Modelling and Predicting the Number of Airline Passengers

Assignment 3 – 02417 Time Series Analysis – Anders Hørsted (s082382)

In this report a model of the monthly number of airline passengers in the U.S. is build. The data set used to build the model is the actual number of passengers for every month between January 1995 and March 2002. To be able to give an estimate of the precision of the model, the data set is separated into two parts: A training set containing data for the period January 1995 to June 2001, and a test set containing data for the period July 2001 to March 2002. Throughout the modelling phase we work as if only the training set is available. The test set is used when the final model have been build, to compare the predictions of the final model, and the actual numbers in the test set. From now and until the section about measuring the model performance the training set is just referred to as "the data set" or "the data"

Data exploration

In this section the data set is introduced. First a plot of the data is created and shown in figure 1.

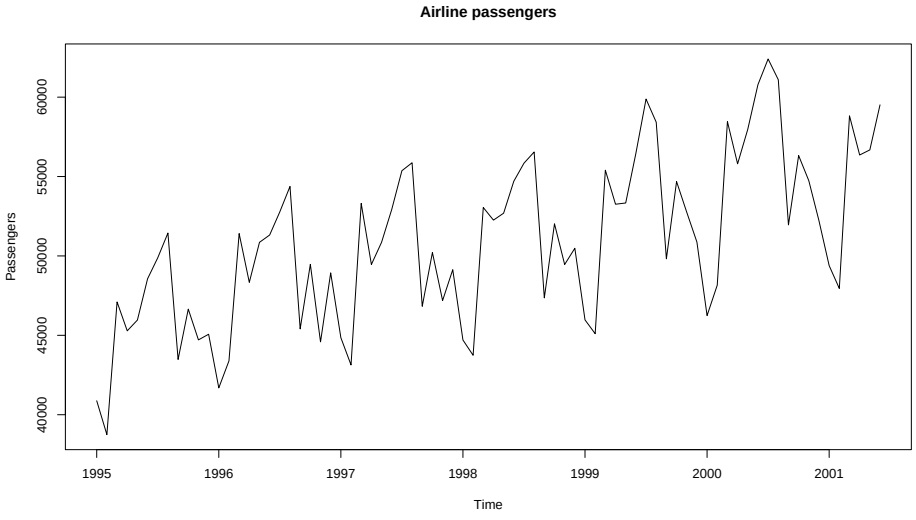


Figure 1: Plot of data set used for modelling

From the plot a general upward trend is recognized. This isn't surprising since the U.S. economy got stronger in the period and therefore more airline passengers should be

expected. Also a regular seasonal pattern can be seen which isn't surprising either. In the summer months e.g. we would expect more passengers than for the other months etc. The conclusion of these two observations is that the time series is non-stationary and this is something that should be coped with during the modelling phase. To support that the time series is non-stationary the estimated autocorrelation function (ACF) and partial autocorrelation function (PACF) are now plotted (see figure 2)

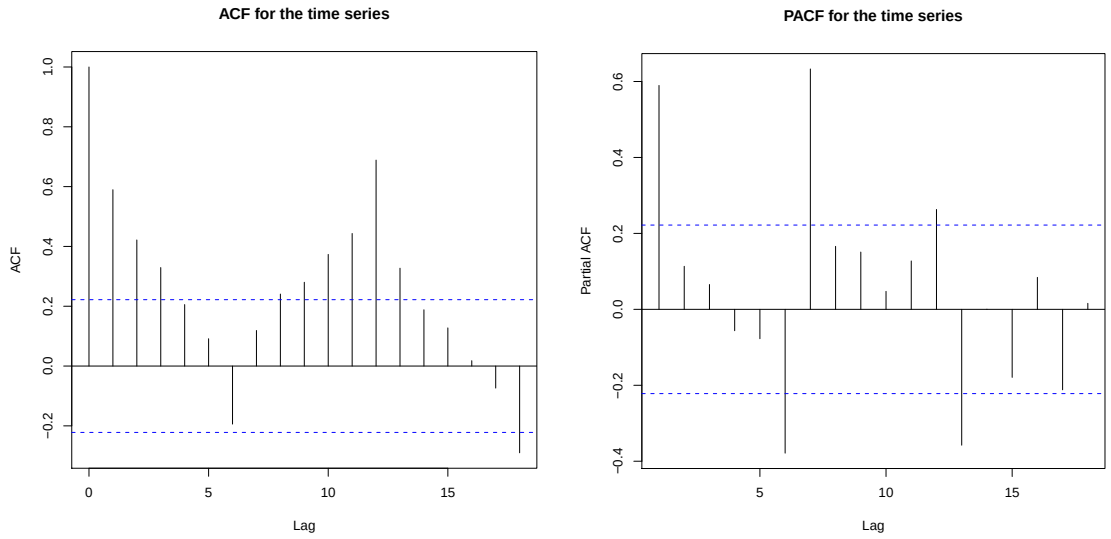


Figure 2: ACF and PACF for the original time series

Neither the ACF or the PACF are decreasing fast toward zero which further confirms that the time serie is non-stationary.

Building the model

It is time to start building the model. Since the time series is non-stationary a natural first step is to take the first order difference of the series. With Y_t the original series a new series is defined as

$$W_t = \nabla Y_t$$

The ACF and PACF are calculated for this new series and are shown in figure 3.

From the ACF of the differenced series it is seen that the estimated correlation for most lags are not significantly different from zero, except for the lags 6, 12, 18, 24. Also lag 6 and 18 are negative and 12 and 24 is positive. This confirms the 12-period seasonality that was observed in the plot of the original time series. A natural next step is therefore to do a seasonal differencing giving a new series as $Z_t = \nabla_{12} W_t = \nabla \nabla_{12} Y_t$

The ACF and PACF of the new series are shown in figure 4.

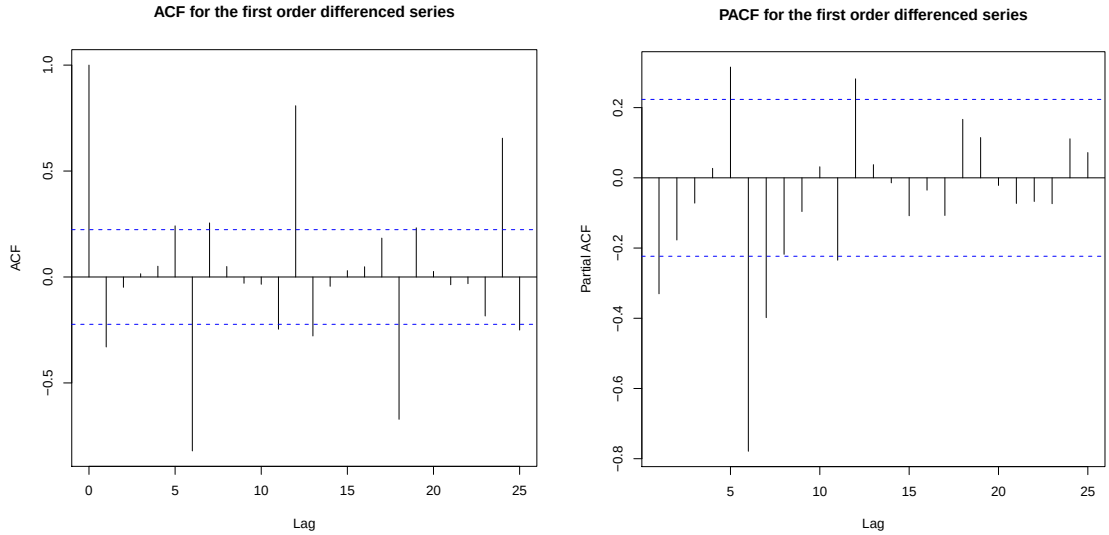


Figure 3: ACF and PACF for the differenced series ∇Y_t

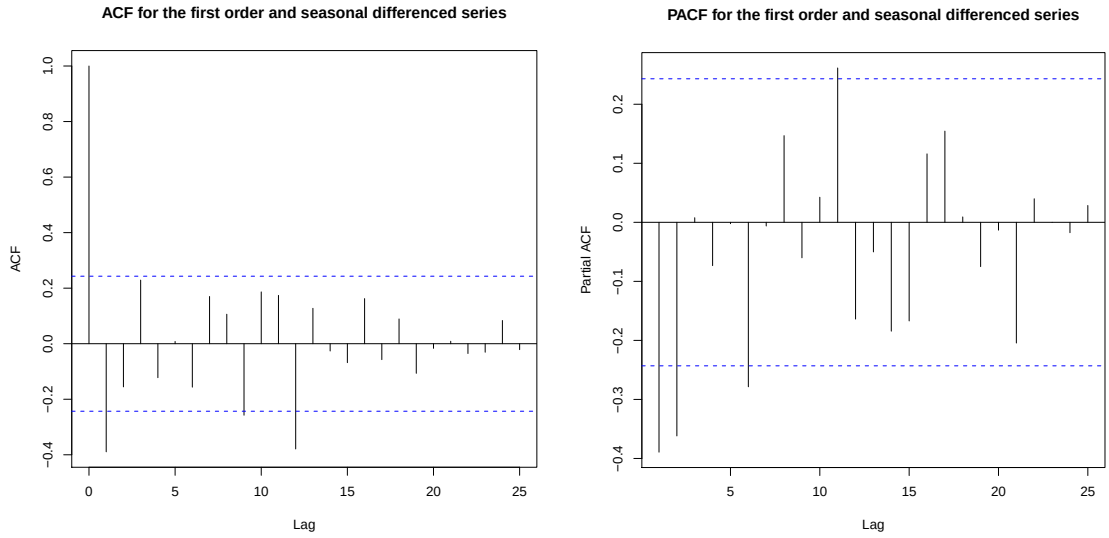


Figure 4: ACF and PACF for the series $\nabla\nabla_{12}Y_t$

It is seen that most lags of the ACF is zero, except for lag 1 and 12. For the PACF the lags 1, 2, 6 and 11 are different from zero. Based on this it looks as if it is possible to model the original series Y_t as a multiplicative $(p, 1, q) \times (P, 1, Q)_{12}$ seasonal model. Since only lag 1 and 12 are different from 0 in the ACF a model with $q = 1$ and $Q = 1$, could be tried. Ignoring lag 6 and 11 in the PACF for now $p = 2$ and $P = 0$ is attempted.

The $(2,1,1) \times (0,1,1)_{12}$ seasonal model

Using the `arima` function in R a $(2,1,1) \times (0,1,1)_{12}$ seasonal model is now fitted to the original time series. The resulting model is given by

$$(1 + 0.20B + 0.27B^2)Z_t = (1 - 0.40B)(1 - 0.46B^{12})\varepsilon_t \quad \Leftrightarrow$$
$$(1 + 0.20B + 0.27B^2)(1 - B)(1 - B^{12})Y_t = (1 - 0.40B)(1 - 0.46B^{12})\varepsilon_t$$

To check the model the residuals are checked to see if they resemble white noise as they should for an adequate model. First the residuals are plotted and a QQ-plot of the residuals are also created. Both are shown in figure 5.

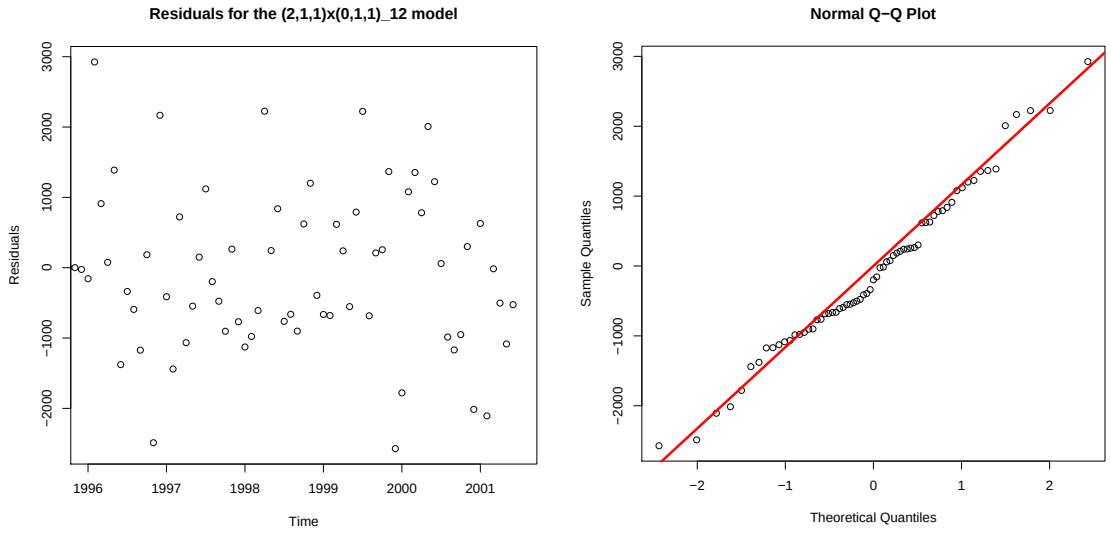


Figure 5: Normal plot and QQ-plot of the residuals for the fitted $(2,1,1) \times (0,1,1)_{12}$ seasonal model

From both plots it looks as if the residuals are pretty close to white noise. Doing a sign test on the residuals gives a p-value of 0.27 which also supports that the residuals are white noise. The ACF for the residuals is plotted and shown in figure 6

The ACF also supports that the residuals are white noise, since only lag 0 is different from 0. By all means it looks as if the chosen model is able to describe the original time series and so it only remains to check whether a simpler model exists that explains the data just as well. From the estimated parameters of the model along with the standard error of the estimates (see figure 7) it looks as if at least some of the parameters are redundant and it is expected that a simpler model exists. The primary candidate for changes is the normal AR(2)-part of the model which could possible be changed to 0 or 1. Therefore a $(0,1,1) \times (0,1,1)_{12}$ model is now tested.

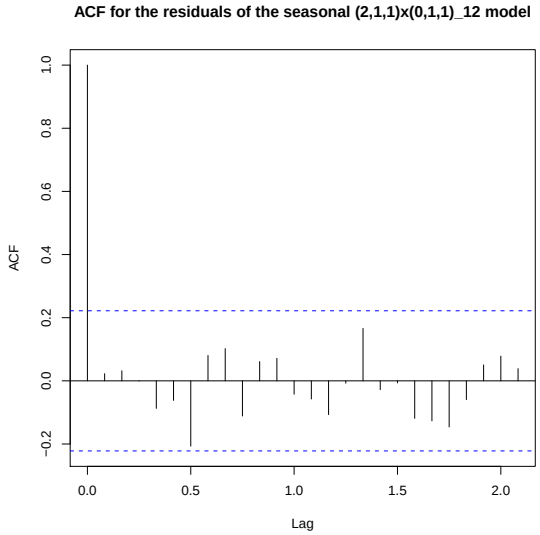


Figure 6: ACF for the $(2,1,1) \times (0,1,1)_{12}$ seasonal model

```
Coefficients:
          ar1      ar2      ma1     sma1
      -0.2000  -0.2726  -0.3961  -0.4567
s.e.   0.3613   0.2091   0.3810   0.1250
```

Figure 7: Estimated coefficients and standard errors of the estimates, for the $(2,1,1) \times (0,1,1)_{12}$ seasonal model

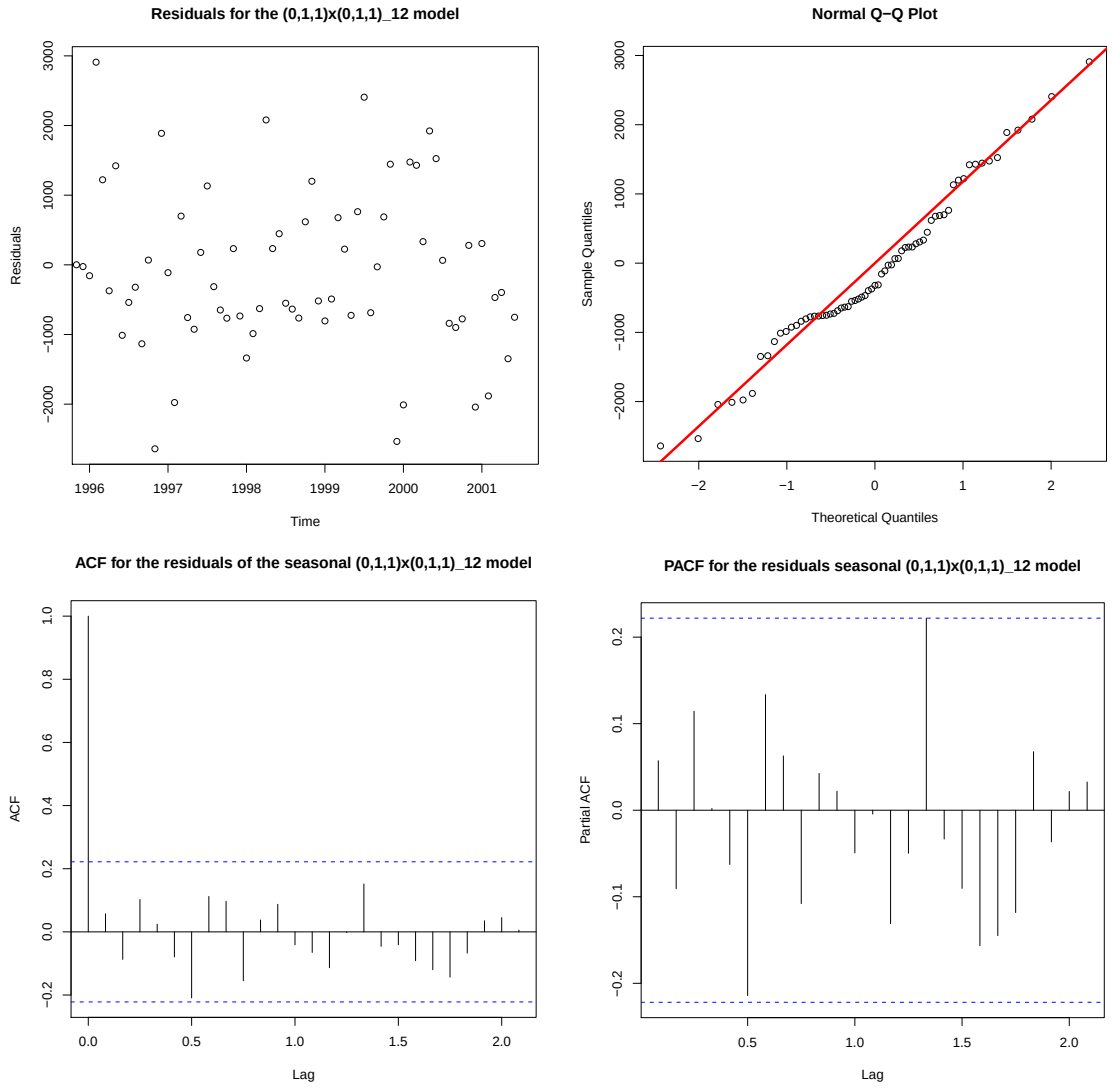
The $(0,1,1) \times (0,1,1)_{12}$ seasonal model

Parameters for the $(0,1,1) \times (0,1,1)_{12}$ seasonal model is now estimated from the original time series which gives

$$\begin{aligned} Z_t &= (1 - 0.66B)(1 - 0.45B^{12}) \quad \Leftrightarrow \\ (1-B)(1-B^{12})Y_t &= (1 - 0.66B)(1 - 0.45B^{12}) \end{aligned} \tag{1}$$

By doing all the same model checks as for the previous model we obtain the results shown in figure 8.

From the plots it seems as if the simpler $(0,1,1) \times (0,1,1)_{12}$ model is also able to explain the model and from the coefficients and their standard errors it looks as if this is as simple as the model can be. To further support that the simpler model is to be preferred the AIC is seen (see appendices A.4 and A.6) to be lower for the simpler model. The conclusion is that the $(0,1,1) \times (0,1,1)_{12}$ model with coefficients as in equation (1) is accepted as the best model to describe the data. The final test for the chosen model is to predict 9 months ahead in time and then compare with the actual values.




```
Coefficients:
          ma1     sma1
      -0.6627  -0.4531
s.e.   0.1033   0.1221
```

Figure 8: Diagnostic plots for the $(0,1,1) \times (0,1,1)_{12}$ seasonal model

Prediction with the $(0,1,1) \times (0,1,1)_{12}$ model

Note that in this section the data that in the previous section was just called the data set is now referred to as the training set. The data set with airline passenger numbers for July 2001 to March 2002 is called the test set.

Predictions for the months July 2001 to March 2002 are now made using the `predict.Arima` function in R. The method is set to Maximum Likelihood. The predictions are plotted along with the training set data and is shown in figure 9.

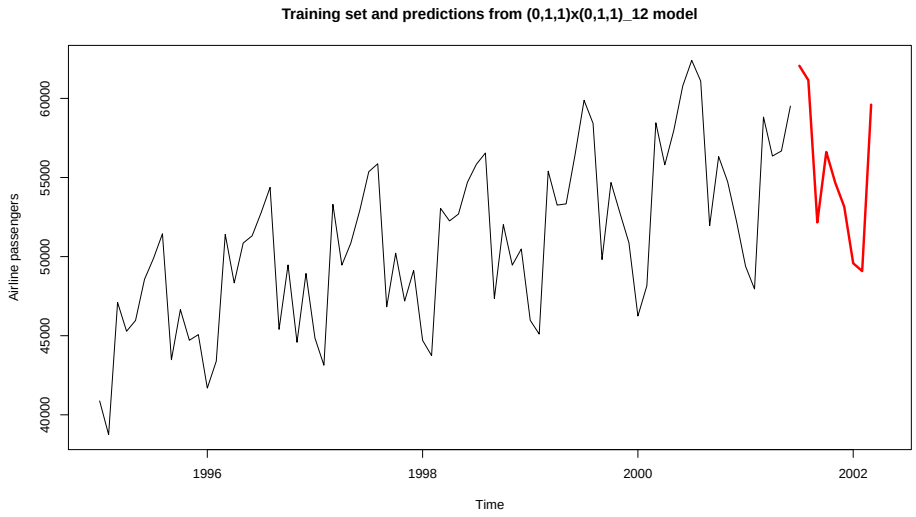


Figure 9: The training data set and model predictions combined

From the figure it looks as if the model is capturing the overall pattern in the series. Now it is tested how well the model fits the real numbers in the test set. The predictions along with the real numbers from the test set is therefore plotted in figure 10.

Except for the two first months the predictions are seen to be quite far from the true values. This is a bit surprising since it was just seen that the predictions captured the overall pattern very well. Comparing the predictions and the real data it is seen that the error is very large in the autumn 2001 and then becomes smaller in 2002. It seems as if the real data is deviating from its regular pattern in the autumn 2001.

This is confirmed in figure 11 where it is seen that a dramatic decrease in the number of airline passengers happened in september 2001. This can of cause be explained by the terrorist attacks on the World Trade Center on September 11th. People were probably not that keen on flying in the days/weeks following 9/11. Although the 9/11 no doubt had great impact on the number of airline passengers, it is interesting to note that already around december 2001 the predictions captures the overall pattern in the real data very well. The mean level is just 6000-8000 too high for the predictions. The impact of 9/11 could probably be captured in an intervention model as described in [1] but this doesn't change the fact that the intervention model is useable only for the expected events. A statistician working for an airline company in June 2001 would probably have build a

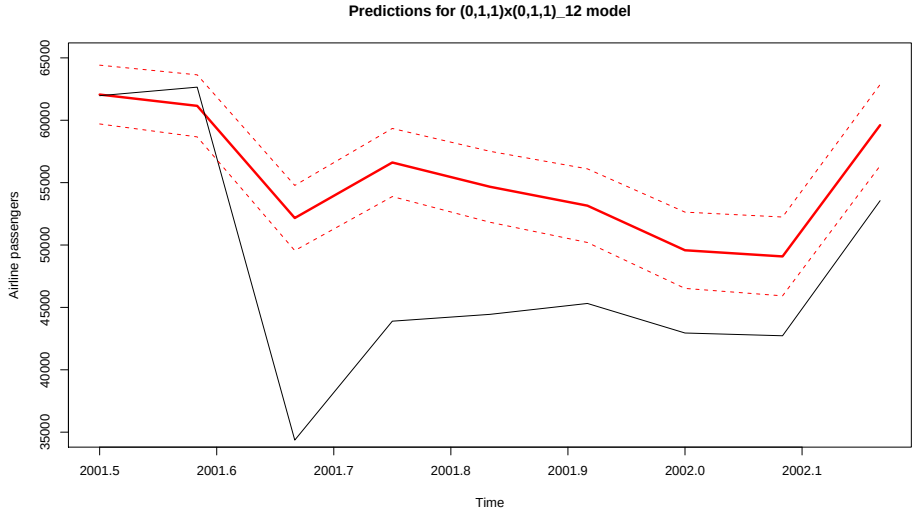


Figure 10: The predictions along with the real numbers from the test set

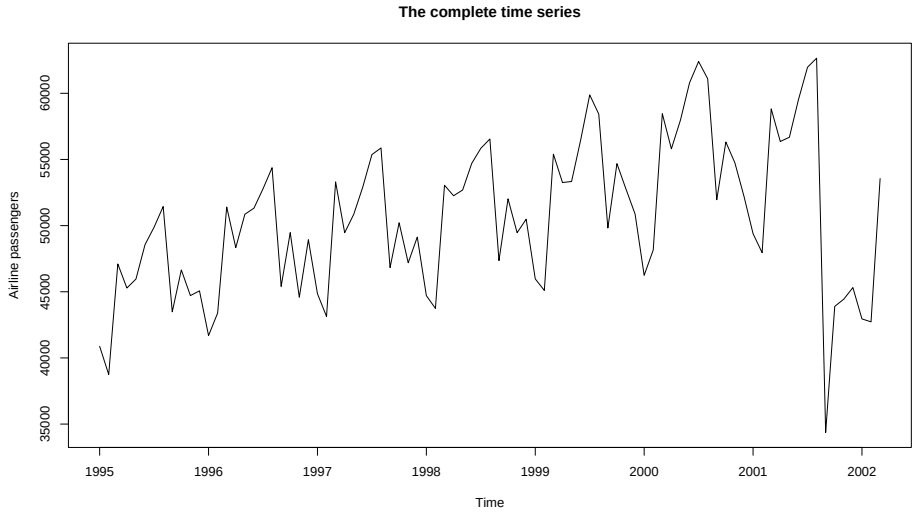


Figure 11: The full data set. (Both training set and test set)

model comparable with the model in this report and would have made predictions just as wrong. For an easy read, popular science discussion about the extreme events and the impact for especially business analysis read [2]

A Appendices

All R code used for this assignment is included here. All source code incl. latex code for this report can be found at `https://github.com/alphabits/dtu-fall-2011/tree/master/02417/assignment-3`

A.1 Load data script

```
dat = read.table("../data/assignment3data.dat")
dat = dat[,1]
dat.ts = ts(dat, start=1995, freq=12)
train = dat[1:78]
train.ts = ts(train, start=1995, freq=12)
test = dat[-(1:78)]
test.ts = ts(test, start=c(2001, 7), freq=12)
```

A.2 Helper functions

```
plot.and.save = function(filename, width, height, plotfunction, ...) {
    save.the.plot = exists('SAVEPLOTS') && SAVEPLOTS
    if (save.the.plot) {pdf(sprintf('../plots/%s', filename), width, height)}
    plotfunction(...)
    if (save.the.plot) {dev.off()}
}

diagnose = function(ts) {
    par(mfrow=c(2,1))
    acf(ts)
    pacf(ts)
}

sign.test = function(res) {
    n = length(res)
    binom.test(sum(1*(res[2:n]*res[1:(n-1)] < 0)), n-1)
}

save.sign.test = function(res, filename) {
    sink(sprintf('../tables/%s', filename))
    print(sign.test(res))
    sink()
}

qq = function(model) {
    plot.qq.res(residuals(model), model$sigma2)
}

plot.qq.res = function(res, sigma) {
    qqnorm(res)
    lines((-4):4,((-4):4)*sqrt(sigma), type="l", lwd=3, col='red')
}

save.model.summary = function(model, filename) {
    sink(sprintf('../tables/%s', filename))
    print(model)
    sink()
```

```
}
```

A.3 Main script

```
source('loaddata.R')
source('functions.R')

SAVEPLOTS = TRUE

############################
### Differencing the data ###
############################

plot.and.save('trainingset.pdf', 12, 7,
              plot, train.ts, main='Airline passengers', xlab='Time',
              ylab='Passengers')
plot.and.save('acf-trainingset.pdf', 7, 7,
              acf, train, main='ACF for the time series')
plot.and.save('pacf-trainingset.pdf', 7, 7,
              pacf, train, main='PACF for the time series')

train.d = diff(train)
plot.and.save('acf-onediff.pdf', 7, 7,
              acf, train.d, lag.max=25,
              main='ACF for the first order differenced series')
plot.and.save('pacf-onediff.pdf', 7, 7,
              pacf, train.d, lag.max=25,
              main='PACF for the first order differenced series')

train.d2 = diff(train.d, lag=12)
plot.and.save('acf-seasondiff.pdf', 7, 7,
              acf, train.d2, lag.max=25,
              main='ACF for the first order and seasonal differenced series')
plot.and.save('pacf-seasondiff.pdf', 7, 7,
              pacf, train.d2, lag.max=25,
              main='PACF for the first order and seasonal differenced series')
plot(train.d2, type="l")


###############
### Model 1 ###
###############

m1 = arima(train.ts, order=c(2,1,1),
           seasonal=list(order=c(0,1,1), period=12),
           method="ML")
save.model.summary(m1, 'model1.txt')
m1.r = residuals(m1)
m1.r.trim = m1.r[12:78]

plot.and.save('acf-m1.pdf', 7, 7,
              acf, m1.r, lag.max=25,
              main='ACF for the residuals of the seasonal (2,1,1)x(0,1,1)_12 model')
plot.and.save('pacf-m1.pdf', 7, 7,
              pacf, m1.r, lag.max=25,
              main='PACF for the residuals seasonal (2,1,1)x(0,1,1)_12 model')

plot.and.save('residuals-m1.pdf', 7, 7,
              plot, m1.r, type="p", xlim=c(1996,2001.5), ylab='Residuals',
```

```
              main='Residuals for the (2,1,1)x(0,1,1)_12 model')

plot.and.save('qq-residuals-m1.pdf', 7, 7,
              plot.qq.res, m1.r.trim, m1$sigma2)

save.sign.test(m1.r.trim, 'signtest-m1.txt')

hist(m1.r.trim, probability=T, col='blue')
# The Ljung-Box test is a refinement of the Box-Pierce test
# that is described under the heading "Portmanteau lack-of-fit test"
# on page 175 in course text book. See also:
# http://en.wikipedia.org/wiki/Ljung-Box_test
tsdiag(m1)

sink('../tables/aic-scores.txt')
for (i in 0:5) {
    for (j in 1:3) {
        mi = arima(train.ts, order=c(i,1,j),
               seasonal=list(order=c(0,1,1), period=12),
               method="ML")
        print(sprintf('(%d,1,%d): %.02f', i, j, mi$aic))
    }
}
sink()

###############
### Model 2 ###
###############

m2 = arima(train.ts, order=c(0,1,1), seasonal=list(order=c(0,1,1), period=12),
           include.mean=T, method="ML")
save.model.summary(m2, 'model2.txt')
m2.r = residuals(m2)
m2.r.trim = m2.r[12:78]
m2.p = predict(m2, n.ahead=9)

plot.and.save('acf-m2.pdf', 7, 7,
              acf, m2.r, lag.max=25,
              main='ACF for the residuals of the seasonal (0,1,1)x(0,1,1)_12 model')
plot.and.save('pacf-m2.pdf', 7, 7,
              pacf, m2.r, lag.max=25,
              main='PACF for the residuals seasonal (0,1,1)x(0,1,1)_12 model')

plot.and.save('residuals-m2.pdf', 7, 7,
              plot, m2.r, type="p", xlim=c(1996,2001.5), ylab='Residuals',
              main='Residuals for the (0,1,1)x(0,1,1)_12 model')

plot.and.save('qq-residuals-m2.pdf', 7, 7,
              plot.qq.res, m2.r.trim, m2$sigma2)

save.sign.test(m2.r.trim, 'signtest-m2.txt')

plot.predictions = function(pred, ...) {
    p = pred$pred
    se = pred$se
    plot(p, type="l", ylim=c(35000, 65000), col="red",
         ylab='Airline passengers', lwd=3, ...)
    lines(test.ts, type="l")
    lines(p + 2*se, lty=2, col="red")
    lines(p - 2*se, lty=2, col="red")
}
```

```
plot.with.trainingset = function() {
    plot(train.ts, ylab='Airline passengers', xlim=c(1995, 2002.25),
         main='Training set and predictions from (0,1,1)x(0,1,1)_12 model')
    lines(m2.p$pred, col="red", lwd=3)
}

plot.and.save('test-and-prediction.pdf', 12, 7,
              plot.predictions, m2.p,
              main='Predictions for (0,1,1)x(0,1,1)_12 model')

plot.and.save('training-and-prediction.pdf', 12, 7,
              plot.with.trainingset)

plot.and.save('full-data-set.pdf', 12, 7,
              plot, dat.ts, main='The complete time series',
              ylab='Airline passengers')
```

A.4 Summary for $(2,1,1) \times (0,1,1)_{12}$ model

```
Call:
arima(x = train.ts, order = c(2, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 12), method = "ML")

Coefficients:
          ar1      ar2      ma1     sma1
      -0.2000  -0.2726  -0.3961  -0.4567
s.e.   0.3613   0.2091   0.3810   0.1250

sigma^2 estimated as 1351640:  log likelihood = -552.72,  aic = 1115.44
```

A.5 Sign test for $(2,1,1) \times (0,1,1)_{12}$ model

```
    Exact binomial test

data:  sum(1 * (res[2:n] * res[1:(n - 1)] < 0)) and n - 1
number of successes = 28, number of trials = 66, p-value = 0.2678
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.3034019 0.5521060
sample estimates:
probability of success
             0.4242424
```

A.6 Summary for $(0,1,1) \times (0,1,1)_{12}$ model

```
Call:
arima(x = train.ts, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1),
    period = 12), include.mean = T, method = "ML")
```

```
Coefficients:
          ma1     sma1
      -0.6627  -0.4531
s.e.   0.1033   0.1221

sigma^2 estimated as 1388435:  log likelihood = -553.57,  aic = 1113.14
```

A.7 Sign test for $(0,1,1) \times (0,1,1)_{12}$ model

```
    Exact binomial test

data:  sum(1 * (res[2:n] * res[1:(n - 1)] < 0)) and n - 1
number of successes = 30, number of trials = 66, p-value = 0.5386
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.3314394 0.5818632
sample estimates:
probability of success
             0.4545455
```

References

[1] Henrik Madsen, *Time Series Analysis*. Chapman & Hall/CRC, 1st Edition, 2008.

[2] Nassim Nicholas Taleb, *The Black Swan*. Random House Trade Paperbacks 2nd Edition, 2010.