

Econometrics II - Assignment 5

Think of something new

13 Mar 2022

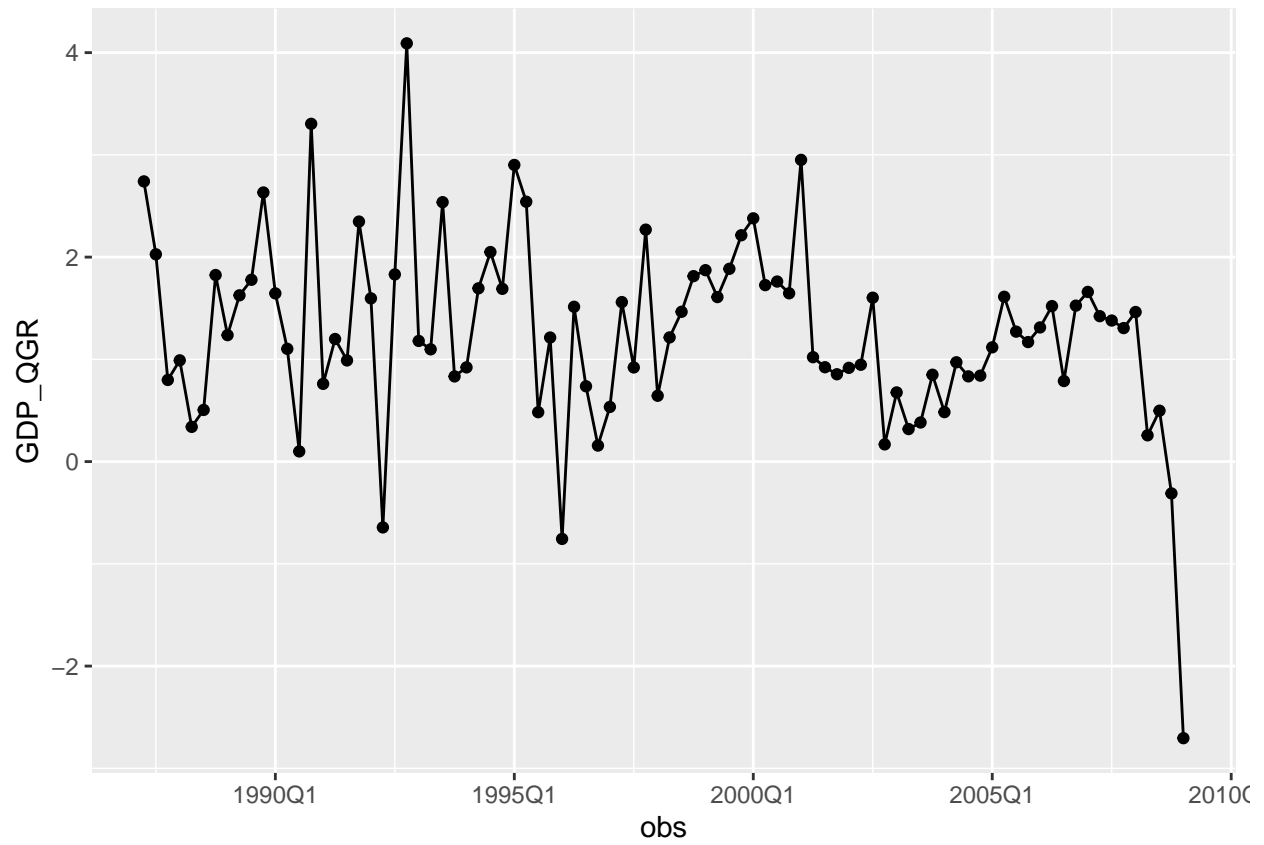
Question 1

```
# Load data
data <- read.csv("data_assign_p1.csv")
```

1.) Plot the sample of Dutch GDP quarterly growth rates that you have at your disposal. Report the 12-period sample ACF and PACF functions and comment on their shape. What does the sample ACF tells you about the dynamic properties of GDP quarterly growth rates?

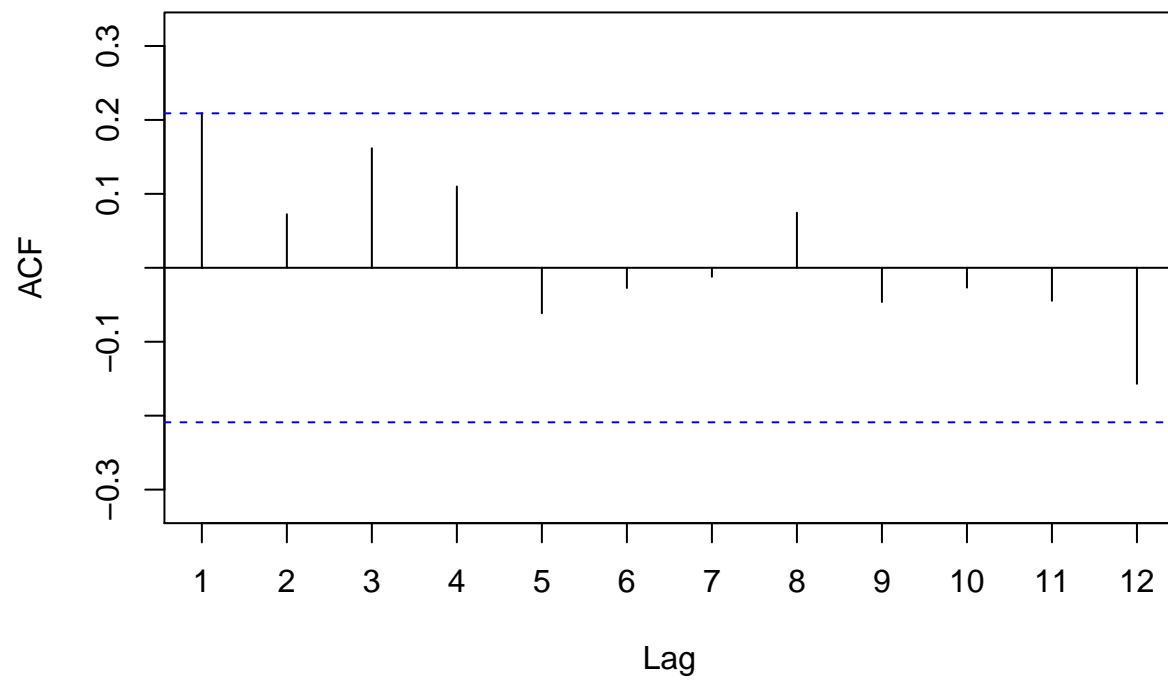
```
fmt <- "%YQ%q"
data$obs <- as.yearqtr(data$obs, format = fmt)
```

```
ggplot(data, aes(obs, GDP_QGR)) +
  geom_point() +
  geom_line() +
  scale_x_yearqtr(format = fmt)
```

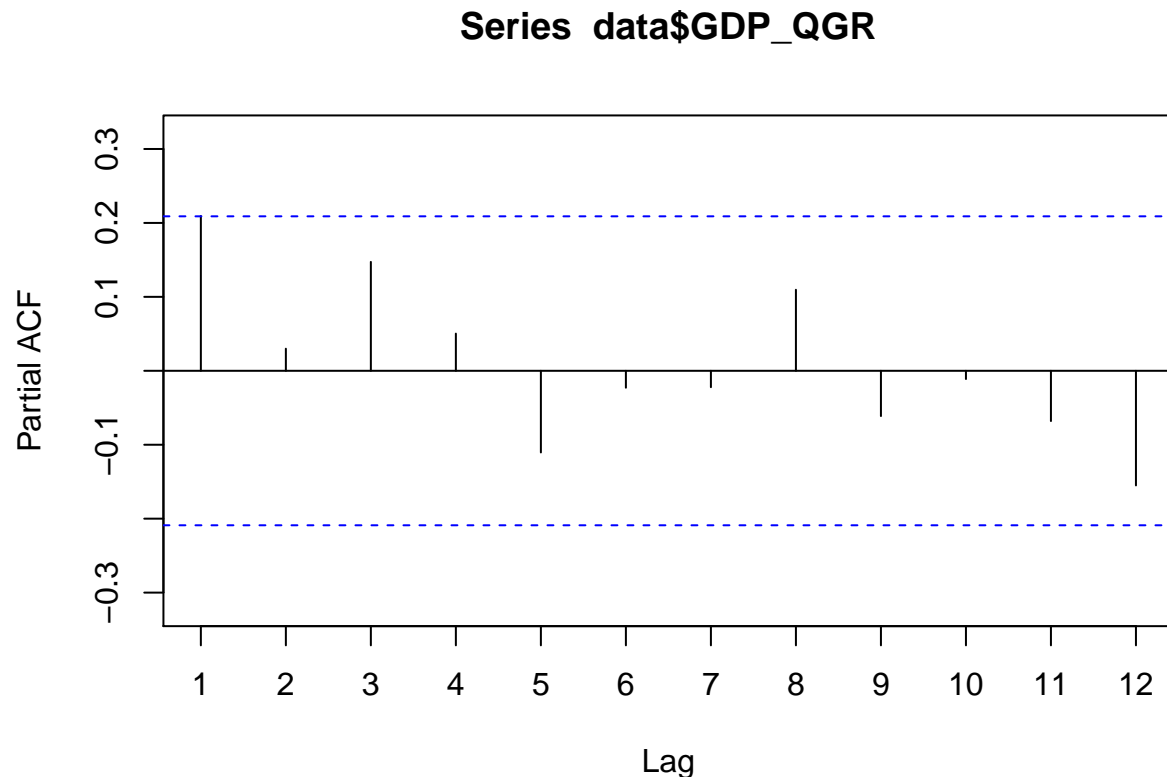


```
Acf(  
  data$GDP_QGR,  
  lag.max = 12  
)
```

Series data\$GDP_QGR



```
Pacf(  
  data$GDP_QGR,  
  lag.max = 12  
)
```



The ACF is the autocorrelation function and measures the autocorrelation between X_t and $X_{t-h} \forall h$, in our case for $h = \{1, 2, \dots, 12\}$. The autocorrelation is an indicator for memory: the higher the correlation, the higher the time dependence. Hence, the ACF also is a good first indicator for the selection of the lags of our time series model (corresponding to the Box-Jenkins approach). The partial autocorrelation function (PACF) goes one step further, controlling for other lags. Looking at our ACF and PACF graphs, we can see that the first lag seem to have a significant effect on our GDP while higher lags have a rather small autocorrelation with X_t .

2.) Estimate an AR(p) model for the same time-series. Please use the general-to-specific modeling approach by starting with a total $p = 4$ lags and removing insignificant lags sequentially. Report the final estimated AR(p) model, working at a 5% significance level. Comment on the estimated coefficients. What do these coefficients tell you about the dynamic properties of the GDP quarterly growth rate?

```
ar4 <- arima(data$GDP_QGR, order = c(4,0,0))
coeftest(ar4)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value Pr(>|z|)
## ar1      0.241960   0.117749   2.0549   0.03989 *
## ar2      0.029955   0.119538   0.2506   0.80213
## ar3      0.189023   0.119463   1.5823   0.11359
## ar4      0.086413   0.120151   0.7192   0.47201
## intercept 1.214421   0.203636   5.9637 2.466e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
ar3 <- arima(data$GDP_QGR, order = c(3,0,0))
coeftest(ar3)
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error z value Pr(>|z|)
## ar1         0.255536   0.116737  2.1890  0.02860 *
## ar2         0.029868   0.120377  0.2481  0.80404
## ar3         0.200007   0.119049  1.6800  0.09295 .
## intercept 1.228330   0.179893  6.8281 8.605e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
ar2 <- arima(data$GDP_QGR, order = c(2,0,0))
coeftest(ar2)
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error z value Pr(>|z|)
## ar1         0.263192   0.118876  2.2140  0.02683 *
## ar2         0.057902   0.121458  0.4767  0.63356
## intercept 1.249173   0.139580  8.9495 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
ar1 <- arima(data$GDP_QGR, order = c(1,0,0))
coeftest(ar1)
```

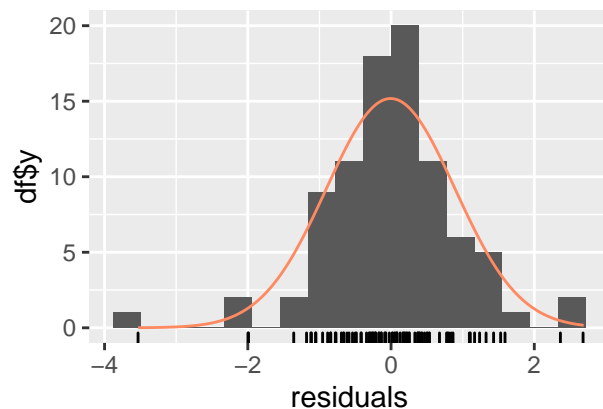
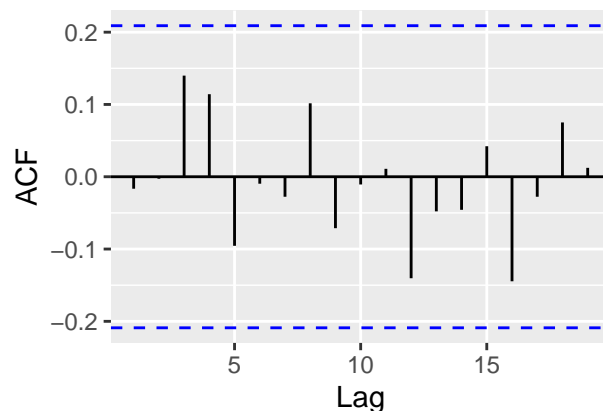
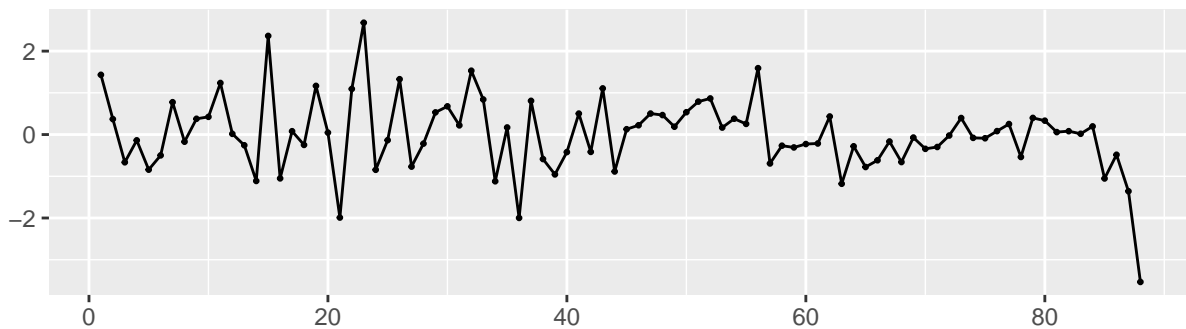
```
##
## z test of coefficients:
##
##           Estimate Std. Error z value Pr(>|z|)
## ar1         0.27191    0.11775  2.3092  0.02093 *
## intercept 1.25269    0.13036  9.6091 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As only the estimator of the first lag has a significant effect (which can be observed in all model specification), our final model specification is an AR(1) model. The estimator is around 0.27191. Hence, an increase in the previous period's GDP growth by 1%, increases the next period's GDP growth by approximately 0.22%.

3.) Check the regression residuals of the estimated AR(p) model for autocorrelation by plotting the estimated residual ACF function. Does the model seem well specified?

```
checkresiduals(ar1)
```

Residuals from ARIMA(1,0,0) with non-zero mean



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,0,0) with non-zero mean
## Q* = 5.5739, df = 8, p-value = 0.6948
##
## Model df: 2.   Total lags used: 10
```

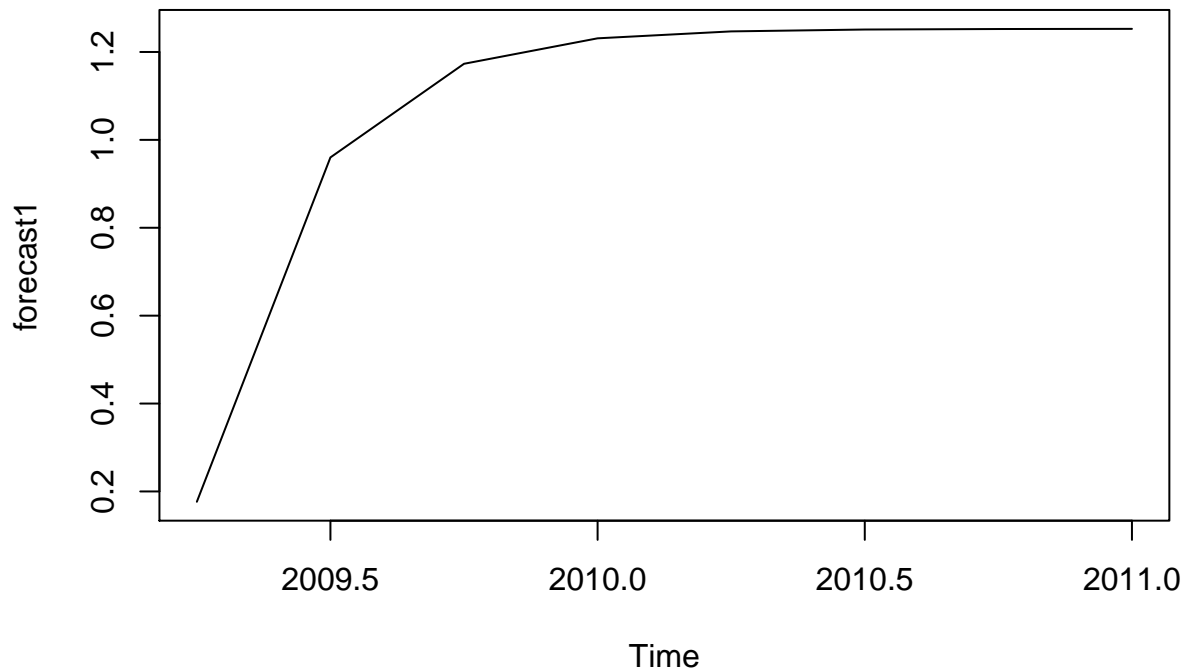
There is no autocorrelation between the regression residuals of the estimated AR(1) model. Therefore, the model seems to be well specified.

4.) Make use of your estimated AR model to produce a 2-year (8 quarters) forecast for the Dutch GDP quarterly growth rate that spans until the first quarter of 2011. Report the values you obtained and explain how you derived them.

```
seriesdata <- ts(data$GDP_QGR, start= c(1987, 2), end = c(2009, 1), frequency = 4)
forecast1 <- ts(predict(ar1, n.ahead=8)$pred, start = c(2009,2), frequency = 4)
forecast1
```

```
##           Qtr1      Qtr2      Qtr3      Qtr4
## 2009      0.1765306 0.9600700 1.1731221
## 2010 1.2310530 1.2468050 1.2510881 1.2522527
## 2011 1.2525694
```

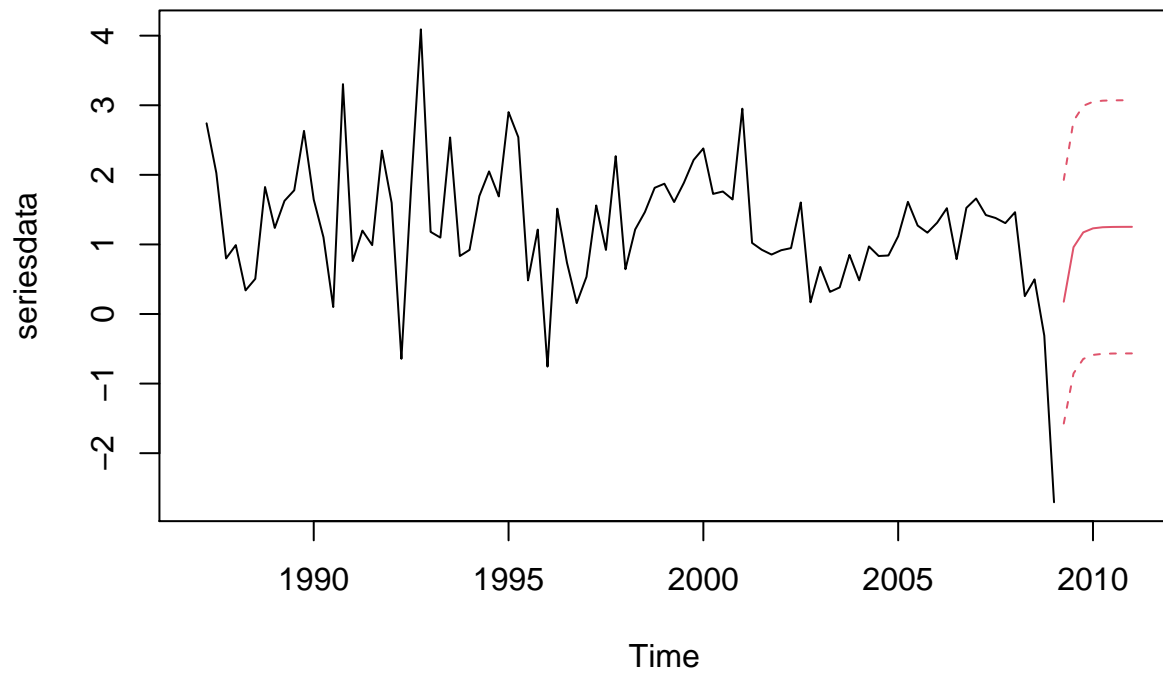
```
ts.plot(forecast1, type="l")
```



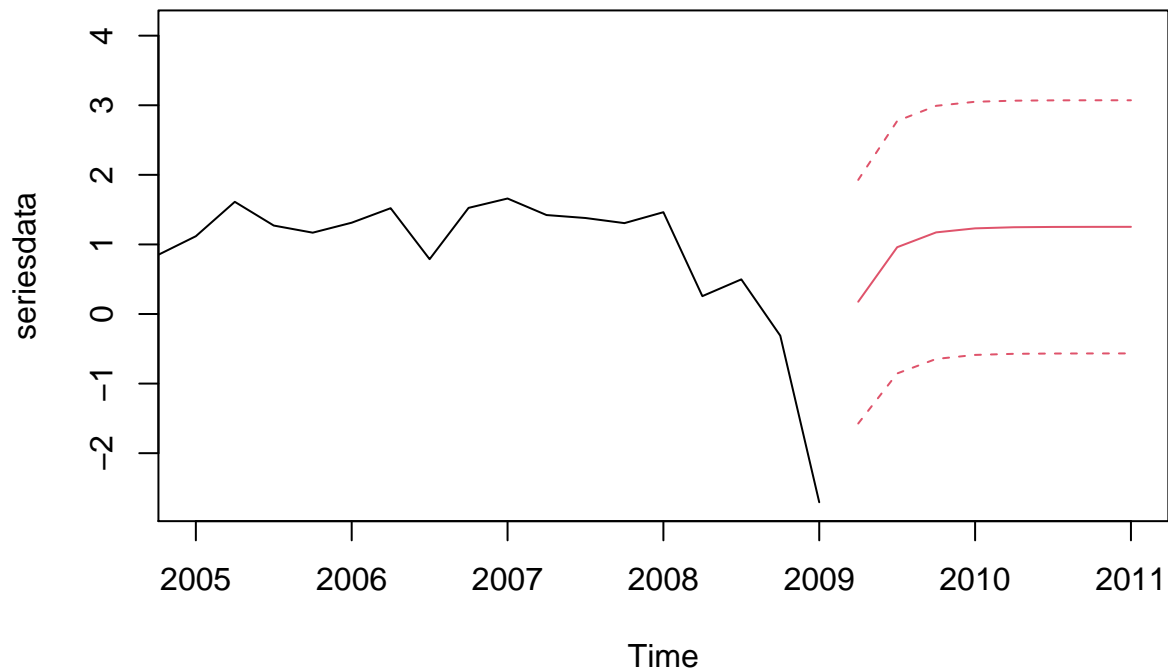
We used the function `predict()` to forecast the next 8 quarters based on the AR(1) model. It gives us a time series of predictions, using finite-history prediction (<https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/predict.Arima>). The algorithm is based on the Kalman filter (https://www.jstor.org/stable/pdf/2347987.pdf?casa_token=YM8e7AtkQeUAAAAA:iKQiv7RxaEzEBcvzWiMcCKERAXdYmC8XT1hnKuhGYsgsr53gT48e2qJex4rkupEafCimogVYWp1sLz9KmYjUiH-nVpbYCgoiOfnbVC-k4UypJenakvI).

5.) Suppose that the innovations in your AR model are iid Gaussian. Produce 95% confidence intervals for your 2-year forecast. Furthermore, comment on the following statement issued by government officials: “Given the available GDP data, we believe that there is a low probability that the Dutch GDP growth rate will remain negative in the second quarter of 2009.”

```
ts.plot(seriesdata, xlim = c(1987, 2011))
se1 <- ts(predict(ar1, n.ahead=8)$se, start = c(2009,2), frequency = 4)
points(forecast1, type="l", col=2)
points(forecast1 - 1.96*se1, type="l", col=2, lty=2)
points(forecast1 + 1.96*se1, type="l", col=2, lty=2)
```



```
ts.plot(seriesdata, xlim = c(2005, 2011))
sel <- ts(predict(ar1, n.ahead=8)$se, start = c(2009,2), frequency = 4)
points(forecast1, type="l", col=2)
points(forecast1 - 1.96*sel, type="l", col=2, lty=2)
points(forecast1 + 1.96*sel, type="l", col=2, lty=2)
```

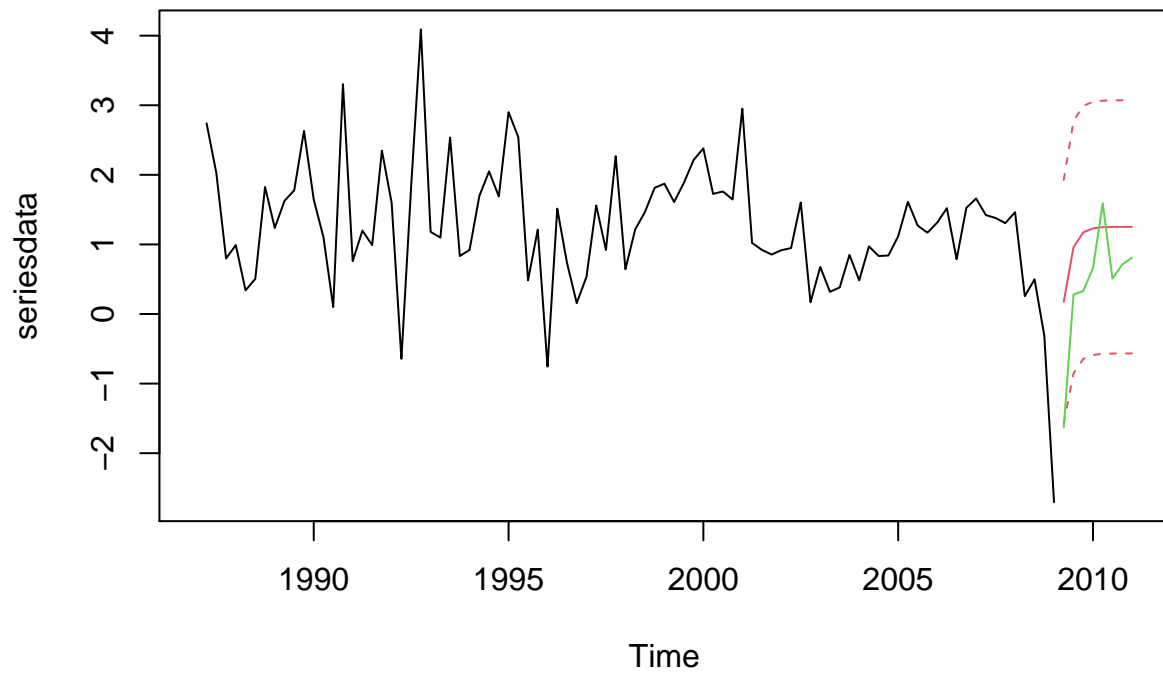
```
t_stat = (forecast1[1]-0)/se1[1]
n = length(data$obs)
pt(t_stat, n-2, lower.tail = FALSE)
```

```
## [1] 0.4218944
```

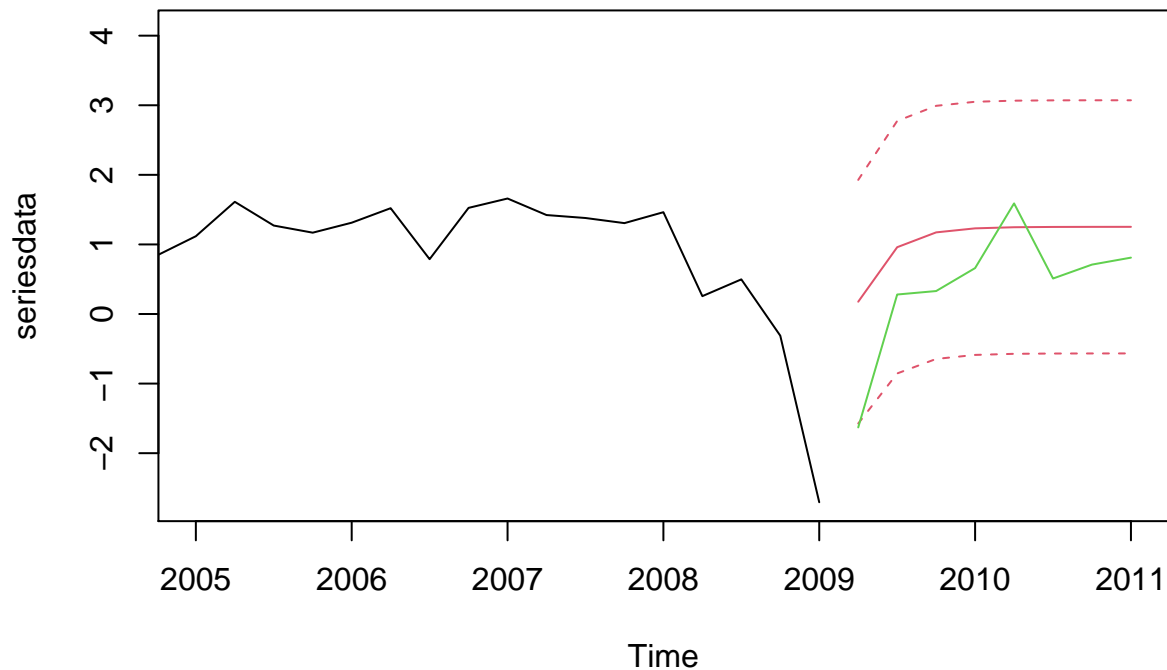
6.) Do you find the assumption of iid Gaussian innovations reasonable? How does this affect your answer to the previous question?

7.) Suppose that 2 years have passed since you delivered your forecasts to the government, in the first quarter of 2009. Compare your point forecasts and confidence bounds with the following actual observed values for the 12 quarters from 2009q2 to 2011q1. Please comment on the accuracy of your forecasts.

```
ts.plot(seriesdata, xlim = c(1987, 2011))
re <- ts(c(-1.63, 0.28, 0.33, 0.66, 1.59, 0.51, 0.71, 0.81), start = c(2009, 2), frequency = 4)
points(forecast1, type="l", col=2)
points(forecast1 - 1.96*se1, type="l", col=2, lty=2)
points(forecast1 + 1.96*se1, type="l", col=2, lty=2)
points(re, type="l", col=3)
```

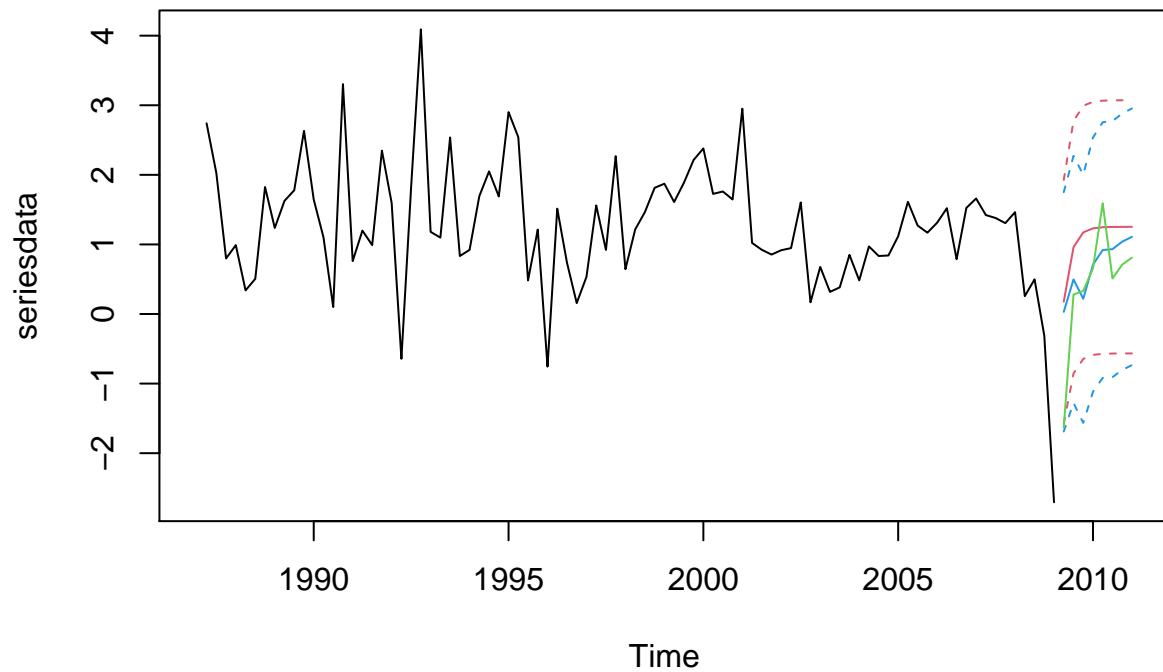


```
ts.plot(seriesdata, xlim = c(2005, 2011))
re <- ts(c(-1.63, 0.28, 0.33, 0.66, 1.59, 0.51, 0.71, 0.81), start = c(2009, 2), frequency = 4)
points(forecast1, type="l", col=2)
points(forecast1 - 1.96*se1, type="l", col=2, lty=2)
points(forecast1 + 1.96*se1, type="l", col=2, lty=2)
points(re, type="l", col=3)
```

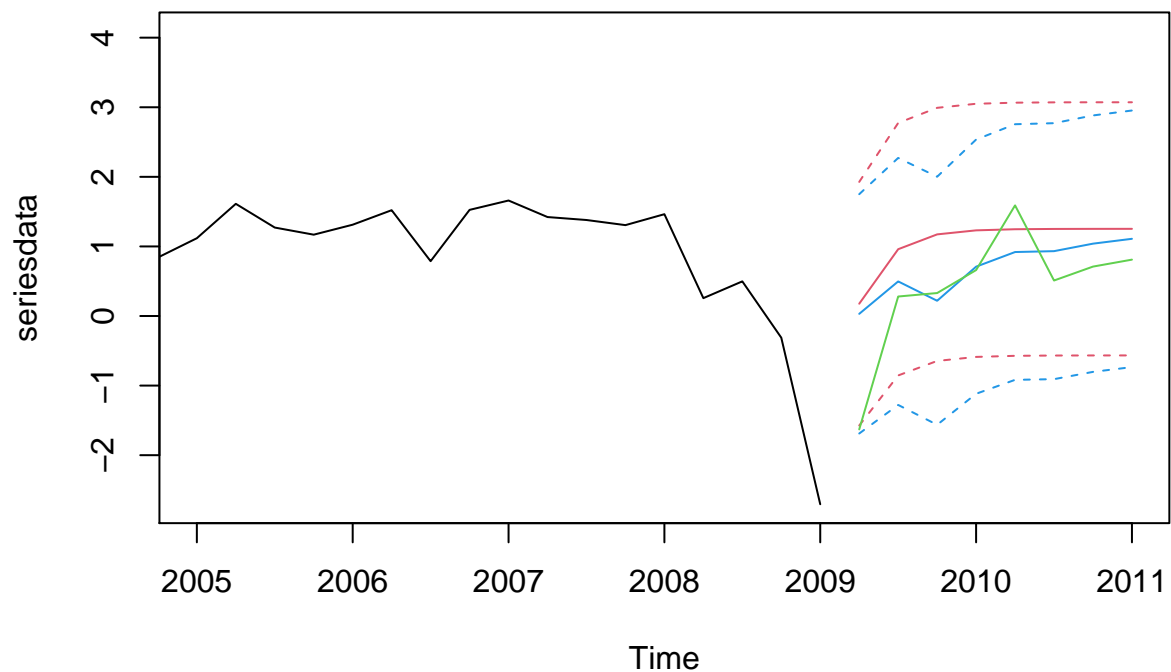


8.) Repeat question 2 above, but this time using a 10% significance level for the general-to-specific modeling approach. Use the newly estimated model to produce a 2-year (8 quarters) point forecast of the Dutch GDP quarterly growth rate. Comment on the accuracy of the forecast generated by the newly estimated AR model. Is it better than the model you estimated before?

```
ts.plot(seriesdata, xlim = c(1987,2011))
forecast3 <- ts(predict(ar3, n.ahead=8)$pred, start = c(2009,2), frequency = 4)
se3 <- ts(predict(ar3, n.ahead=8)$se, start = c(2009,2), frequency = 4)
points(forecast1, type="l", col=2)
points(forecast1 - 1.96*se1, type="l", col=2, lty=2)
points(forecast1 + 1.96*se1, type="l", col=2, lty=2)
points(forecast3, type="l", col=4)
points(forecast3 - 1.96*se3, type="l", col=4, lty=2)
points(forecast3 + 1.96*se3, type="l", col=4, lty=2)
points(re, type="l", col=3)
```



```
ts.plot(seriesdata, xlim = c(2005,2011))
forecast3 <- ts(predict(ar3, n.ahead=8)$pred, start = c(2009,2), frequency = 4)
se3 <- ts(predict(ar3, n.ahead=8)$se, start = c(2009,2), frequency = 4)
points(forecast1, type="l", col=2)
points(forecast1 - 1.96*se1, type="l", col=2, lty=2)
points(forecast1 + 1.96*se1, type="l", col=2, lty=2)
points(forecast3, type="l", col=4)
points(forecast3 - 1.96*se3, type="l", col=4, lty=2)
points(forecast3 + 1.96*se3, type="l", col=4, lty=2)
points(re, type="l", col=3)
```



```
rmse(re, forecast1)
```

```
## [1] 0.8602256
```

```
rmse(re, forecast3)
```

```
## [1] 0.6752132
```

```
error1 <- re - forecast1
```

```
error3 <- re - forecast3
```

```
dm.test(error1, error3)
```

```
##
```

```
## Diebold-Mariano Test
```

```
##
```

```
## data: error1error3
```

```
## DM = 2.6014, Forecast horizon = 1, Loss function power = 2, p-value =
```

```
## 0.03535
```

```
## alternative hypothesis: two.sided
```