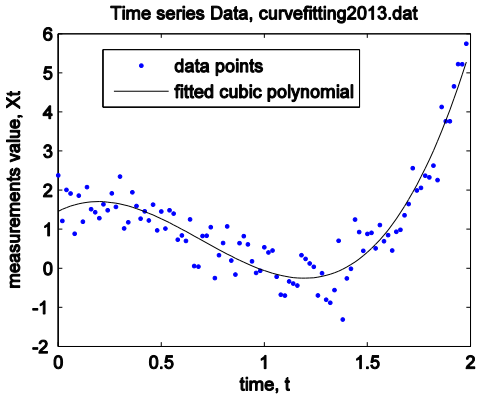
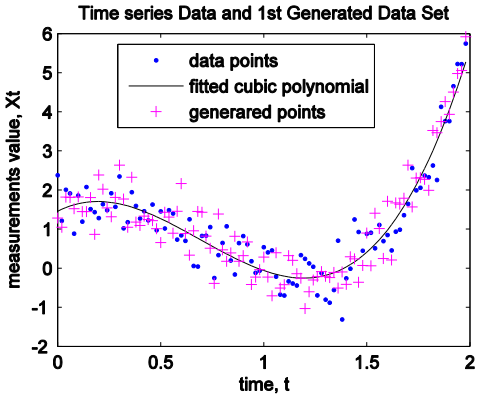
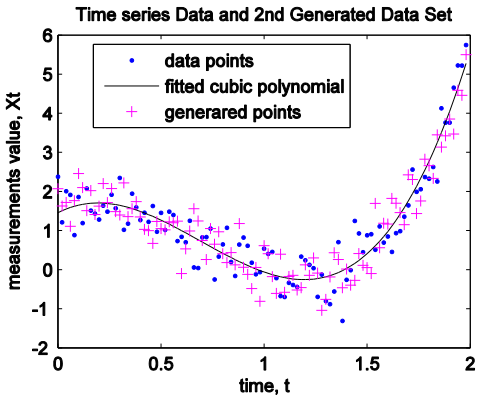
2.1 Curve Fitting and Model Selection

(a)





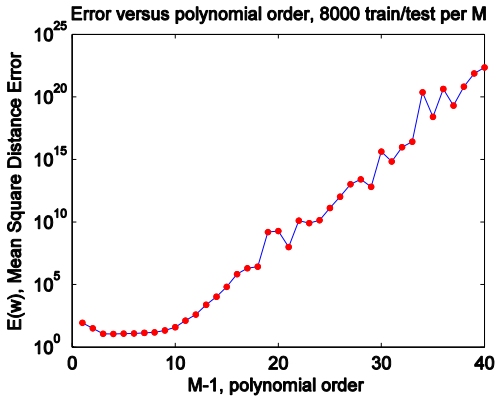


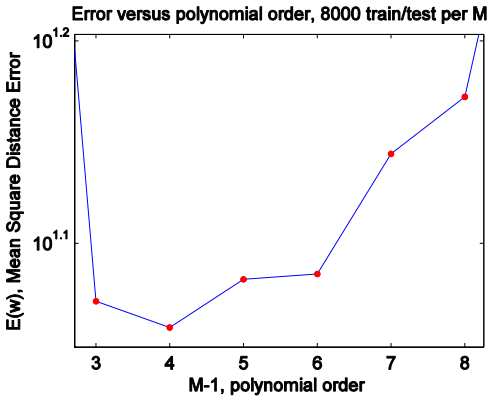
The equation for the cubic curve fitted to curvefittingdata2013.dat is:

$$X_{MLE}(t) = 1.45 + 2.72t - 8.15t^2 + 3.92t^3$$

The unbiased variance of the data points about the cubic curve is $\sigma^2 = 0.206$.

(b)





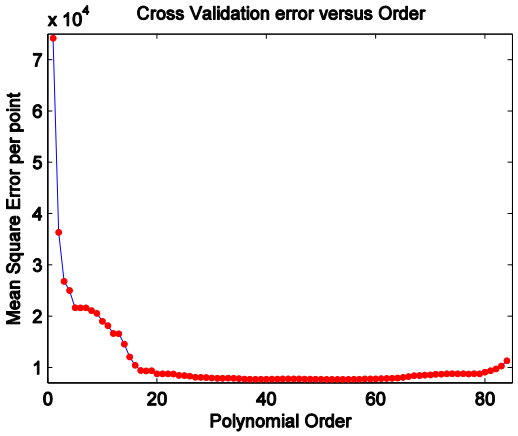
The above graphs were calculated numerically using cross validation. The data set of 100 points was randomly divided into two sets of 50 points, the polynomial was fitted to the first set of 50 points,

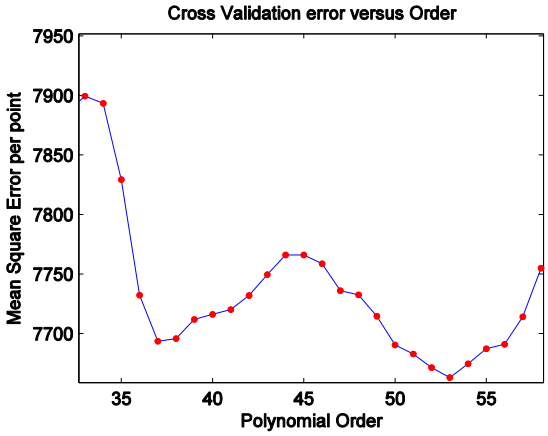
then the square distance error was calculated using the remaining 50 points. This process was repeated 8000 times for each order of the polynomial. Upon close inspection the error is minimised when a 4^{th} order polynomial is fitted parametised by 5 coefficients, M=5. This results is consistent having run the algorithm multiple times.

(c)

FTSE Data

Applying the cross validation to the ftse dataset of 3128 data points suggests that the least square error is minimised by a polynomial of order between 30 and 65. However square error can be greatly reduced and the long term rise of the data can be captured using a polynomial of order 4 or 5 or an exponential curve. Here I shall present the two polynomial curves and cases for choosing either one.

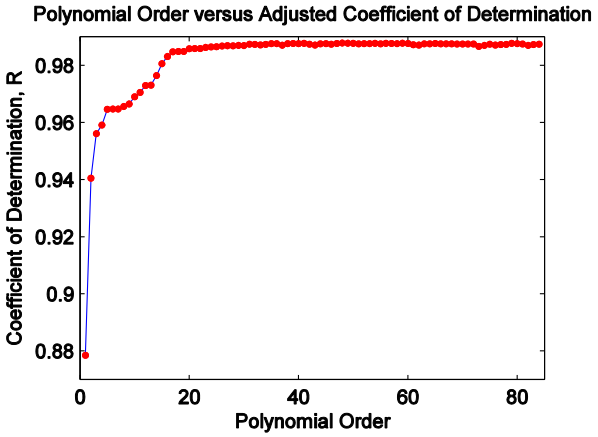


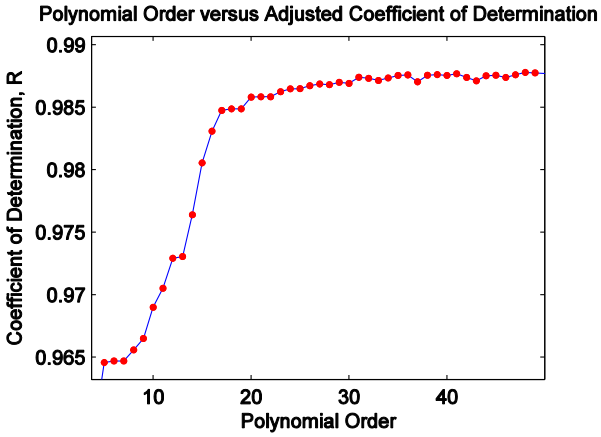


In the above graphs, cross validation was performed splitting the data in half, training and testing 4000 times for each polynomial order. Error is largely reduced by order 5, and reduced further by order 17 and we see that the polynomial order is apparently minimized at 53 using this technique.

The data set is financial data over 12 years, in reality the boom-bust economic cycle is typically 5 to 10 years, government elections and new economic policy decisions are every few years, annual seasonal effects, weekly scandals, and other effects perturb the ftse value giving the data set unpredictable structure on many length scales.

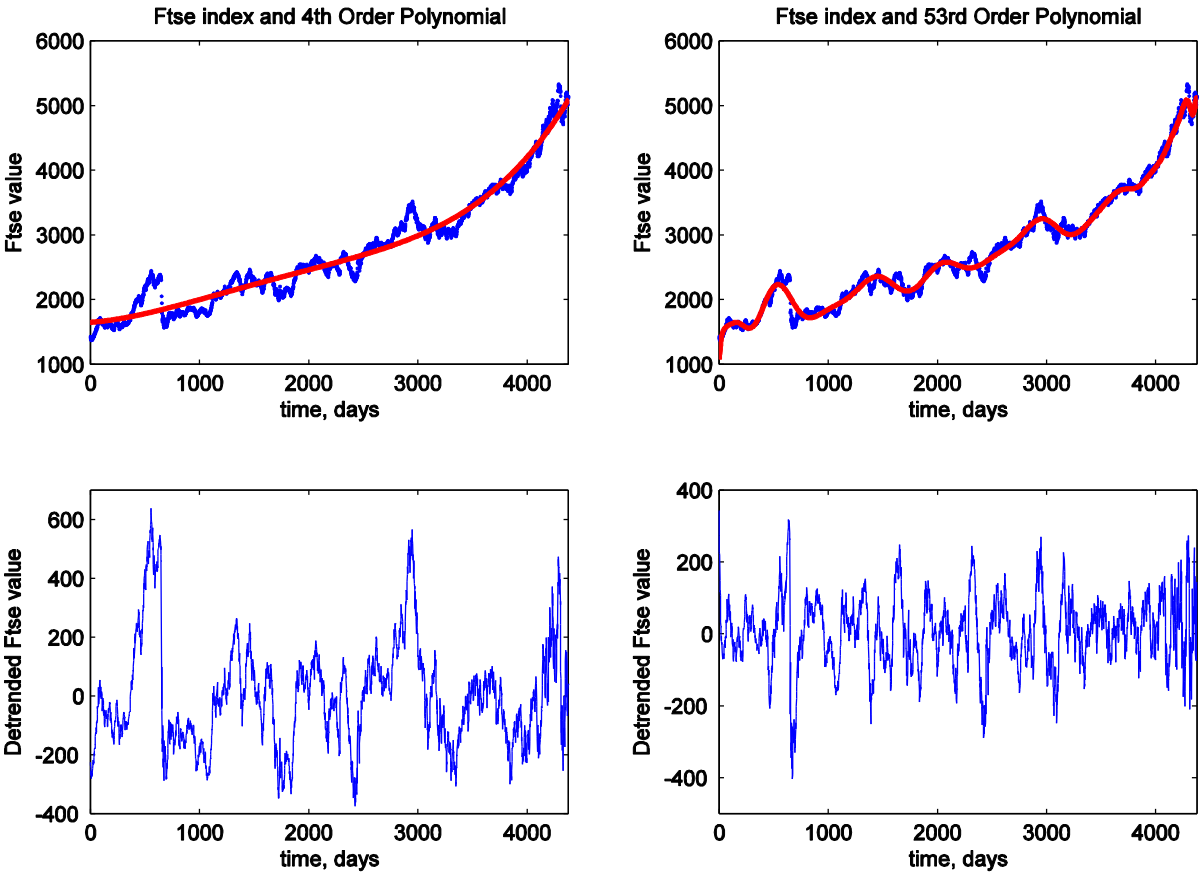
One may fit a polynomial of arbitrarily increasing order and approximate closer and closer the features of smaller scales, ever reducing error. The effects of local features may be studied however the polynomial will quickly go to infinity outside of the interval of the data set, extrapolation is not possible. A lower order polynomial will ignore the short scale structure and may provide reasonable results beyond the scope of the data.





Inspecting the above graphs, like the error, the adjusted coefficient of determination reaches a high level by order 5, then 17 and at order 27 it is no longer consistently monotonically increasing, the length scale is of the order of numerical errors.

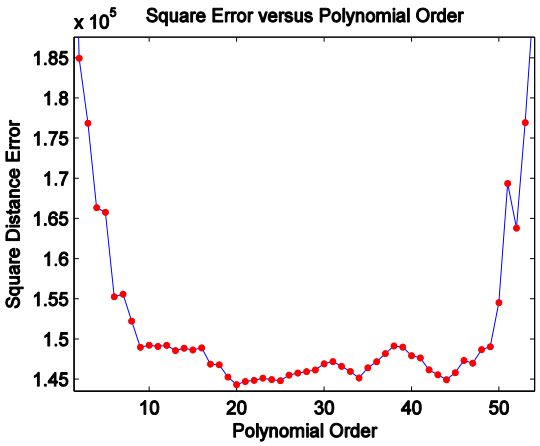
Below are plots of the ftse data with 4th order and 53rd order polynomials and the resulting detrended data.

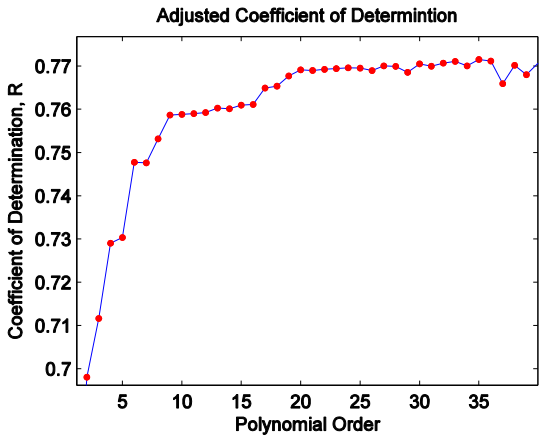


We see that the 53rd order polynomial has captured more of the local structure. The standard deviation of the 4th order detrended data is 178.4, and for the 53rd order 97.3. The higher order

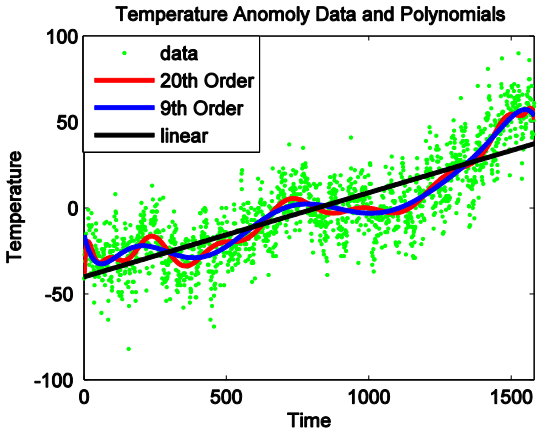
polynomial captures more local structure, however I think higher order polynomials would be hypothetically more accurate and in this case numerical errors, ill-conditioned matrices, limit the fitting of a higher order polynomial, I do not think the minute differences in the square error curve are particularly accurate beyond 17th order. Comparing the stationarity of the 5th and 53rd order detrended data, the 5th has more temporary highs and lows on longer time scales and a wider variance than the 53rd which has had more of those features removed.

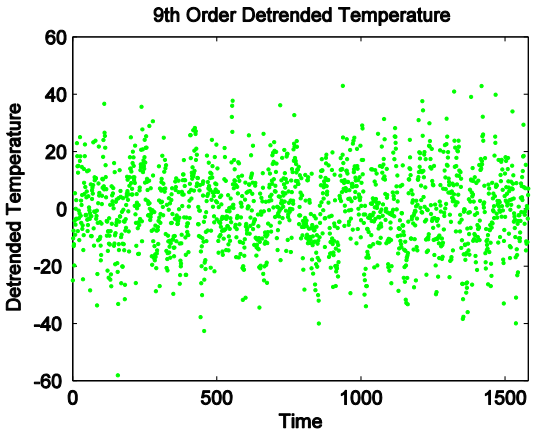
Temperature Anomoly Data



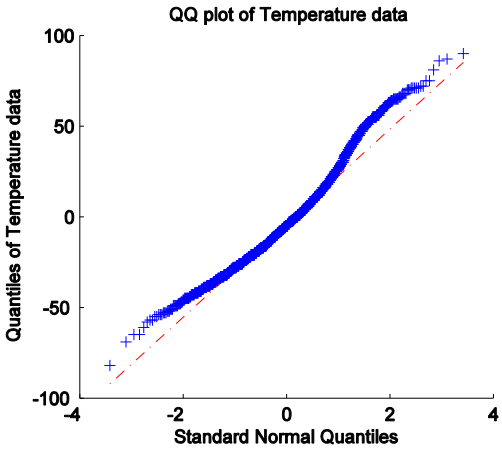


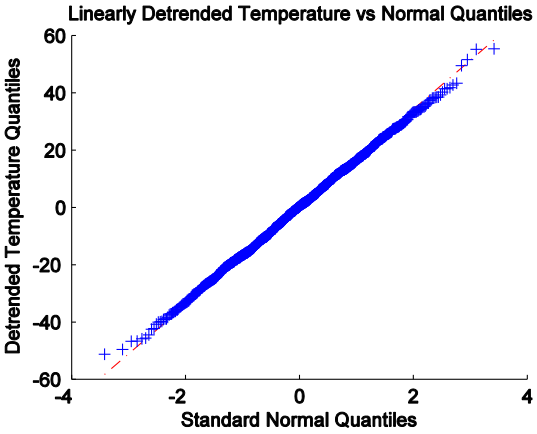
The best fit polynomial is 20th order however the error largely reduced by a 9th order polynomial as is also shown by the determination coefficient.





For linearly detrended data the standard devation is 16.5, for the 9th order detrended data it is lower at 13.6 and 20th order it is largely unchanged at 13.3. The stationarity of the detrended data is like that of an independent and identically distributed Gaussian random variable, the QQ plot below shows that the quantiles of the linearly detrended temperature data match those of the normal distribution, this relationship is true for the 9th and 20th order detrended data too but this is not true for the original data with the trend.

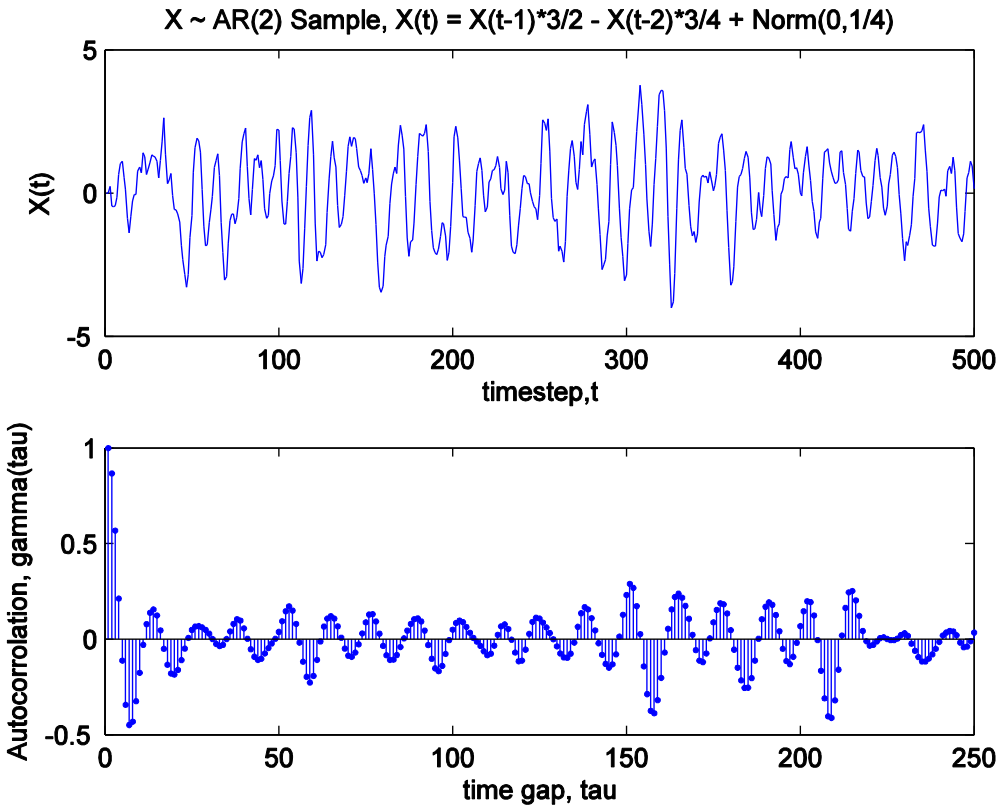


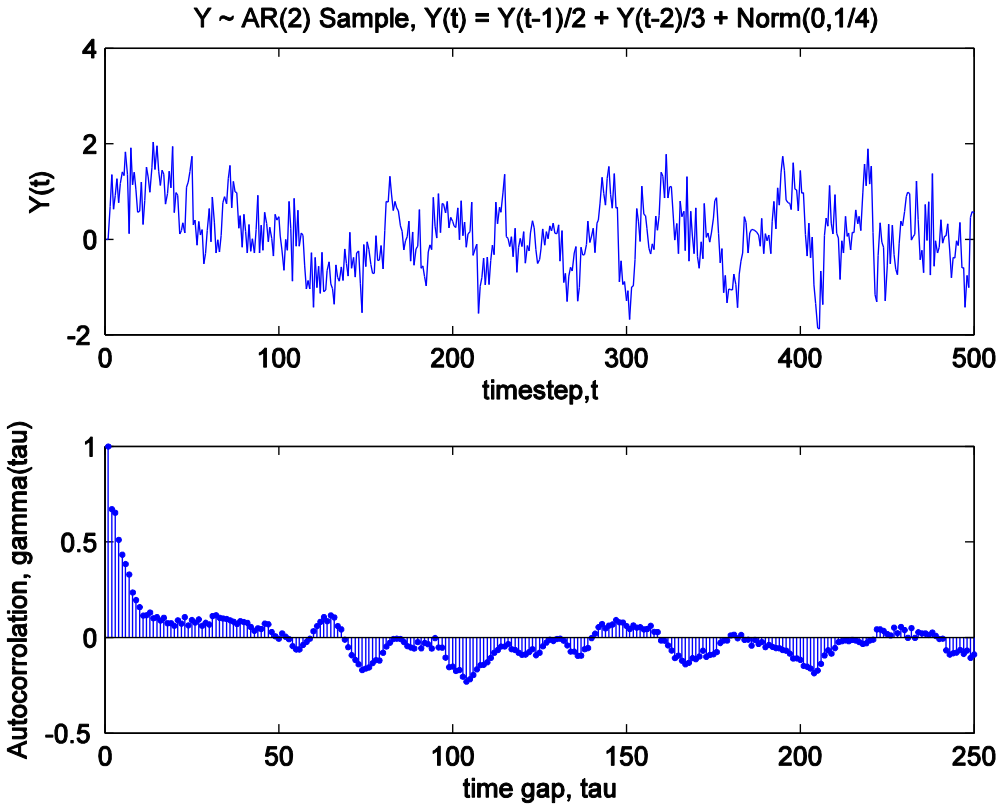


Although this is time series data not random variables, the detrending is intended to remove the time dependence, and the above graphs illustrate this change and the resulting data is Gaussian.

2. **Autoregressive Models**

(a)





(b)

Assuming the first two values are fixed, we may use the remaining data to set up a regression problem for the parameters of the AR(2) time series, c, ϕ_1, ϕ_2. We can find these parameters by mminising the square error bewteen the data and the AR(2) sequence without noise:

$$E = \frac{1}{2}\sum_{i=0}^{N}(X_i - c - \phi_1 X_{i-1} - \phi_2 X_{i-2})^2$$

If the first two terms of the sequence are known then they will not conrtribute to the error and the sum will start from $i = 2$. To minimise the error simply differwentiate with rtespect to the parameters and set to zero:

$$\nabla E = |0\rangle = \begin{pmatrix} \sum_{i=2}^{N}(X_i - c - \phi_1 X_{i-1} - \phi_2 X_{i-2}) * 1 \\ \sum_{i=2}^{N}(X_i - c - \phi_1 X_{i-1} - \phi_2 X_{i-2}) X_{i-1} \\ \sum_{i=2}^{N}(X_i - c - \phi_1 X_{i-1} - \phi_2 X_{i-2})\, X_{i-2} \end{pmatrix}$$

$$= \begin{pmatrix} \sum_{i=2}^{N}(X_i) * 1 \\ \sum_{i=2}^{N}(X_i) X_{i-1} \\ \sum_{i=2}^{N}(X_i)\, X_{i-2} \end{pmatrix} - c \begin{pmatrix} \sum_{i=2}^{N} 1 \\ \sum_{i=2}^{N} X_{i-1} \\ \sum_{i=2}^{N} X_{i-2} \end{pmatrix} - \phi_1 \begin{pmatrix} \sum_{i=2}^{N} X_{i-1} \\ \sum_{i=2}^{N} X_{i-1} X_{i-1} \\ \sum_{i=2}^{N} X_{i-1} X_{i-2} \end{pmatrix} - \phi_2 \begin{pmatrix} \sum_{i=2}^{N} X_{i-2} \\ \sum_{i=2}^{N} X_{i-1} X_{i-2} \\ \sum_{i=2}^{N} X_{i-2} X_{i-2} \end{pmatrix}$$

$$= \begin{pmatrix} \sum_{i=2}^{N}(X_i) * 1 \\ \sum_{i=2}^{N}(X_i) X_{i-1} \\ \sum_{i=2}^{N}(X_i)\, X_{i-2} \end{pmatrix} - \begin{pmatrix} \sum_{i=2}^{N} 1 & \sum_{i=2}^{N} X_{i-1} & \sum_{i=2}^{N} X_{i-2} \\ \sum_{i=2}^{N} X_{i-1} & \sum_{i=2}^{N} {X_{i-1}}^2 & \sum_{i=2}^{N} X_{i-1} X_{i-2} \\ \sum_{i=2}^{N} X_{i-2} & \sum_{i=2}^{N} X_{i-1} X_{i-2} & \sum_{i=2}^{N} {X_{i-2}}^2 \end{pmatrix} \begin{pmatrix} c \\ \phi_1 \\ \phi_2 \end{pmatrix}$$

$$\nabla E = \Psi^T |X\rangle - \Psi\Psi^T |\Phi\rangle = |0\rangle$$

Where $|X\rangle$ is the vector of data points, $|\Phi\rangle$ is the vector of parameters, and Ψ is the design matrix with elements given by the basis functions of each parameter evaluated at the time step, the basis function of the first parameter is simply a constant, so we set is to 1, the basis function of the second parameter is the previous data point, the the basis function of the final parameters is the 2nd previous data point.

$$\Psi_{i,j} = \begin{cases} 1 & for\ j = 1 \\ X_{i-1} & for\ j = 2 \\ X_{i-2} & for\ j = 3 \end{cases}$$

where *i* runs from 2 to N, and $j \in \{1,2,3\}$. The equation for the gradient of the least square error is simply a set of simultaneous linear equations that may be rearranged to find the parameters:

$$\nabla E = \Psi^T|X\rangle - \Psi\Psi^T|\Phi\rangle = |0\rangle$$

$$\Psi\Psi^T|\Phi\rangle = \Psi^T X\rangle$$

$$|\Phi\rangle = (\Psi\Psi^T)^{-1}\Psi^T|X\rangle$$

$(\Psi\Psi^T)^{-1}\Psi^T$ is known as the Moore-Penrose Pseudo Inverse matrix.

(c)

Implementing the above calculation on the generated datasets provided estimates for the parameters:

For the first dataset: $\begin{pmatrix} c \\ \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 3/2 \\ -3/4 \end{pmatrix} \quad \begin{pmatrix} \hat{c} \\ \hat{\phi}_1 \\ \hat{\phi}_2 \end{pmatrix} = \begin{pmatrix} 0.0186 \pm 0.028 \\ 1.4951 \pm 0.018 \\ -0.7281 \pm 0.017 \end{pmatrix}$

For the second dataset: $\begin{pmatrix} c \\ \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1/2 \\ 1/3 \end{pmatrix} \quad \begin{pmatrix} \hat{c} \\ \hat{\phi}_1 \\ \hat{\phi}_2 \end{pmatrix} = \begin{pmatrix} 0.0289 \pm 0.026 \\ 0.4255 \pm 0.063 \\ 0.3656 \pm 0.061 \end{pmatrix}$

Errors in these parameters were calculated using two techniques, firstly the parameters were calculated using 200 consecutive points within the data set, ranging from 1-200 up to 301-500. The above errors are two times standard error of the 300 calculated sets multiplied by a factor of $\sqrt{200}/\sqrt{500}$. This is clearly a rough calculation and shows that the order of magnitude of the uncertainty is less than the value, except in the case of the constant which was originally zero. Hence the parameter values are within the uncertainty of the original values used in the generation of the sequence.

Secondly the ARIMA and 'estimate' functions were used in Matlab which gave the following almost identical results:

For the first dataset: $\begin{pmatrix} \hat{c} \\ \hat{\phi}_1 \\ \hat{\phi}_2 \\ \hat{\sigma}^2 \end{pmatrix} = \begin{pmatrix} 0.0185 \pm 0.022 \\ 1.4951 \pm 0.033 \\ -0.7281 \pm 0.032 \\ 0.247 \pm 0.016 \end{pmatrix}$

For the second dataset: $\begin{pmatrix} \hat{c} \\ \hat{\phi}_1 \\ \hat{\phi}_2 \\ \hat{\sigma}^2 \end{pmatrix} = \begin{pmatrix} 0.029 \pm 0.023 \\ 0.426 \pm 0.039 \\ 0.366 \pm 0.038 \\ 0.258 \pm 0.016 \end{pmatrix}$

The errors calculated by the 'estimate' function are smaller, and suggest that the parameter values are a good approximation.

(d)

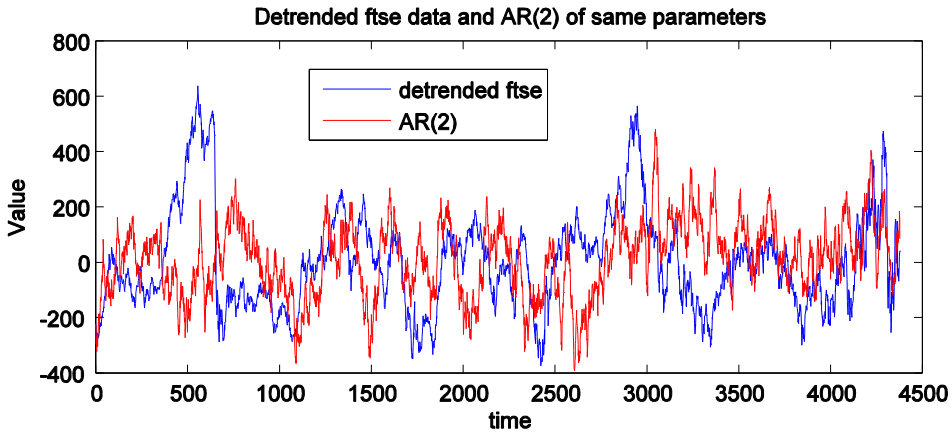
The ftse data was detrended using a 4th order polynomial. This data set is does not have values for weekends, 2 of 7 days a week, so I have assumed that the Friday and Monday values are consecutive.

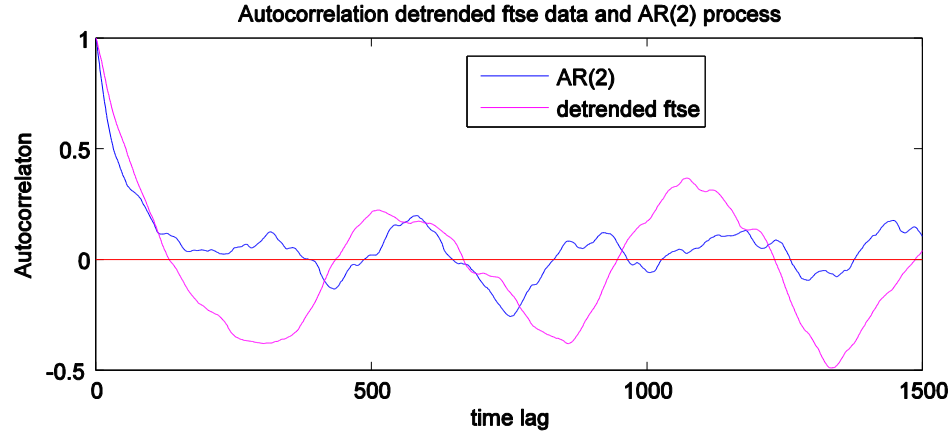
Applying the design matrix calculation produced the following parameters:

$$\begin{pmatrix} \hat{c} \\ \hat{\phi}_1 \\ \hat{\phi}_2 \\ \hat{\sigma}^2 \end{pmatrix} = \begin{pmatrix} 0.0811 \pm 0.90 \\ 1.077 \pm 0.015 \\ -0.087 \pm 0.014 \\ 596.1 \pm 13 \end{pmatrix}$$

Where $\hat{\sigma}^2$ was simply calculated by taking the variance of the differences between each data point and the would-be AR(2) generated term without noise, errors are from the Matlab estimate function, which produced almost identical values for the parameters. The detrended ftse values are typically between -200 and 200, $\hat{\phi}_1 \times 200 \approx 215$, and $\hat{\sigma} \approx 24$, however $\hat{\phi}_2 \times 200 \approx 16$, suggesting the sequence is driven by noise and only the previous term in regions within -200, 200, like a random walk, AR(1) or Markov process and so I do not think AR(2) is a particularly good model for ftse data.

The autocorrelation of the ftse data, quickly drops to zero for short time lags, suggesting it is stationary however there appears not to be other structure on longer time scales which would require an AR(q) process with a greater lag, q>>2.





3. Extracting a Signal from Noise

(a) $$X_t = Acos(2\pi\omega t + \phi) + \xi_t$$

Using the following identity

$$\cos(u + v) = \cos(u)\cos(v) - \sin(u)\sin(v)$$

where $u = 2\pi\omega,\ v = \phi$, we get:

$$X_t = Acos(\phi)cos(2\pi\omega t) - Asin(\phi)\sin(2\pi\omega) + \xi_t$$

$$X_t = B_1 cos(2\pi\omega t) + B_2 \sin(2\pi\omega) + \xi_t$$

where $B_1 = Acos(\phi),\ B_2 = -Asin(\phi)$.

(b)

Two methods were used to calculate the coefficients A and ϕ, firstly using the Moore-Penrose Pseudo inverse matrix to find B_1 and B_2, and secondly using the matlab curve fitting tool to find A and ϕ directly.

Method 1: Using the following Matlab code:

```
%timevals and datavals are vectors of the source data

%vectors of base functions, cos and sin, with omega=1/50
phi1 = cos(timevals*2*pi/50);
phi2 = sin(timevals*2*pi/50);

%the design matrix
PHI=[phi1, phi2];

%pseudoinverse matrix
MPPinverse=(PHI'*PHI)^-1*PHI';

B=MPPinverse*datavals
```

Gave the output B = -0.6301, -2.1550.

Therefore: $Acos(\phi) = -0.6301\ \ Asin(\phi) = 2.1550$

$$tan(\phi) = \frac{2.1550}{-0.6301} \ \rightarrow\ \phi = -1.2863$$

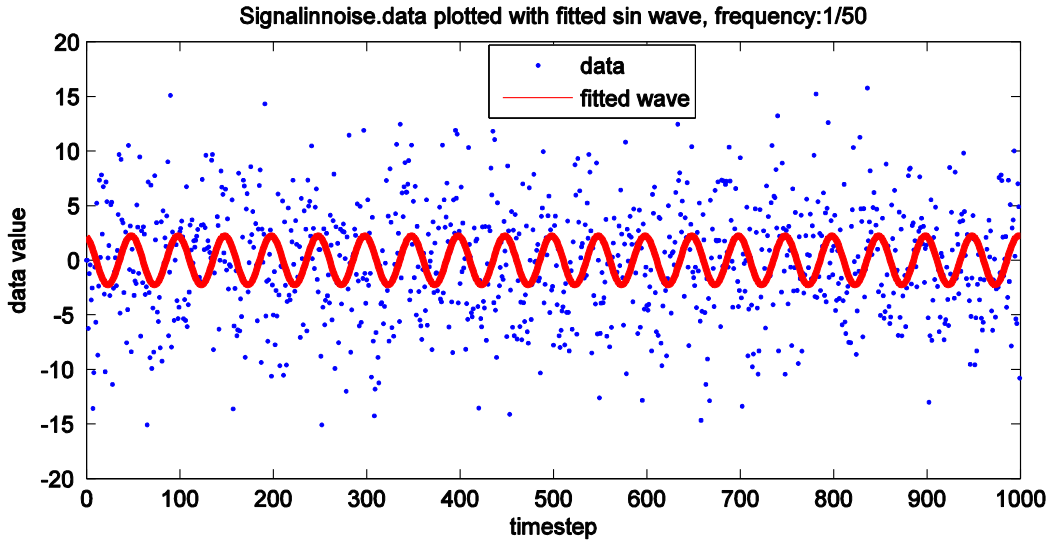
$$A = \frac{2.1550}{\sin(-1.2863)} = -2.2453$$

Method 2: Curve fitting tool

Given the data and the function $y = A * \cos(x\frac{2\pi}{50} + c)$ the curve fitting tool returned:

$$A = -2.245,\ \ c = 4.997 = \phi + 2\pi$$

(c)



The mean of the data points is -0.15, near the centre and the standard devatiation of the sample is 5.25, qqplots of the y values of the data, with and without the wave, show that the noise is Gaussian. The amplitude of the wave is 2.24, less than 1/2 of a standard deviation of the data suggesting this fit is implausible. However, calculating the amplitude for other frequencies, taking the discrete Fourier transform, shows that 1/50 = 0.02 has the highest amplitude, which is maximum at 0.428 standard deviations of the data.