Time Series Analysis: All Lectures - Part 2

Dr. Esam Mahdi

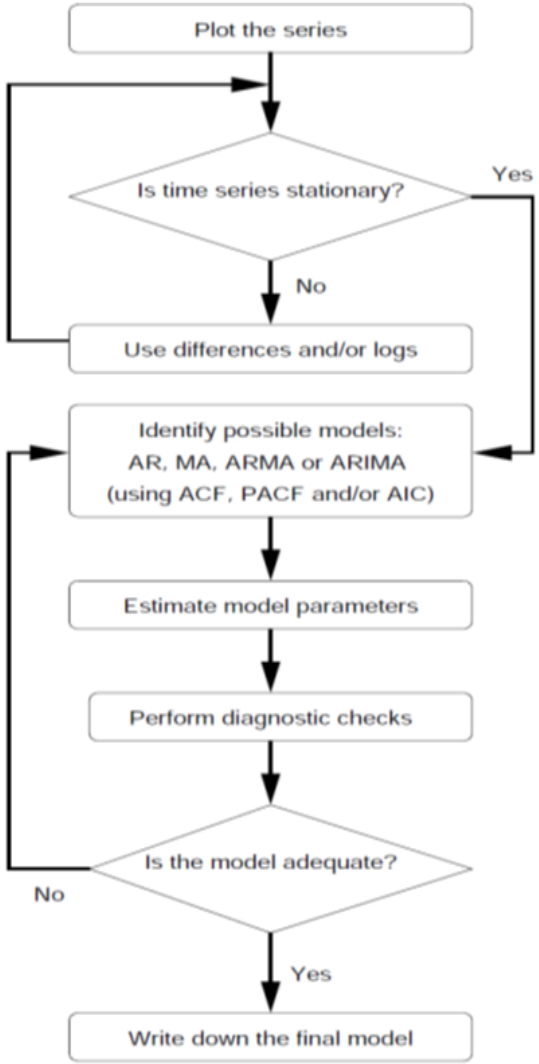
Islamic University of Gaza - Department of Mathematics

April 9, 2017

Modeling approach - Box-Jenkins methodology

In general, there are three major steps in time series analysis:

1. **Identification:** One goal of the analysis is to identify all patterns, in the sequence of numbers over time, accounted for in the model.
2. **Estimation:** In this step, we fit a suitable model, so that we can use it for predicting value of observations in the near future.
3. **Diagnostic checking:** In this step, the goodness of fit tests and residual scores are examined to check the adequacy of the fitted model and to determine if there are still patterns in the data that are not accounted for.



A flowchart for fitting ARIMA (Box–Jenkins) type models.

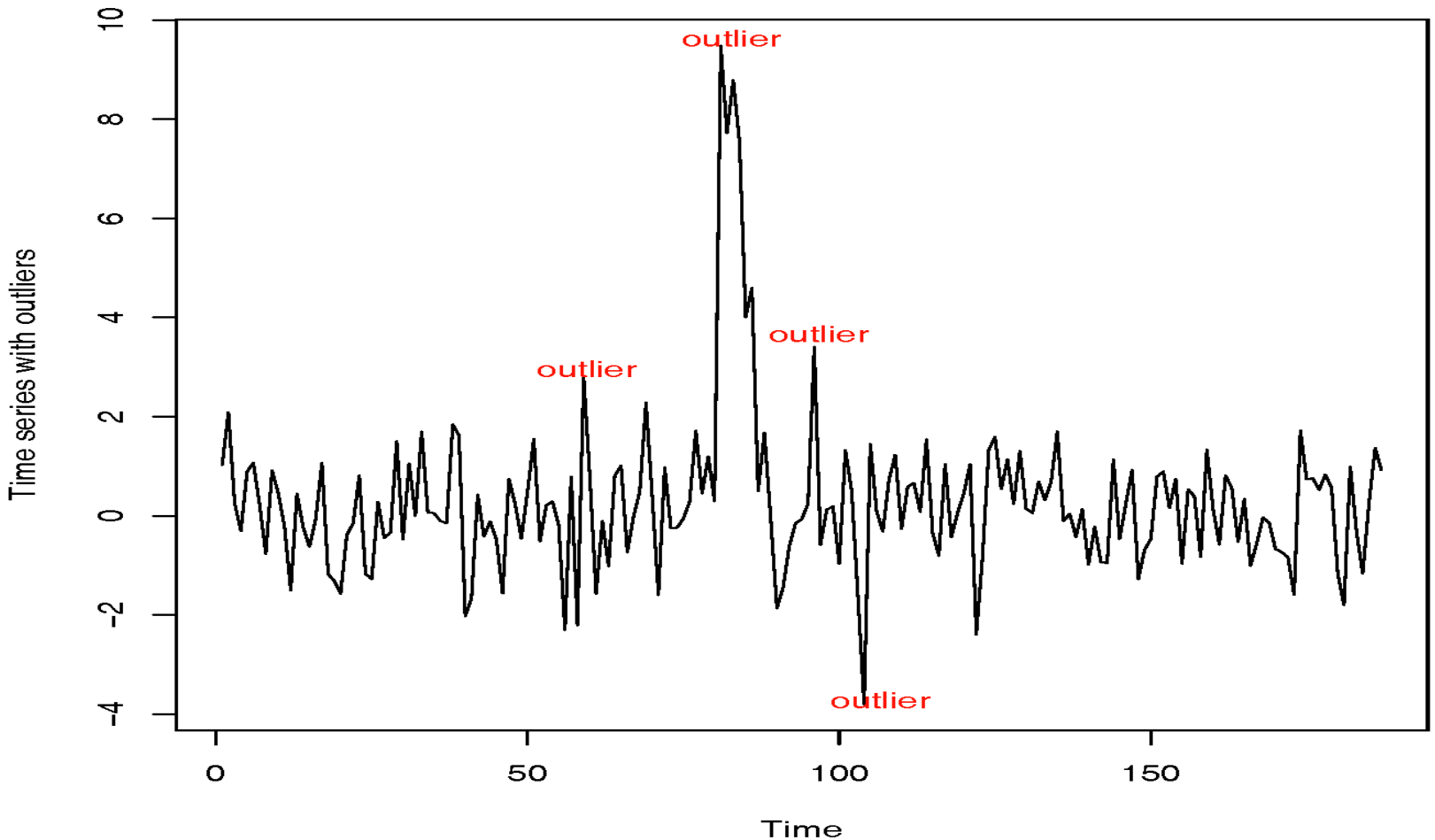
Box-Jenkins model identification

1. Detecting stationarity and seasonality (or periodicity):
   - Examine the time plot of the series.
   - Examine the autocorrelation and partial autocorrelation plots (ACF and PACF) to detect seasonality, random walks, etc.
   - Test for non-stationarity: Dickey-Fuller test statistics.
2. Differencing to achieve the stationarity:
3. Seasonal differencing: Although, we can include the order of the seasonal terms in the model specification to the ARIMA estimation software, seasonal differencing may help in the model identification of the non-seasonal component of the model.
4. Identify p and q: Once stationarity and seasonality have been addressed, the next step is to identify the order of ARMA , where the plots of the sample ACF and the sample PACF are useful tools for identifying the orders p and q.

Examine the time plot of the series

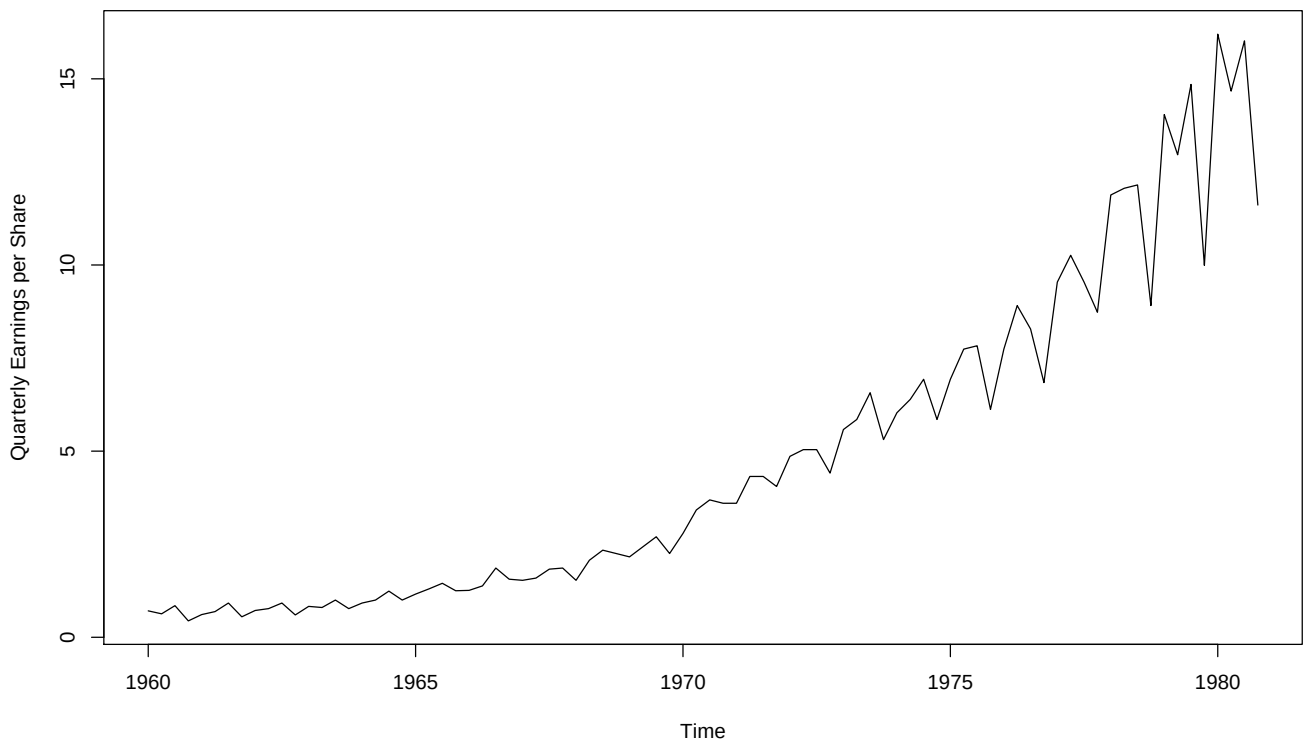
- One of the most important steps in a preliminary time series analysis is to plot the data; i.e., create a time plot. This is achieved with the generic `ts.plot()` R function.
- A time series plot not only emphasizes patterns and features of the data but can also expose **outliers, missing, and erroneous values**.
- **Outliers** are observations that are highly inconsistent with the remainder of the time-series data. They can greatly affect the results of the analysis.
- The plot of a non-stationary series may shows a trend, seasonality, or changing variance. In this case, transform the series to be a stationary is needed. You may detrend (remove the trend) the series by implementing logs, differencing, etc.

Outliers detection: Sample chart



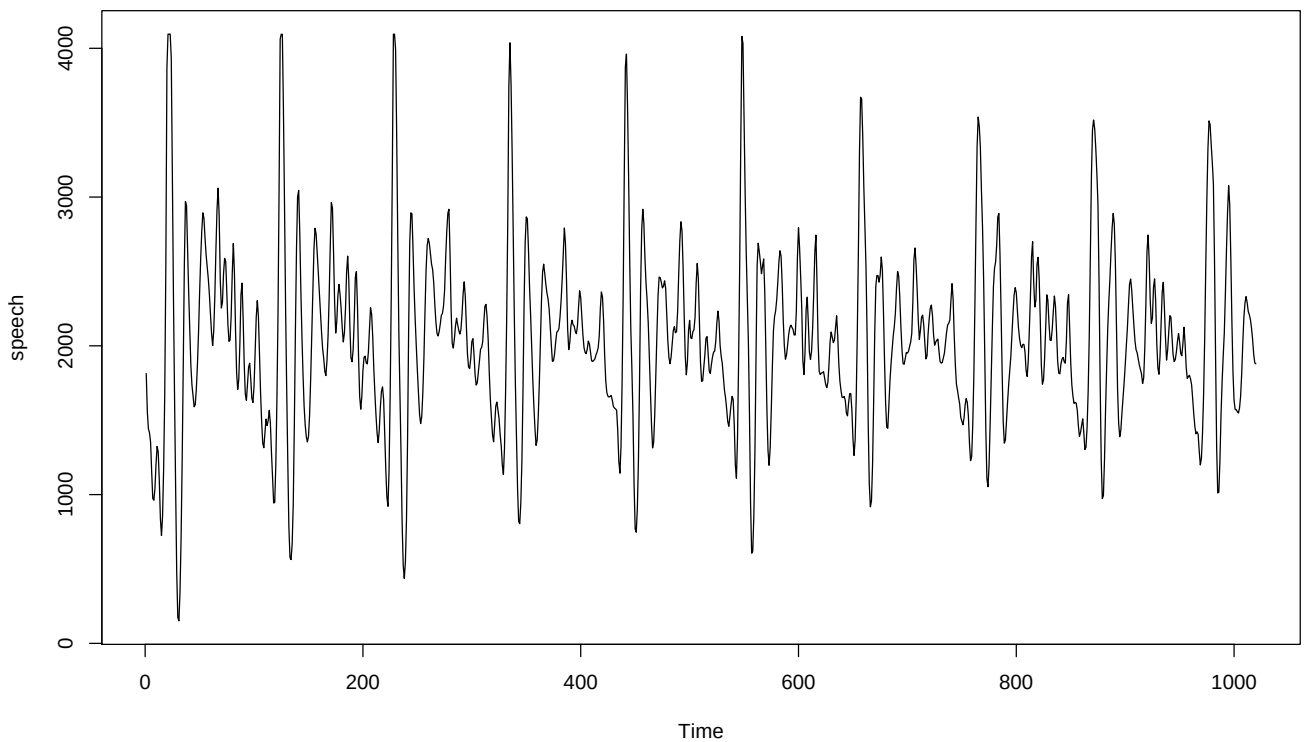
Plot of the series - model identification stage: example 1

Quarterly earnings per share for 1960Q1 to 1980Q4 of the U.S. company, Johnson & Johnson, Inc. (Upward trend).



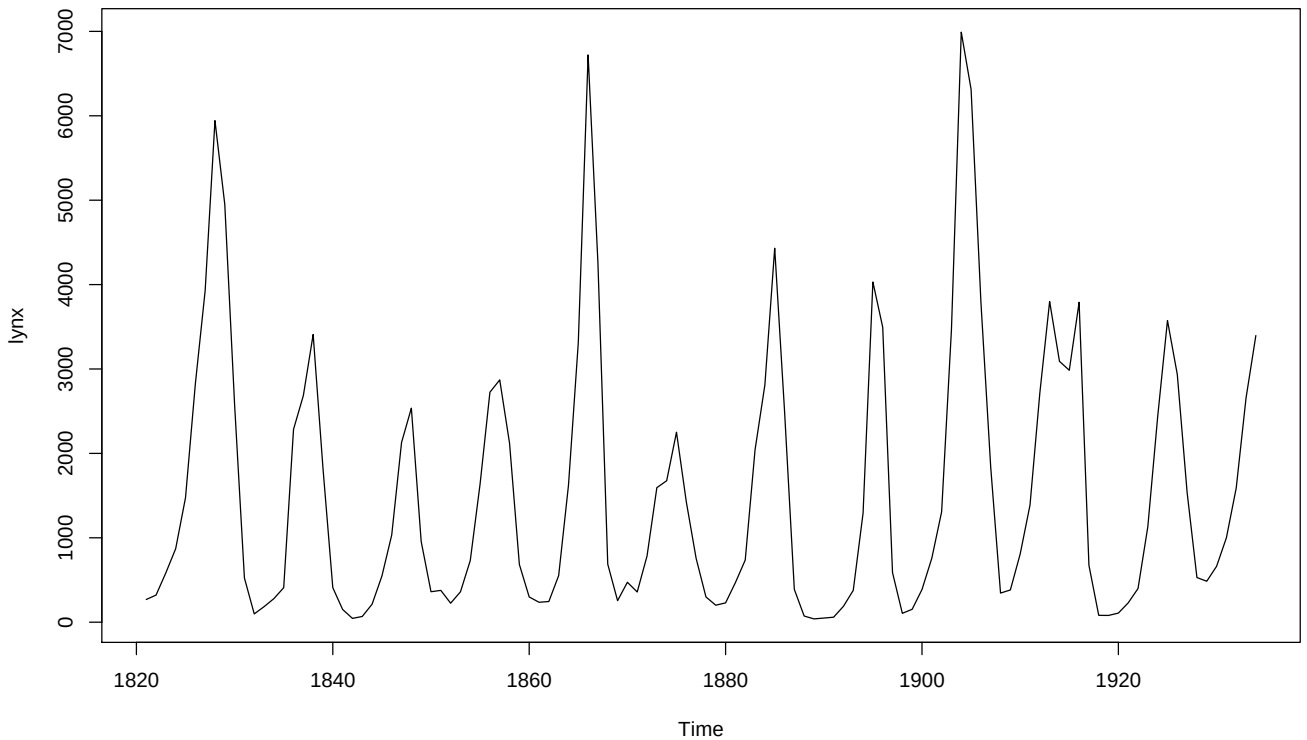
Plot of the series - model identification stage: example 2

A small .1 second (1000 points) sample of recorded speech for the phrase "aaa...hhh". (Regular repetition of small wavelets).



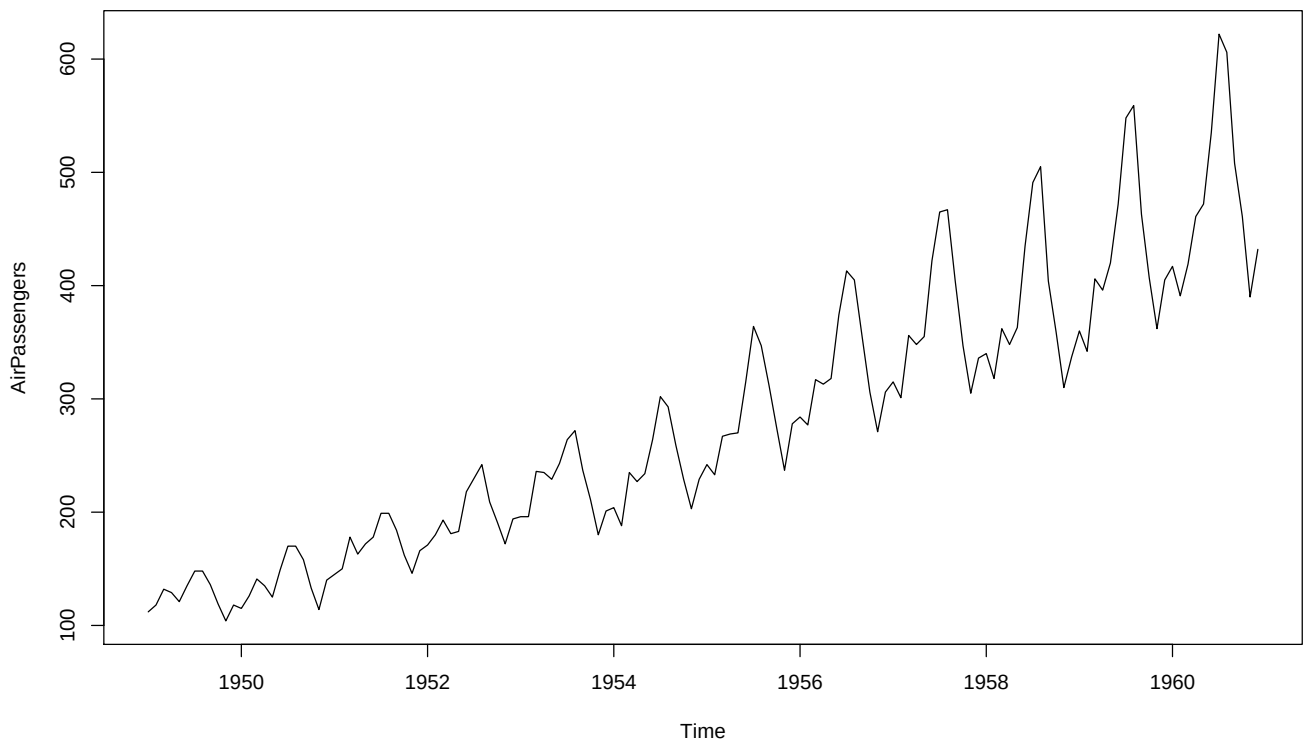
Plot of the series - model identification stage: example 3

Annual numbers of lynx trappings in McKenzie river in Northwest Territories of Canada over the years 1821-1934. (Aperiodic cycles of approximately 10 years).



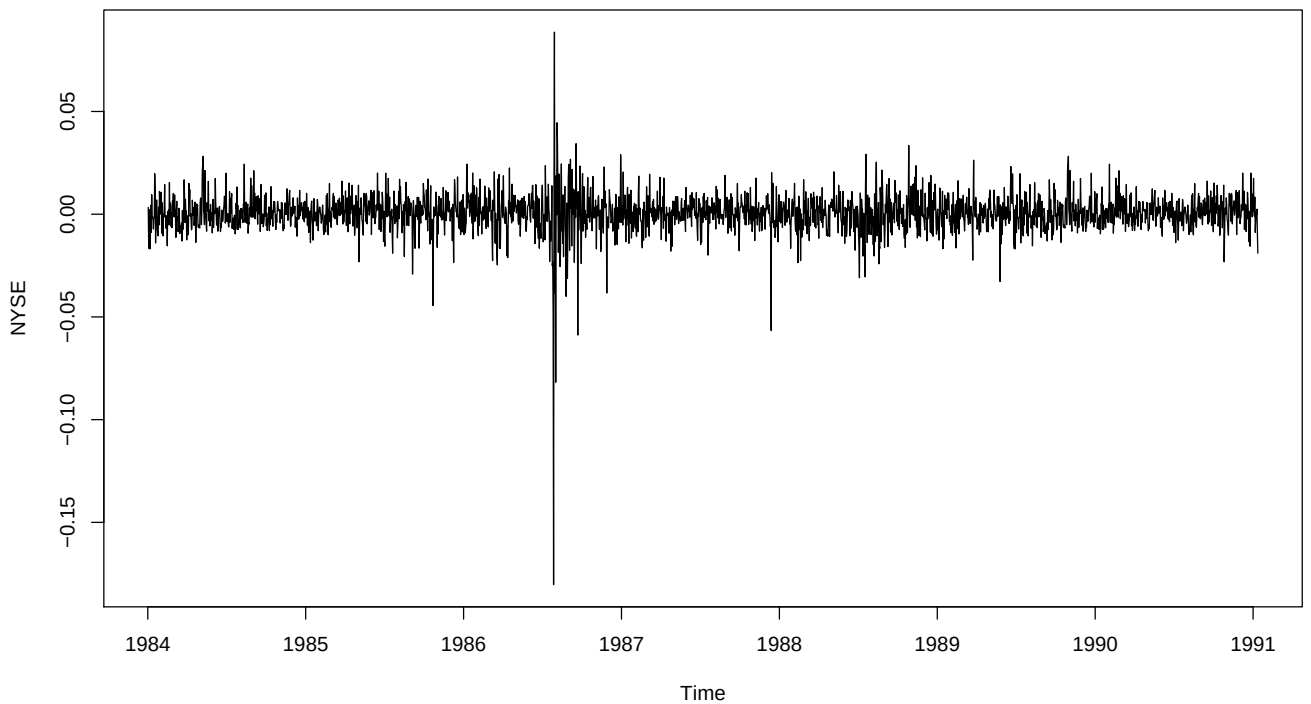
Plot of the series - model identification stage: example 4

Monthly Airline Passenger Numbers 1949-1960. (Seasonality appears to increase with the general trend).



Plot of the series - model identification stage: example 5

Returns of the New York Stock Exchange (NYSE) from February 2, 1984 to December 31, 1991. (Average return of approximately zero, however, volatility (or variability) of data changes over time).



Examine ACF & PACF for ARIMA model identification

- The ACF and PACF functions are used to identify the appropriate ARMA (p, q) model, where the orders p and q can be identified by using the sample autocorrelation and sample partial autocorrelation function.
- The following table summarizes the behaviour of the theoretical ACF and PACF of the ARMA models.

	ACF	PACF
White noise	All zeros	All zeros
AR (p)	Tails off as exponential decay (or damped sin wave)	Cuts off after lag p
MA (q)	Cuts off after lag q	Tails off as exponential decay (or damped sin wave)
ARMA (p, q)	Tail off after lag $(q - p)$ (sometimes in an oscillating manner)	Tail off after lag $(p - q)$ (sometimes in an oscillating manner)
Random walk	No decay to zero	All zero after lag 1

ACF and PACF for a simulated AR(1)

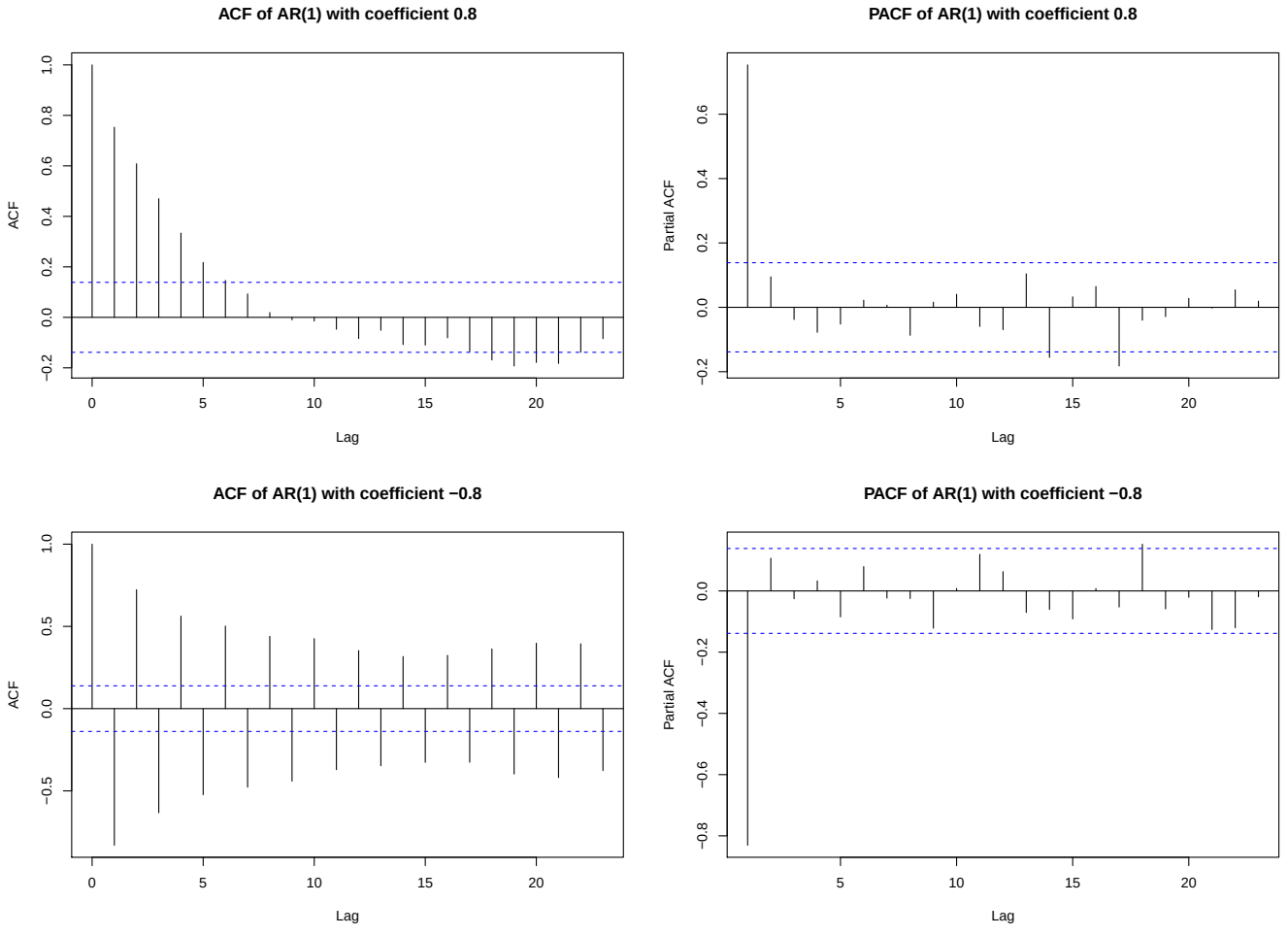


Figure : A simulated AR(1) process with coefficients ± 0.8

ACF and PACF for simulated MA (1)

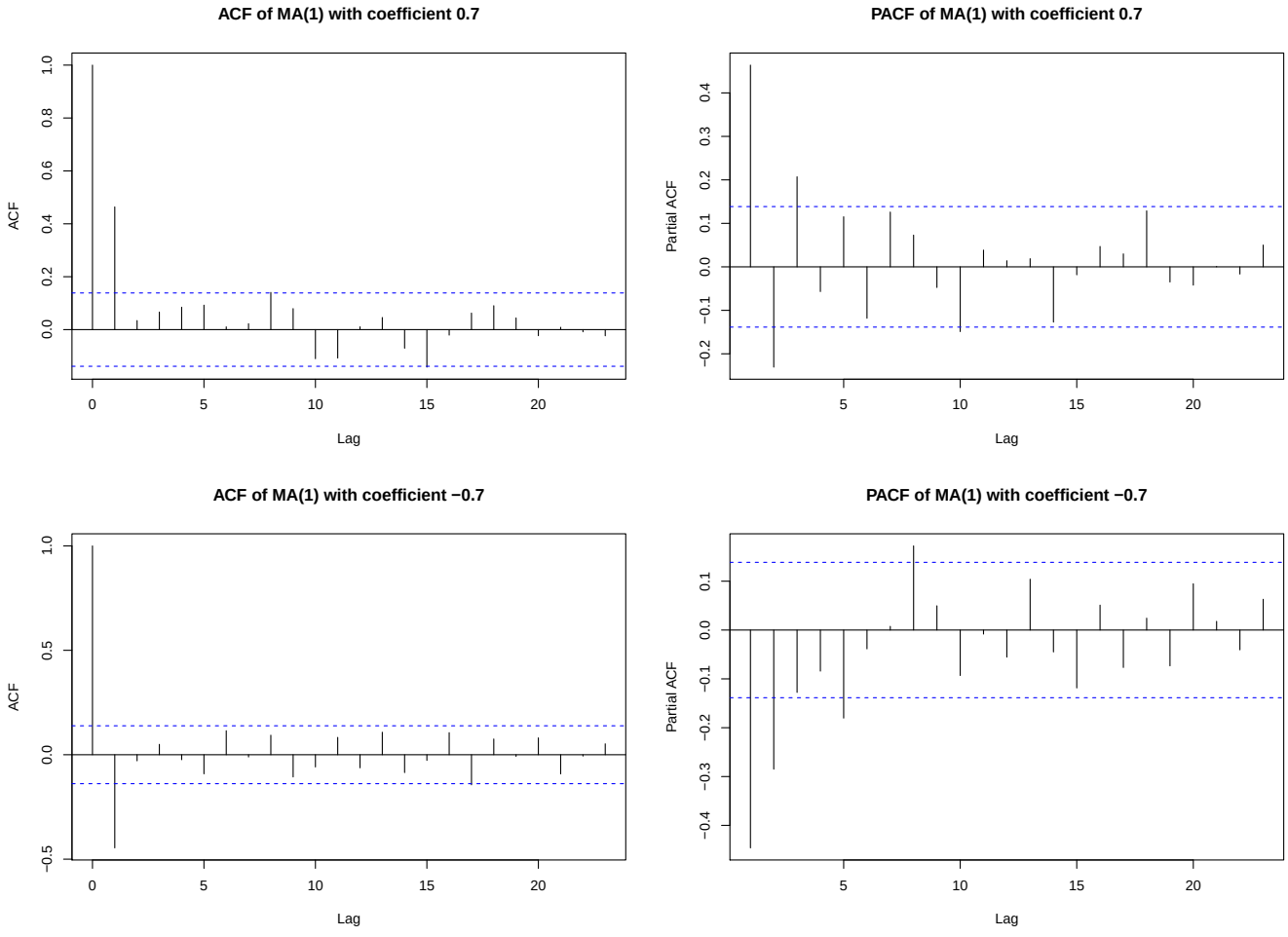


Figure : A simulated MA (1) process with coefficients ± 0.7

ACF and PACF for simulated ARMA (1, 1)

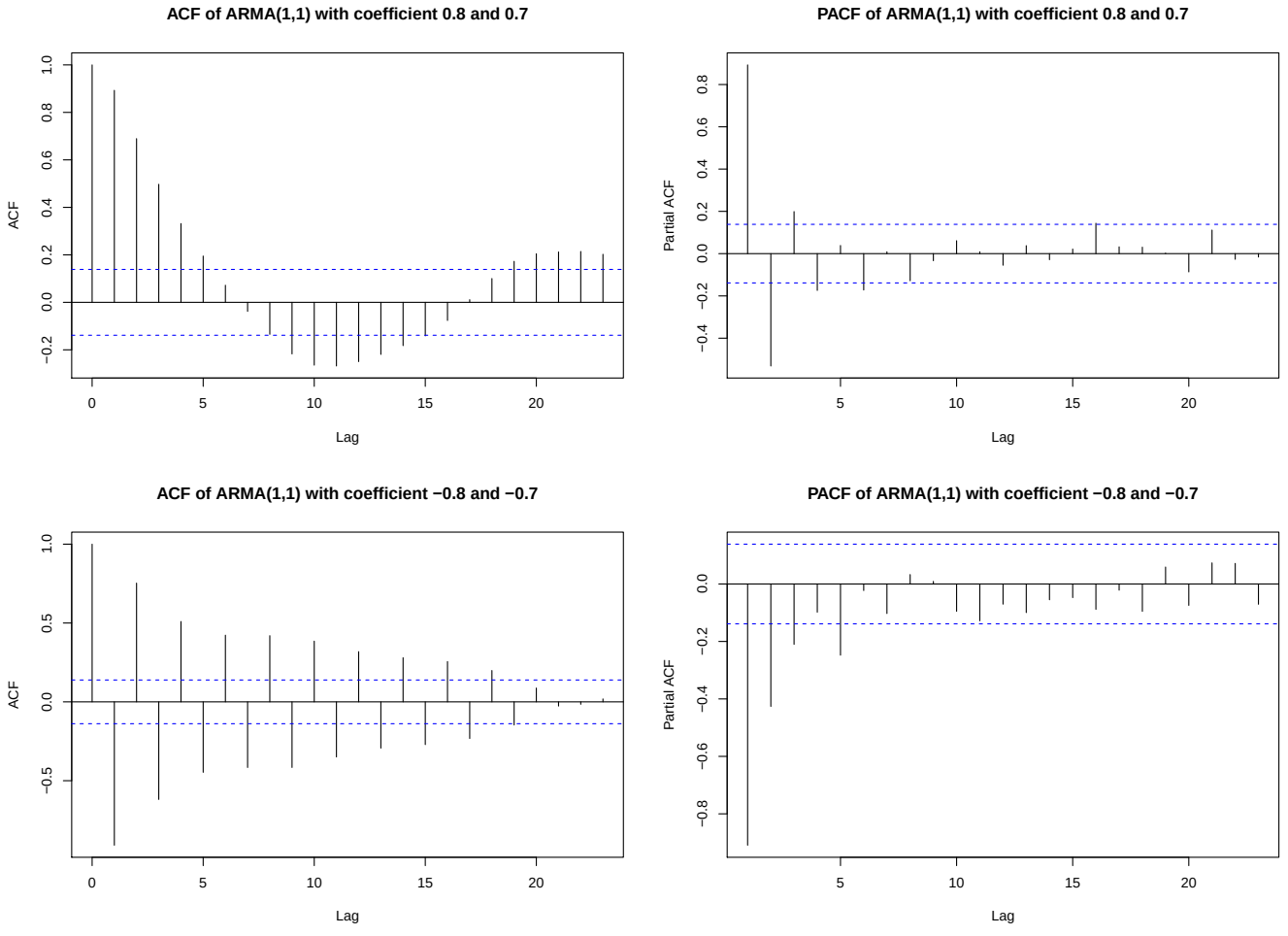


Figure : A simulated ARMA (1,1) process with coefficients $\pm 0.8, \pm 0.7$

A simulated ACF and PACF of white noise series

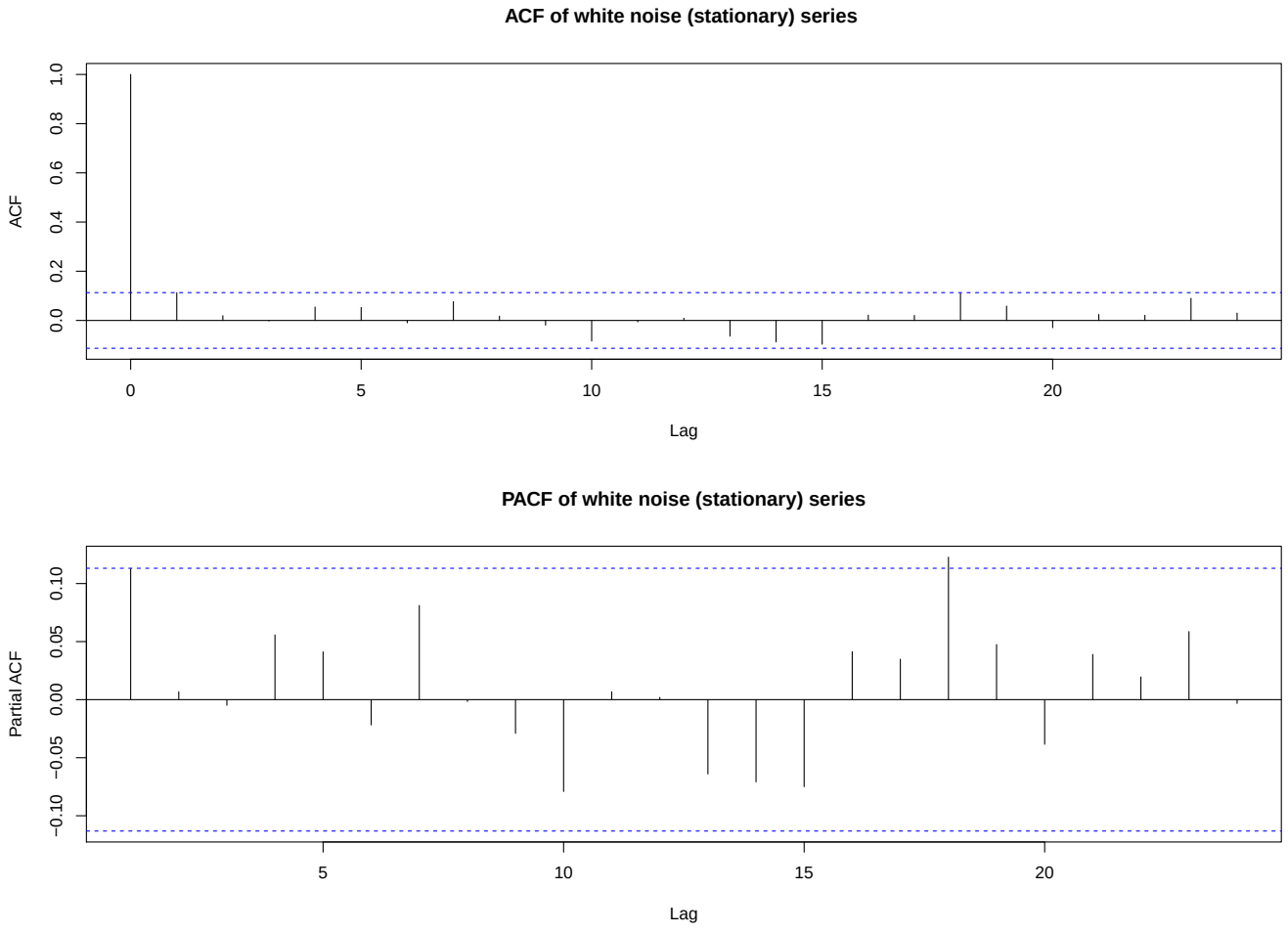


Figure : A simulated ACF and PACF for white noise series.

A simulated ACF and PACF of a random walk

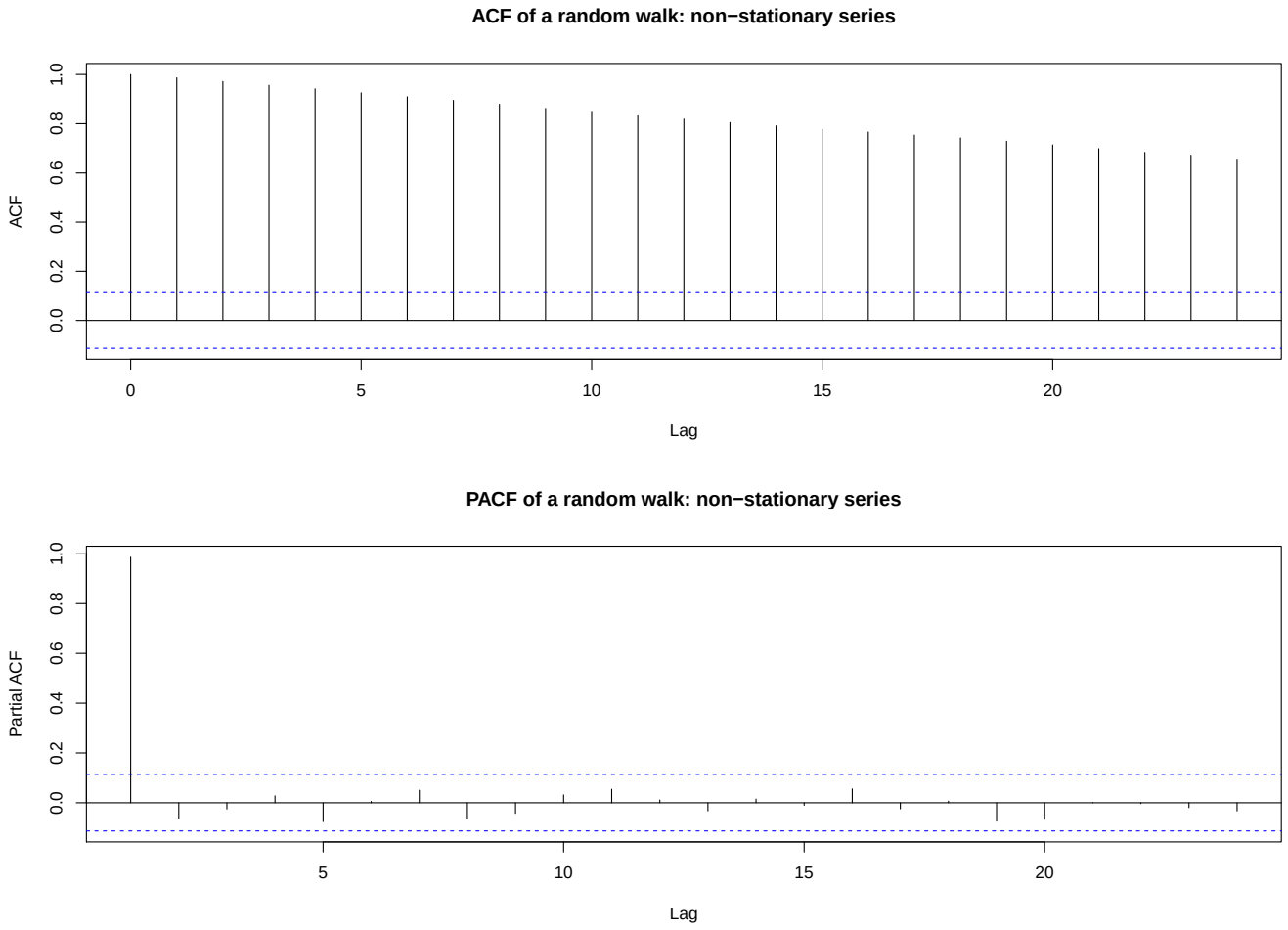


Figure : A simulated ACF and PACF of a random walk.

Examine ACF & PACF for SARIMA model identification

One can use the sample ACF for model identification
You may use the following notes to guide you in identifying the orders of the ARIMA $(p, d, q) \times (P, D, Q)_s$ models.

- The pure SAR or pure SMA behavior is similar to the pure AR or pure MA behavior, except that the pattern appears across multiples of lag s in the ACF and PACF .
- An SAR model usually occurs when the ACF at the seasonal period is positive, whereas an SMA model usually occurs when the seasonal ACF is negative. Try to avoid mixing SAR and SMA terms in the same model.
- For a strong and consistent seasonal pattern, you must use a seasonal differencing, where $d + D$ should not exceed 2.

A simulated ACF and PACF of SAR model

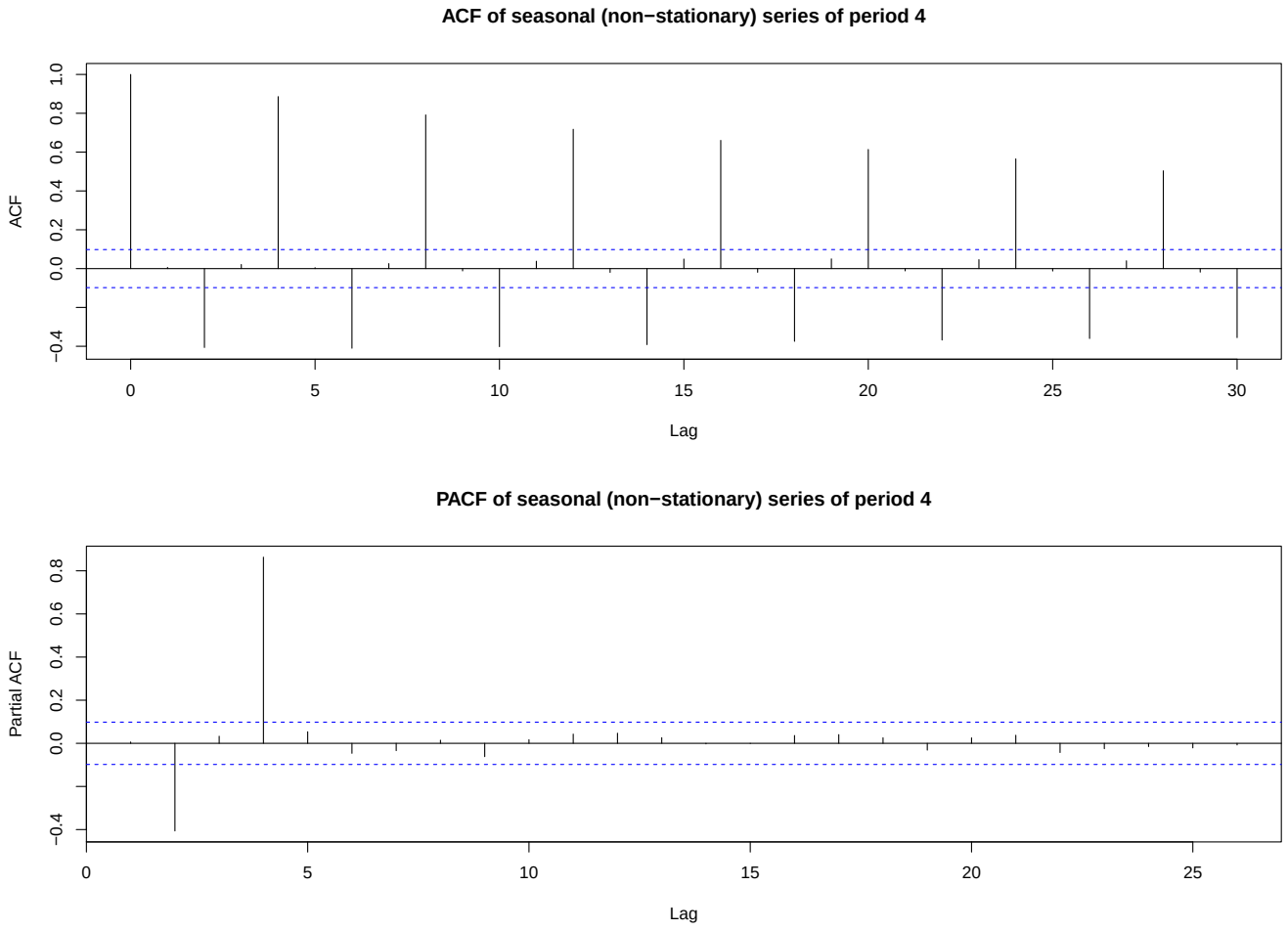


Figure : A simulated ACF and PACF of a SAR (1) model.

Maximum likelihood and least squares estimation for ARMA process

Suppose that $\{X_t\}$ is the casual ARMA process

$$X_t = \phi_1 X_{t-1} + \ldots + \phi_p X_{t-p} + \theta_0 \epsilon_t + \ldots + \theta_q \epsilon_{t-q},$$

where $\epsilon_t \sim N(0, \sigma^2)$ and $\theta_0 = 1$.

- Computer programs are generally needed to obtain estimates of $\Phi = (\phi_1, \ldots, \phi_p)^T, \Theta = (\theta_1, \ldots, \theta_q)^T$, and σ^2.
- In R, the function arima() can be used for fitting an ARIMA models to a univariate time series.
- This function implements three fitting methods:
  - The Conditional Sum of Squares (CSS) Estimation,
  - The Maximum Likelihood Estimation (MLE),
  - The Conditional Sum of Squares Estimation to find initial values, then the Maximum Likelihood Estimation (CSS -MLE).

  The default fitting method is the CSS -MLE method.

Main techniques for model verification

After identification and estimation of the parameters in a fitted model, the next step is to check the adequacy of this fitted model. Two main techniques can be used in diagnostic checking.

- **Goodness of Fit**: In this technique, one can use the t tests or/and Akaiake Information Criterion (AIC) or/and Bayesian Information Criterion (BIC) likelihood ratio tests for testing the significantly of adding or removing some parameters from the fitted model.
- **Residual analysis**: In time series analysis, we assume that the series is stationary with Gaussian White Noise innovations. This means that a good fitted model must produce residuals that are approximately normally distributed & uncorrelated in time.

Goodness of fit - Likelihood ratio test

A Likelihood Ratio Test, denoted by Λ, is a statistical measure used to compare the fit of two models under a null hypothesis, H_0 (full model), and an alternative hypothesis, H_1 (restricted model),

$$\Lambda = -2\ln(\text{likelihood function of } H_0) + 2\ln(\text{likelihood function of } H_1)$$

The probability distribution of Λ is approximately a χ^2 distribution with degrees of freedom equal to the number of parameters under the null model, H_0, minus that number of parameters under the alternative model, H_1.

Information criterion (AIC)

- Similar to Λ, the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC) are goodness of fit tests for selecting the best model among a set of models.

$$\begin{aligned} AIC &= -2\ln(L) + 2k \\ BIC &= -2\ln(L) + k\ln(n) \end{aligned}$$

where L is the value of the likelihood function evaluated at the parameter estimates, n is the number of observations, and k is the number of estimated parameters $(p+q)$.

Information criterion and model selection

The following 3 steps summarize the general approach for model selection in time series analysis based on the the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC):

- **Step 1:** For a given data, fit a set of different ARMA models, say AR (1), ..., AR (p), MA (1), ..., MA (q), & ARMA (1,1), ARMA (2,1), ARMA (1,2), ..., ARMA (p, q) where $p, q \in \mathbb{Z}^+$.
- **Step 2:** For each model, compute the AIC or/and BIC .
- **Step 3:** Select the parsimonious model with the least AIC or/and BIC value.
- A lower AIC or/and BIC value indicates a better fit.

Example: information criterion and model selection

- The exchange rates for British pounds sterling to New Zealand dollars for the period January 1991 to March 2000 are available from the link `http://staff.elena.aut.ac.nz/Paul-Cowpertwait/ts/pounds_nz.dat`.
- The data are mean values taken over quarterly periods of three months, with the first quarter being January to March and the last quarter being October to December.
- In the R code (see the next three slides) the fitted MA(1), AR(1) and ARMA$(1,1)$ models are compared using the AIC. The ARMA$(1,1)$ model provides the best fit to the data, followed by the AR(1) model, with the MA(1) model providing the poorest fit.

```
R> pound<-"http://staff.elena.aut.ac.nz/Paul-Cowpertwait
 + /ts/pounds_nz.dat"
R> Z <- read.table(pound, header = T)
R> Z.ts <- ts(Z, st = 1991, fr = 4)
R> plot(Z.ts,ylab="Quarterly exchange rate in $NZ/pound")
```

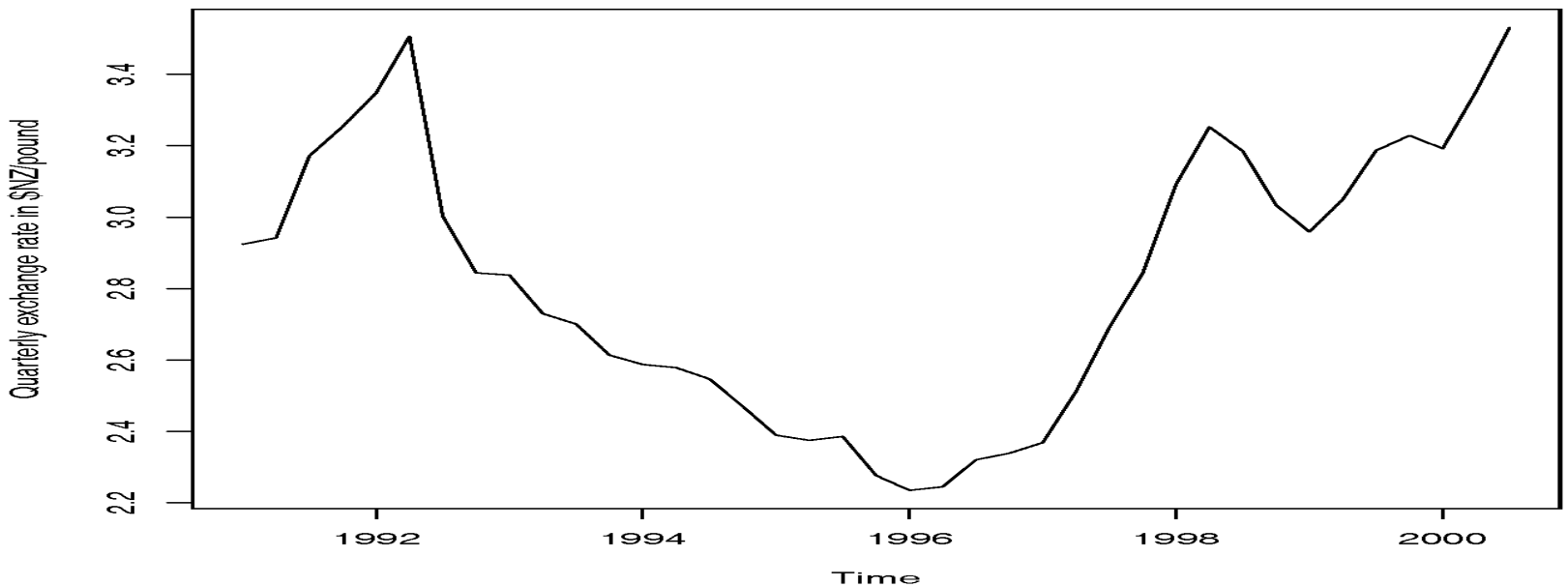


Figure : Exchange rate for British pounds sterling to New Zealand dollars.

```
R> par(mfrow=c(1,2))
R> acf(Z, main="Autocorrelation Function")
R> pacf(Z, main="Partial Autocorrelation Function")
```

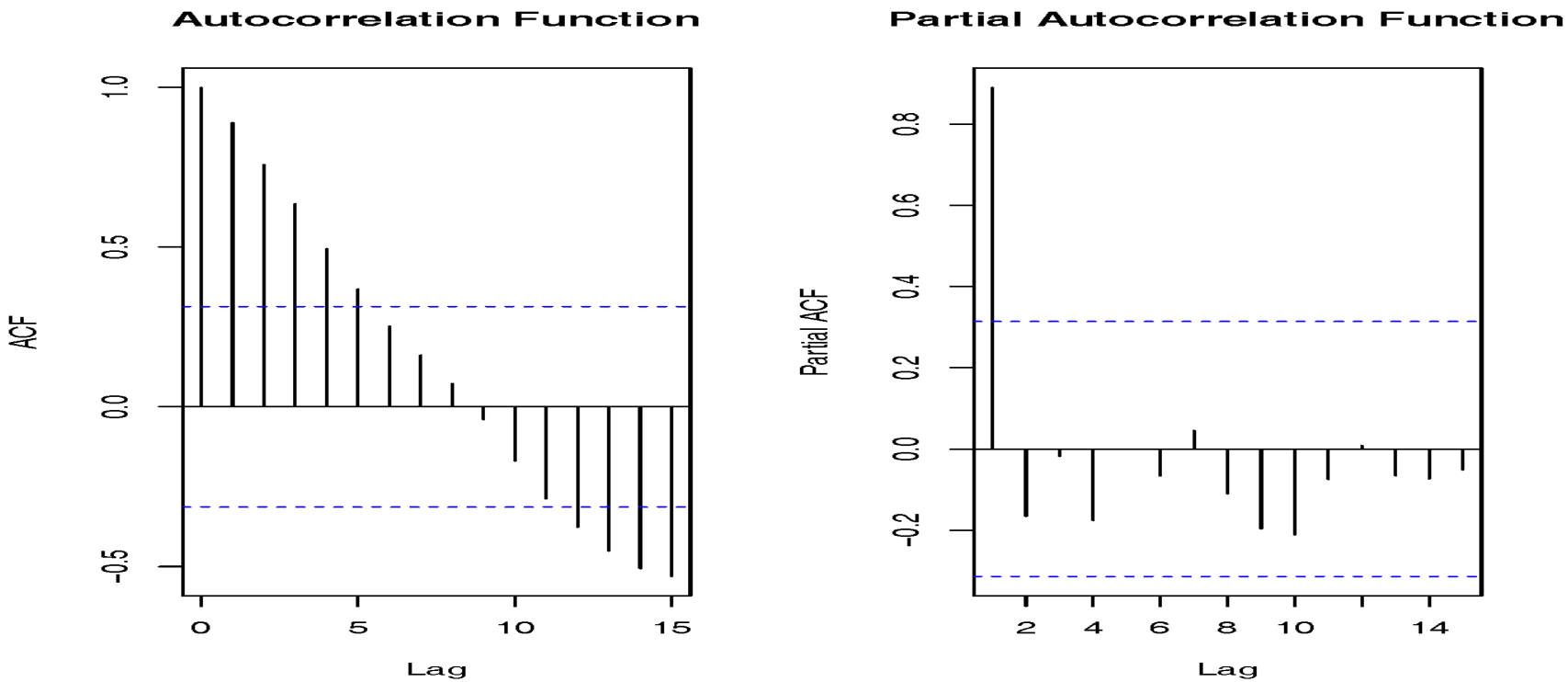


Figure : ACF and PACF exchange rate data.

```
R> x.ma <- arima(Z.ts, order = c(0, 0, 1))
R> x.ar <- arima(Z.ts, order = c(1, 0, 0))
R> x.arma <- arima(Z.ts, order = c(1, 0, 1))
R> AIC(x.ma); AIC(x.ar); AIC(x.arma)
[1] -3.526895
[1] -37.40417
[1] -42.27357
R> acf(resid(x.arma),main="ACF of residual for ARMA(1,1)")
R> pacf(resid(x.arma),main="PACF of residual for ARMA(1,1)"
```

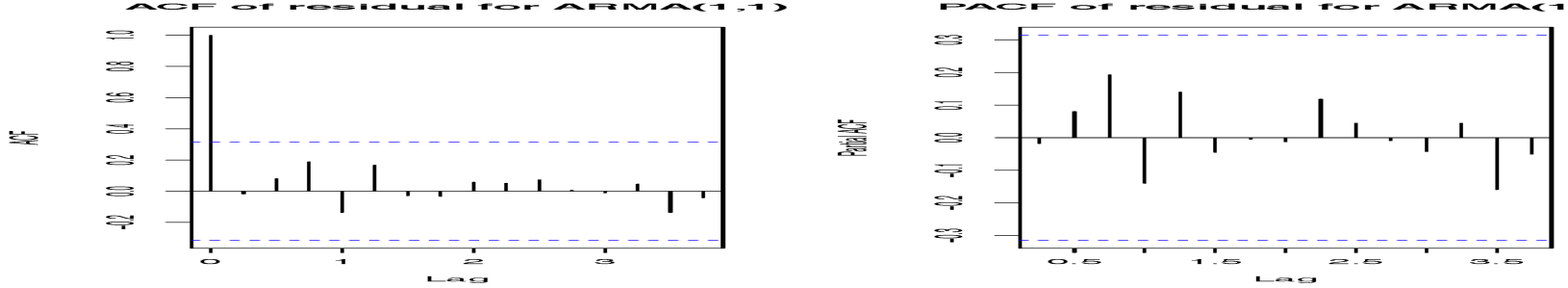


Figure : ACF & PACF of residual series for ARMA $(1, 1)$ model fitted to exchange rate data.

Residual analysis: testing for randomness

- Calculate the residuals, e_t, from the fitted model and plot their ACF and PACF to check that whether they are consistent with White Noise or not.
- The null hypothesis: H_0 : Residuals series is a White Noise sequence (i.e., $\rho_e(h) = 0, \forall h = 1, 2, \ldots, m$, for some $m > 1$).
- The alternative hypothesis: H_1 : Residuals series is not a White Noise sequence (i.e., $\rho_e(h) \neq 0$, for some $h = 1, 2, \ldots, m$).

Portmanteau test statistics

- Box and Pierce (1970) proved that the asymptotic distribution of the residual autocorrelations, $\rho_e(h), h = 1, 2, \ldots$, can be utilized to check the validity of the White Noise assumption under the ARMA (p, q) models.
- Under the null hypothesis that the process is White Noise, instead of testing randomness at each distinct lag, Box and Pierce (1970) introduced the *portmanteau test* on residuals autocorrelations up to lag m.
- Ljung and Box (1978) improved the Box and Pierce (1970) test by introducing a new portmanteau test.
- Peňa and Rodríguez (2006) proposed the generalized variance portmanteau test statistic and showed that their test statistic is more powerful than Ljung and Box test.

Box-Pierce and Ljung-Box portmanteau tests

- Box-Pierce (1970) portmanteau test statistic is

$$Q_m = n \sum_{h=1}^{m} \hat{\rho}^2(h) \sim \chi^2_{(m-p-q)},$$

- Ljung-Box (1978) portmanteau test statistic is

$$\tilde{Q}_m = n(n+2) \sum_{h=1}^{m} \frac{\hat{\rho}^2(h)}{n-h} \sim \chi^2_{(m-p-q)},$$

where n is the sample size, $\hat{\rho}(h)$ is the sample autocorrelation of the residual series at lag h, for $(p+q) < m < n$, and m is the number of lags being tested.

- The decision rule is to reject H_0 if Q_m (*or* $\tilde{Q}_m$) $\geq \chi^2_{\alpha,(m-p-q)}$, where $\chi^2_{\alpha,(m-p-q)}$ denotes the $100(1-\alpha)$th percentile of a chi-squared distribution with $m-p-q$ degrees of freedom.

Generalized variance portmanteau test

Peňa-Rodríguez (2006) portmanteau test statistic:

$$\tilde{D}_m = -n(m+1)^{-1} \log |\, \hat{\mathcal{R}}_m \,| \sim \Gamma(\alpha, \beta)$$

where $\Gamma(\alpha, \beta)$ stands for the gamma distribution of shape α and rate β, $|\cdot|$ is the determinant, and $\hat{\mathcal{R}}_m$ is the residual correlation matrix of order $m+1$,

$$\hat{\mathcal{R}}_m = \begin{pmatrix} 1 & \hat{\rho}(1) & \dots & \hat{\rho}(m) \\ \hat{\rho}(1) & 1 & \dots & \hat{\rho}(m-1) \\ \vdots & \dots & \ddots & \vdots \\ \hat{\rho}(m) & \hat{\rho}(m-1) & \dots & 1 \end{pmatrix}.$$

$$\alpha = \frac{3(m+1)[m-2(p+q)]^2}{2[2m(2m+1)-12(m+1)(p+q)]},$$

$$\beta = \frac{3(m+1)[m-2(p+q)]}{2m(2m+1)-12(m+1)(p+q)}$$

Portmanteau tests in R

- In R, the function Box.test() can be used to compute the p-value of Q_m and $\tilde{Q}_m$ test statistics.
- The R package **portes** contains a set of portmanteau diagnostic checks for univariate and multivariate time series, where the functions BoxPierce(), LjungBox() and gvtest() in this package can be used to compute the p-value of $Q_m, \tilde{Q}_m$ and $\tilde{D}_m$ statistics respectively.
- The decision rule is then to reject H_0 (Residuals series is a White Noise sequence) if the p-value is less than or equal to α (usually %5), the significance level.

Example: portmanteau test statistics

Example: The following table represents the sample autocovariances at lags $h = 0, 1, \ldots, 4$ for a realization of a time series of length $n = 50$:

h	0	1	2	3	4
$\hat{\gamma}(h)$	0.93673	-0.05292	-0.00140	0.06516	-0.01058

At the %5 level of significant, use the `Ljung-Box` portmanteau test to prove or disprove that the realization are coming from a White Noise series.

Solution: Note that $p = q = 0$, $m = 4$, and the critical value at $\alpha = \%5$ is $\chi^2_{(0.05,4)} = 0.711$. (see the next slide)

Example: portmanteau test statistics (Cont.)

h	0	1	2	3	4
$\hat{\gamma}(h)$	0.93673	-0.05292	-0.00140	0.06516	-0.01058
$\hat{\rho}(h)$	1.000	-0.056	-0.001	0.070	-0.011
$\hat{\rho}^2(h)$	1.000000	0.003136	0.000001	0.004900	0.000121

Ljung and Box portmanteau test statistic

$$\tilde{Q}_4 = 50(52)\sum_{h=1}^{4}\frac{\hat{\rho}^2(h)}{50-h} = 0.44436,$$

which is less than $\chi^2_{(0.05,4)} = 0.711$. Hence, we fail to reject the null hypothesis (The series is a White Noise series) and conclude that the realization are coming from a White Noise series.

Another example: portmanteau test statistics

The built in R function `FitAR()` in the package **FitAR** is used to select the best of fit model (AR(2) model), based on the BIC criterion, to the logarithms of Canadian lynx trappings from 1821 to 1934. Data is available from the R package **datasets** under the name `lynx`.

```
> library("FitAR")
> lynxData <- log(lynx)
> p <- SelectModel(lynxData, ARModel = "AR",
 + Criterion = "BIC", Best = 1)
> p
```

```
[1] 2
```

```
>Fitlynx <- FitAR(lynxData, p, ARModel = "AR")
```

Example: portmanteau test statistics (Cont.)

The `Ljung-Box` and `Peña-Rodríguez` portmanteau tests, available from the R package **portes**, are applied on the residuals of the fitted model at lag values 5, 10, 15, 20, 25, and 30, which indicate that the AR(2) model is not an adequate model.

```
> library("portes")
> LB <- LjungBox(Fitlynx)
> round(LB, 2)
```

Lags	Statistic	df	pvalue
5	7.03	3	0.07
10	17.17	8	0.03
15	24.93	13	0.02
20	34.23	18	0.01
25	39.13	23	0.02
30	44.36	28	0.03

```
> GV <- gvtest(Fitlynx)
> round(GV, 2)
```

Lags	Statistic	df	pvalue
5	5.98	2.09	0.05
10	10.04	5.86	0.12
15	21.45	9.61	0.01
20	31.81	13.37	0.00
25	38.76	17.12	0.00
30	43.94	20.87	0.00

Diagnostic plots for time series fits

- If the model fits well, the standardized residuals should show no obvious patterns (behave as an i.i.d. sequence with mean zero and variance one), where no outliers exceeding 3 standard deviations in magnitude.
- The R function tsdiag() produces diagnostic plots for time series fits. Generally plot the standardized residuals, the autocorrelation function of the residuals, and the p-values of a Ljung-Box Portmanteau test for all lags.

Example A regular time series giving the luteinizing hormone in blood samples at 10 mins intervals from a human female available from R under the name lh with 48 samples.
We fit an AR(3) model and use the R function tsdiag() to produce diagnostic plots for residuals of fitted model.

```
R>fit3.lh<-arima(lh,c(3,0,0));tsdiag(fit3.lh)
```

(see the next slide)

Diagnostic plots for time series fits

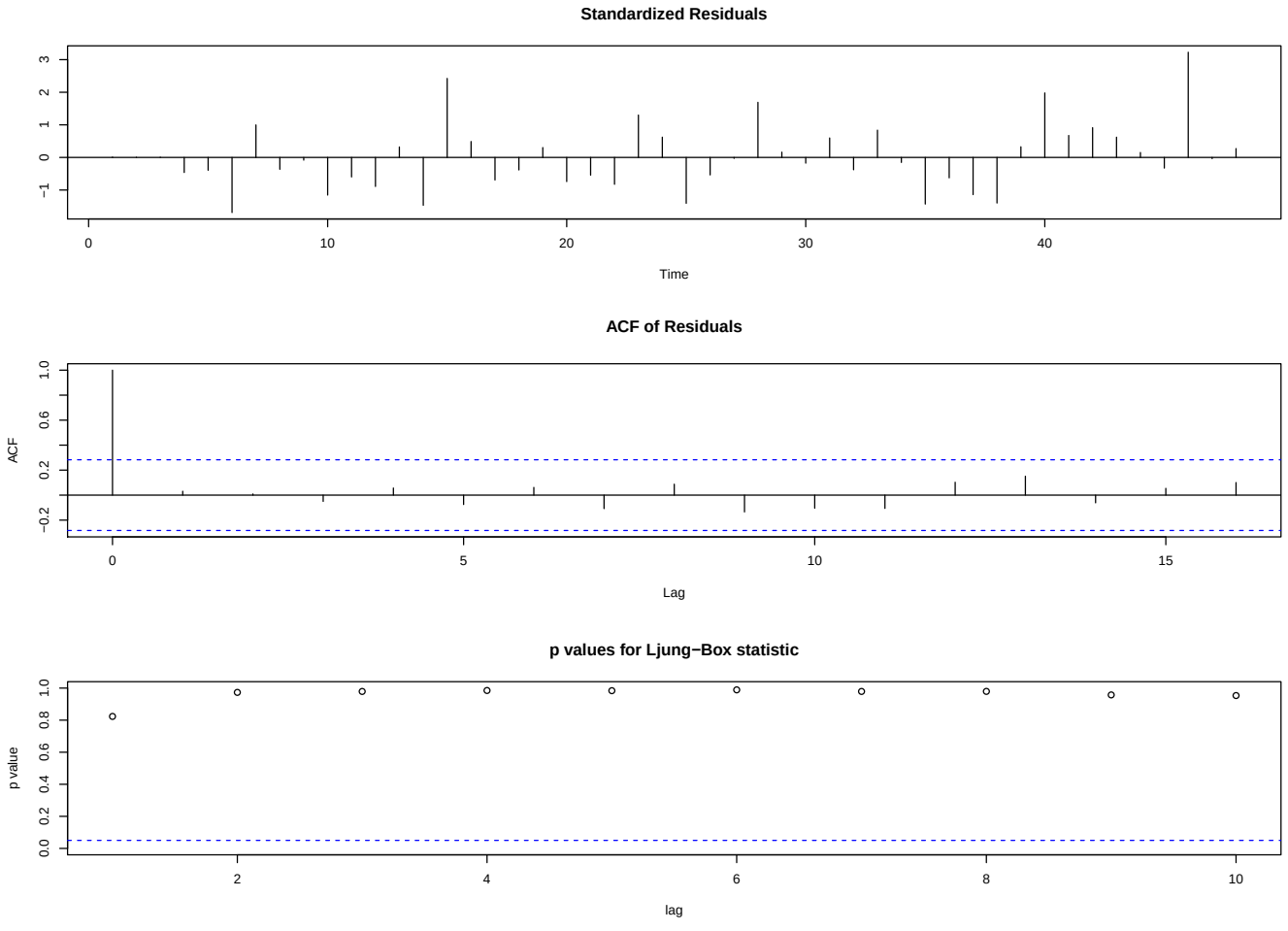


Figure : Diagnostic Plots for Time-Series Fit.

Normality of residuals

- In time series analysis, the residual is assumed to be normally distributed, so that the Maximum Likelihood Estimation (MLE) method provides estimates for the model's parameters.
- To test whether this assumptions is valid or not, we use the Quantile-Quantile plot of the residuals (denoted by Q-Q plot). If the residuals do have a normal distribution, the points in the plot will lie close to the diagonal line.
- Under the null assumption of normality (H_0 : data has normal distribution), the p-value of the Shapiro-Wilk test statistic yields to reject this assumption if it is greater than or equals to the level of significance (usually %5).

Normality of residuals

- In R, the function qqnorm() produces a normal Q-Q plot of points (X-axis is the theoretical quantiles, whereas Y-axis is the sample quantiles) and the function qqline() draw the diagonal line for normal Q-Q plots produced by qqnorm(). The function qqplot() produces a Q-Q plot of two datasets.
- The R function shapiro.test() produces the value of Shapiro-Wilk test and its p-value for testing the hypothesis null H_0 : data is normally distributed versus the alternative one H_1: data is not normally distributed.

Q-Q plot of residuals

```
R> fit.lh <- ar(lh)
R> resid.lh <- fit.lh$resid[-(1:3)]
R> qqnorm(resid.lh);  qqline(resid.lh)
```

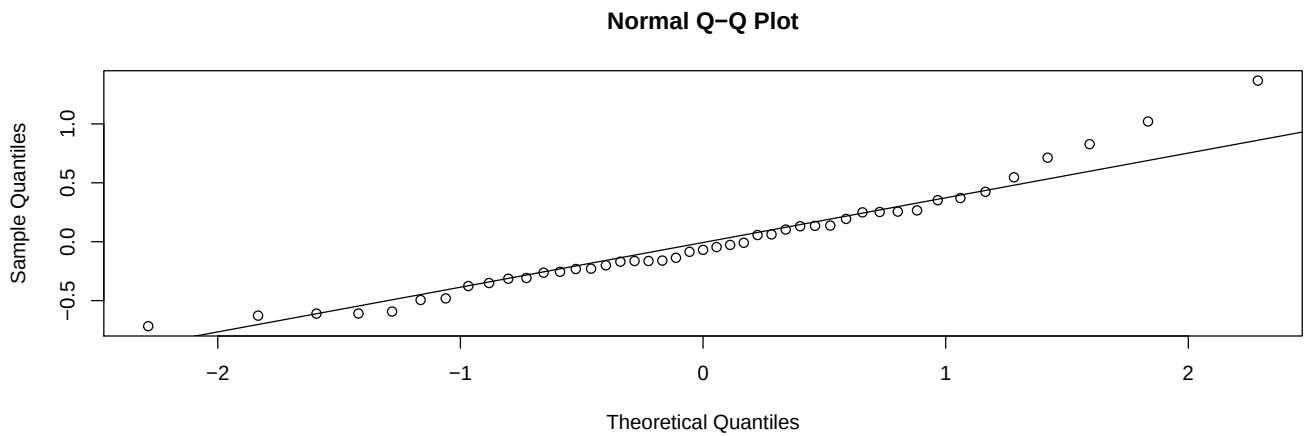


Figure : Q-Q, plot of the residuals.

Clearly, the Q-Q plot of the residuals indicates that the residuals are approximately normally distributed.

Forecasts for ARMA models

- Consider the time series $\{X_t\}$, where the values of $\{X_t\}$ are known up to time $t = n$. The forecast of X_t at $t = n + 1$ is not an observed value of the series and is written as $\hat{X}_{n+1|n}$.
- $\hat{X}_{n+1|n}$ is called *a one-step ahead forecast of* X_{n+1}, *since the forecast is one-step ahead of the available data* $X_n, x_{n-1}, \ldots$.
- In general, the *k-step ahead forecast is* $\hat{X}_{n+k|n}$, *and we may express it as a linear combination of the form:*

$$\hat{X}_{n+k|n} = a_0 + a_1 X_n + a_2 X_{n-1} + \ldots$$

- Our objective is to produce an optimum forecast with minimum Mean Square Error (MSE) forecast, where MSE is defined as $\text{MSE} = \mathbb{E}(X_{n+k} - \hat{X}_{n+k|n})^2$.
- Note a k-step ahead forecast can be written in many ways such as $\hat{X}_{n+k|n}$ or $\hat{X}_{n|n-k}$ or $\hat{X}_{n-2|n-k-2}$.

Forecasting AR models

Consider the AR(p) model:

$$X_n = \phi_1 X_{n-1} + \phi_2 X_{n-2} + \ldots + \phi_p X_{n-p} + \epsilon_n, \text{ where } \epsilon_n \sim WN(0, \sigma_\epsilon^2),$$

Suppose a one-step ahead forecast ($\hat{X}_{n+1|n}$) is required if the information about the time series $\{X_n\}$ is known up to time n.

$$\begin{aligned} X_{n+1} &= \phi_1 X_n + \phi_2 X_{n-1} + \ldots + \phi_p X_{n-p+1} + \epsilon_{n+1} \\ X_{n+1|n} &= \phi_1 X_{n|n} + \phi_2 X_{n-1|n} + \ldots + \phi_p X_{n-p+1|n} + \epsilon_{n+1|n} \\ \hat{X}_{n+1|n} &= \phi_1 \hat{X}_{n|n} + \phi_2 \hat{X}_{n-1|n} + \ldots + \phi_p \hat{X}_{n-p+1|n} + \hat{\epsilon}_{n+1|n} \end{aligned}$$

Now, since information is known up to time n, the values of $\hat{X}_{n|n}, \hat{X}_{n-1|n}, \ldots, \hat{X}_{n-p+1|n}$ are known exactly: they are the values of X at time $n, n-1, \ldots, n-p+1$, respectively (i.e., $\hat{X}_{n|n} = X_n, \hat{X}_{n-1|n} = X_{n-1}, \ldots, \hat{X}_{n-p+1|n} = X_{n-p+1}$). For the unknown value of $\hat{\epsilon}_{n+1|n}$, it can be replaced by its mean value, which is zero. (see the next slide)

Forecasting AR models

The one-step ahead forecast is given by:

$$\hat{X}_{n+1|n} = \phi_1 X_n + \phi_2 X_{n-1} + \ldots + \phi_p X_{n-p+1}$$

The error in making the forecast is $\epsilon_{n+1} = X_{n+1} - \hat{X}_{n+1|n}$.
In general, the k-step ahead forecasts can be written as:

$$\hat{X}_{n+k|n} = \phi_1 \hat{X}_{n+k-1|n} + \phi_2 \hat{X}_{n+k-2|n} + \ldots + \phi_p \hat{X}_{n+k-p|n} + \hat{\epsilon}_{n+k|n}$$

As before, the value of $\hat{\epsilon}_{n+k|n}$ is replaced by its mean value zero.

$$\hat{X}_{n+k|n} = \phi_1 \hat{X}_{n+k-1|n} + \phi_2 \hat{X}_{n+k-2|n} + \ldots + \phi_p \hat{X}_{n+k-p|n}$$

The error in making the forecast is $X_{n+k} - \hat{X}_{n+k|n}$.

Example: forecasting AR(1) model

Example: Consider the AR(1) model: $X_t = 0.6X_{t-1} + \epsilon_t$, where we observed $X_{100} = 0.9$. Find the forecasts for $X_{101}, X_{102}, X_{103}$ and X_{104}.

Solution:

$$\begin{aligned}
\hat{X}_{101|100} &= 0.6X_{100} = 0.6 \times 0.9 = 0.54 \\
\hat{X}_{102|100} &= 0.6\hat{X}_{101|100} = 0.6 \times 0.54 = 0.324 \\
\hat{X}_{103|100} &= 0.6\hat{X}_{102|100} = 0.6 \times 0.324 = 0.1944 \\
\hat{X}_{104|100} &= 0.6\hat{X}_{103|100} = 0.6 \times 0.1944 = 0.11664
\end{aligned}$$

The general form of the k-step ahead forecasts for the AR(1) model is:

$$\hat{X}_{t+k|t} = \phi^k X_t, k \geq 1$$

Example: forecasting AR(2) model

Example: Consider the AR(2) model: $X_t = 0.4X_{t-1} - 0.2X_{t-2} + \epsilon_t$, where we observed $X_{199} = 1.2$ and $X_{200} = 0.9$. Find the forecasts for $X_{201}, X_{202}, X_{203}$ and X_{204}.

Solution: We use the k-step ahead forecasts for the AR(p), where $p = 2$, $\phi_1 = 0.4, \phi_2 = -0.2$, $n = 200$, and $k = 1, 2, 3, 4$.

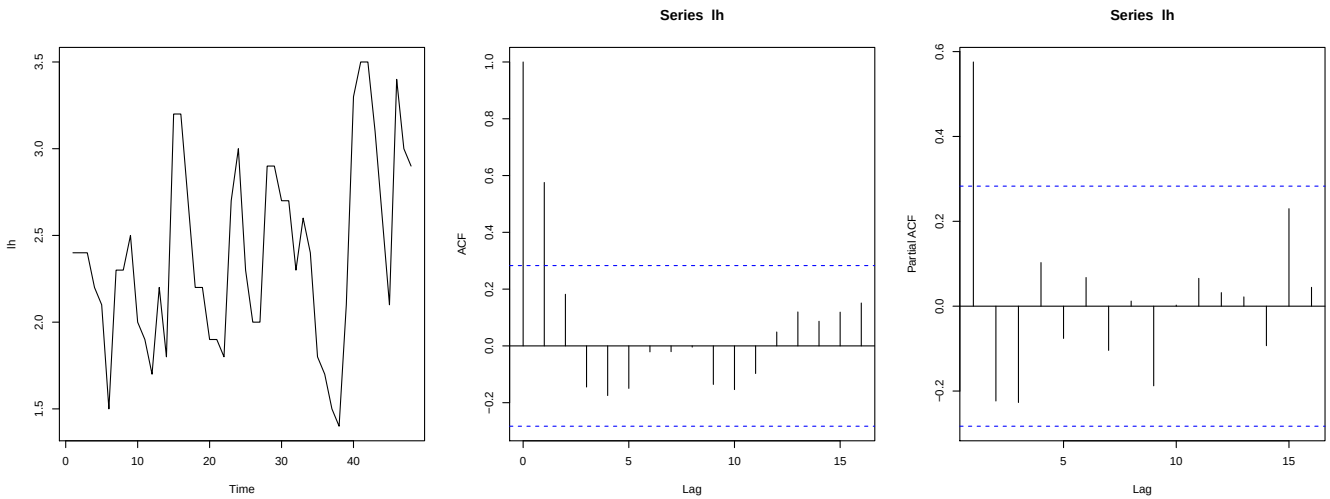
$$\hat{X}_{200+k|200} = 0.4\hat{X}_{200+k-1|200} - 0.2\hat{X}_{200+k-2|200}$$

$$\begin{aligned}
\hat{X}_{201|200} &= 0.4X_{200} - 0.2X_{199} = 0.4(0.9) - 0.2(1.2) = 0.12 \\
\hat{X}_{202|200} &= 0.4\hat{X}_{201|200} - 0.2X_{200} = 0.4(0.12) - 0.2(0.9) = -0.132 \\
\hat{X}_{203|200} &= 0.4\hat{X}_{202|200} - 0.2\hat{X}_{201|200} \\
&= 0.4(-0.132) - 0.2(0.12) = -0.0768 \\
\hat{X}_{204|200} &= 0.4\hat{X}_{203|200} - 0.2\hat{X}_{202|200} \\
&= 0.4(-0.0768) - 0.2(-0.132) = -0.00432
\end{aligned}$$

Luteinizing hormone in blood samples data

Consider the regular time series data available from R with the name lh. The data is 48 samples giving the luteinizing hormone in blood samples at 10 minute intervals from a human female.

```
R> par(mfrow=c(1,3))
R> plot(lh); acf(lh); pacf(lh)
```



The ACF and PACF suggest to fit MA(1) or AR(1).

A fitting model for luteinizing hormone in blood samples

An AR(p) model can be fitted to data in R using the ar() function. The function ar() uses the smallest AIC (which is 64.18482) to choose the order, p, of the bestfitting AR(p) model, AR(3).

```
R> ar(lh); lh.MA1 <- arima(lh, order = c(0, 0, 1))

Call:
ar(x = lh)

Coefficients:
      1         2         3
 0.6534   -0.0636   -0.2269

Order selected 3  sigma^2 estimated as  0.1959
R> AIC(lh.MA1)
[1] 68.10389
```

Diagnostics for luteinizing hormone data AR(3) model

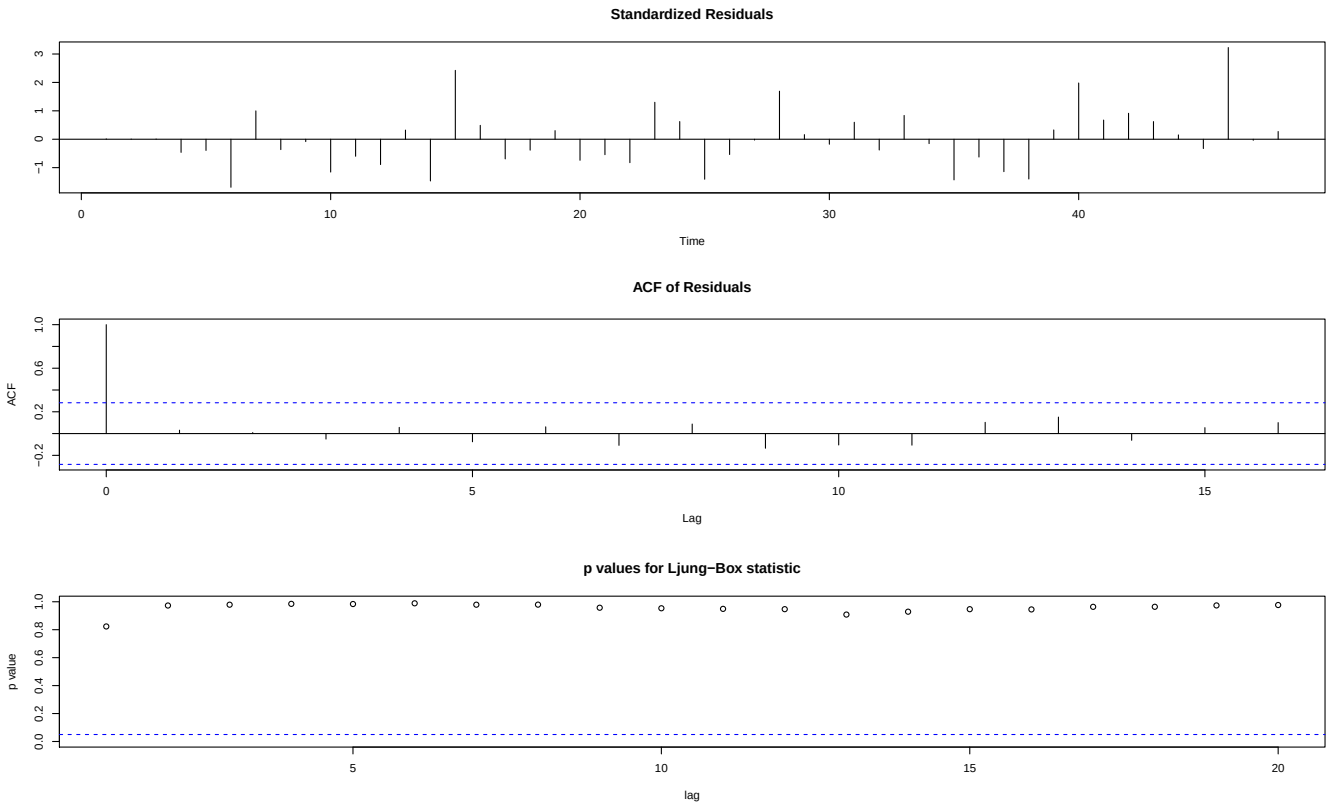


Figure : Diagnostics for luteinizing hormone Data (AR(3) Model).

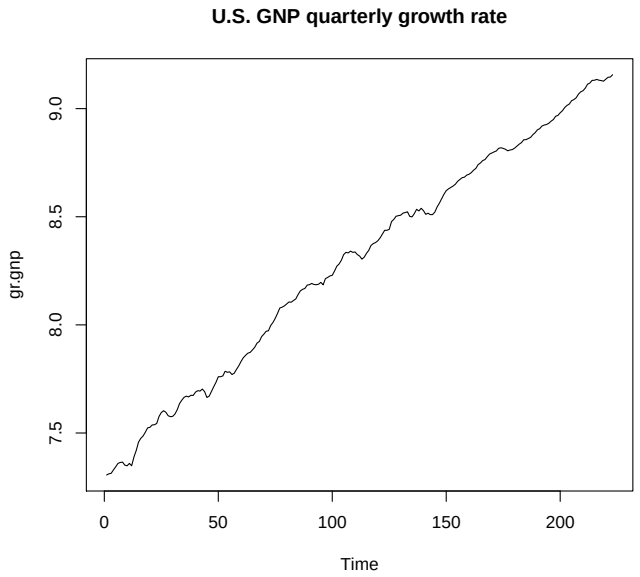
The model is: $X_t = 0.6534X_{t-1} - 0.0636X_{t-2} - 0.2269X_{t-3} + \epsilon_t$

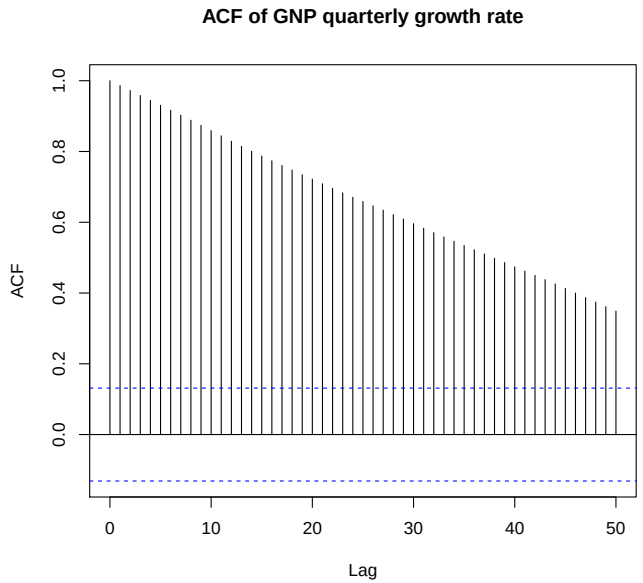
Real U.S. gross national product, GNP , quarterly data

- The quarterly U.S. GNP data from 1947 (Q1) to 2002 (Q3) was obtained from the Federal Reserve Bank of St. Louis, where it has been seasonally adjusted.
- Data is available as a time series (i.e., with `ts` object) from R package **astsa** with the name `gnp`.
- Recall that with financial data $\{X_t\}$, we define the *return* or *growth rate* to be $\{Y_t\}$, where $Y_t = \Delta[\log(X_t)]$.
- The plots of the U.S. GNP quarterly growth rate suggest that the GNP quarterly is not-stationarity (look like unit root series) with strong increasing trend. Thus the first difference of the data is needed to detrend the series.

Real U.S. GNP quarterly growth rate

```
R> library(astsa)
R> par(mfrow=c(1,2))
R> gr.gnp <- ts(log(gnp))
R> plot(gr.gnp,main="U.S. GNP quarterly growth rate")
R> acf(gr.gnp,50,main="ACF of GNP quarterly growth rate")
```





Real U.S. GNP quarterly growth rate (Cont.)

```
R> par(mfrow=c(1,3))
R> gnp.gr <- ts(diff(log(gnp)))
R> plot(gnp.gr,main="U.S. GNP quarterly growth rate")
R> acf(gnp.gr,24)
R> pacf(gnp.gr,24)
```

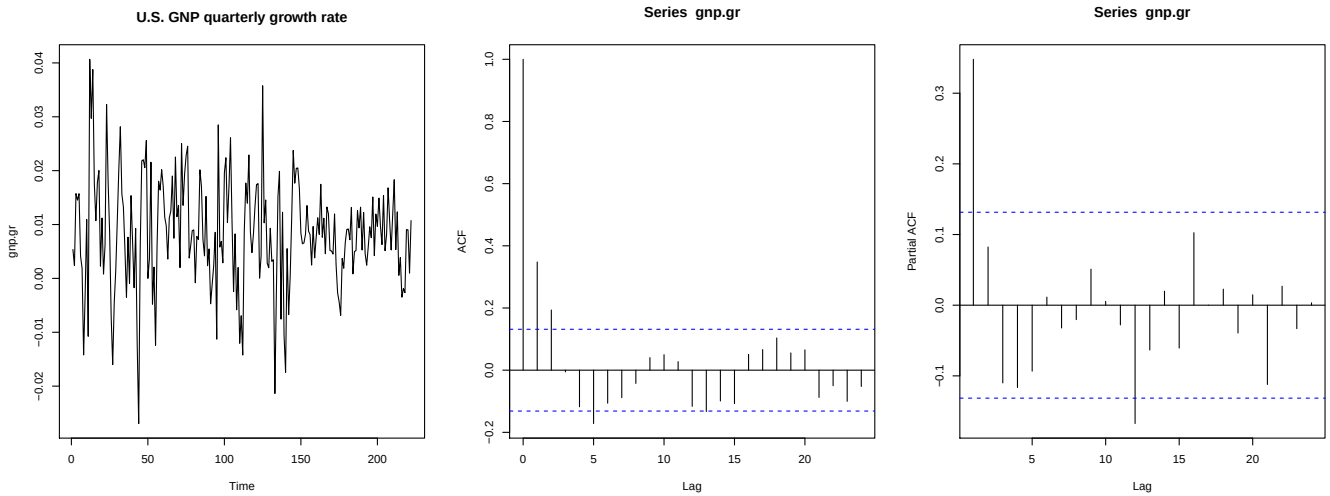


Figure : U.S. GNP quarterly growth rate.

Real U.S. GNP quarterly growth rate (Cont.)

- Inspecting the sample ACF and PACF , indicate that the ACF is cutting off at lag 2 and the PACF is tailing off. This would suggest the GNP growth rate follows an MA (2) process, or log GNP follows an ARIMA (0, 1, 2) model.

$$X_t = \mu + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2},$$

where X_t is the U.S. log GNP at time t and $\epsilon \sim WN(0, \sigma^2)$.

- Another close look at these plots appears that the ACF is tailing off and the PACF is cutting off at lag 1. This suggests an AR (1) model for the growth rate, or ARIMA (1, 1, 0) for log GNP .

$$X_t = \mu + \phi(X_{t-1} - \mu) + \epsilon_t,$$

where X_t is the U.S. log GNP at time t and $\epsilon \sim WN(0, \sigma^2)$.

AR(1) fit model for U.S. GNP quarterly growth rate

```
R> gnpgr.AR1 <- arima(gnp.gr, order = c(1, 0, 0))
R> gnpgr.AR1

Call:
arima(x = gnp.gr, order = c(1, 0, 0))

Coefficients:
         ar1  intercept
      0.3467     0.0083
s.e.  0.0627     0.0010

sigma^2 estimated as 9.03e-05:  log likelihood = 718.61,  a
```

The model is: $X_t = 0.0083 + 0.3467(X_{t-1} - 0.0083) + \epsilon_t,$

or

$$X_t = 0.0054 + 0.3467X_{t-1} + \epsilon_t$$

MA (2) fit model for U.S. GNP quarterly growth rate

```
R> gnpgr.MA2 <- arima(gnp.gr, order = c(0, 0, 2))
R> gnpgr.MA2

Call:
arima(x = gnp.gr, order = c(0, 0, 2))

Coefficients:
         ma1     ma2  intercept
      0.3028  0.2035     0.0083
s.e.  0.0654  0.0644     0.0010

sigma^2 estimated as 8.919e-05:  log likelihood = 719.96,
```

The model is: $X_t = 0.0083 + \epsilon_t + 0.3028\epsilon_{t-1} + 0.2035\epsilon_{t-2}$

Diagnostics for GNP growth rate (AR(1) model)

```
R> tsdiag(gnpgr.AR1, gof.lag=20)
```

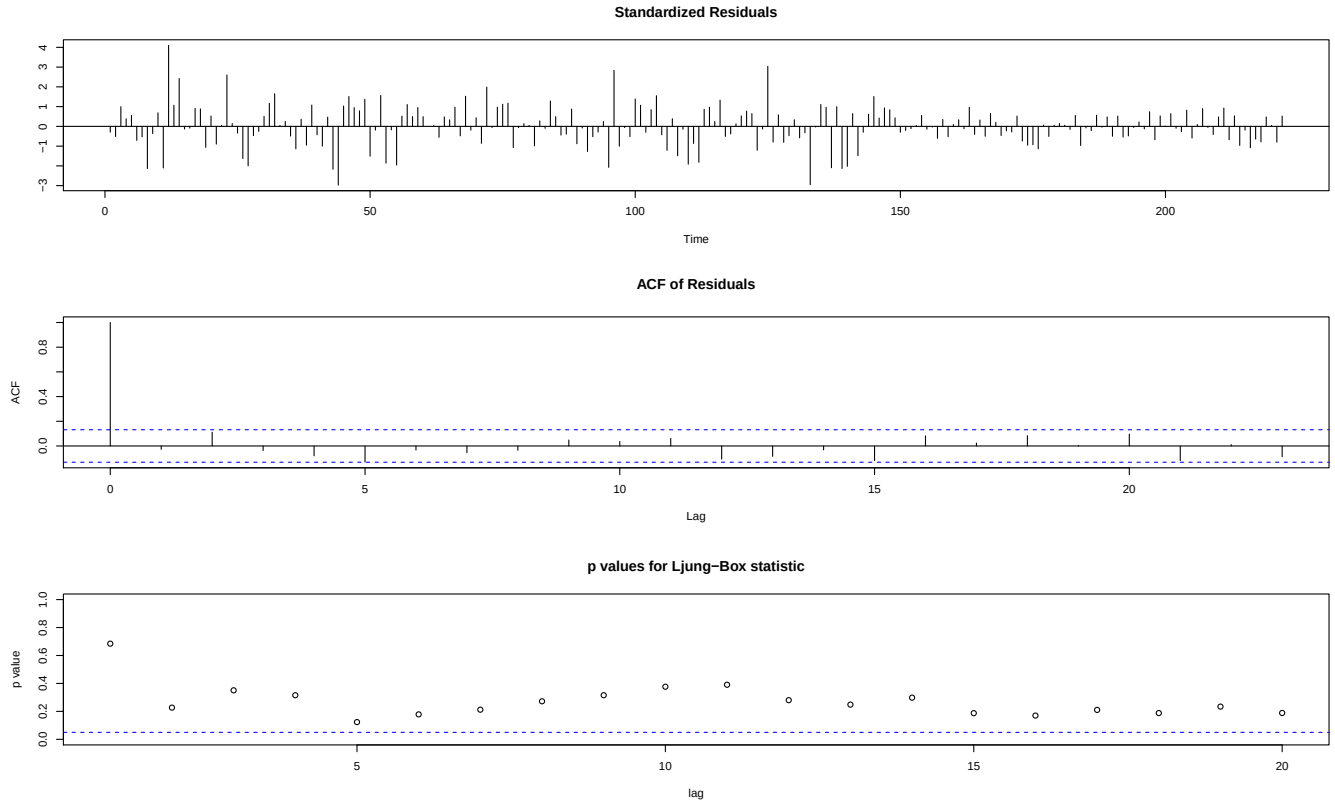


Figure : Diagnostics for U.S. GNP quarterly growth rate (AR(1) Model).

Diagnostics for GNP growth mate (MA (2) model)

```
R> tsdiag(gnpgr.MA2, gof.lag=20)
```

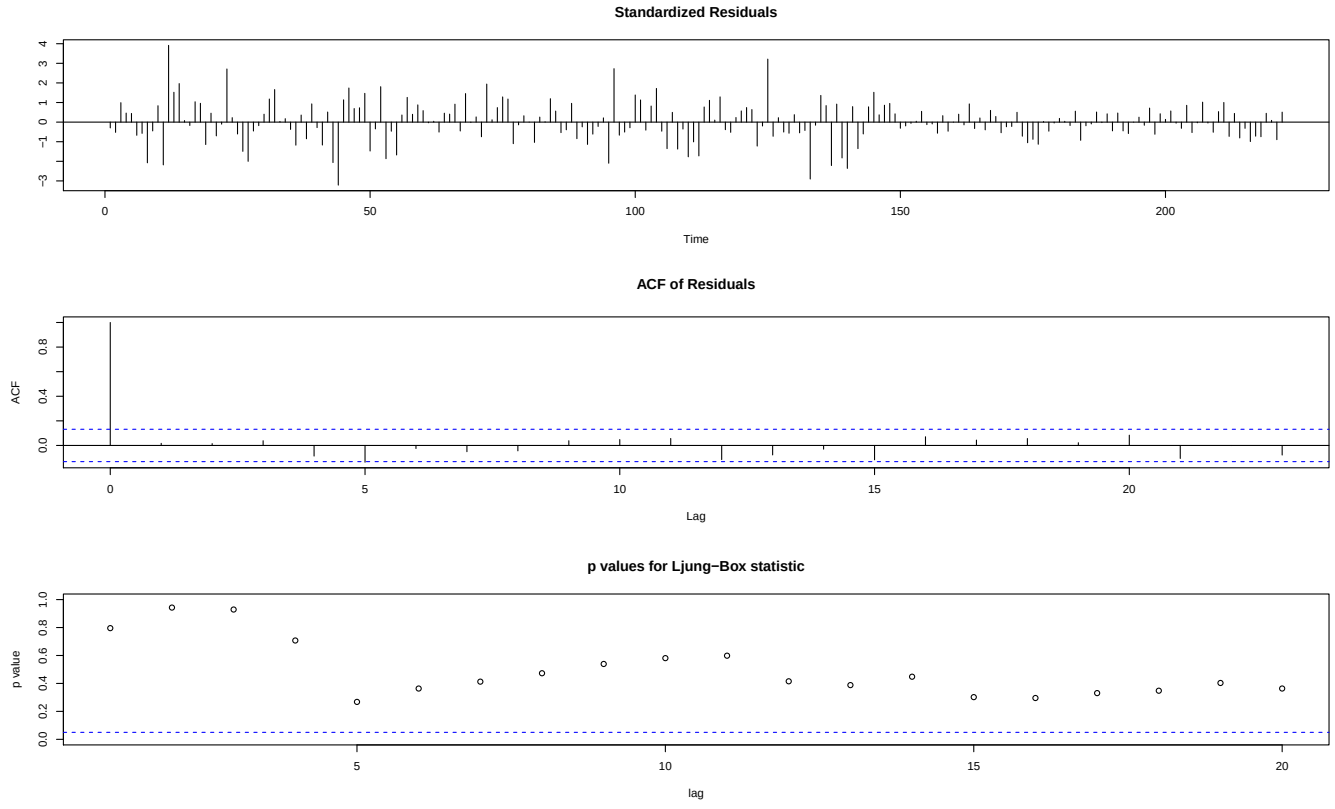
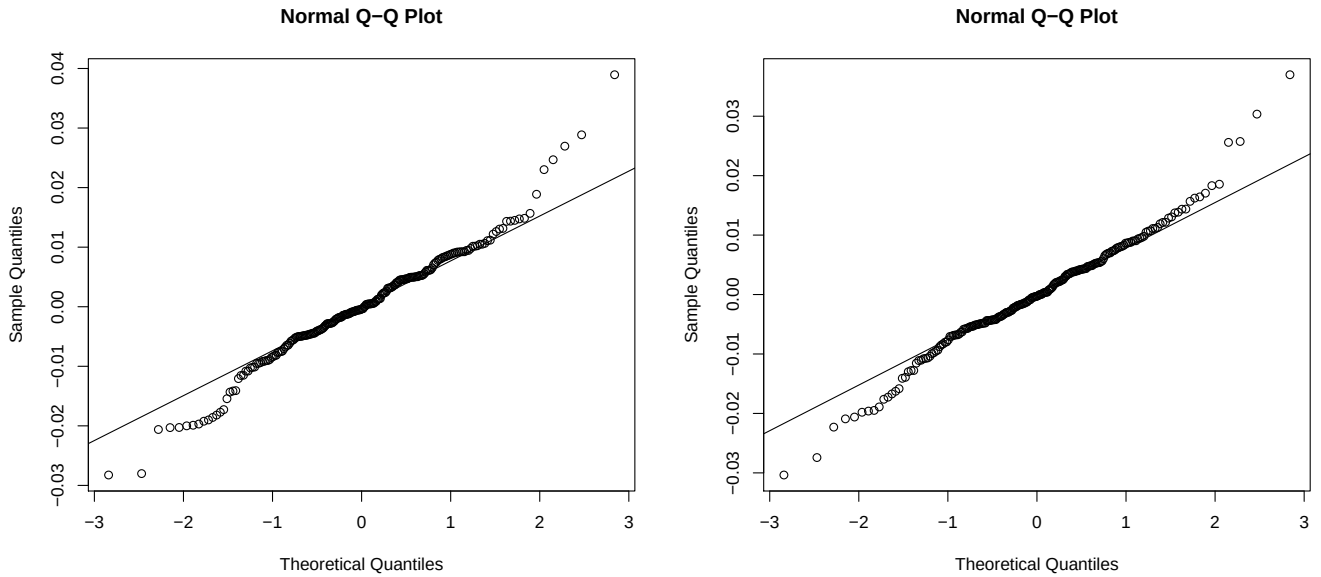


Figure : Diagnostics for U.S. GNP quarterly growth rate (MA (2) Model).

Diagnostics for GNP growth rate

```
R> par(mfrow=c(1,2))
R> qqnorm(gnpgr.AR1$resid);qqline(gnpgr.AR1$resid)
R> qqnorm(gnpgr.MA2$resid);qqline(gnpgr.MA2$resid)
R> tshAR1<-shapiro.test(gnpgr.AR1$resid)
R> tshMA2<-shapiro.test(gnpgr.MA2$resid)
```



Diagnostics for GNP growth rate

- Inspection of the time plot of the standardized residuals in the previous figures show no obvious patterns. However, there are some outliers values exceeding 3 standard deviations in magnitude. The ACF of the standardized residuals shows no apparent departure from the model assumptions, and the portmanteau statistic is never significant at the lags shown.
- Running a Shapiro-Wilk test for normality yields a p-values of 0.0007 and 0.003 for AR(1) and MA(2) respectively, which indicates the residuals are not normal. Hence, the model appears to fit well except for the fact that a distribution with heavier tails than the normal distribution should be employed.

Model choice for the U.S. GNP series

- To choose the final model, we compare the AIC or BIC for both models.

```
R> AIC(gnpgr.AR1)

[1] -1431.221

R> AIC(gnpgr.MA2)

[1] -1431.929
```

- The AIC prefers the MA (2) fit.

Thus, the final fitted model is the MA (2):

$$X_t = 0.0083 + \epsilon_t + 0.3028\epsilon_{t-1} + 0.2035\epsilon_{t-2}$$

List of some useful R functions

R-Function	Description
`ar()`	fit an autoregressive time series model to the data, by default selecting the complexity by AIC .
`arima()`	fit an ARIMA model to a univariate time series.
`AIC()`	Akaike information criterion for selection model.
`BIC()`	Bayesian information criterion for selection model.
`Box.test()`	compute the Box-Pierce or Ljung-Box portmanteau tests.
`BoxPierce()`	compute the univariate or multivariate Box-Pierce portmanteau test (**p**ortes package).
`LjungBox()`	compute the univariate or multivariate Ljung-Box portmanteau test (**p**ortes package).
`gvtest()`	compute the univariate Peňa-Rodríguez or multivariate Mahdi-McLeod portmanteau test (**p**ortes).
`tsdiag()`	diagnostic plots for time series fits.
`qqplot()`	produces a Q-Q plot of two datasets.
`qqnorm()`	produces a normal Q-Q plot of points.
`qqline()`	draw the diagonal line for normal Q-Q plots produced by `qqnorm()`.
`layout()`	divides the device up into as many rows and columns as there are in matrix `mat`, with the column-widths and the row-heights specified in the respective arguments.
`boxplot()`	produce box-and-whisker plot(s) of the given (grouped) values.
`start()`	extract and encode the times the first observation were taken.
`end()`	extract and encode the times the last observation were taken.
`frequency()`	returns the number of samples per unit time.
`polyroot()`	finds zeros of polynomials and roots of the characteristic equation to check for stationarity.
`det()`	calculate the Determinant of a Matrix.
`matrix()`	create a matrix from a given set of values.

Homework 1

Use R with set.seed(873) to generate a time series of length 300 from the ARMA $(1, 1)$

$$X_t = 0.7X_{t-1} + \epsilon_t + 0.3\epsilon_{t-1}.$$

- Plot the ACF up to lag 50 and explain what you get.
- Plot the PACF up to lag 50 and explain what you get.
- Plot the ACF up to lag 50 and explain what you get.
- Fit the models: ARIMA (2,0,0), ARIMA (0,0,2) ARIMA (1,0,1) and compute the AIC or/and BIC to select the best model.
- Perform diagnostic checking for the selected model.

Homework 2

- Consider the AR(1) model: $X_t = -0.8X_{t-1} + \epsilon_t$, where we observed $X_{50} = 0.5$. Find the forecasts for $X_{51}, X_{52}, X_{53}, X_{54}$, and X_{55}.
- Consider the AR(2) model: $X_t = 1.25X_{t-1} - 0.25X_{t-2} + \epsilon_t$, where we observed $X_{806} = 1.2$ and $X_{807} = 0.9$. Find the forecasts for $X_{808}, X_{809}, X_{810}, X_{811}$, and X_{812}.
- Consider the AR(3) model: $X_t = -0.4X_{t-1} + 0.6X_{t-2} - 0.1X_{t-3} + \epsilon_t$, where we observed $X_{402} = 1$ $X_{403} = 0.89$, and $X_{404} = 1.1$. Find the forecasts for $X_{405}, X_{406}, X_{407}, X_{408}, X_{409}, X_{410}, X_{411}$, and X_{412}.