbf{H} &= 0 \\
\mathbf{H}^T (\mathbf{x} - \mathbf{H}\boldsymbol{\theta}) &= 0 \\
\mathbf{H}^T \mathbf{x} &= \mathbf{H}^T \mathbf{H}\boldsymbol{\theta} \\
(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x} &= \boldsymbol{\theta}
\end{aligned} \tag{19}$$

Thus, $\boldsymbol{\theta}$ in eqn 19 is the least squares estimator for the coefficients a . \mathbf{H} is a hermitian matrix

The above uses autocorrelation which windows the data, so the estimates might not model the process correctly.

3.3.2 Observation Matrix

The observation matrix \mathbf{H} is deterministic because the autocorrelation functions are not swayed by noise present. Thus its parameters would not change if noise changed, unless the parameters which generate the signal itself changes.

3.3.3 Sunspots LSE

The format of the coefficients in table 3 is $y = a_1 + a_2y[n-1] + a_3y[n-2] + \dots + a_{11}y[n-10]$. Table 3 was generated with the original sunspot data without removing the mean or normalising the variance. If we were to remove the mean, the general shape of the curve would still be the same, but the y intercept a_1 would decrease because the original data has been shifted downwards. The matlab function `lpc` performs the same calculation as eqn 19 to find the OLS AR coefficients.

Coefficients	1	2	3	4	5	6	7	8	9	10
a_1	1	1	1	1	1	1	1	1	1	1
a_2	-0.8212	-1.3783	-1.2953	-1.3011	-1.3018	-1.3044	-1.2760	-1.2361	-1.1959	-1.1952
a_3	-	0.6783	-0.5097	0.4856	0.4885	0.4994	0.4597	0.4447	0.4243	0.4243
a_4	-	-	0.1223	0.1836	0.1912	0.1602	0.1859	0.1541	0.1609	0.1608
a_5	-	-	-	-0.0473	-0.0676	-0.1469	-0.1749	-0.1351	-0.1523	-0.1520
a_6	-	-	-	-	0.0156	0.2269	0.1394	0.0971	0.1210	0.1205
a_7	-	-	-	-	-	-0.1623	0.0661	-0.0385	-0.0658	-0.0652
a_8	-	-	-	-	-	-	-0.1751	0.01154	0.0368	0.0362
a_9	-	-	-	-	-	-	-	-0.2276	-0.0093	-0.0109
a_{10}	-	-	-	-	-	-	-	-	-0.1766	-0.1721
a_{11}	-	-	-	-	-	-	-	-	-	-0.0038

Table 3: LSE coefficients for different model orders

3.3.4 Model order

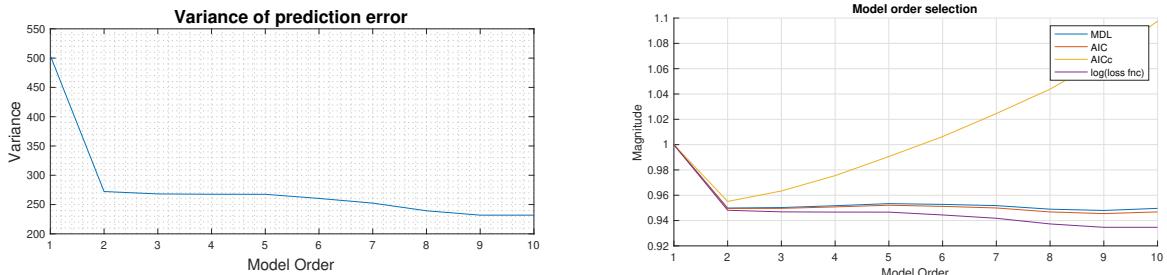


Figure 53: Variance and Error of prediction

The figure above shows the variance of the prediction error as compared to the model order. The variance does not decrease much after model order 2, this data is similar to section 3.2 where the best predictor would also be model order 2. We can also see on the right where the optimal order would be 2, as additional complexity would not give significant improvements.

3.3.5 Sunspot power spectra

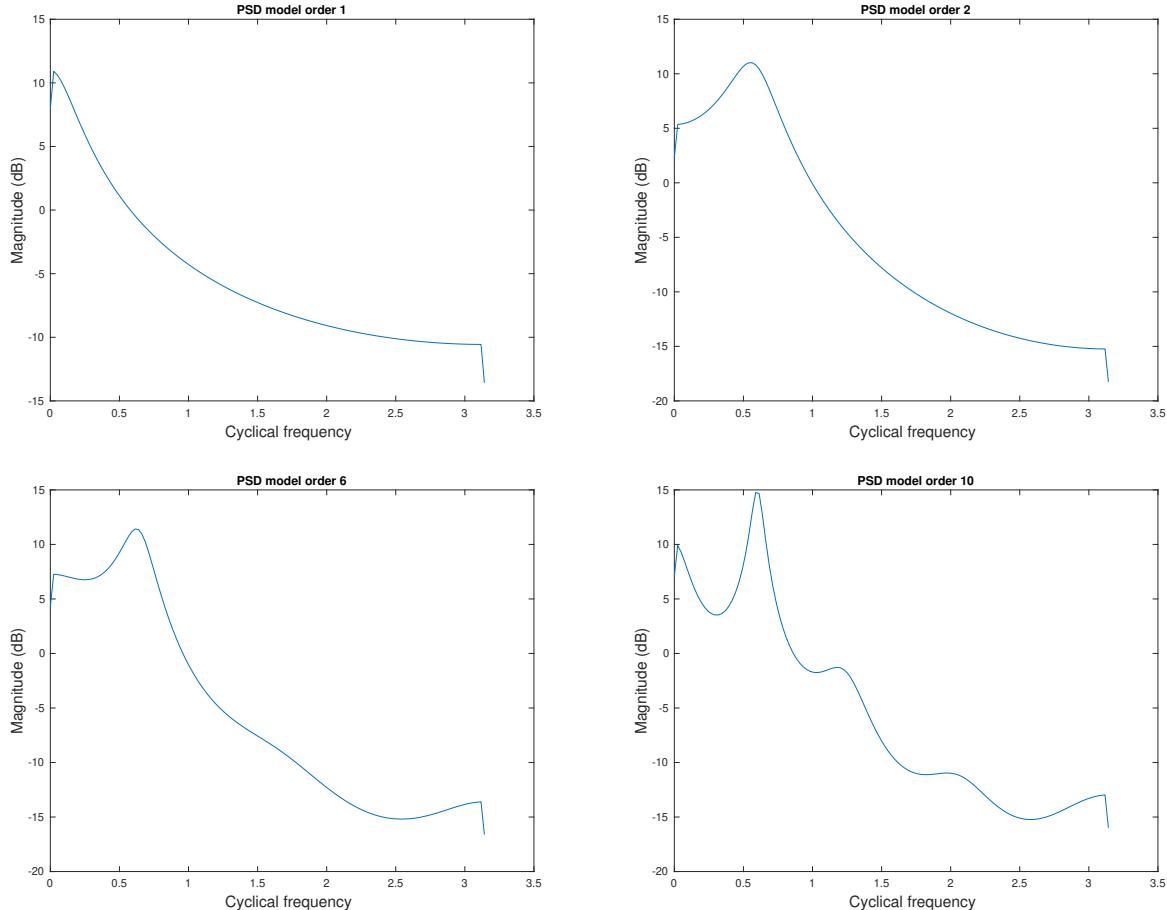


Figure 54: Sunspot PSD estimates

As the model order increases, the power spectrum becomes more detailed with small peaks at frequencies apart from 0.6. Undermodelling such as model order 1 has only 1 peak close to 0, which would not capture all the needed data available.

3.3.6 Sunspot approximate model error

3.4 Spectrogram for time-frequency analysis

3.4.1 Random landline number

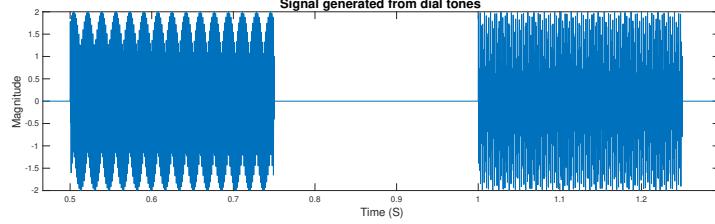


Figure 55: Time domain when 2 and 0 are pressed

Fig 55 shows a small sample of 2 and 0 from the entire landline time domain. The generated landline number is 02047031013, and lasts for 5.25 seconds. The sampling rate of 32768 Hz is far higher than the maximum frequency in the dial pad which is 1477 Hz, so everything would be preserved. This prevents aliasing from occurring for any sampled frequency up till 16384 Hz which is the Nyquist frequency.

3.4.2 Spectrogram of landline

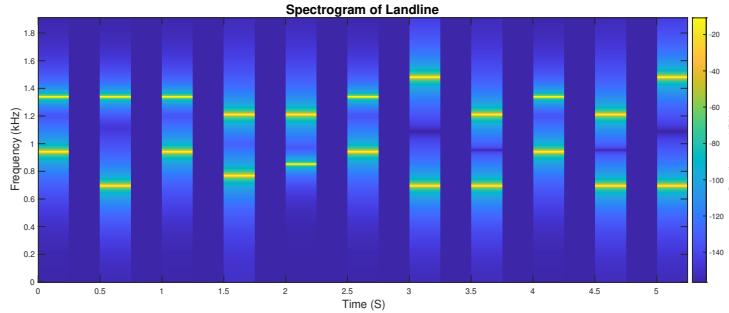


Figure 56: Spectrogram of entire dialled landline sequence

A hanning window of size 8192 was used as each press of 0.25 sec. Fig 56 was made with the function `spectrogram(y,hann(8192),0,[],fs,'yaxis')`. The figure shows clear distinction between keys pressed and breaks, the bright yellow bars indicate the frequency components present in the signal while at the breaks there is no strong frequencies.

The gradual decrease in frequency strength is associated with the spectral leakage due to windowing, the ideal graph would just show 2 distinct peaks for each time a button is pressed. Frequency resolution of FFT is computed as F_s/N , in our windowed case the resolution would be 4Hz, so any peak which is not divisible by 4 would have a magnitude lesser than expected.

3.4.3 Key classification

It is possible to identify a sequence by plotting the signal on a spectrogram, and assigning a threshold to the magnitudes. We would record any signal whose magnitude is above the threshold, but the result would be a range and not a definite number as the peaks decrease gradually on both sides of the frequency. We then refer back to the dial pad to find out the specific number pressed at each sequence.

3.4.4 Noisy signal

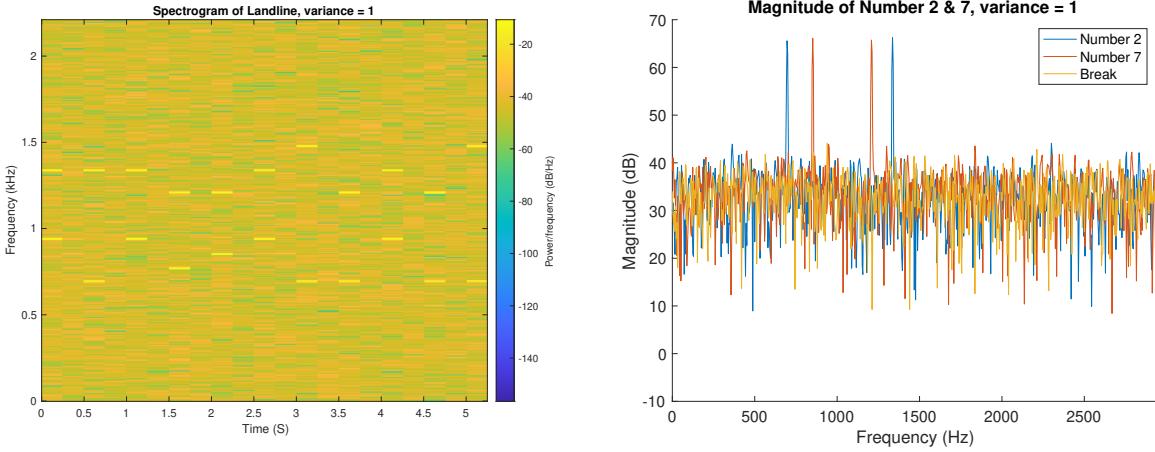


Figure 57: Spectrogram of number 2 & 7, WGN variance 1

Fig 57 shows the spectrogram of the same landline sequence as the above but with WGN of variance = 1. It is still able to show peaks of the signals frequency components as the magnitude is still larger than the noise. The parts with the break is no longer distinct as compared to its surroundings and the graphs produced in the sections above.

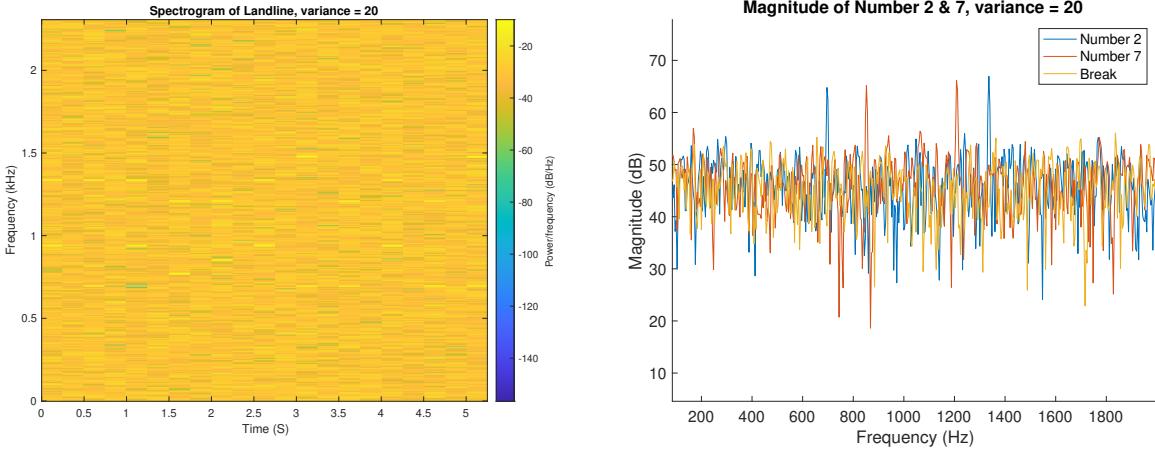


Figure 58: Spectrogram of number 2 & 7, WGN variance 20

Fig 58 has been consumed with WGN of variance = 20, the magnitude of the peak is still constant but the surrounding noise have increased in magnitude from around 30dB to 50dB. This is close to the limit of thresholding as we can just about differentiate this signal from its surroundings. It is really hard (might still be possible) to distinguish peaks in the spectrogram, so the magnitude plot would give a clearer view.

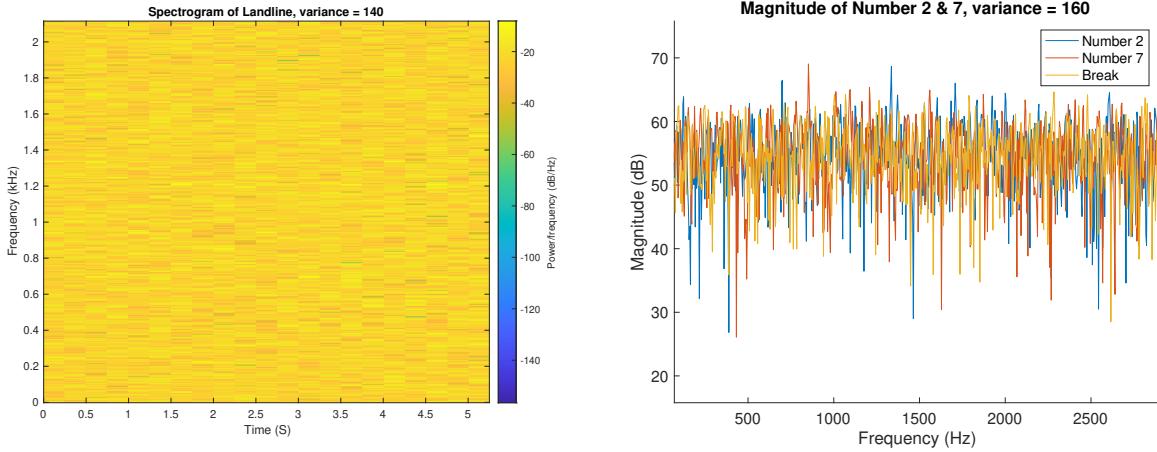


Figure 59: Spectrogram of number 2 & 7, WGN variance 160

Fig 59 shows the signal with WGN of variance = 140, both the spectrogram and magnitude plot have been overwhelmed by the magnitude of the noise. We cannot discern individual peaks thus, any signal with noise above this variance will not be useful for analysis.

3.5 Respiratory sinus arrhythmia from RR-Intervals

3.5.1 Periodograms

Fig 60 was made by putting the zero mean RRI trials into the function pgm which I made.

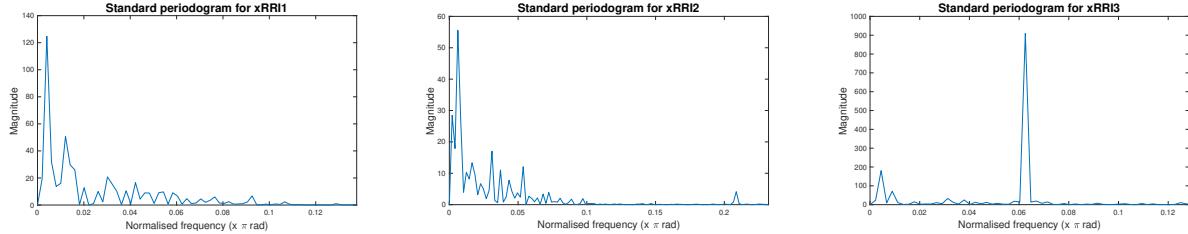


Figure 60: Standard Periodogram

By using a window of 150s and 50s respectively the figures below were generated. The signals had to be zero padded after windowing to increase frequency resolution of FFT in pgm.

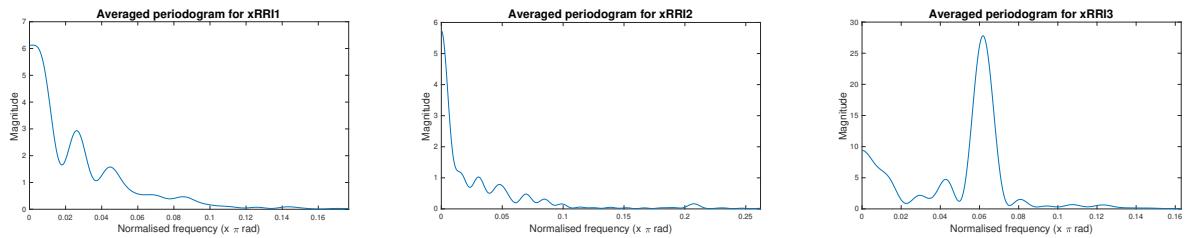


Figure 61: Averaged periodogram, 150s window

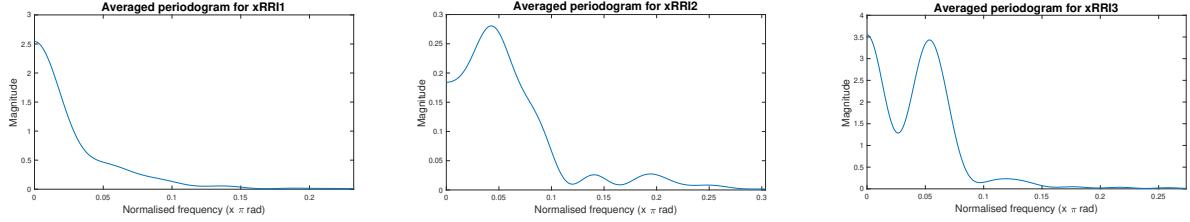


Figure 62: Averaged periodogram, 50s window

3.5.2 Difference between PSD estimates

The RRI data frequency component shows the heart rate of the test subject. Trial 3 had the largest frequency component compared to the other 2, this means his heart rate was beating quickly, taking 7.5 breaths a min was too little for the subject thus his heart had to work harder to supply oxygen to the body. RRI2 had the lowest frequency component, thus, heart rate. Breathing 25 breaths a min was giving the body a lot of oxygen so the heart need not pump as much.

4 Optimal filtering - fixed and adaptive

4.1 Wiener Filter

4.1.1 Cross and Autocorrelation

The signal to noise ratio is given by $\frac{\mathbb{E}(z^2)}{\mathbb{E}(\eta^2)} = \frac{1}{0.01} = 100 = 20dB$. The coefficients of the wiener filter are shown in table 4 below. W_{opt} of the original data matches the filter coefficients, while coefficients of the normalised data has been scaled by a factor of around 5.

Coefficients	1	2	3	4	5
W_{opt} Original Data	0.9979	1.9968	2.9965	1.9969	1.0074
W_{opt} Normalised Data	0.2161	0.4368	0.6487	0.4375	0.2165

Table 4: Wiener filter coefficients

4.1.2 Varying noise variance

As the variance of the noise increases the accuracy of the coefficients tend to decrease. Because the noise drowns out the original signal as SNR decreases, lower quality signal.

Variance	0.1	0.8	2	5	8	10
SNR (dB)	10.1548	3.4425	1.5162	0.7917	0.4343	0.6908
Coeff. 1	0.2285	0.2388	0.2488	0.1381	0.2277	0.1181
Coeff. 2	0.4726	0.5125	0.3912	0.5580	0.6251	0.6155
Coeff. 3	0.7191	0.6889	0.7036	0.6549	0.6615	1.0724
Coeff. 4	0.4843	0.4912	0.3792	0.6062	0.3268	0.5217
Coeff. 5	0.2310	0.2589	0.1799	0.2181	0.1957	0.4164

Table 5: Signal noise ratios

The number of Wiener filter coefficients generated if the order is greater than the filter order is shown in table 6 below. There would still be the same number of matching filter coefficients on the last 5 coefficients. The extra (first 2) would have very small values, almost negligible. This shows that over modelling would not contribute much to the accuracy of the filtered data.

Coefficients	1	2	3	4	5	6	7
W_{opt} Normalised Data	2.0643 e-06	1.9103e-04	0.2161	0.4368	0.6487	0.4375	0.2165

Table 6: Wiener filter coefficients, $N_w > 4$

4.1.3 Wiener complexity

The autocorrelation function preforms N multiplications and N summations, this comes up to $2N$ for each lag, and we have $N+1$ lags, so this becomes $2N(N+1)$ for the ACF. This value is also similar for the cross correlation function. Matrix inversion has a $O(n^3)$ complexity and multiplication in our case of a $N \times N$ and $N \times 1$ matrix has $O(n^2)$. The total becomes $O(5n^2 + n^3)$.

4.2 Least mean squares algorithm

As the variance of the noise increases, it is expected that the accuracy of the LMS estimator would decrease as explained above. The accuracy can be compared to table 4 where the variance of the noise is very low (0.01) and can be seen as accurate, and table 5 with different variance.

LMS allows the data to be non stationary while Wiener has to be stationary, LMS does not process all the data at once, windowing can be seen in the function. Because of this, the LMS coefficients would always be a little less accurate than Wiener, but the advantage is that it can adapt to changing signals.

Variance	0.1	0.8	2	5	8	10
Coeff. 1	0.2207	0.2770	0.2004	0.2566	0.3872	0.2758
Coeff. 2	0.4515	0.5924	0.5158	0.2633	0.3118	0.4344
Coeff. 3	0.6685	0.6486	0.9331	0.8283	0.5543	0.7997
Coeff. 4	0.4788	0.3899	0.4475	0.4153	0.8640	0.1970
Coeff. 5	0.2554	0.2319	0.3469	0.4574	0.2010	0.4613

Table 7: LMS coefficients with varying noise variance

4.2.1 Changing adaptation gain

Fig 63 right shows an adaptation gain of 0.01. In general, as the adaptation gain increases, the faster the coefficients converge to steady state. If the gain is big, the coefficients would jump (overshoot) around the optimal value as the step size is to big for it to converge correctly. Fig 58 right shows that if the gain is too big, the coefficients would diverge instead.

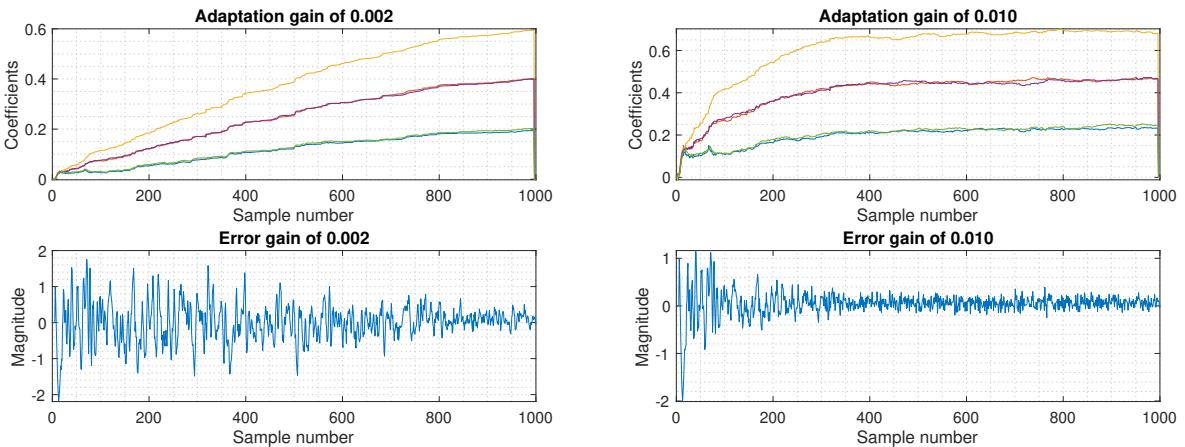


Figure 63: Adaptation gain and error

A small step size (gain) does not allow enough time for the coefficients to converge at optimality if the sample number is too small, this is non ideal as we need a large sample (longer time) for the LMS to learn. A good balance between convergence time and accuracy is needed thus a gain of around 0.02 would be a good fit.

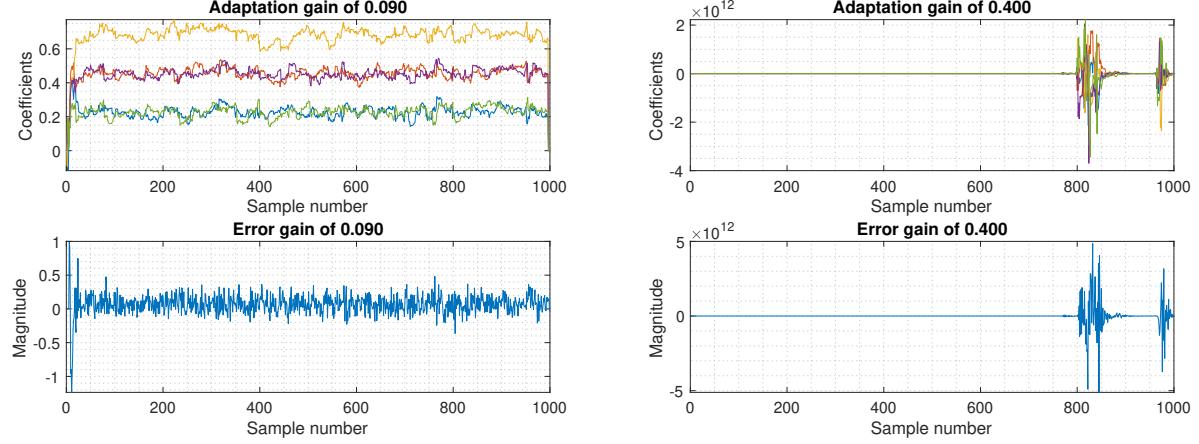


Figure 64: Adaptation gain and error

4.2.2 LMS computational complexity

The convolution has N_w+1 multiplications and N_w additions, so its complexity is $O(N_w)$. The error is just a simple subtraction thus adds no substantial complexity. The multiplication of the error, gain and \mathbf{x} is $2O(N_w+1)$. So the total is around $O(N_w)$ in complexity.

4.3 Gear shifting

The learning rate (gain) is not fixed and changes based on the previous error, adapting to the signal to improve coefficient estimation. It changes based on the gradient descend algorithm. If the error is large, the gain would have a large change, this would not be the case when the error is small. The equation below was found in a IEEE paper¹

$$\mu(n+1) = \mu(n) + \rho e(n+1) \mathbf{x}^T(n) \mathbf{x}(n-1) \quad (20)$$

4.4 Identification of AR processes

The algorithm works correctly if the 2nd order coefficients a_1 and a_2 converge to -0.9 and -0.2 respectively. This is because we have to multiply the original coefficients by a -ve for matlab `filter` to correctly AR filter \mathbf{x} .

¹ Mathews, V. and Xie, Z., 1993. A stochastic gradient adaptive filter with gradient adaptive step size. IEEE Transactions on Signal Processing, 41(6), pp.2075-2087.

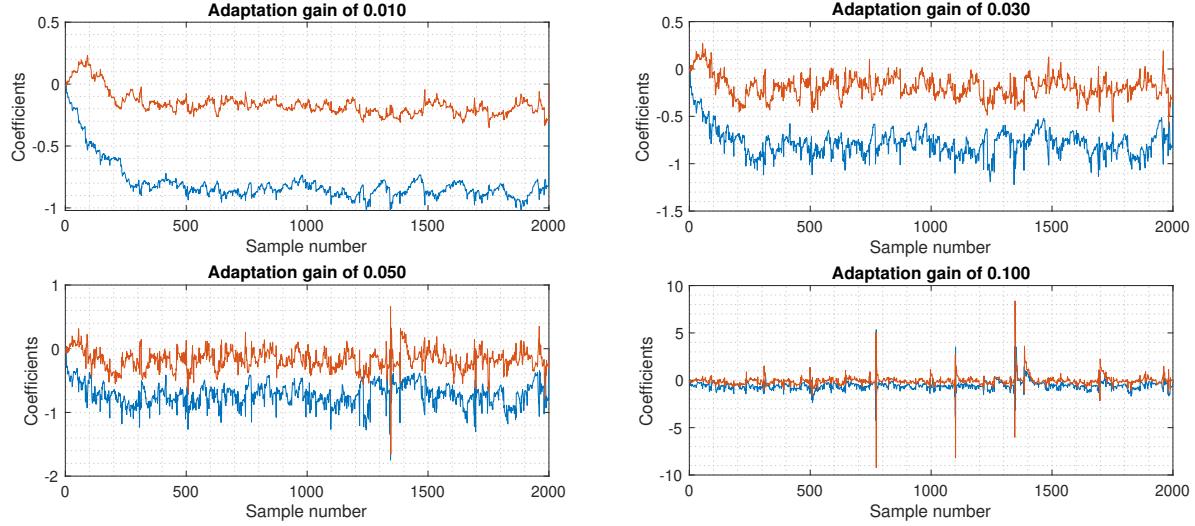


Figure 65: Adaptation gain and convergence

When the adaptation gain is low, the system accurately predicts the model parameters, speed of convergence increases with gain, but it would fluctuate more around the proper coefficients making it harder to predict the actual value. This was explained sections above where, if the step size is too huge, the weights would just jump around the correct value and not reach it. While if the gain is low, weights would converge to the correct value, but it would take a longer time to get there as more steps are required. As gain increases further, the system would be somewhat unstable with sudden peaks in predicted coefficients, because weights would just overshoot the correct value straight away.

4.5 Speech recognition

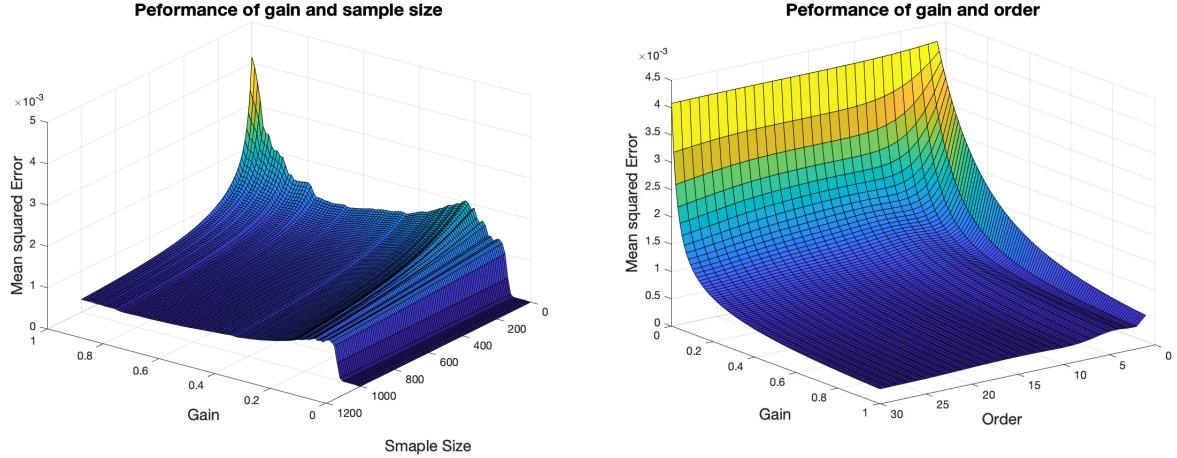


Figure 66: Effects of gain, order and sample size on MSE

A adaptation rate of 1 seems to be the most optimum as the MSE is the minimum. In both figures 66 (representing X) the error increases as gain becomes smaller. The model order does not seem to play a huge role in accuracy, but logically as sample size increases the accuracy would increase.

4.5.1 Optimal filter length

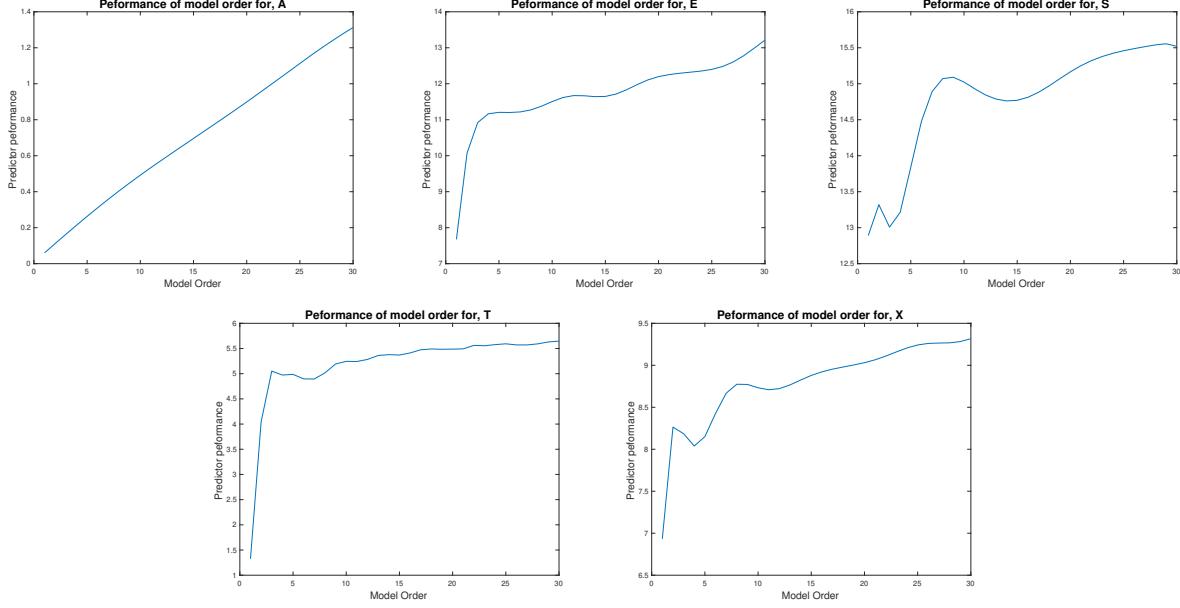


Figure 67: Prediction performance versus model order at gain of 0.5

Each alphabet would have a different optimal model order needed to predict it with confidence. In the figures above, the higher the y axis the better the performance, as the error variance is smaller. For "A" there is no optimal order as performance keeps increases as order increases, unless the optimal is above 30. We would often choose the best results at a lower order to save on computational complexity.

Alphabet	A	E	S	T	X
Optimal Model order	30	10	9	9	8

Table 8: Speech optimal model order

4.6 Sign algorithms

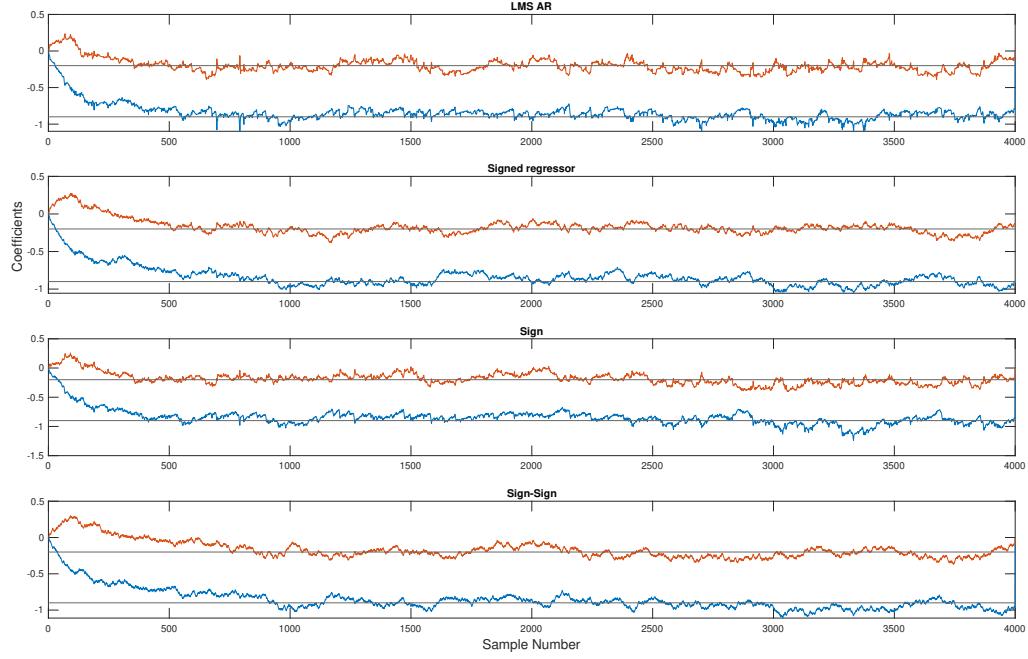


Figure 68: Comparison between LMS algorithms

The sign algorithm had the quickest convergence time, but seemed to vary much more as compared to the regressor and sign-sign. Algorithms which has the signum function would have reduced computational complexity because sign takes either -1 or 1, but a drawback is the rate of convergence is decreased. Sign-sign had the longest convergence, after signed regressor, both had less variance at steady state. The normal LMS algorithm had the most variation at steady state.

5 MLE for signal frequency

5.1

From eqn 45 of the coursework:

$$J(\boldsymbol{\theta}) = \sum_{n=1}^{N-1} (x[n] - A\cos(2\pi f_o n + \phi))^2 \quad (21)$$

Expanding cos with trigonometric identities and substituting $\alpha_1 = A\cos(\phi)$, $\alpha_2 = -A\sin(\phi)$:

$$\begin{aligned} A\cos(2\pi f_o n + \phi) &= A\cos(\phi)\sin(2\pi f_o n) - A\sin(\phi)\cos(2\pi f_o n) \\ &= \alpha_1\sin(2\pi f_o n) + \alpha_2\cos(2\pi f_o n) \end{aligned} \quad (22)$$

Substituting eqn 22 back into eqn 21:

$$J(\boldsymbol{\theta}) = \sum_{n=0}^{N-1} (x[n] - \alpha_1\sin(2\pi f_o n) + \alpha_2\cos(2\pi f_o n))^2 \quad (23)$$

α_1 and α_2 can be treated as linear terms as they do not contain the variable n . 2 new variables are introduced, $\mathbf{c} = [1, \cos(2\pi f_o), \dots]^T$ and $\mathbf{s} = [0, \sin(2\pi f_o), \dots]^T$, which corresponds to the summation of all the terms.

$$J'(\alpha_1, \alpha_2, f_o) = (\mathbf{x} - \alpha_1 \mathbf{c} - \alpha_2 \mathbf{s})^T (\mathbf{x} - \alpha_1 \mathbf{c} - \alpha_2 \mathbf{s}) \quad (24)$$

Concatenating the vectors $\boldsymbol{\alpha} = [\alpha_1, \alpha_2]^T$ and $\mathbf{H} = [\mathbf{c}, \mathbf{s}]$

$$J'(\boldsymbol{\alpha}, f_o) = (\mathbf{x} - \mathbf{H}\boldsymbol{\alpha})^T (\mathbf{x} - \mathbf{H}\boldsymbol{\alpha}) \quad (25)$$

5.2

Minimising $J'(\boldsymbol{\alpha}, f_o)$ is setting its derivative to 0. Expanding eqn 25:

$$\begin{aligned} J'(\boldsymbol{\alpha}, f_o) &= (\mathbf{x} - \mathbf{H}\boldsymbol{\alpha})^T (\mathbf{x} - \mathbf{H}\boldsymbol{\alpha}) \\ &= \mathbf{x}^T \mathbf{x} - \boldsymbol{\alpha}^T \mathbf{H}^T \mathbf{x} - \mathbf{x}^T \mathbf{H} \boldsymbol{\alpha} - \boldsymbol{\alpha}^T \mathbf{H}^T \mathbf{H} \boldsymbol{\alpha} \end{aligned} \quad (26)$$

Setting the derivative of $J'(\boldsymbol{\alpha}, f_o)$ to 0, to find the minimum:

$$\begin{aligned} \frac{\partial(J'(\boldsymbol{\alpha}, f_o))}{\partial \boldsymbol{\alpha}} &= \frac{\partial(\mathbf{x}^T \mathbf{x} - \boldsymbol{\alpha}^T \mathbf{H}^T \mathbf{x} - \mathbf{x}^T \mathbf{H} \boldsymbol{\alpha} - \boldsymbol{\alpha}^T \mathbf{H}^T \mathbf{H} \boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} \\ &= -\frac{\partial(\boldsymbol{\alpha}^T \mathbf{H}^T \mathbf{x})}{\partial \boldsymbol{\alpha}} - \frac{\partial(\mathbf{x}^T \mathbf{H} \boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} + \frac{\partial(\boldsymbol{\alpha}^T \mathbf{H}^T \mathbf{H} \boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} \\ &= -\mathbf{H}^T \mathbf{x} - (\mathbf{x}^T \mathbf{H})^T + \mathbf{H}^T \mathbf{H} \boldsymbol{\alpha} + (\mathbf{H}^T \mathbf{H})^T \boldsymbol{\alpha} \\ &= -2\mathbf{H}^T \mathbf{x} + 2\mathbf{H}^T \mathbf{H} \boldsymbol{\alpha} \\ \therefore \hat{\boldsymbol{\alpha}} &= (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x} \end{aligned} \quad (27)$$

So now we have found in the above the minimising solution for $J'(\boldsymbol{\alpha}, f_o)$. Substituting eqn 27 back into eqn 25:

$$\begin{aligned} J'(\hat{\boldsymbol{\alpha}}, f_o) &= (\mathbf{x} - \mathbf{H}\hat{\boldsymbol{\alpha}})^T (\mathbf{x} - \mathbf{H}\hat{\boldsymbol{\alpha}}) \\ &= (\mathbf{x} - \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x})^T (\mathbf{x} - \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x}) \\ &= \mathbf{x}^T \mathbf{x} - \mathbf{x}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x} - \mathbf{x}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x} + \mathbf{x}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x} \end{aligned}$$

We know that $(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{H} = \mathbf{I}$:

$$\begin{aligned} \therefore J'(\hat{\boldsymbol{\alpha}}, f_o) &= \mathbf{x}^T \mathbf{x} - 2\mathbf{x}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x} + \mathbf{x}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x} \\ &= \mathbf{x}^T \mathbf{x} - \mathbf{x}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x} \end{aligned} \quad (28)$$

It can be seen that eqn 28 contains $\mathbf{x}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x}$ which we have to maximise. And this is linked to the minimising solution $\hat{\boldsymbol{\alpha}}$, as seen in the equations 26-28.

5.3

Maximising $\mathbf{x}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x}$:

$$\begin{aligned} \mathbf{x}^T \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{x} &= \mathbf{x}^T [c \ s] \left(\begin{bmatrix} c \\ s \end{bmatrix} \begin{bmatrix} c & s \end{bmatrix} \right)^{-1} \begin{bmatrix} c \\ s \end{bmatrix} \mathbf{x} \\ &= [x^T c \ x^T s] \begin{bmatrix} c^T c & c^T s \\ s^T c & s^T s \end{bmatrix}^{-1} \begin{bmatrix} c^T x \\ s^T x \end{bmatrix} \\ &= \begin{bmatrix} c^T x \\ s^T x \end{bmatrix}^T \begin{bmatrix} c^T c & c^T s \\ s^T c & s^T s \end{bmatrix}^{-1} \begin{bmatrix} c^T x \\ s^T x \end{bmatrix} \end{aligned} \quad (29)$$

5.4

If $f_o = 0$, vector s is composed of sin, so the whole vector would just be 0. The same goes for when $f_o = 0.5$. The inverse $(\mathbf{H}^T \mathbf{H})^{-1}$ can be shown to be:

$$\begin{aligned} \begin{bmatrix} \mathbf{c}^T \mathbf{c} & \mathbf{c}^T \mathbf{s} \\ \mathbf{s}^T \mathbf{c} & \mathbf{s}^T \mathbf{s} \end{bmatrix}^{-1} &= \begin{bmatrix} \frac{N}{2} & 0 \\ 0 & \frac{N}{2} \end{bmatrix}^{-1} \\ &= \frac{4}{N^2} \begin{bmatrix} \frac{N}{2} & 0 \\ 0 & \frac{N}{2} \end{bmatrix} \end{aligned} \quad (30)$$

Substituting eqn 30 into eqn 29:

$$\begin{aligned} &= \begin{bmatrix} \mathbf{c}^T \mathbf{x} \\ \mathbf{s}^T \mathbf{x} \end{bmatrix}^T \begin{bmatrix} \mathbf{c}^T \mathbf{c} & \mathbf{c}^T \mathbf{s} \\ \mathbf{s}^T \mathbf{c} & \mathbf{s}^T \mathbf{s} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{c}^T \mathbf{x} \\ \mathbf{s}^T \mathbf{x} \end{bmatrix} \\ &= \frac{4}{N^2} \begin{bmatrix} \mathbf{c}^T \mathbf{x} \\ \mathbf{s}^T \mathbf{x} \end{bmatrix}^T \begin{bmatrix} \frac{N}{2} & 0 \\ 0 & \frac{N}{2} \end{bmatrix} \begin{bmatrix} \mathbf{c}^T \mathbf{x} \\ \mathbf{s}^T \mathbf{x} \end{bmatrix} \\ &= \frac{4}{N^2} [\mathbf{c}^T \mathbf{x} \quad \mathbf{s}^T \mathbf{x}] \begin{bmatrix} \frac{N}{2} \mathbf{c}^T \mathbf{x} \\ \frac{N}{2} \mathbf{s}^T \mathbf{x} \end{bmatrix} \\ &= \frac{2}{N} ((\mathbf{c}^T \mathbf{x})^2 + (\mathbf{s}^T \mathbf{x})^2) \end{aligned} \quad (31)$$

Eulers formula can be used to rewrite the periodogram from the coursework into sin and cosines:

$$\begin{aligned} \frac{1}{N} \left| \sum_{n=0}^{N-1} x[n] e^{-j2\pi f \frac{n}{N}} \right|^2 &= \frac{1}{N} \left| \sum_{n=0}^{N-1} x[n] (\cos(2\pi f \frac{n}{N}) + j \sin(2\pi f \frac{n}{N})) \right|^2 \\ &= \frac{1}{N} \sum_{n=0}^{N-1} \left(\sqrt{(x[n] \cos(2\pi f \frac{n}{N}))^2 + (x[n] \sin(2\pi f \frac{n}{N}))^2} \right)^2 \\ &= \frac{1}{N} \sum_{n=0}^{N-1} (x[n] \cos(2\pi f \frac{n}{N}))^2 + (x[n] \sin(2\pi f \frac{n}{N}))^2 \end{aligned} \quad (32)$$

We can now see that eqn 31 and eqn 32 above are similar, thus, the MLE can be obtained from periodogram maximisation. The summation in eqn 32 represents the vectorised form in eqn 31.

5.5

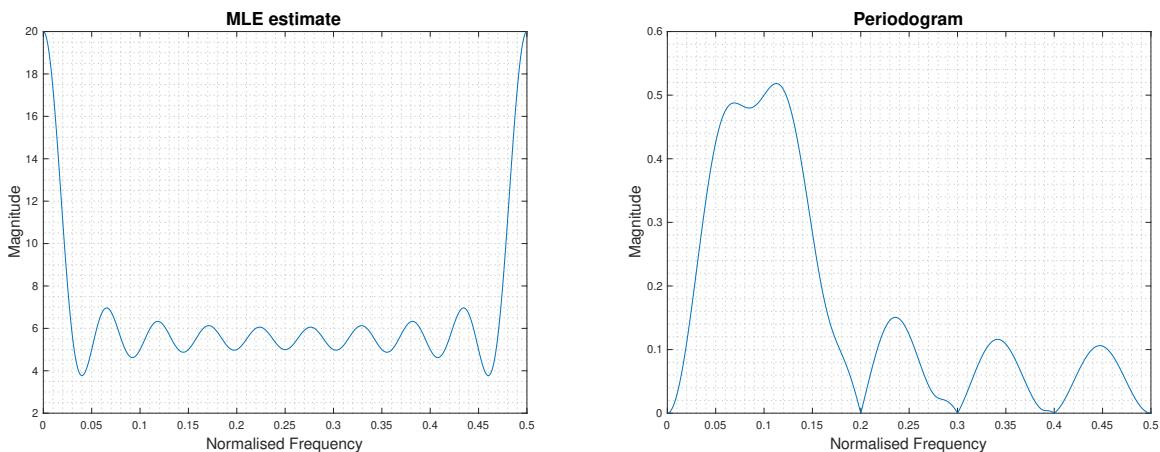


Figure 69: Comparison between MLE and Periodogram

When f_0 approaches both 0 or 0.5, the MLE estimate peaks to its maximum value because the vectors $\mathbf{s} = 0$. This also occurs in the periodogram where \sin becomes 0.