Demo Time Series Analysis

AMS 316 Weiwei Tao

Dec 1, 2021

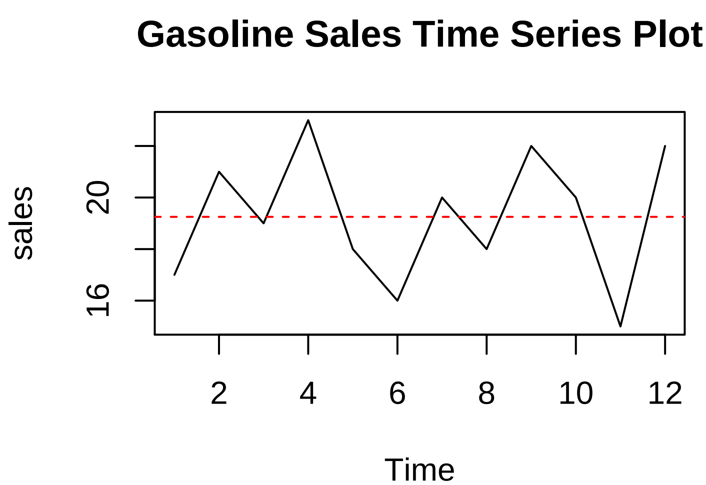
Construct a time series plot

A time series is a sequence of observations on a variable measured at successive points in time or over successive periods of time.

The pattern of the data is an important factor in understanding how the time series has behaved in the past.

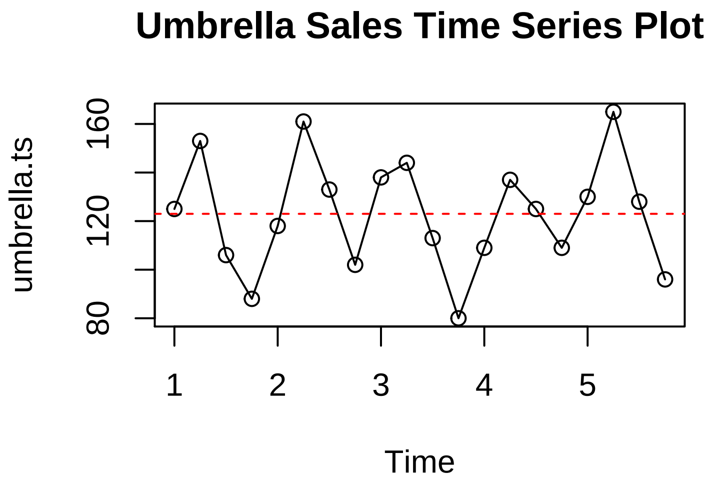
If such behavior can be expected to continue in the future, we can use the past pattern to guide us in selecting an appropriate forecasting method.

To identify the underlying pattern in the data, a useful first step is to construct a time series plot.



Time series with seasonal pattern

A time series plot for a stationary time series will always exhibit a horizontal pattern. But simply observing a horizontal pattern is not sufficient evidence to conclude that the time series is stationary.



White Noise and Random Walk

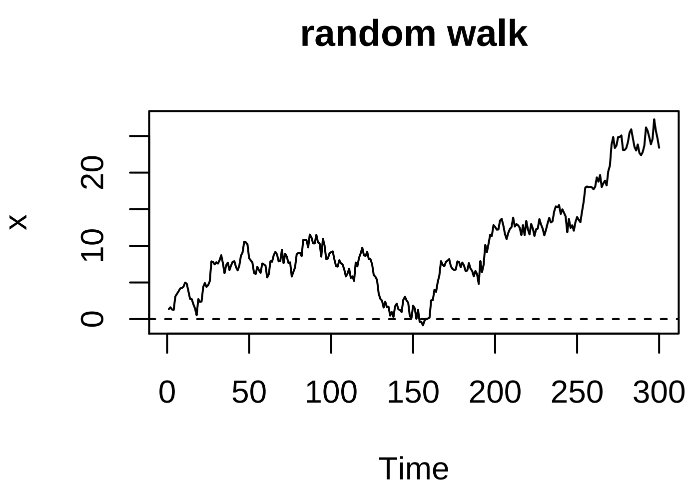
A simple kind of generated series might be a collection of uncorrelated random variables, w_t, with mean 0 and finite variance σ_w^2.

The time series generated from uncorrelated variables is used as a model for noise in engineering applications where it is called white noise. A particularly useful white noise series is Gaussian white noise.

The random walk is given by

$$x_t = x_{t-1} + w_t$$

for $t = 1, 2, \cdots$ with initial condition of $x_0 = 0$.



MA Model

MA model is always stationary.

Properties of MA(1) Model $X_t = \mu + w_t + \theta_1 w_{t-1}$.

- $E(X_t) = \mu$
- $Var(X_t) = \sigma_w^2(1 + \theta_1^2)$

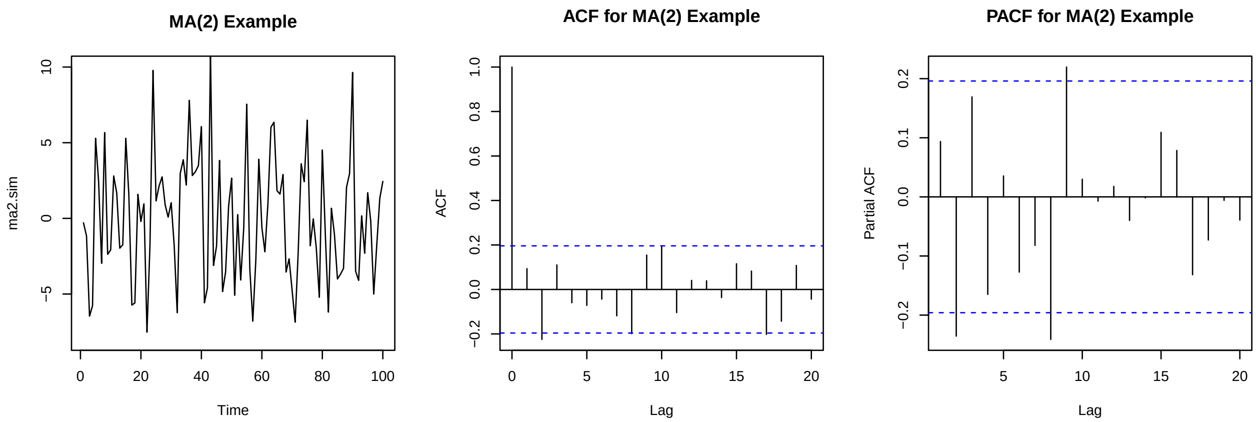
- ACF is:

$$\rho(1) = \frac{\theta_1}{1+\theta_1^2}$$

  and $\rho(k) = 0$ for $k > 0$.

For an MA model, the theoretical PACF does not shut off, but instead tapers toward 0 in some manner.

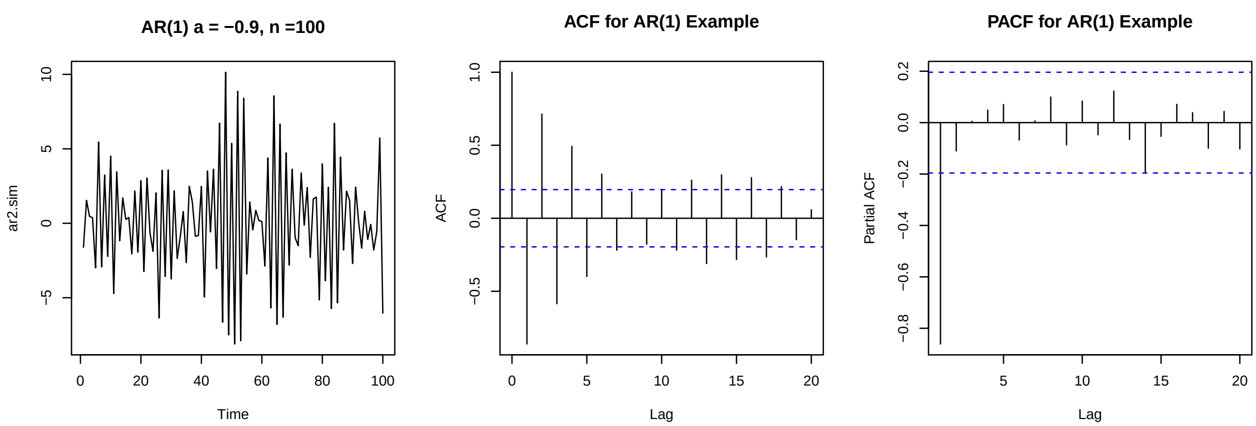
A clearer pattern for an MA model is in the ACF. The ACF will have non-zero autocorrelations only at lags involved in the model.



AR Model

AR models have theoretical PACFs with non-zero values at the AR terms in the model and zero values elsewhere.

The ACF will taper to zero in some fashion.



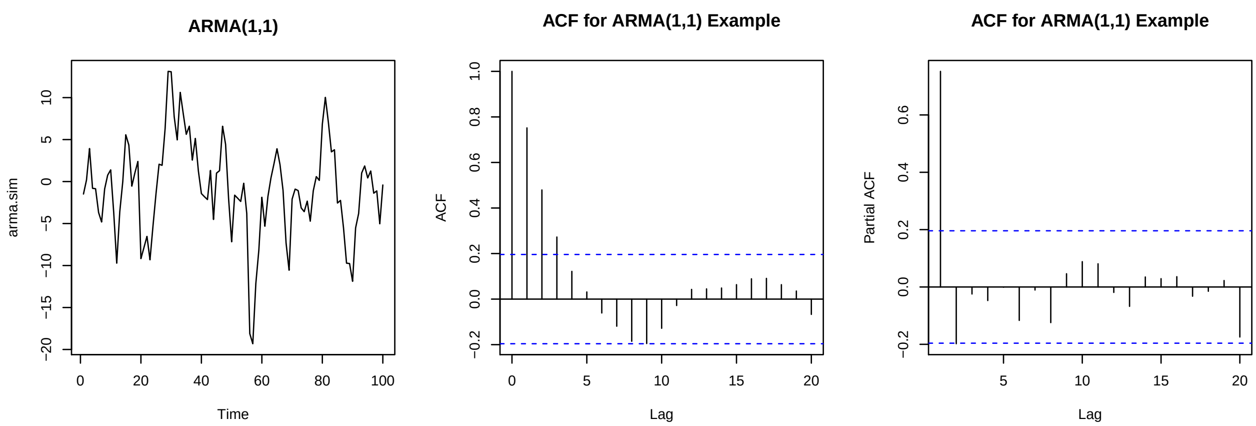
ARIMA Model

ARIMA models, also called Box-Jenkins models, are models that may possibly include autoregressive terms, moving average terms, and differencing operations. Various abbreviations are used:

- When a model only involves autoregressive terms it may be referred to as an AR model.

- When a model only involves moving average terms, it may be referred to as an MA model.
- When no differencing is involved, the abbreviation ARMA may be used.

ARMA models (including both AR and MA terms) have ACFs and PACFs that both tail off to 0. These are the trickiest because the order will not be particularly obvious.



Steps for Time Series Modeling

1. Guess that one or two terms of each type may be needed and then see what happens when you estimate the model.
2. After you've made a guess (or two) at a possible model and once the model has been estimated, do the following:

- Look at the significance of the coefficients.
- Look at the ACF of the residuals. For a good model, all autocorrelations for the residual series should be non-significant. If this isn't the case, you need to try a different model.
- Look at Box-Pierce (Ljung) tests for possible residual autocorrelation at various lags

If more than one model works?

- Possibly choose the model with the fewest parameters.
- Examine standard errors of forecast values. Pick the model with the generally lowest standard errors for predictions of the future.
- Compare models with regard to statistics such as the MSE (the estimate of the variance of the wt), AIC, and BIC. Lower values of these statistics are desirable.

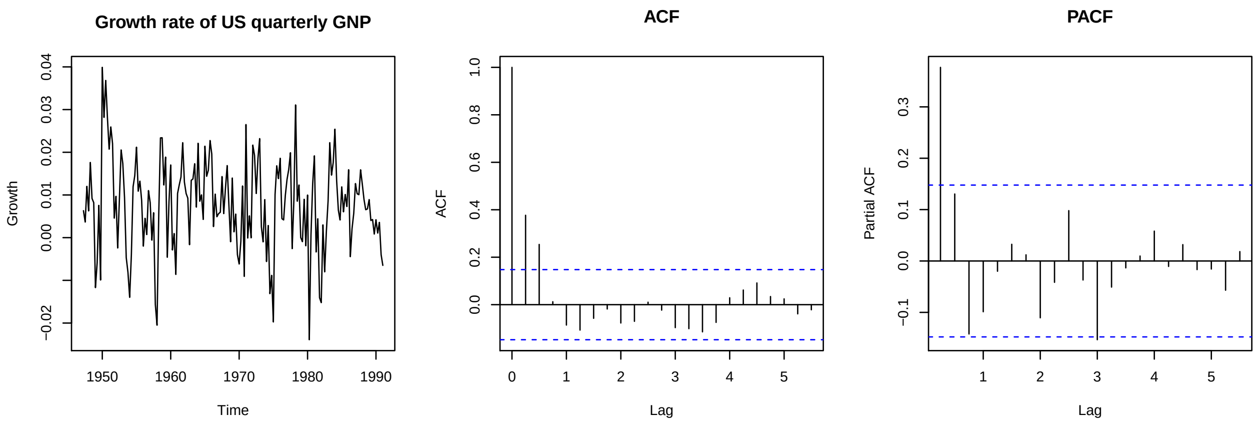
Example 1: Growth rate of US quarterly GNP

Read in the data and generate time series plot, ACF and PACF plots.

Guess: AR(1) model or AR(2)?

```
##          V1
## 1 0.00632
## 2 0.00366
```

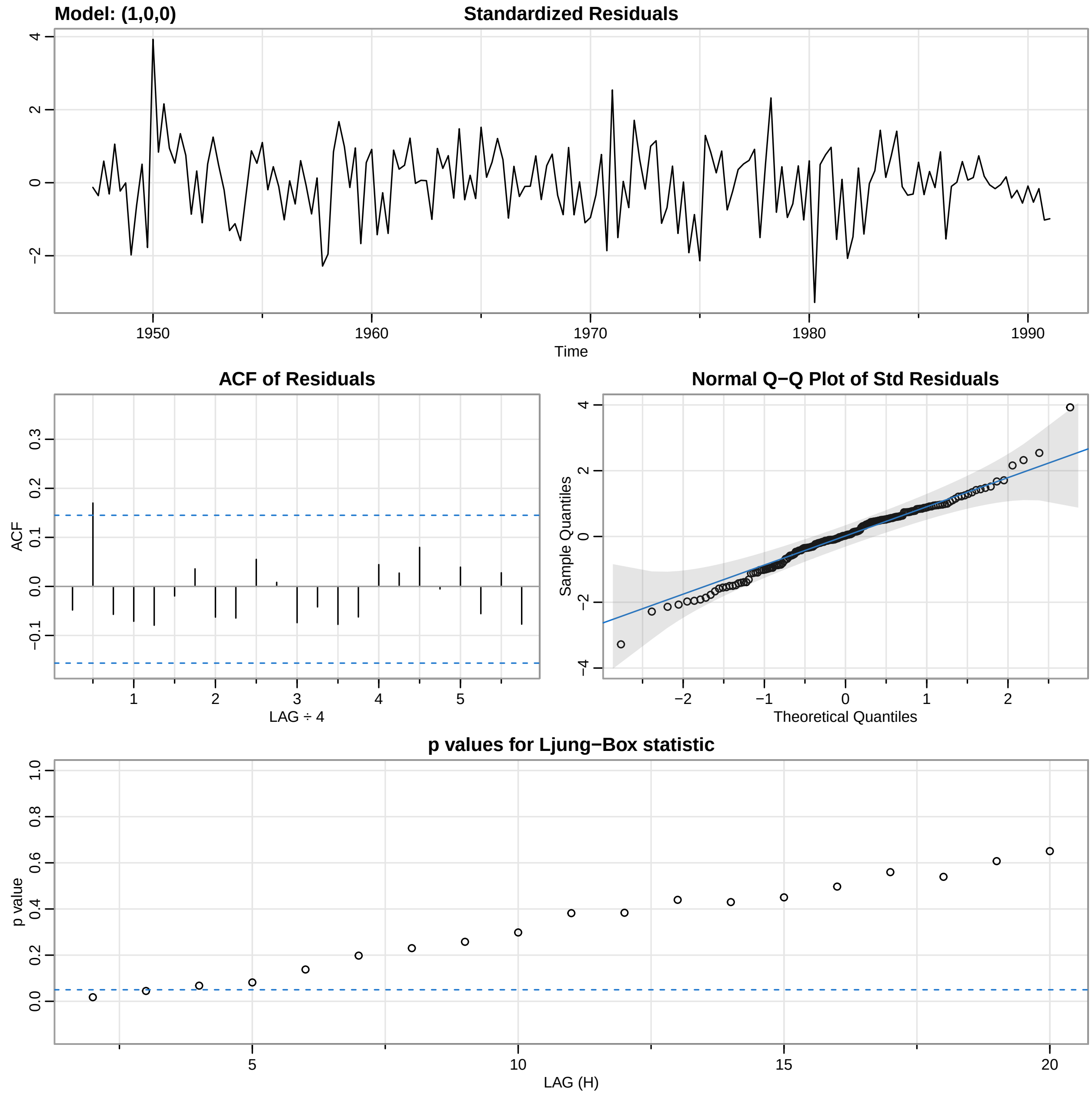
```
## 3 0.01202
## 4 0.00627
## 5 0.01761
## 6 0.00918
```



Statistics Diganosis

The Ljung-Box statistic, also called the modified Box-Pierce statistic. This statistic can be used to examine residuals from a time series model in order to see if all underlying population autocorrelations for the errors may be 0.

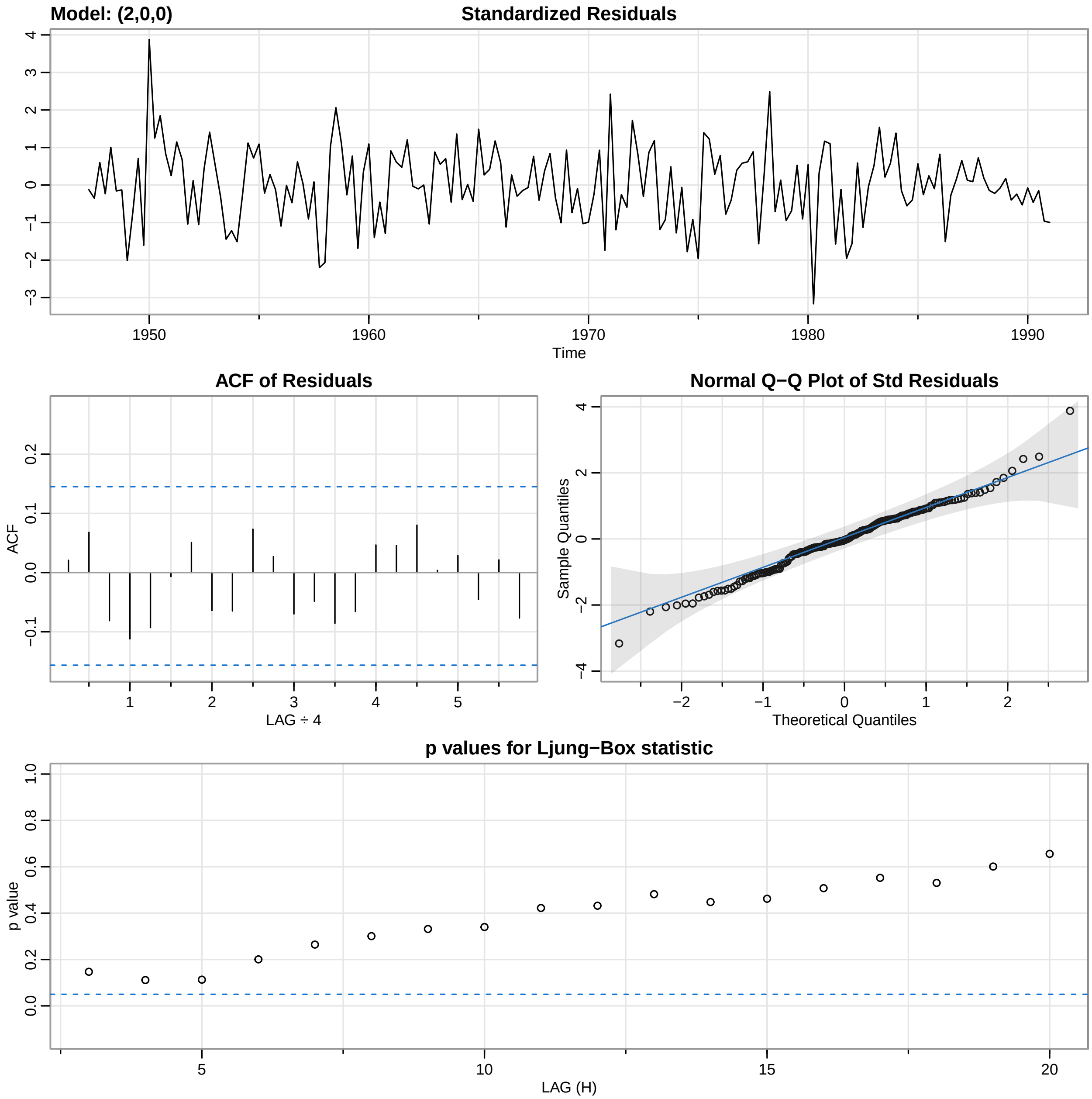
```
## initial  value -4.534991
## iter   2 value -4.612427
## iter   3 value -4.612436
## iter   4 value -4.612437
## iter   5 value -4.612439
## iter   5 value -4.612439
## iter   5 value -4.612439
## final  value -4.612439
## converged
## initial  value -4.614795
## iter   2 value -4.614798
## iter   3 value -4.614798
## iter   4 value -4.614798
## iter   4 value -4.614798
## iter   4 value -4.614798
## final  value -4.614798
## converged
```




```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,
##     optim.control = list(trace = trc, REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1   xmean
##       0.3787  0.0077
## s.e.  0.0698  0.0012
##
## sigma^2 estimated as 9.801e-05:  log likelihood = 562.47,  aic = -1118.94
##
```

```
## $degrees_of_freedom
## [1] 174
##
## $ttable
##       Estimate     SE t.value p.value
## ar1     0.3787 0.0698  5.4226       0
## xmean   0.0077 0.0012  6.4092       0
##
## $AIC
## [1] -6.357629
##
## $AICc
## [1] -6.357235
##
## $BIC
## [1] -6.303586
```

```
## initial  value -4.532539
## iter   2 value -4.606981
## iter   3 value -4.618641
## iter   4 value -4.618930
## iter   5 value -4.618943
## iter   6 value -4.618943
## iter   7 value -4.618944
## iter   7 value -4.618944
## iter   7 value -4.618944
## final  value -4.618944
## converged
## initial  value -4.623701
## iter   2 value -4.623706
## iter   3 value -4.623709
## iter   4 value -4.623710
## iter   5 value -4.623711
## iter   5 value -4.623711
## iter   5 value -4.623711
## final  value -4.623711
## converged
```



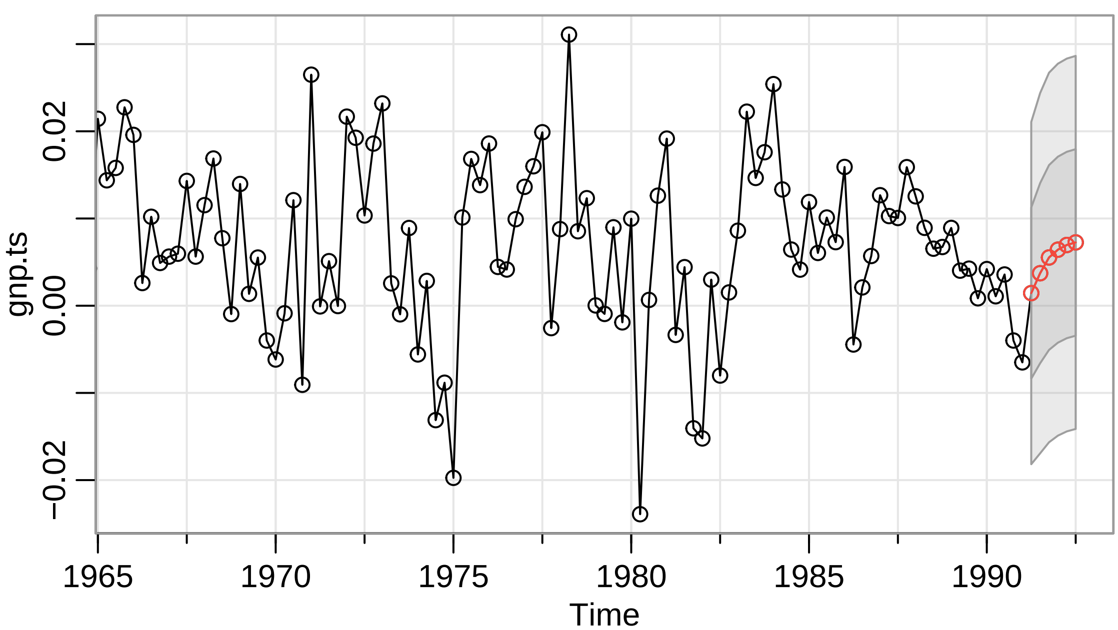

```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,
##     optim.control = list(trace = trc, REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1     ar2   xmean
##       0.3288  0.1331  0.0076
## s.e.  0.0746  0.0748  0.0014
##
## sigma^2 estimated as 9.626e-05:  log likelihood = 564.04,  aic = -1120.08
##
```

```
## $degrees_of_freedom
## [1] 173
##
## $ttable
##       Estimate     SE t.value p.value
## ar1     0.3288 0.0746  4.4084  0.0000
## ar2     0.1331 0.0748  1.7800  0.0768
## xmean   0.0076 0.0014  5.5863  0.0000
##
## $AIC
## [1] -6.36409
##
## $AICc
## [1] -6.363297
##
## $BIC
## [1] -6.292033
```

Predicition

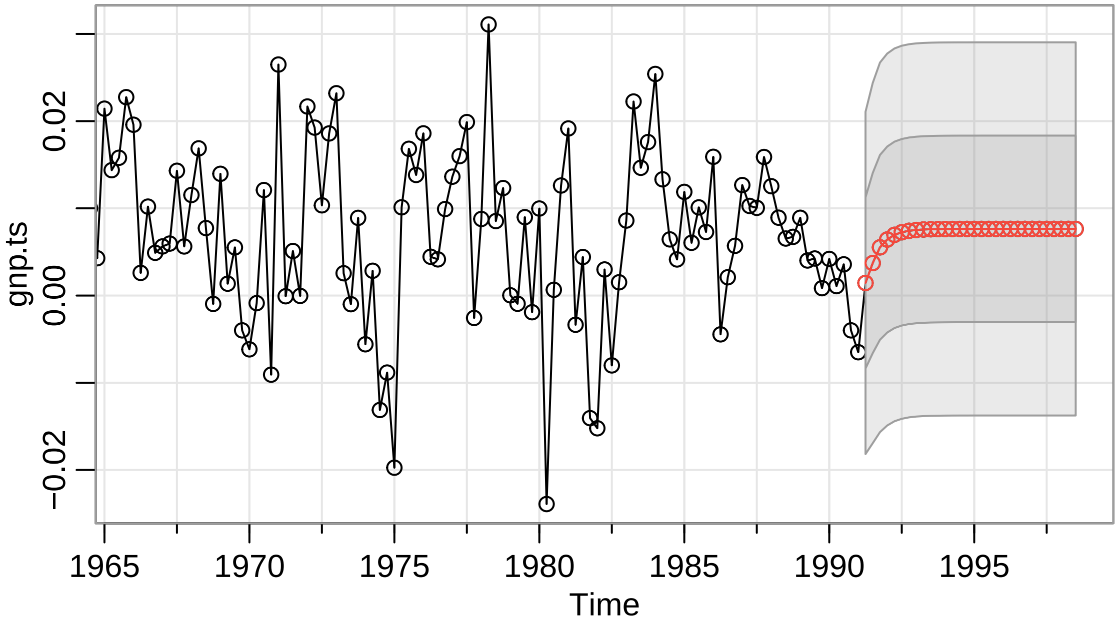
For a stationary series and model, the forecasts of future values will eventually converge to the mean and then stay there.

Note below what happened with the stride length forecasts, when we asked for 30 forecasts past the end of the series. [Command was sarima.for (stridelength, 30, 2, 0, 0)]. The forecast got to mean and then stayed there.




```
## $pred
##             Qtr1        Qtr2        Qtr3        Qtr4
## 1991             0.001445249 0.003723583 0.005529874
## 1992 0.006426920 0.006962205 0.007257563
##
## $se
```

```
##            Qtr1        Qtr2        Qtr3        Qtr4
## 1991             0.009810979 0.010327669 0.010595231
## 1992 0.010663777 0.010687502 0.010694785
```




```
## $pred
##            Qtr1        Qtr2        Qtr3        Qtr4
## 1991             0.001445249 0.003723583 0.005529874
## 1992 0.006426920 0.006962205 0.007257563 0.007425898
## 1993 0.007520546 0.007574064 0.007604254 0.007621301
## 1994 0.007630923 0.007636355 0.007639421 0.007641152
## 1995 0.007642129 0.007642681 0.007642992 0.007643168
## 1996 0.007643267 0.007643323 0.007643355 0.007643373
## 1997 0.007643383 0.007643389 0.007643392 0.007643394
## 1998 0.007643395 0.007643395 0.007643395
##
## $se
##            Qtr1        Qtr2        Qtr3        Qtr4
## 1991             0.009810979 0.010327669 0.010595231
## 1992 0.010663777 0.010687502 0.010694785 0.010697140
## 1993 0.010697886 0.010698124 0.010698200 0.010698224
## 1994 0.010698232 0.010698234 0.010698235 0.010698235
## 1995 0.010698235 0.010698235 0.010698235 0.010698235
## 1996 0.010698235 0.010698235 0.010698235 0.010698235
## 1997 0.010698235 0.010698235 0.010698235 0.010698235
## 1998 0.010698235 0.010698235 0.010698235
```

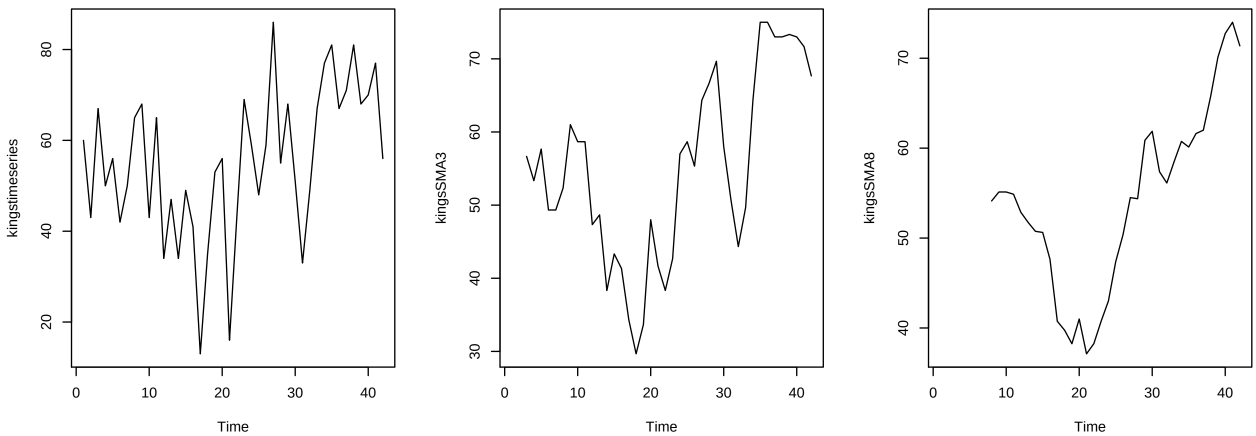
Example 2: Ages at Death of the Kings of England (Non-stationary with Trend)

Performs the Augmented Dickey-Fuller test for the null hypothesis of a unit root of a univarate time series x (equivalently, x is a non-stationary time series).

Recall that non-seasonal time series consist of a trend component and a random component. Decomposing the time series involves tying to separate the time series into these individual components.

One way to do this is using some smoothing method, such as a simple moving average. The SMA() function in the TTR R package can be used to smooth time series data using a moving average.

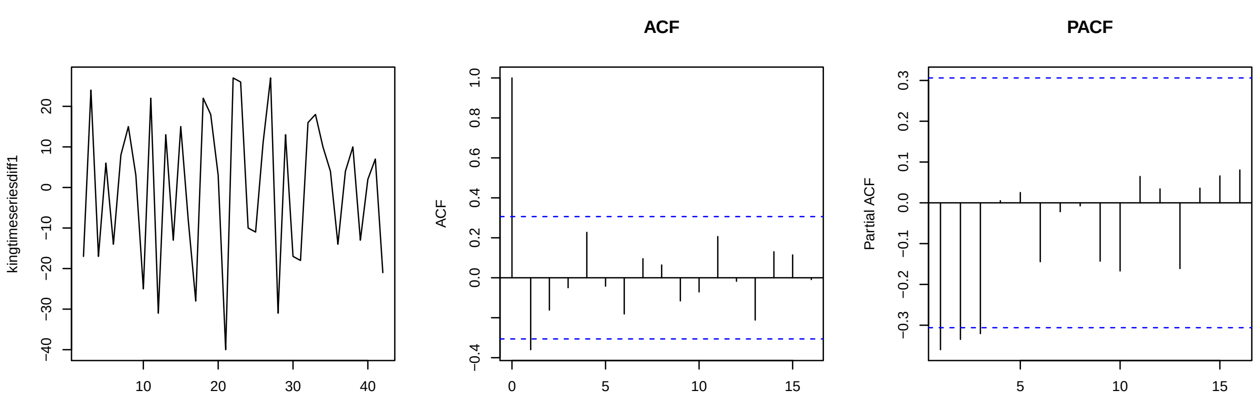
```
##
##  Augmented Dickey-Fuller Test
##
## data:  kingstimeseries
## Dickey-Fuller = -2.1132, Lag order = 3, p-value = 0.529
## alternative hypothesis: stationary
```



ADF test shows that the time series is not stationary.

In order to make it staionary, we can differencing the time series.

```
##
##  Augmented Dickey-Fuller Test
##
## data:  kingtimeseriesdiff1
## Dickey-Fuller = -4.0754, Lag order = 3, p-value = 0.01654
## alternative hypothesis: stationary
```



Since the correlogram is zero after lag 1, and the partial correlogram tails off to zero after lag 3, this means that the following ARMA (autoregressive moving average) models are possible for the time series of first differences:

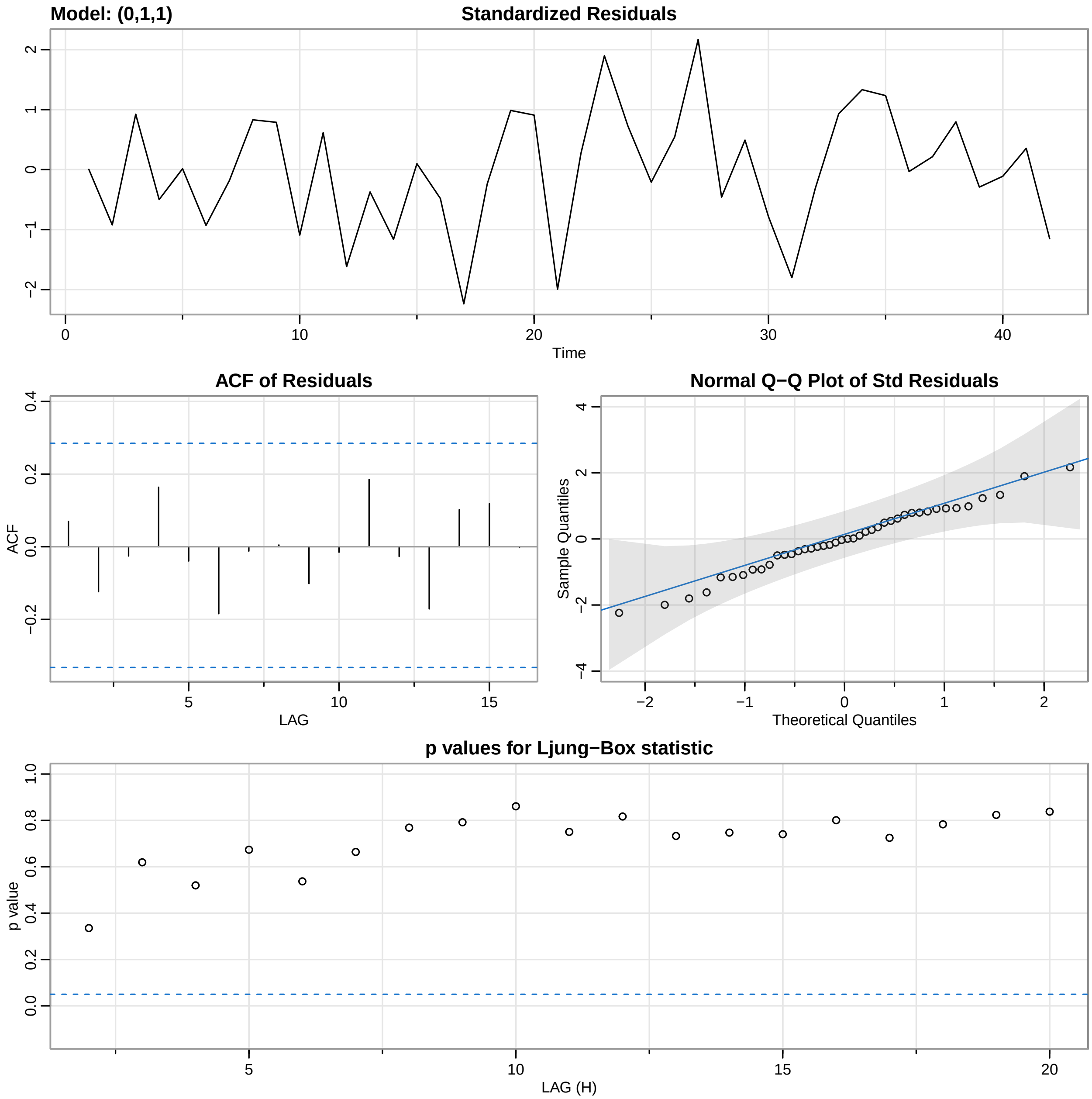
- an ARMA(3,0) model, that is, an autoregressive model of order p=3, since the partial autocorrelogram is zero after lag 3, and the autocorrelogram tails off to zero (although perhaps too abruptly for this model to be appropriate)
- an ARMA(0,1) model, that is, a moving average model of order q=1, since the autocorrelogram is zero after lag 1 and the partial autocorrelogram tails off to zero
- an ARMA(p,q) model, that is, a mixed model with p and q greater than 0, since the autocorrelogram and partial correlogram tail off to zero (although the correlogram probably tails off to zero too abruptly for this model to be appropriate)

Rule of Thumb

Pick the one with fewest parameter.

Pick ARIMA(0,1,1) to include both differencing and MA(1).

```
## initial  value 2.897384
## iter   2 value 2.786631
## iter   3 value 2.747600
## iter   4 value 2.734244
## iter   5 value 2.723824
## iter   6 value 2.721840
## iter   7 value 2.721500
## iter   8 value 2.721091
## iter   9 value 2.720780
## iter  10 value 2.720774
## iter  11 value 2.720773
## iter  11 value 2.720773
## final  value 2.720773
## converged
## initial  value 2.725322
## iter   2 value 2.725281
## iter   3 value 2.725130
## iter   3 value 2.725130
## iter   3 value 2.725130
## final  value 2.725130
## converged
```




```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     xreg = constant, transform.pars = trans, fixed = fixed, optim.control = list(trace = trc,
##         REPORT = 1, reltol = tol))
##
## Coefficients:
##           ma1  constant
##       -0.7463    0.3882
## s.e.   0.1278    0.6522
##
## sigma^2 estimated as 228.2:  log likelihood = -169.91,  aic = 345.81
##
```

```
## $degrees_of_freedom
## [1] 39
##
## $ttable
##          Estimate     SE t.value p.value
## ma1       -0.7463 0.1278 -5.8373  0.0000
## constant   0.3882 0.6522  0.5952  0.5552
##
## $AIC
## [1] 8.434478
##
## $AICc
## [1] 8.44218
##
## $BIC
## [1] 8.559861
```

Example 3: Australian beer production (Non-stationary with both trend and seasonality)

Decomposition procedures are used in time series to describe the trend and seasonal factors in a time series.

The following two structures are considered for basic decomposition models:

- Additive: = Trend + Seasonal + Random
- Multiplicative: = Trend * Seasonal * Random

Steps in Decomposition

1. The first step is to estimate the trend:

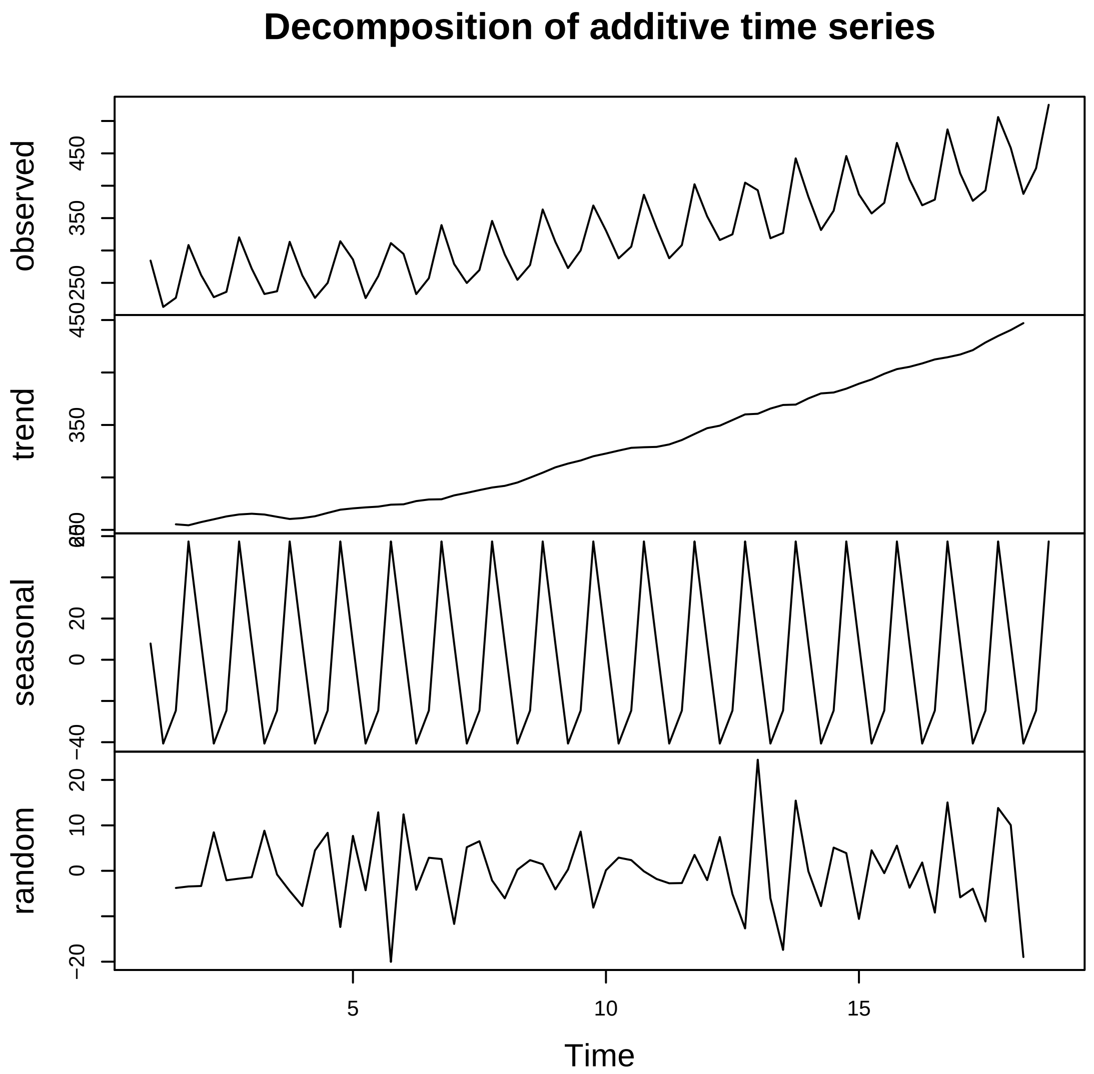
- moving averages
- The second approach is to model the trend with a regression equation.

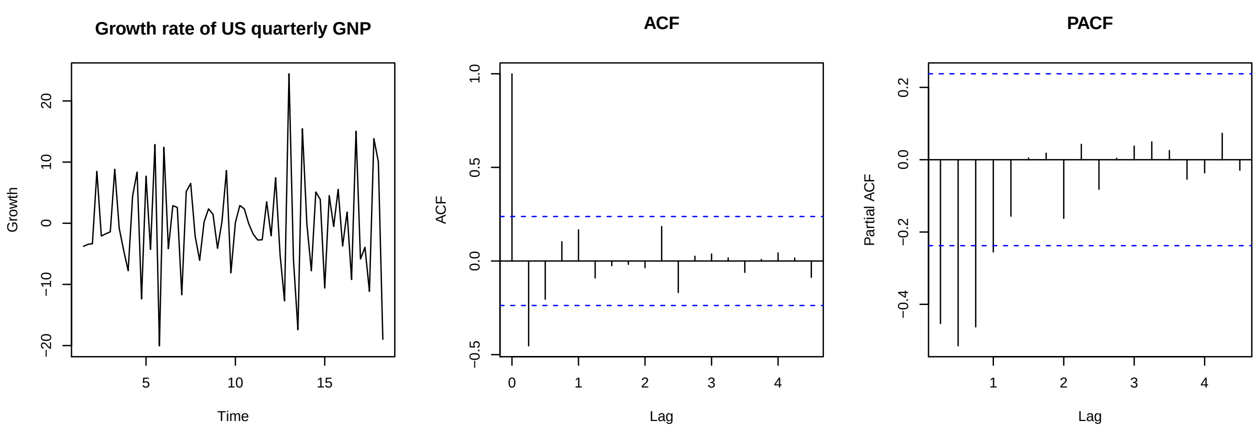
2. The second step is to "de-trend" the series. For an additive decomposition, this is done by subtracting the trend estimates from the series. For a multiplicative decomposition, this is done by dividing the series by the trend values.

3. Next, seasonal factors are estimated using the de-trended series.

4. The final step is to determine the random (irregular) component

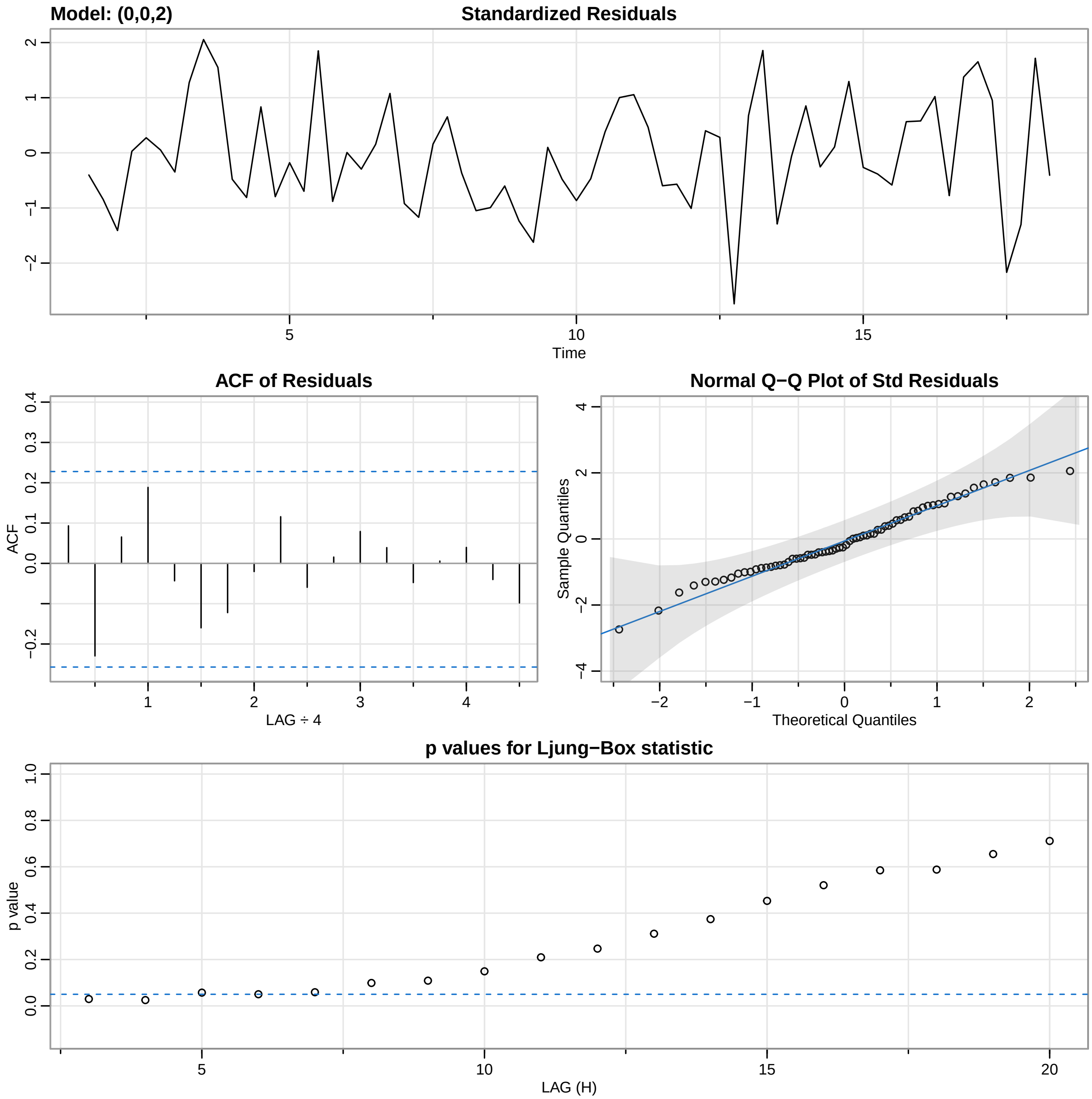
A seasonal time series, in addition to the trend and random components, also has a seasonal component. Decomposing a seasonal time series means separating the time series into these three components. In R we can use the decompose() function to estimate the three components of the time series.






```
## initial  value 2.112391
## iter   2 value 1.871910
## iter   3 value 1.813240
## iter   4 value 1.798554
```

```
## iter   5 value 1.782690
## iter   6 value 1.773187
## iter   7 value 1.767042
## iter   8 value 1.735763
## iter   9 value 1.660720
## iter  10 value 1.650986
## iter  11 value 1.644716
## iter  12 value 1.640025
## iter  13 value 1.638287
## iter  14 value 1.637806
## iter  15 value 1.637647
## iter  16 value 1.637543
## iter  17 value 1.637527
## iter  18 value 1.637526
## iter  18 value 1.637526
## iter  18 value 1.637526
## final  value 1.637526
## converged
## initial  value 1.586587
## iter   2 value 1.582694
## iter   3 value 1.564951
## iter   4 value 1.552802
## iter   5 value 1.549052
## iter   6 value 1.517650
## iter   7 value 1.511941
## iter   8 value 1.508632
## iter   9 value 1.508629
## iter  10 value 1.508622
## iter  11 value 1.508436
## iter  12 value 1.508424
## iter  12 value 1.508424
## iter  12 value 1.508424
## final  value 1.508424
## converged
```




```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,
##     optim.control = list(trace = trc, REPORT = 1, reltol = tol))
##
## Coefficients:
##           ma1     ma2    xmean
##       -1.7724  0.7724  -0.0436
## s.e.   0.1067  0.0995   0.0070
##
## sigma^2 estimated as 18.16:  log likelihood = -199.06,  aic = 406.12
##
```

```
## $degrees_of_freedom
## [1] 65
##
## $ttable
##       Estimate     SE  t.value p.value
## ma1    -1.7724 0.1067 -16.6036       0
## ma2     0.7724 0.0995   7.7650       0
## xmean  -0.0436 0.0070  -6.2654       0
##
## $AIC
## [1] 5.972372
##
## $AICc
## [1] 5.977887
##
## $BIC
## [1] 6.102932
```