Statistical Models – December 2016 — Assignment 04

Chouliaras Georgios Christos , Jiayang Zhuo
Group 15

I. Computational Problems

Problem 1

(i) In order to analyze the time series of the montly count of sunspots from 1749 to 1997, a graphical representation of the data is essential in order to detect trend and seasonal components. The plot of the time series can be seen in figure (1).

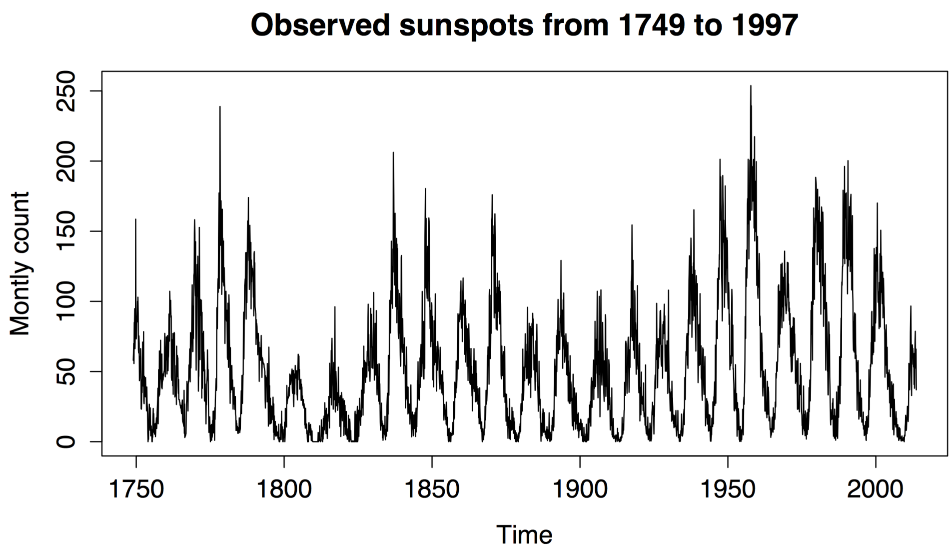


Figure 1: Time series of montly count of observed sunspots for years 1749 to 1997.

From a visual observation of the data it is easily observable that there is a seasonal component as the data show a cyclic behaviour, however it is hard to determine the exact period from this specific plot. Furthermore, it is also hard to detect a general trend from this graph. However, for the years 1880 until 1960 it can be seen that there is an upward trend to the monthly count of the observed sunspots. A way to check whether a time series is stationary is by plotting the autocorrelation function as a function of lag (*correlogram*). The ACF of a stationary time series drops to zero relatively quickly, while the ACF of a non-stationary time series decreases slowly. From figure (2) it can be seen that the autocorrelation function decreases slowly, something that indicates that the time series is non-stationary. Moreover in the right plot, it is easily observed the cyclic behaviour of the time series which indicates seasonality.

(ii) In order to determine the period of the seasonality, a closer plot of the time series with only 44 years is provided in figure (3). As it can be seen from this figure, 4 cycles occurred during

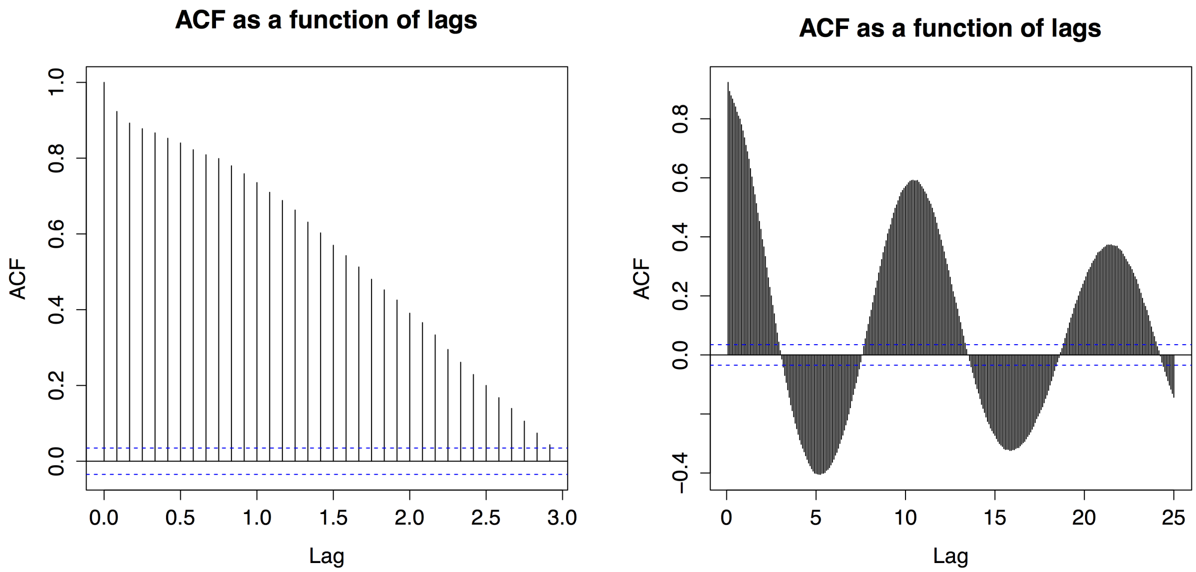


Figure 2: Correlogram of the time series, shows a slowly decreasing ACF and a seasonal pattern.

44 years and this corresponds to 1 cycle per 11 years. Hence, this is an indication that the period is indeed $d = 11$ years. It needs to be noted here, that the same procedure was conducted also to more intervals of 44 years, and again, the same period of 11 years was observed.

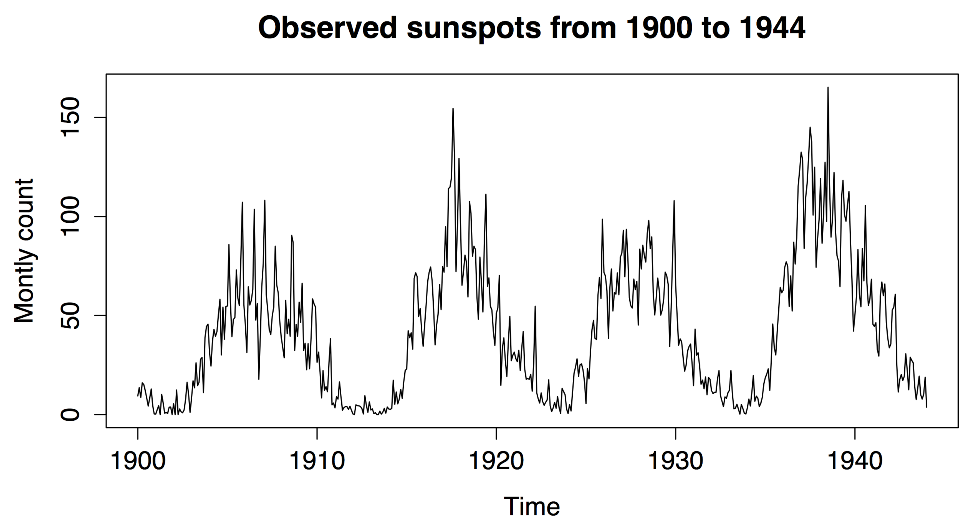


Figure 3: Montly count of the observed sunspots for the year of 1900 to 1944.

(iii) In order to estimate and eliminate the trend and the seasonality two different techniques were used. Firstly, the trend of the time series was obtained using the R function `decompose`.

As it can be seen from figure (4), using this technique an overall trend was obtained which seems to be periodic, hence this trend may also include seasonality. In order to obtain a detrended and deseasonalized time series, the trend was removed from the original time series, and the residual time series can be seen in figure (5).

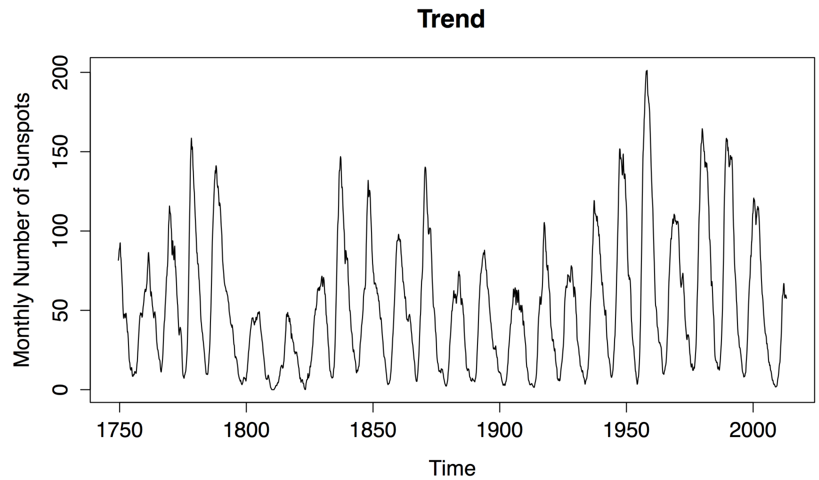


Figure 4: Trend of the time series, obtained by function `decompose`

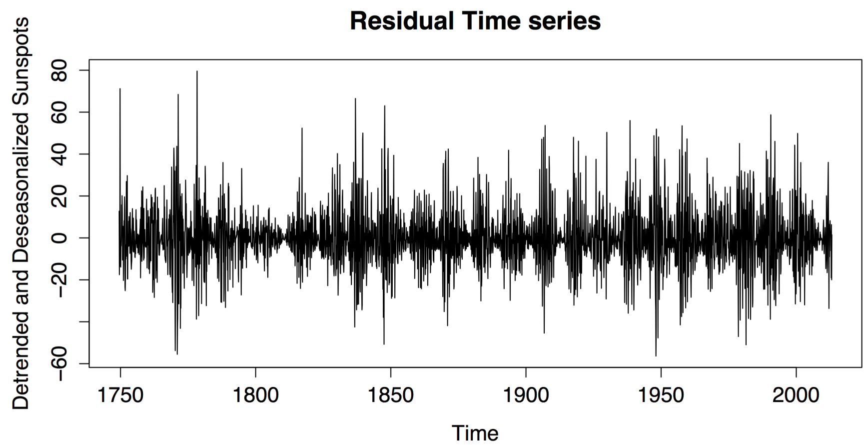


Figure 5: Residual Time Series obtained by function `decompose`

Moreover, the *spline smoothing* technique was conducted which eliminates trend and seasonality together. In order to check which degree of smoothing is the most appropriate for the time series, plots of different values of the parameter *spar*, which determines the degree of smoothing, are presented in figure (6). In this figure, the time series along with the spline smoother for "spar" values of 0.9, 0.5 and 0.1 can be seen. The spline smoothing for spar=0.9 is very smooth and does not capture the form of the time series. The spline smoothing with value of spar=0.5, performs much better, since it is smooth and also follows the data, however it cannot detect most of the peaks and the valleys in the data. On the other hand the spline smoother with spar = 0.1 is pretty rough and follows the data correctly, however a very low value of spar will have as a result, the smoothing to be very good for the given data, but inappropriate for any other data set. From the aforementioned, it can be concluded that the appropriate value for the "spar" parameter is between 0.5 and 0.1 and thus, the spline smoothing for values 0.4, 0.35 and 0.3 have been plotted in figure (7).

In figure (7) it can be seen that for spar=0.4 the spline smoothing is quite smooth but again it

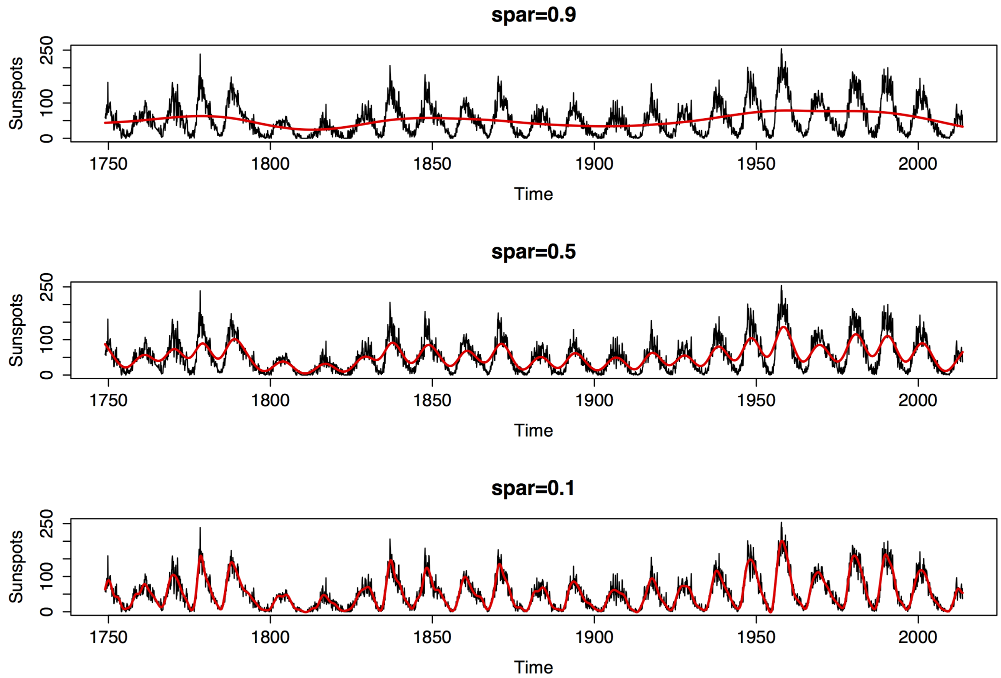


Figure 6: Plot of the time series along with spline smoother for different degrees of smoothing.

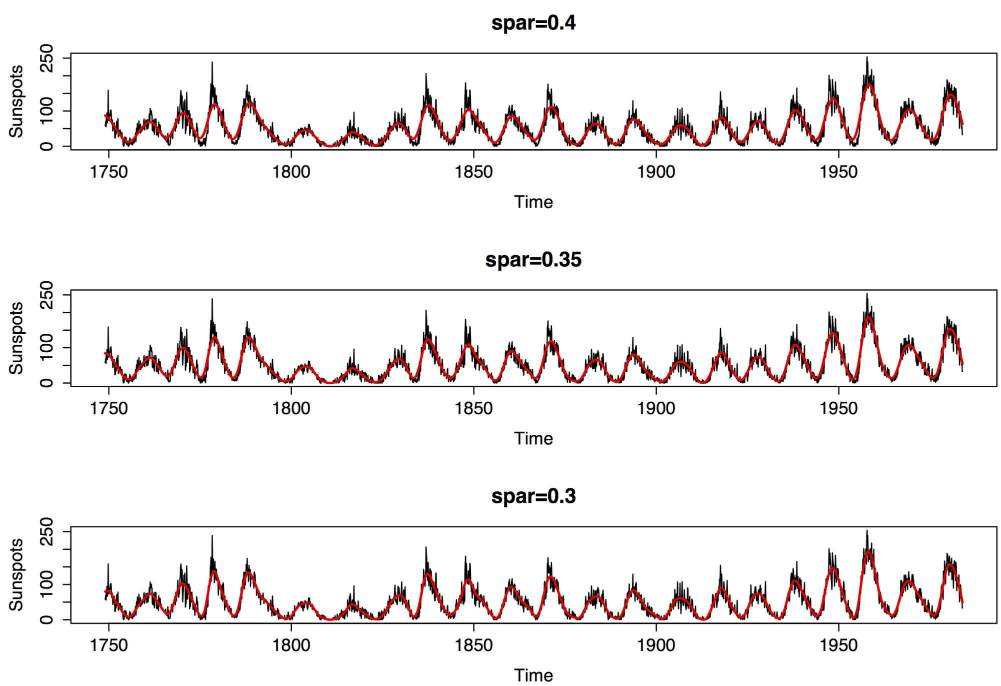


Figure 7: Plot of the time series along with spline smoother for different degrees of smoothing.

does not detect all the valleys and peaks correctly. The spline smoother for spar=0.3 on the other hand is too rough and thus the most appropriate value is spar=0.35 as the spline smoothing is still smooth and also follows the data in a nice way. After choosing the parameter spar as 0.35, the spline smoothing estimated and subtracted from the original time-series. The plot of the detrended and deseasonalized time series can be seen in figure (8). For the detrended and deseasonalized time series, the ACF and PACF also were plotted in figure (9).

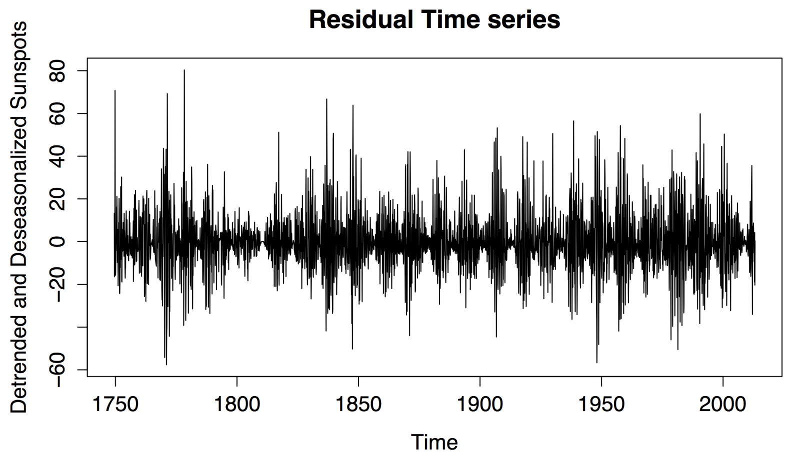


Figure 8: Detrended and deseasonalized time series, using spline smoothing.

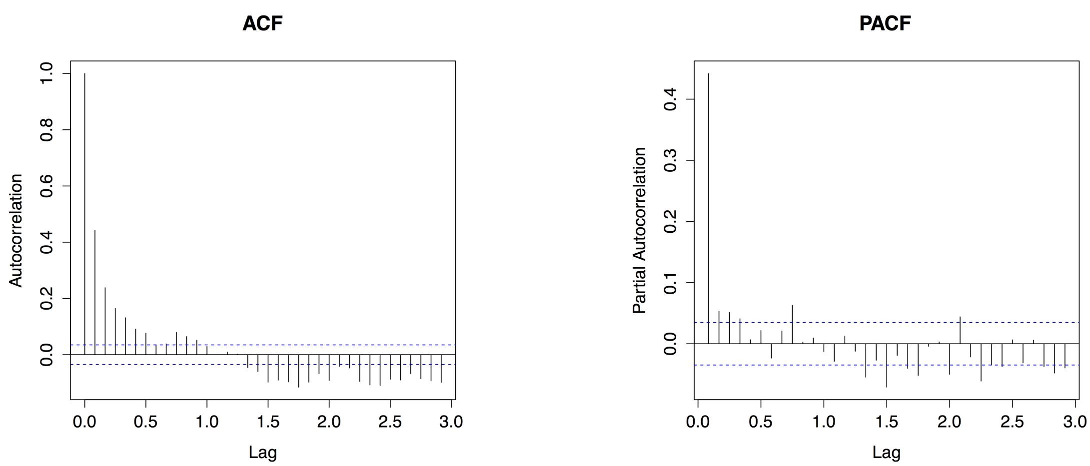


Figure 9: ACF (left) and PACF (right) of the residual time series.

(iv) In figures (5) and (8) can be seen the residual time series which obtained from each technique. Both time series seem to be stationary, as there is no indication of trend or seasonality and the values are also spread around zero. The form of both the residual time series seem to be very similar, indicating that both techniques had similar results. Furthermore, in figure (9) it can be observed, that the correlogram of the autocorrelation function of the residual time series decreases quickly, something that indicates that the time series is stationary. In both residual time series, the *Box-Pierce* test was conducted for many lags in order to check whether the residuals are white noise. The test for both the residuals resulted to very low p - values ($< 2.2 \cdot 10^{-16}$), smaller than the significance level $\alpha = 0.05$, something that indicates that the null hypothesis for independence is being rejected. This means that the residual time series is not white noise, and hence it is not random. This observation leads to the conclusion, that the residual time series might include information, which can be exploited by fitting an appropriate model. From the right plot in figure (9) it can be seen that the PACF cuts off dramatically something that indicates that an Autoregressive model (AR) should be appropriate for modeling the time-series. However, in order to determine the appropriate order of the AR model, several AR models should be tested and the one with the best performance should be selected.

Problem 2

(i) In order to create an ARMA time series, the following parameters determined:

$$\begin{gathered} \text{n} = 500 \\ \text{p} = 2 \\ \text{q} = 1 \\ \alpha_1 = -0.2 \\ \alpha_2 = 0.7 \\ \beta_1 = 0.8 \\ \sigma^2 = 2 \end{gathered}$$

resulting to the time series ARMA(2,1) of the form:

$$X_t = -0.2X_{t-1} + 0.7X_{t-2} + Z_t + 0.8Z_{t-1} \tag{1}$$

where $Z_t \sim WN(0, 2)$. A plot of the time series can be seen in figure (10).

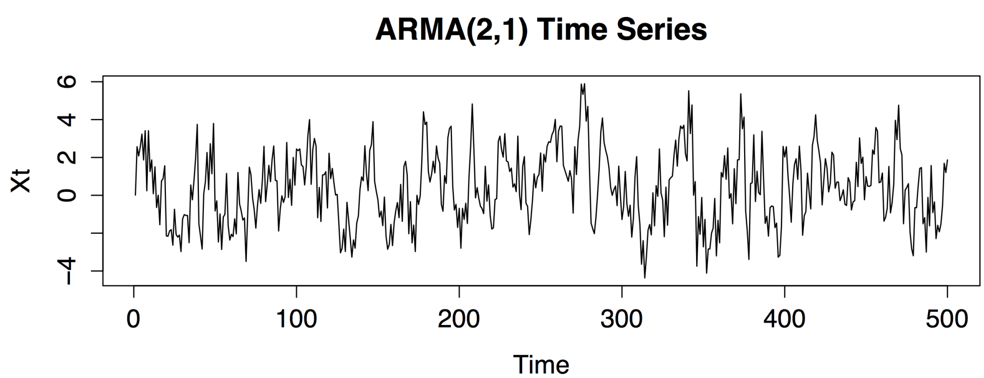


Figure 10: ARMA(2,1) time series

Furthermore, an ARMA(2,1) model was fitted in the generated time series and the estimates of the parameters obtained. The parameter estimates along with 95% confidence intervals for each of the parameters are presented in the table below.

Parameter	Estimate	Real	95 % Confidence Interval
α_1	-0.1395123	-0.2	[-0.247, -0.027]
α_2	0.6940237	0.7	[0.568, 0.719]
β_1	0.7667331	0.8	[0.559, 0.804]
σ^2	1.836	2	

Table 1: Parameter estimates and 95% confidence intervals for the fitted ARMA(2,1) model.

In table (1) it can be seen that the estimates of the parameters are very close to the real parameter values. Moreover, the confidence intervals are quite narrow for all the parameters. These facts indicate a very good fit, however this is not surprising, since the generated time series is an ARMA(2,1), hence the fit of an ARMA(2,1) model was expected to be good. Furthermore, the residuals of the fitted model were obtained and are plotted in figure (11).

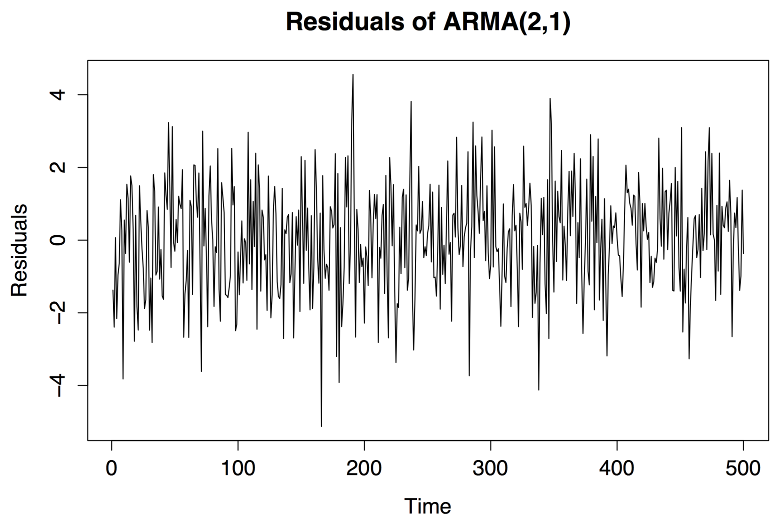


Figure 11: Residuals of ARMA(2,1) fitted model.

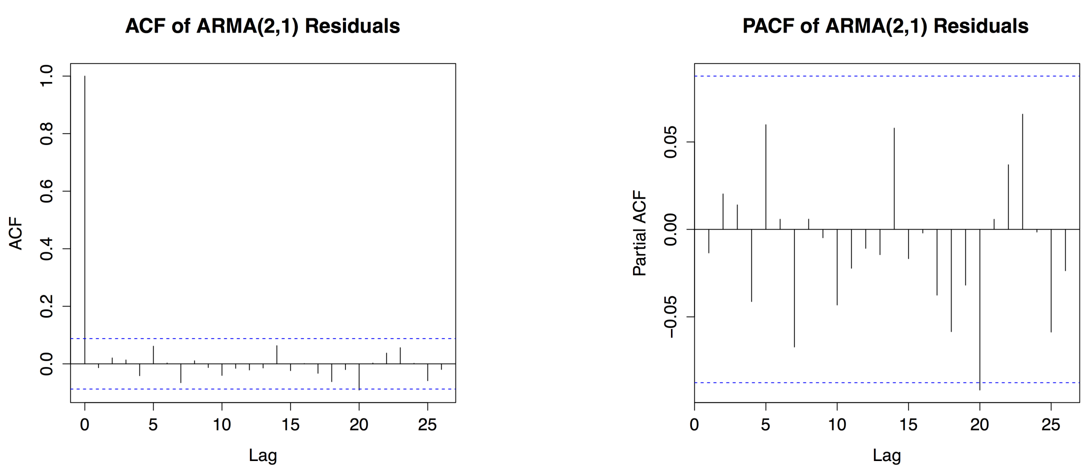


Figure 12: ACF (left) and PACF (right) of the residuals of the ARMA (2,1) fitted model.

From figure (11) it can be seen that the time series of the residuals is completely random and similar to that of white noise, something that indicates a good fit of the model. Moreover, in figure (12) can be seen the ACF and the PACF of the ARMA(2,1) residuals. Both ACF and PACF are almost 0 after lag 0 and this is another indication that the residuals are indeed white noise. Additionaly, the *Box-Pierce* and *Ljung-Box* tests were conducted for many lags in order to verify if the residuals are indeed white noise. Both test produced high values, meaning that they both not reject the null hypothesis of independent time series and hence the residuals are indeed white noise. The plot of the portmanteau test can be seen in the left figure (16).

(ii) Furthermore, an $AR(2)$ model was fitted to the same ARMA(2,1) time series in order to investigate the precision of the parameter estimates. The Yule-walker estimates are provided in the table below.

Parameter	Estimate	Real
α_1	0.4592	-0.2
α_2	0.2711	0.7
σ^2	2.309	2

Table 2: Parameter estimates for the AR(2) model, fitted in the ARMA(2,1) time series.

As it can be seen from the table, the estimates are not good, as they deviate from the real values in a great extent. The reason for this, is that the original time series is an ARMA(2,1) and hence, the bad estimates mean that an AR(2) model is not appropriate for modeling this time series.

A further investigation, was to create a new AR(2) time series with the same parameters, and then fit an AR(2) model, in order to do again a comparison of the Yule Walker estimates and those from the ARMA(2,1) model. The generated AR(2) time series has the following form

$$X_t = -0.2X_{t-1} + 0.7X_{t-2} + Z_t \tag{2}$$

where again $Z_t \sim WN(0, 2)$. The plot of this time series can be seen in figure (13).

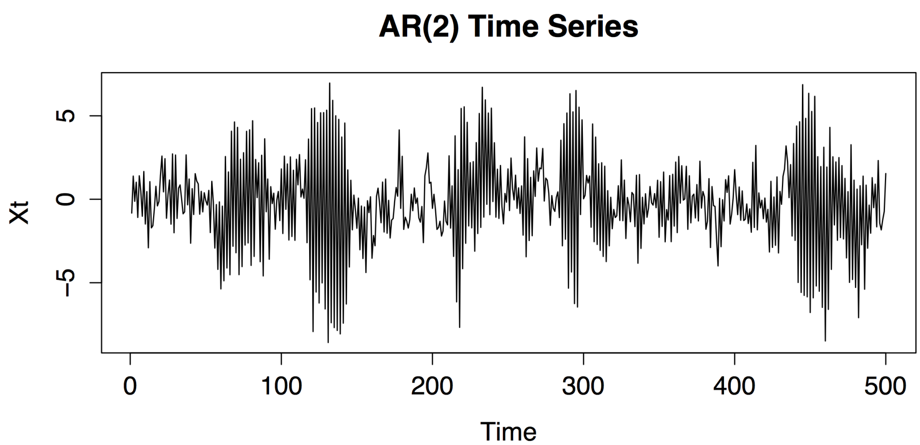


Figure 13: AR(2) time series

Next, an AR(2) model was fitted in this time series and the Yule-Walker estimates were obtained. The estimates along with the 95 % confidence intervals can be seen in the table below.

Parameter	Estimate	Real	95 % Confidence Interval
α_1	-0.2095	-0.2	[-0.260, -0.138]
α_2	0.7046	0.7	[0.651, 0.773]
σ^2	2.061	2	

Table 3: Parameter estimates and 95% confidence intervals for the fitted AR(2,1) model.

From table (3) it can be seen that the Yule Walker estimates are very good, since the estimated values are almost identical to the real ones. Furthermore, the confidence intervals are narrow.

These facts indicate again a very good fit, however it was expected since the original time series was AR(2) and hence an AR(2) model is expected to fit the time series nicely. Through a comparison between the Yule Walker estimates for the AR(2) model and the estimates of the parameters for the ARMA(2,1) model in table (1), it can be seen that the Yule Walker estimates are much better, since they are much closer to the real values. Furthermore, the residuals of the AR(2) model along with the sample ACF and PACF were obtained and are presented in figures (14) and (15).

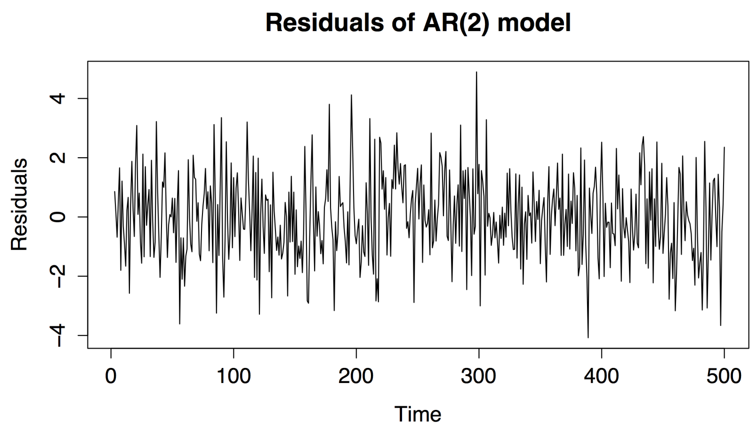


Figure 14: Residuals of AR(2,1) fitted model.

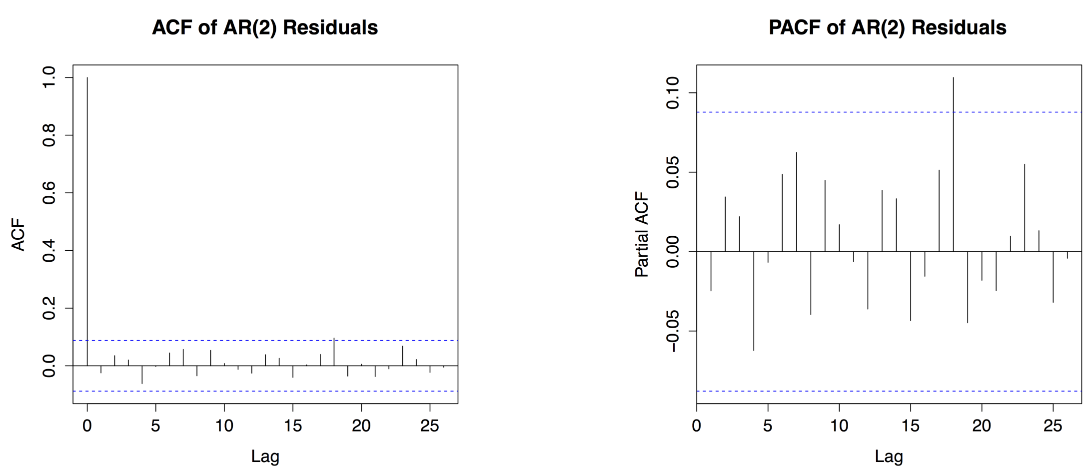


Figure 15: ACF (left) and PACF (right) of the residuals of the AR (2) fitted model.

The residuals of the AR(2) model seem to be random, as there is no indicating pattern and they are also spread around zero. Furthermore, the ACF and the PACF seem to cut-off immediately and this is an indication that the residuals are white noise. In order to verify this, the Portmanteau test was conducted on the residuals for many lags. The Box-Pierce test and the Ljung-Box resulted to p-values greater than the significance level $\alpha = 0.05$ meaning that they do not reject the null hypothesis of independence, and hence the residuals are white noise. The plots of these test can be seen in the right figure of (16). This means that the model fits the data well, however it is noted here again that this was expected, since the AR(2) model was fitted in an AR(2) time series, something that results in a good fit.

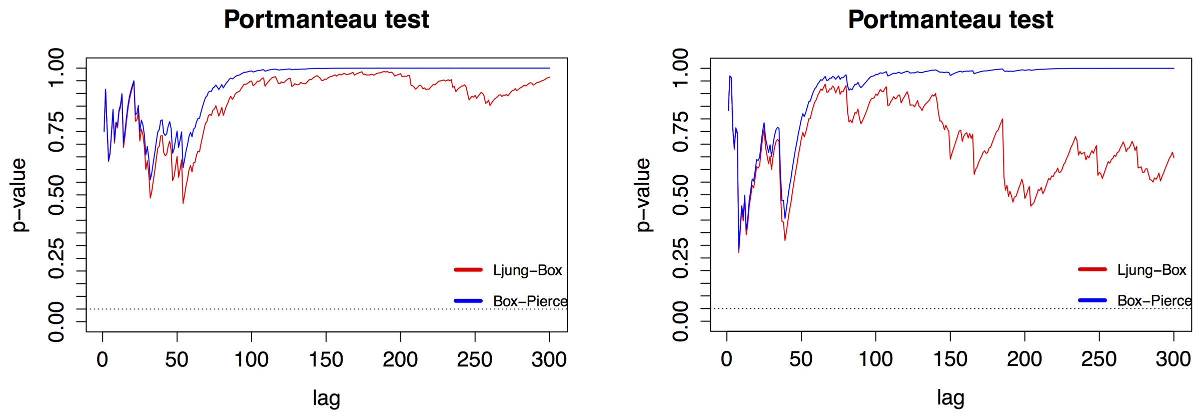


Figure 16: Portmanteau test for ARMA(2,1) model (left) and AR(2) model (right) for many lags.

Problem 3

(i) The daily closing prices of the European stock indices *DAX, SMI, CAC* and *FTSE* from 1991 to 1998 have been plotted and are presented in figure (17).

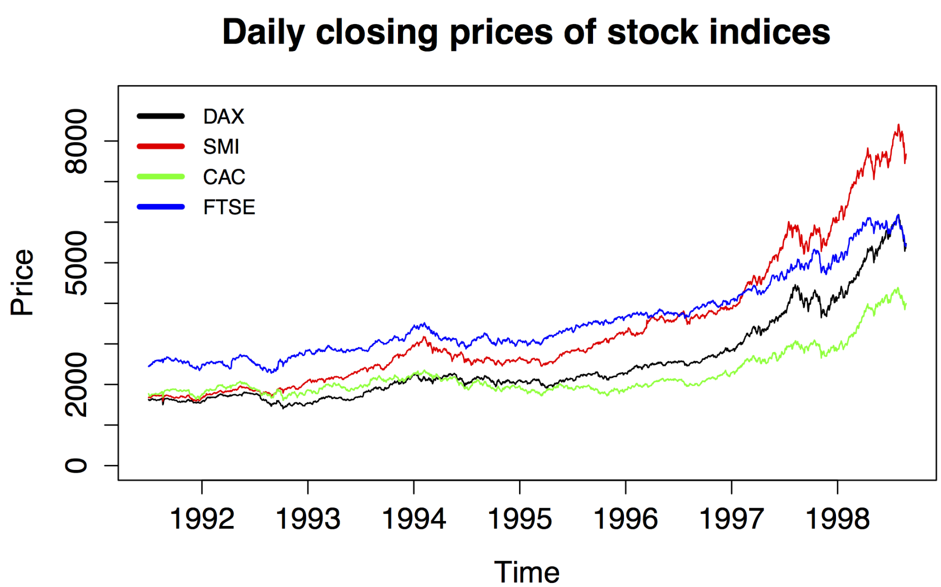


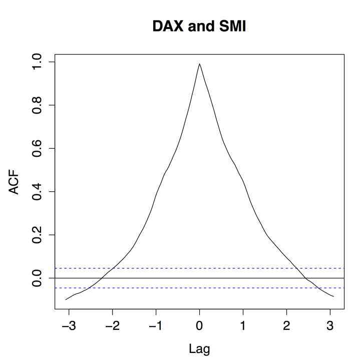
Figure 17: Daily closing prices of four major EU stock indices from 1991 to 1998.

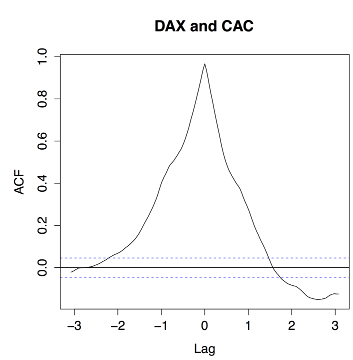
From this figure, comparisons between the the historical evolution of the four time series can be made. Firstly, all the indices show a general upward trend however there are some differences on how fast each index grows. All the indices grow from 1991 to 1994 and then they make a small dip, due to the known bond-crash in 1994. All the indices begin to grow again after the middle of 1995 except of CAC which begins to grow again after the middle of 1996. After 1997 there is a great similarity in the evolution of SMI and DAX (in the middle of 1997 there was a steep rise for both the indices) however the closing prices for SMI are in higher levels. For the same years, there is also a similarity in the evolution of CAC and FTSE however the closing prices of FTSE are in higher levels. In the end of 1997 all the indices show a sudden dip (probably due to the mini-crash at 27 of October 1997) and after that all seem to grow relatively fast. Moreover, all the indices seem to fall again in the end of 1998, while the closing prices of FTSE and DAX seem to coincide in this downward trend in the end of 1998. Another interesting observation, is that all the indices had closing prices under 2000 in 1991, except of FTSE which started at a price about 2500. However, after 1997 the price of FTSE became equal with the one of SMI and after that point it remained in lower levels compared to that of SMI. In general, one could say that the SMI index showed the best course, while CAC had the most steady course, but with the lowest closing price compared to the rest of the indices. In order to further analyze the relationship between the indices the cross-correlations of each pair have been computed and the maximum cross-correlation along with the lag at which it occurred are presented in table (4). Furthermore, the cross-correlation plots are shown in figure (20).

Pair of Indices	Maximum Cross-Correlation	Lag
DAX & SMI	0.9911	0
DAX & CAC	0.9662	0
DAX & FTSE	0.9751	0
SMI & CAC	0.9468	0
SMI & FTSE	0.9899	0
CAC & FTSE	0.9157	0

Table 4: Maximum Cross-Correlation for each pair of indices along with the lag of the maximum.

From the table it can be seen that the most correlated pair is *DAX & SMI*, while the least correlated pair is *CAC & FTSE*. The high correlation between DAX and SMI indices, justifies the fact that these indices show a similar behavior in figure (17). The reason for low correlation between CAC and FTCE (actually it is not low at all, however it is the lowest between the pairs) might be that for the years between 1995 and the middle of 1996, FTSE seems to go upwards, while CAC seems to decline a bit and then remain steady. For all the pairs the maximum cross-correlation occurred at lag 0 and as it can be seen from the plots, the correlation becomes smaller as the lags move away from 0. In the financial context the cross-correlation is a common measure of relationship between indices and also stocks. It is often used as an input to create optimal portfolios, as it refers to the extent to which performances of different asset classes move in relation to each other. Thus, lower correlations among different indices in a portfolio imply that diversification could create the maximum profit. From the plots it can be seen that the indices are in general highly correlated, something that means that their prices tend to respond to market news in the same direction at the same time.





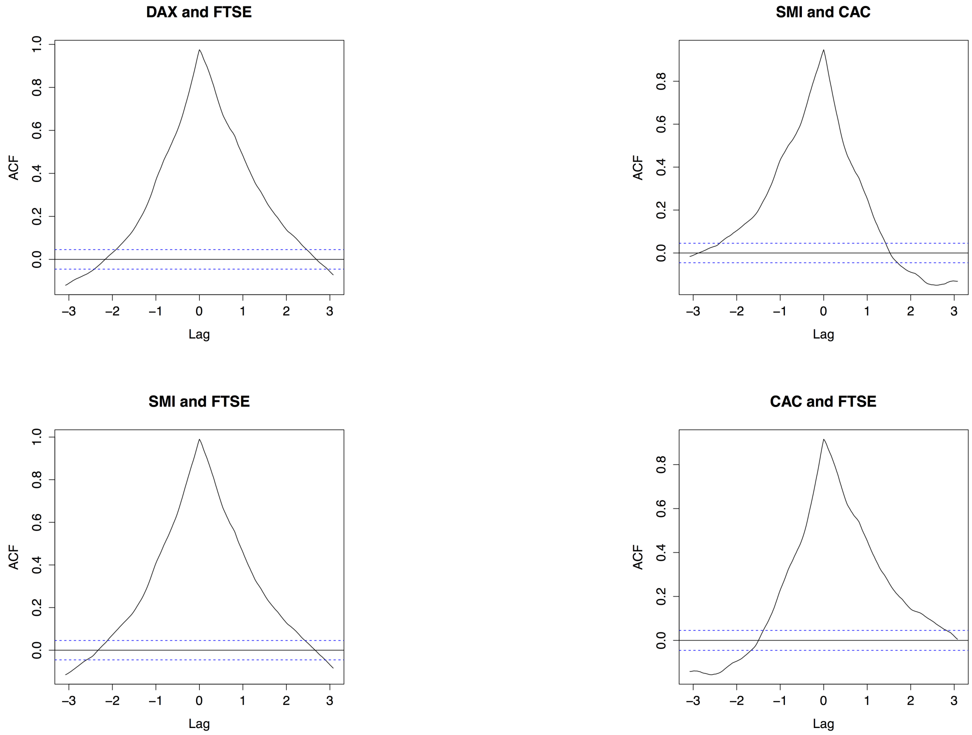


Figure 20: Cross-correlation plots for all the pairs of indices.

II. Theoretical Problems

Problem 1

(i) For the mean function m_t holds:

$$m_t = \mathbb{E}[X_t] = \mathbb{E}[cos(\lambda t + U)] \tag{3}$$

Using the definition of expectation and the fact that the probability density function of a uniformly distributed random variable in $(-\pi, \pi]$ is: $\frac{1}{2\pi}\mathbb{1}_{(-\pi,\pi]}(u)$ we have

$$\begin{aligned}
\mathbb{E}[cos(\lambda t + U)] &= \int_{-\infty}^{\infty} cos(\lambda t + u)\frac{1}{2\pi}\mathbb{1}_{(-\pi,\pi]}(u)du \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} cos(\lambda t + u)du \\
&= \frac{1}{2\pi}\left[sin(\lambda t + u)\right]_{-\pi}^{\pi} \\
&= \frac{1}{2\pi}(sin(\lambda t + \pi) - sin(\lambda t - \pi)) \\
&= \frac{1}{2\pi}(-sin(\lambda t) - (-sin(\lambda t))) \\
&= \frac{1}{2\pi}(-sin(\lambda t) + sin(\lambda t)) \\
&= 0
\end{aligned} \tag{4}$$

Now for the autocovariance function we have

$$\begin{aligned}\gamma_x(s,t) &= Cov(X_s, X_t)\\ &= \mathbb{E}[X_s X_t] - \mathbb{E}[X_s]\mathbb{E}[X_t]\\ &= \mathbb{E}[cos(\lambda s + U)cos(\lambda t + U)]\\ &= \int_{-\infty}^{\infty} cos(\lambda s + u)cos(\lambda t + u)\frac{1}{2\pi}\mathbb{1}_{(-\pi,\pi]}(u)du\\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} cos(\lambda s + u)cos(\lambda t + u)du \qquad (5)\end{aligned}$$

Now using the known formula: $cosa - cosb = \frac{1}{2}(cos(a-b) + cos(a+b))$ we have

$$\begin{aligned}\gamma_x(s,t) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{1}{2}(cos(\lambda s + u - \lambda t - u) + cos(\lambda s + u + \lambda t + u))du\\ &= \frac{1}{4\pi}\int_{-\pi}^{\pi} cos(\lambda(s-t)) + cos(\lambda(s+t) + 2u)du\\ &= \frac{1}{4\pi}\int_{-\pi}^{\pi} cos(\lambda(s-t))du + \frac{1}{4\pi}\int_{-\pi}^{\pi} cos(\lambda(s+t)+2u)du\\ &= \frac{1}{4\pi}cos(\lambda(s-t))[\pi+\pi] + \frac{1}{4\pi}\,[sin(\lambda(s+t)+2u)]_{-\pi}^{\pi}\\ &= \frac{1}{2}cos(\lambda(s-t)) + \frac{1}{4\pi}\,(sin(\lambda(s+t)+2\pi) - sin(\lambda(s+t)-2\pi))\\ &= \frac{1}{2}cos(\lambda(s-t)) + \frac{1}{4\pi}\,(sin(\lambda(s+t)) - sin(\lambda(s+t)))\\ &= \frac{1}{2}cos(\lambda(s-t)) \qquad (6)\end{aligned}$$

Since the autocovariance function depends only on the time difference $(s-t)$ (lag), and also $\mathbb{E}[X_t] = 0$ does not depend on t the time series $\{X_t\}$ is (weakly) stationary.

(ii) For the mean function m_t we have

$$\begin{aligned}m_t &= \mathbb{E}[X_t]\\ &= \mathbb{E}[Acos(\lambda t) + Bsin(\lambda t)]\\ &= \mathbb{E}[Acos(\lambda t)] + \mathbb{E}[Bsin(\lambda t)]\\ &= cos(\lambda t)\mathbb{E}[A] + sin(\lambda t)\mathbb{E}[B]\\ &= 0 \qquad (7)\end{aligned}$$

For the autocovariance function we have

$$\begin{aligned}\gamma_x(s,t) &= Cov(X_s, X_t)\\ &= \mathbb{E}[X_s X_t] - \mathbb{E}[X_s]\mathbb{E}[X_t]\\ &= \mathbb{E}[(Acos(\lambda s) + Bsin(\lambda s))(Acos(\lambda t) + Bsin(\lambda t))]\\ &= \mathbb{E}[A^2cos(\lambda s)cos(\lambda t) + ABsin(\lambda s)cos(\lambda t) + ABcos(\lambda s)sin(\lambda t) + B^2sin(\lambda s)sin(\lambda t)]\\ &= cos(\lambda s)cos(\lambda t)\mathbb{E}[A^2] + sin(\lambda s)cos(\lambda t)\mathbb{E}[AB] + cos(\lambda s)sin(\lambda t)\mathbb{E}[AB] + sin(\lambda s)sin(\lambda t)\mathbb{E}[B^2] \qquad (8)\end{aligned}$$

But

$$\begin{aligned} Var[A] &= \sigma^2 \\ &= \mathbb{E}[A^2] - (\mathbb{E}[A])^2 \\ &= \mathbb{E}[A^2] \end{aligned} \tag{9}$$

The same holds for $Var[B]$. Hence $\mathbb{E}[A^2] = \mathbb{E}[B^2] = \sigma^2$. Moreover since A, B are uncorrelated it holds that $\mathbb{E}[AB] = \mathbb{E}[A]\mathbb{E}[B] = 0$. Thus, equation (8) becomes

$$\begin{aligned} \gamma_x(s,t) &= \sigma^2(cos(\lambda s)cos(\lambda t)) + \sigma^2 sin(\lambda s)sin(\lambda t) \\ &= \sigma^2\left(\frac{1}{2}\left(cos(\lambda s - \lambda t) + cos(\lambda s + \lambda t) + cos(\lambda s - \lambda t) - cos(\lambda s + \lambda t)\right)\right) \\ &= \sigma^2\left(\frac{1}{2}2cos(\lambda s - \lambda t)\right) \\ &= \sigma^2 cos(\lambda(s-t)) \end{aligned} \tag{10}$$

Again, the autocovariance function depends only on lag $(s-t)$ and not on time and also $\mathbb{E}[X_t] = 0$ and does not depend on time, so X_t is (weakly) stationary.

Problem 2

(i) For $\nabla_d X_t$ we have

$$\begin{aligned} \nabla_d X_t &= \nabla_d(a + bt + s_t + Y_t) \\ &= \nabla_d(a) + \nabla_d(bt) + \nabla_d(s_t) + \nabla_d(Y_t) \end{aligned} \tag{11}$$

But $\nabla_d(a) = 0$ since a is a constant and equation (11) becomes

$$\begin{aligned} \nabla_d X_t &= b\nabla_d t + \nabla_d(s_t) + \nabla_d(Y_t) \\ &= b(t-(t-d)) + s_t - s_{t-d} + Y_t - Y_{t-d} \\ &= bd + Y_t - Y_{t-d} \end{aligned} \tag{12}$$

Since $s_t = s_{t-d}$, as s_t has period d.
Now for $\nabla\nabla_d X_t$ we have

$$\begin{aligned} \nabla\nabla_d X_t &= \nabla(bd + Y_t - Y_{t-d}) \\ &= \nabla(bd) + \nabla Y_t - \nabla Y_{t-d} \\ &= Y_t - Y_{t-1} - Y_{t-d} + Y_{t-d-1} \\ &= Y_t - Y_{t-1} - (Y_{t-d} - Y_{t-d-1}) \\ &= \nabla Y_t - \nabla Y_{t-d} \\ &= \nabla(Y_t - Y_{t-d}) \\ &= \nabla\nabla_d Y_t \end{aligned} \tag{13}$$

(ii) $\nabla_d^2 W_t$ can be rewritten as

$$\begin{aligned}
\nabla_d^2 W_t &= \nabla_d \nabla_d W_t \\
&= W_t - W_{t-d} - (W_{t-d} - W_{t-2d}) \\
&= W_t - 2W_{t-d} + W_{t-2d} \\
&= at^2 + bts_t + Y_t - 2(a(t-d)^2 + b(t-d)s_{t-d} + Y_{t-d}) + a(t-2d)^2 + b(t-2d)s_{t-2d} + Y_{t-2d} \\
&= at^2 + bts_t + Y_t - 2a(t^2 - 2td + d^2) - 2b(t-d)s_{t-d} - 2Y_{t-d} \\
&+ a(t^2 - 4td + 4d^2) + b(t-2d)s_{t-2d} + Y_{t-2d} \\
&= a(t^2 - 2t^2 + 4td - 2d^2 + t^2 - 4td + 4d^2) + b(ts_t - 2(t-d)s_{t-d} + (t-2d)s_{t-2d}) + Y_t - 2Y_{t-d} + Y_{t-2d} \\
&= a2d^2 + b(ts_t - ts_{t-d} + ts_{t-2d} - ts_{t-d} + 2d(s_{t-d} - s_{t-2d})) + Y_t - 2Y_{t-d} + Y_{t-2d} \\
&= 2ad^2 + b(t(s_t - s_{t-d}) + t(s_{t-2d} - s_{t-d}) + 2d(s_{t-d} - s_{t-2d})) + Y_t - 2Y_{t-d} + Y_{t-2d} \qquad (14)
\end{aligned}$$

But since s_t is the seasonal component with period d it holds $s_t = s_{t-d}$, $s_{t-d} = s_{t-2d}$, hence equation (14) becomes

$$\begin{aligned}
\nabla_d^2 W_t &= 2ad^2 + Y_t - 2Y_{t-d} + Y_{t-2d} \\
&= 2ad^2 + Y_t - Y_{t-d} - Y_{t-d} + Y_{t-2d} \\
&= 2ad^2 + (Y_t - Y_{t-d}) - (Y_{t-d} - Y_{t-2d}) \\
&= 2ad^2 + \nabla_d Y_t - \nabla_d Y_{t-d} \\
&= 2ad^2 + \nabla_d (Y_t - Y_{t-d}) \\
&= 2ad^2 + \nabla_d^2 Y_t \qquad (15)
\end{aligned}$$

Problem 3

(i) Since for the white noise the autocovariance function is

$$\gamma_Z(h) = \begin{cases} \sigma^2 & ,\text{h} = 0 \\ 0 & ,\text{h} \neq 0 \end{cases}$$

It holds that $\sum_{h\in\mathbb{Z}} |\gamma_Z(h)| = \gamma_Z(0) = Var(Z_t) = \sigma^2 < \infty$. Hence, the spectral density of the white noise is absolutely convergent and is given by

$$\begin{aligned}
f_Z(\lambda) &= \frac{1}{2\pi} \sum_{h\in\mathbb{Z}} \gamma_Z(h) e^{-ih\lambda} \\
&= \frac{\gamma_Z(0)}{2\pi} \\
&= \frac{\sigma^2}{2\pi} \qquad (16)
\end{aligned}$$

for $\lambda \in (-\pi, \pi]$.

(ii) The autocovariance function of the $MA(1)$ time series $X_t = (\alpha Z_t + \beta Z_{t-1})$ is given by

$$\gamma X(h) = \begin{cases} \sigma^2(\alpha^2+\beta^2) & ,h=0 \\ \sigma^2\alpha\beta & ,h=\pm 1 \\ 0 & ,\text{otherwise} \end{cases} \tag{17}$$

So, $\sum_{h\in\mathbb{Z}} |\gamma_Z(h)| = \sigma^2(\alpha^2+\beta^2) + 2\sigma^2\alpha\beta < \infty$ and the spectral density $f_X(\lambda)$ is absolutely convergent uniformly in $\lambda \in \mathbb{R}$ and is given by:

$$\begin{aligned} f_x(\lambda) &= \frac{1}{2\pi}\sum_{h\in\mathbb{Z}} \gamma_X(h)e^{-ih\lambda} \\ &= \frac{1}{2\pi}(\gamma_X(0)e^0 + \gamma_X(1)e^{-i\lambda} + \gamma_X(-1)e^{i\lambda}) \\ &= \frac{1}{2\pi}(\sigma^2(\alpha^2+\beta^2) + \sigma^2\alpha\beta e^{-i\lambda} + \sigma^2\alpha\beta e^{i\lambda}) \\ &= \frac{1}{2\pi}(\sigma^2(\alpha^2+\beta^2) + \sigma^2\alpha\beta(e^{-i\lambda} + e^{i\lambda})) \end{aligned} \tag{18}$$

Using the *Euler* formula $e^{ia} = cos(a) + isin(a)$, $a \in \mathbb{R}$ we have

$$\begin{aligned} e^{-i\lambda} + e^{i\lambda} &= cos(\lambda) - isin(\lambda) + cos(\lambda) + isin(\lambda) \\ &= 2cos(\lambda) \end{aligned}$$

so, equation (18) becomes:

$$\begin{aligned} f_X(\lambda) &= \frac{1}{2\pi}(\sigma^2(\alpha^2+\beta^2) + 2\sigma^2\alpha\beta cos(\lambda)) \\ &= \frac{\sigma^2}{2\pi}(\alpha^2+\beta^2+2\alpha\beta cos(\lambda)) \end{aligned} \tag{19}$$

(iii) In order for $\{X_t\}$ to be a linear filter of $\{Z_t\}$ it needs to be written in the form:

$$X_t = \sum_{k\in\mathbb{Z}} \psi_k Z_{t-k} \tag{20}$$

Indeed, X_t can be written in this form

$$\begin{aligned} X_t &= \alpha Z_t + \beta Z_{t-1} \\ &= \sum_{k\in\mathbb{Z}} \psi_k Z_{t-k} \end{aligned} \tag{21}$$

where

$$\psi_k = \begin{cases} \alpha & ,k=0 \\ \beta & ,k=1 \\ 0 & ,\text{otherwise} \end{cases} \tag{22}$$

The transfer function is

$$\begin{aligned}\psi(\lambda) &= \sum_{k\in\mathbb{Z}} \psi_k e^{-i\lambda} \\ &= \psi_0 e^0 + \psi_1 e^{-i\lambda} \\ &= \psi_0 + \psi_1 e^{-i\lambda} \\ &= \alpha + \beta e^{-i\lambda}\end{aligned} \tag{23}$$

Next, it needs to be shown that

$$f_X(\lambda) = |\psi(\lambda)|^2 f_Z(\lambda) \tag{24}$$

We have

$$|\psi(\lambda)|^2 f_Z(\lambda) = |\alpha + \beta e^{-i\lambda}|^2 \frac{\sigma^2}{2\pi} \tag{25}$$

using again the Euler formula for $e^{-i\lambda}$ it becomes

$$\begin{aligned}|\psi(\lambda)|^2 f_Z(\lambda) &= |\alpha + \beta(cos\lambda - isin\lambda)|^2 \frac{\sigma^2}{2\pi} \\ &= |\alpha + \beta cos\lambda - i\beta sin\lambda|^2 \frac{\sigma^2}{2\pi}\end{aligned} \tag{26}$$

Now, using the fact that for a complex number $z = a + ib$ it holds $|z|^2 = a^2 + b^2$ we have

$$\begin{aligned}|\psi(\lambda)|^2 f_Z(\lambda) &= ((\alpha + \beta cos\lambda)^2 + (\beta sin\lambda)^2)\frac{\sigma^2}{2\pi} \\ &= (\alpha^2 + 2\alpha\beta cos\lambda + \beta^2 cos^2\lambda + \beta^2 sin^2\lambda)\frac{\sigma^2}{2\pi} \\ &= (\alpha^2 + 2\alpha\beta cos\lambda + \beta^2(cos^2\lambda + sin^2\lambda))\frac{\sigma^2}{2\pi} \\ &= (\alpha^2 + \beta^2 + 2\alpha\beta cos\lambda)\frac{\sigma^2}{2\pi} \\ &\overset{(19)}{=} f_X(\lambda)\end{aligned} \tag{27}$$

Hence, the formula $f_X(\lambda) = |\psi(\lambda)|^2 f_Z(\lambda)$ holds. $\square$

Appendix

```
#### EXERCISE 1 ######

library(forecast)


data = sunspot.month

plot(data, main = 'Observed sunspots from 1749 to 1997',
 xlab = 'Time', ylab = 'Montly count')
acf(data,main = 'ACF as a function of lags')
x.acf = acf(data,main = 'ACF as a function of lags',lag.max = 300)
acf(data,main = 'ACF as a function of lags',lag.max = 300)

plot(x.acf,xaxt = 'n')
axis(1, seq(0, length(x.acf$acf), 11))
abline(1,0)

#plot a smaller part to identify seasonality
plot(window(data,start = 1900,end = 1944),
     main = 'Observed sunspots from 1900 to 1944',
     ylab = 'Montly count')

decomp = decompose(data)

decomp2 = decompose(window(data,start = 1900,end = 1944))


#remove trend using decompose

decomp_all = decompose(data)
trend1 = decomp_all$trend
plot(trend1, main = 'Trend', ylab='Monthly Number of Sunspots')

stat = decomp_all$random
plot(stat, main = 'Residual Time series',ylab ='Detrended and Deseasonalized Sunspots')



#check residuals
Box.test(stat,type="Box-Pierce")

y = seq(from = 0, to = 1,by = 0.05)
box = matrix(300)
ljung = matrix(300)
for (i in c(1:300)){
  box[i] = Box.test(stat,type="Box-Pierce", lag =i)$p.value # the Portmanteau test
```

```
  ljung[i] =Box.test(stat,type="Ljung-Box", lag =i)$p.value # the Portmanteau test
}

plot(ljung, col = 'red', main = 'Portmanteau test',
type='l',ylab = 'p-value',xlab = 'lag',ylim = c(0,1),yaxt='n')
lines(box, col = 'blue', main = 'Box-Pierce test',type='l')
axis(side = 2,at = y)
legend('bottomright',
       legend = c('Ljung-Box','Box-Pierce'),lwd =4, lty =c(1,1),cex = 0.7,bty = 'n',
       col = c('red','blue'))
abline(0.05,0,lty=3)




#spline smoothing
par(mfrow=c(3,1))
plot.ts(data,ylab="Sunspots",main="spar=0.9")
SM9=smooth.spline(data,spar=0.9)
lines(SM9,col="red",lwd=2)
plot.ts(data,ylab="Sunspots",main="spar=0.5")
SM5=smooth.spline(data,spar=0.5)
lines(SM5,col="red",lwd=2)
plot.ts(data,ylab="Sunspots",main="spar=0.1")
SM1=smooth.spline(data,spar=0.1)
lines(SM1,col="red",lwd=2)



#spline smoothing
par(mfrow=c(3,1))
plot.ts(data,ylab="Sunspots",main="spar=0.4")
SM4=smooth.spline(data,spar=0.4)
lines(SM4,col="red",lwd=2)
plot.ts(data,ylab="Sunspots",main="spar=0.35")
SM35=smooth.spline(data,spar=0.35)
lines(SM35,col="red",lwd=2)
plot.ts(data,ylab="Sunspots",main="spar=0.3")
SM3=smooth.spline(data,spar=0.3)
lines(SM3,col="red",lwd=2)


detrended = data - SM35$y
acf(detrended, main = 'ACF', ylab = 'Autocorrelation')
pacf(detrended, main = 'PACF',ylab= 'Partial Autocorrelation')
plot(detrended, main = 'Residual Time series',
 xlab = 'Time', ylab = 'Detrended and Deseasonalized Sunspots')
Box.test(detrended,type="Box-Pierce")
```

```
Box.test(detrended,type="Ljung-Box")



### EXERCISE 2 ######


library(forecast)
library(xtable)
library(FitAR)

n = 500
p=2
q=1

var = 2


ts=list(ar=c(-0.2,0.7),ma=0.8)
arma=arima.sim(n,model=ts,sd=sqrt(var))
plot(arma,main="ARMA(2,1) Time Series",ylab = 'Xt')


fit=arima(arma,order=c(p,0,q),include.mean=F)

#95% confidence interval
confint(fit)


summary(fit)
xtable(summary(fit))


residuals = fit$residuals
plot(residuals, main ='Residuals of ARMA(2,1)', ylab = 'Residuals')
Box.test(residuals,type="Box-Pierce")$p.value # the Portmanteau test
Box.test(residuals,type="Ljung-Box")$p.value # the Portmanteau test

y = seq(from = 0, to = 1,by = 0.05)
box = matrix(300)
ljung = matrix(300)
for (i in c(1:300)){
  box[i] = Box.test(residuals,type="Box-Pierce", lag =i)$p.value # the Portmanteau test
  ljung[i] =Box.test(residuals,type="Ljung-Box", lag =i)$p.value # the Portmanteau test
}

plot(ljung, col = 'red', main = 'Portmanteau test',type='l'
,ylab = 'p-value',xlab = 'lag',ylim = c(0,1),yaxt='n')
```

```
lines(box, col = 'blue', main = 'Box-Pierce test',type='l')
axis(side = 2,at = y)
legend('bottomright',
       legend = c('Ljung-Box','Box-Pierce'),lwd =4, lty =c(1,1),cex = 0.7,bty = 'n',
       col = c('red','blue'))
abline(0.05,0,lty=3)

acf(residuals, main = 'ACF of ARMA(2,1) Residuals')
pacf(residuals, main = 'PACF of ARMA(2,1) Residuals')


###AR MODEL fitted in the same ARMA time series


fitar=ar(arma, order.max=p, method = c('yule-walker'))
resAR = fitar$resid
b1=Box.test(resAR,type="Box-Pierce")$p.value # the Portmanteau test

fit.ar = arima(arma, order = c(p,0,0), include.mean = FALSE, init = fitar$ar)
summary(fitar)
fitar


##Create AR time series and estimate again

ts2=list(ar=c(-0.2,0.7))
ar=arima.sim(n,model=ts2,sd=sqrt(var))
plot(ar,main="AR(2) Time Series",ylab = 'Xt')



fitar2=ar(ar, order.max=p, method = c('yule-walker'))
fit.ar = arima(ar, order = c(p,0,0),include.mean = FALSE)
confint(fit.ar)


residuals2 = fitar2$resid

plot(residuals2, main ='Residuals of AR(2) model', ylab = 'Residuals')
Box.test(residuals2,type="Box-Pierce")$p.value # the Portmanteau test
Box.test(residuals2,type="Ljung-Box")$p.value # the Portmanteau test



y = seq(from = 0, to = 1,by = 0.05)
box = matrix(300)
ljung = matrix(300)
for (i in c(1:300)){
```

```
  box[i] = Box.test(residuals2,type="Box-Pierce", lag =i)$p.value # the Portmanteau test
  ljung[i] =Box.test(residuals2,type="Ljung-Box", lag =i)$p.value # the Portmanteau test
}

plot(ljung, col = 'red', main = 'Portmanteau test',type='l',
ylab = 'p-value',xlab = 'lag',ylim = c(0,1),yaxt='n')
lines(box, col = 'blue', main = 'Box-Pierce test',type='l')
axis(side = 2,at = y)
legend('bottomright',
       legend = c('Ljung-Box','Box-Pierce'),lwd =4, lty =c(1,1),cex = 0.7,bty = 'n',
       col = c('red','blue'))
abline(0.05,0,lty=3)


acf(residuals2, main = 'ACF of AR(2) Residuals',na.action = na.omit)
pacf(residuals2, main = 'PACF of AR(2) Residuals',na.action = na.omit)









##### EXERCISE 3 #######


library(forecast)
library(xtable)

data = EuStockMarkets

y=seq(from=0,to=8000,by=1000)


plot.ts(data)

plot.ts(data[,1], main = 'Daily closing prices of stock indices',
ylim=c(0,9000), ylab ='Price', yaxt ='n')
axis(side =2,at=y)
lines(data[,2],col = 'red')
lines(data[,3],col = 'green')
lines(data[,4],col = 'blue')
legend('topleft',
       legend = c('DAX', 'SMI','CAC','FTSE'),lwd =4, lty =c(1,1),cex = 0.7,bty = 'n',
       col = c('black','red','green','blue'))
acf(data)

ccf(data[,1],data[,2], main = 'DAX and SMI',type = 'correlation',lag.max = 800)
```

```
ccf(data[,1],data[,3], main = 'DAX and CAC')
ccf(data[,1],data[,4], main = 'DAX and FTSE')
ccf(data[,2],data[,3], main = 'SMI and CAC')
ccf(data[,2],data[,4], main = 'SMI and FTSE')
ccf(data[,3],data[,4], main = 'CAC and FTSE')

c1 = ccf(data[,1],data[,2],main = 'DAX and SMI',lag.max = 800)
max1 = max(c1$acf)
lag1 = c1$lag[max(c1$acf)]

c2 = ccf(data[,1],data[,3],main = 'DAX and CAC',lag.max = 800)
max2 = max(c2$acf)
lag2 = c2$lag[which.max(c2$acf)]

c3 = ccf(data[,1],data[,4],main = 'DAX and FTSE',lag.max = 800)
max3 = max(c3$acf)
lag3 = c3$lag[which.max(c3$acf)]

c4 = ccf(data[,2],data[,3], main = 'SMI and CAC',lag.max = 800)
max4 = max(c4$acf)
lag4 = c4$lag[which.max(c4$acf)]

c5 = ccf(data[,2],data[,4], main = 'SMI and FTSE',lag.max = 800)
max5 = max(c5$acf)
lag5 = c5$lag[which.max(c5$acf)]

c6 = ccf(data[,3],data[,4],main = 'CAC and FTSE',lag.max = 800)
max6 = max(c6$acf)
lag6 = c6$lag[which.max(c6$acf)]

max1
max2
max3
max4
max5
max6

plot(c1, main = 'DAX and SMI', type ='l')
plot(c2, main = 'DAX and CAC', type ='l')
plot(c3, main = 'DAX and FTSE', type ='l')
plot(c4, main = 'SMI and CAC', type ='l')
plot(c5, main = 'SMI and FTSE', type ='l')
plot(c6, main = 'CAC and FTSE', type ='l')
```