Econometrics III
Assignment Part 1 & 2
Tinbergen Insitute

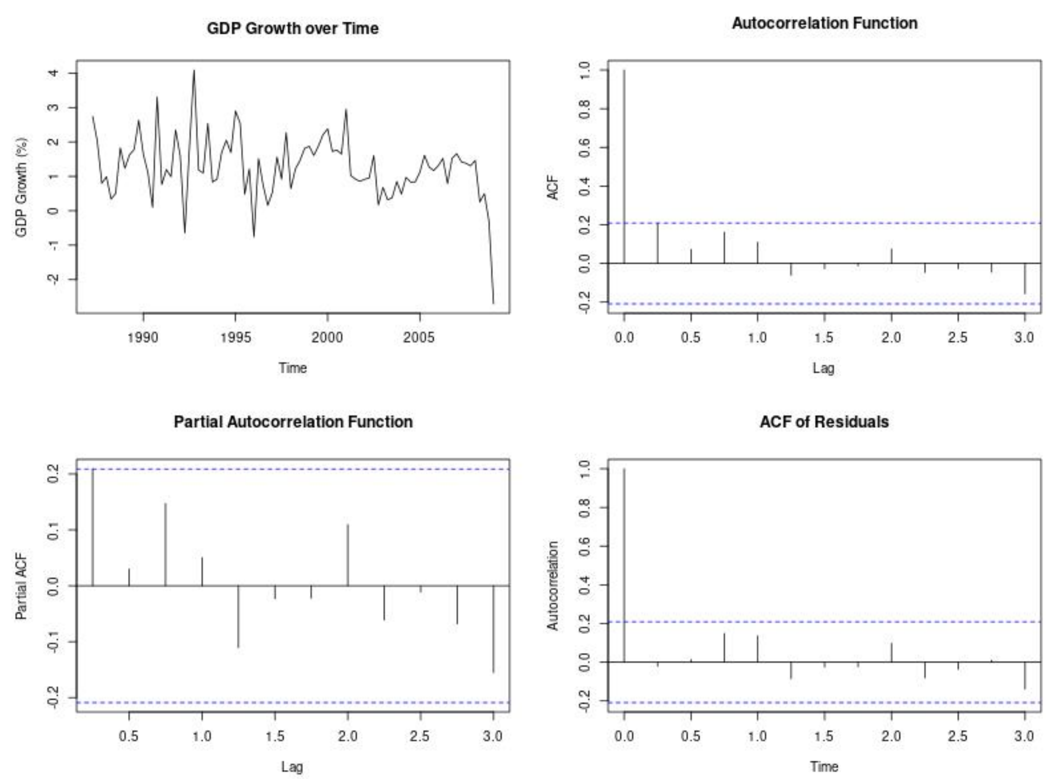
Stanislav Avdeev
590050sa
stnavdeev@gmail.com

Bas Machielsen
590049bm
590049bm@eur.nl

March 22, 2021

Question 1

Part 1




```r
# Part 1: Plot the Dutch GDP, ACF, and PACF
df <- readr::read_csv("./data/data_assign_p1.csv")

df <- ts(df$GDP_QGR,
         frequency = 4,
         start = c(1987, 2))
```

In the above figure, upper left, we observe the growth rates of Dutch GDP from 1987 to 2009.

```r
autoc <- acf(df,
    lag.max = 12, plot = F)

Box.test(df,
         lag = 12,
         type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  df
## X-squared = 12.106, df = 12, p-value = 0.4372
```

```r
pautoc <- pacf(df,
     lag.max = 12, plot = F)
```

In the above figure, we also plot the ACF and PACF, which is the ACF controlled for the other lagged correlations. The ACF tells us that the correlation of GDP growth with its lags is very low - hinting at very little time-dependence in this time-series. More precisely, the estimated correlation coefficients are not higher than 0.2 (for the lag of 1 period). If the GDP is indeed generated by an $AR(p)$ process, the estimates show that the process has low ϕ's (in absolute value), indicating a low time dependence.

Part 2

```r
ar4 <- dynlm(df ~ L(df, 1) + L(df, 2) + L(df, 3) + L(df, 4))
ar3 <- dynlm(df ~ L(df, 1) + L(df, 3) + L(df, 4))
ar2 <- dynlm(df ~ L(df, 1) + L(df, 3))
ar1 <- dynlm(df ~ L(df, 1))

ar1_m <- forecast::Arima(df, c(1,0,0), method = "CSS")

stargazer(ar4, ar3, ar2, ar1,
          type = "latex", header = FALSE,
          font.size = "small", column.sep.width = "0pt",
          omit.stat = c("adj.rsq", "ser", "f"))
```

The table below shows us the coefficients of the $AR(p)$ models with 1 to 4 lags included. Using a general-to-specific approach, we eliminate insignificant lags ($\alpha = 0.05$ level) at each step. Column (1) shows that only the first lag is significant at the 0.1 level. The second lag has the lowest ψ coefficient and the highest s.e., thus, we eliminate it first. Doing this iteratively, we are left with the first lag in the model which is significant at the 0.05 level. Investigating the coefficient, we observe there is a 1-period persistence of the time series on its lagged value. The *degree* of persistence, however, is not large, as the magnitude of the coefficient is fairly small (0.26).

Part 3

In the fourth plot above, we show the autocorrelation of the residuals. We observe that none of the coefficients attain significance. Hence, we think that the model is well-specified. Formally, we compute the Durbin-Watson test statistic, of which the output is shown below. The null hypothesis of no autocorrelation is not rejected. In an unreported plot, we also investigate the *partial* autocorrelation of residuals, in which there is also no sign of autocorrelation.

```r
durbinWatsonTest(ar1)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1     -0.02081405       1.85759    0.43
##  Alternative hypothesis: rho != 0
```

Table 1:

	Dependent variable:			
	df			
	(1)	(2)	(3)	(4)
L(df, 1)	0.232*	0.240*	0.257**	0.267**
	(0.124)	(0.122)	(0.119)	(0.117)
L(df, 2)	0.055			
	(0.126)			
L(df, 3)	0.203	0.210*	0.210*	
	(0.126)	(0.124)	(0.120)	
L(df, 4)	0.094	0.092		
	(0.125)	(0.124)		
Constant	0.479	0.533*	0.631***	0.896***
	(0.299)	(0.271)	(0.238)	(0.181)
Observations	84	84	85	87
R^2	0.099	0.097	0.089	0.058
Note:	*p<0.1; **p<0.05; ***p<0.01			

```r
autocred <- acf(ar1_m$resid, 12,
                plot = FALSE)
```

Part 4

```r
# Part 4: Forecast AR model for 2 years
df_pred <- predict(ar1_m, n.ahead = 8)$pred
```

We derived the forecasts by making use of the conditional expectation as the forecast minimizing the forecast error under quadratic loss. The conditional expectation of the forecast for $T+1$ is: $\mathbb{E}[X_{T+1}|X_1,\ldots,X_T] = \mathbb{E}[\phi X_T+\alpha+\epsilon_{T+1}|X_1,\ldots,X_T] = \phi X_T+\alpha$. Similarly, the forecast for $T+2$ equals $\phi\mathbb{E}[X_{T+1}+\alpha] = \phi(\alpha+\phi X_T)+\alpha$. Generalizing this pattern, we then end up with the following recursive forecasts (for $j>1$):

$$\mathbb{E}[X_{T+j}] = \alpha + \alpha \cdot \left(\sum_{i=1}^{j-1} \phi^i\right) + \phi^j X_T$$

Part 5

We construct the forecasts using a 95% symmetric confidence interval around the forecast.

```r
# Part 5: Produce CI
phi <- ar1_m$coef[1]
sigma <- sqrt(ar1_m$sigma2)

df_ciu <- predict(ar1_m, n.ahead = 8)$pred + predict(ar1_m, n.ahead = 8)$se*1.96
df_cil <- predict(ar1_m, n.ahead = 8)$pred - predict(ar1_m, n.ahead = 8)$se*1.96
```

We can reproduce the forecasts manually by using a standard confidence interval around the aforementioned predicted value, $CI(\mathbb{E}[X_{T+j}]) = \alpha + \alpha \cdot \left(\sum_{i=1}^{j-1} \phi^i\right) + \phi^j X_T \pm 1.96 \cdot \sigma$, where $\sigma_j^2 = \sigma_\epsilon^2 \cdot (1+\phi^2+\cdots+\phi^{2(j-1)})$.

Part 6

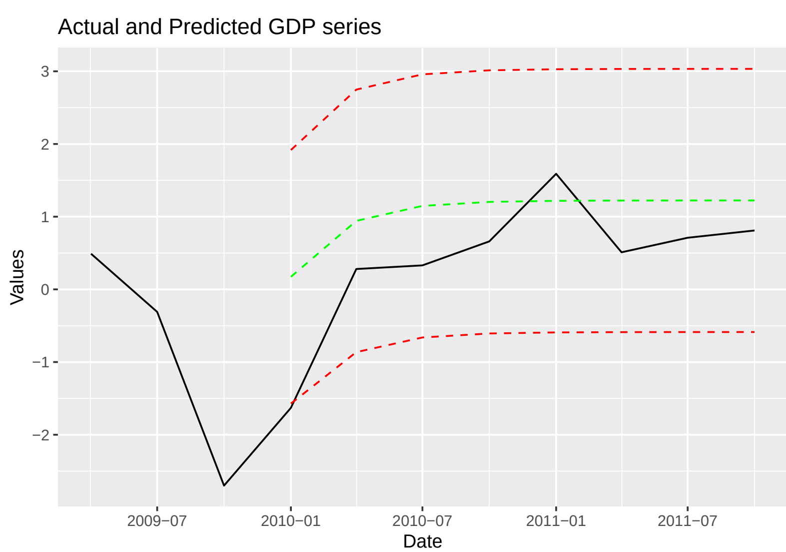
We check for normality of the residuals using the Jarque-Bera test:

```
jb.norm.test(ar1_m$resid)
```

```
##
##  Jarque-Bera test for normality
##
## data:  ar1_m$resid
## JB = 31.722, p-value = 0.001
```

According to the test, we have to reject H0, indicating that normality of the innovations is violated. In the figure below, we plot the forecasts (green, CIs in red) next to the actual data (black).

```
true <- ts(c(0.49, -0.31, -2.7, -1.63, 0.28, 0.33, 0.66, 1.59, 0.51, 0.71, 0.81),
         frequency = 4,
         start = c(2009, 2))

data <- data.frame(date = as.Date(time(true)),
           actual_values = as.matrix(true),
           prediction = c(rep(NA,3), as.matrix(df_pred)),
           upper = c(rep(NA,3), as.matrix(df_ciu)),
           lower = c(rep(NA,3),as.matrix(df_cil)))

data %>%
  ggplot(aes(x = date)) + geom_line(aes(y = actual_values)) +
  geom_line(aes(y = upper), color = "red", lty = "dashed") +
  geom_line(aes(y = lower), color = "red", lty = "dashed") +
  geom_line(aes(y = prediction), color = "green", lty = "dashed") +
  ylab("Values") + xlab("Date") +
  ggtitle("Actual and Predicted GDP series")
```



The forecasts show a tendency to revert to the mean of the AR process, which happens quite quickly after the drop. The drop has thus only a small effect on the forecasts. On the other hand, the limited T makes that the confidence bounds are quite large: on the one hand, all realized values fall within the 95% confidence bounds, which is a sign of accuracy, but on the other hand, the standard errors of the estimates are quite large, making the estimates less informative and perhaps less useful to policy makers.

Question 2

Part 1

First, we plot the unemployment and GDP series.

```r
plot(df2)
```

Secondly, we implement the general-to-specific approach for the ADL in R:

```r
recursive_reg <- function(dataset, equation){

  initial_reg <- dynlm(formula = as.formula(equation), data = dataset)

  p_values <- initial_reg %>%
    summary() %>%
    .$coefficients %>%
    .[,4]

  while(max(p_values) > 0.05){

    variable_drop <- which.max(p_values) %>%
      labels()

    equation <- equation %>%
      str_c("-", fixed(variable_drop))

    # reestimate the model and print the results
    initial_reg <- dynlm(formula = as.formula(equation), data = dataset)

    p_values <- initial_reg %>%
    summary() %>%
    .$coefficients %>%
    .[,4]

    print(summary(initial_reg))
  }

  initial_reg
```

```
}
```

Thirdly, we estimate the AR-models:

```
ar <- recursive_reg(dataset = df2, equation = "GDP_QGR ~ L(GDP_QGR, 1) +
              L(GDP_QGR, 2) + L(GDP_QGR, 3) + L(GDP_QGR, 4)")
```

Fourthly, we estimate the ADL-models:

```
adl <- recursive_reg(dataset = df2, equation = 'UN_RATE ~ L(UN_RATE, 1) +
              L(UN_RATE, 2) + L(UN_RATE, 3) + L(UN_RATE, 4) +
              L(GDP_QGR, 0) + L(GDP_QGR, 1) + L(GDP_QGR, 2) +
              L(GDP_QGR, 3) + L(GDP_QGR, 4)')
```

```
stargazer(ar, adl, type = "latex", header = FALSE,
          font.size = "small", column.sep.width = "0pt",
          omit.stat = c("adj.rsq", "ser", "f"),
          column.labels = c("AR for GDP", "ADL for Unemp"))
```

Table 2:

	Dependent variable:	
	equation	
	AR for GDP	ADL for Unemp
	(1)	(2)
L(UN_RATE, 1)		1.400^{***}
		(0.029)
L(UN_RATE, 3)		-0.432^{***}
		(0.029)
L(GDP_QGR, 1)	0.353^{***}	-0.027^{**}
	(0.092)	(0.013)
L(GDP_QGR, 3)	0.251^{***}	
	(0.092)	
Constant	0.400^{***}	0.208^{***}
	(0.144)	(0.052)
Observations	104	104
R^2	0.235	0.992
Note:		*p<0.1; **p<0.05; ***p<0.01

Part 3

Assuming strict exogeneity, we can interpret the coefficient estimates as being causal, in which case, taking $\delta X = 1$, we see that the long-run multiplier $\bar{Y} = \frac{\gamma_1}{1-\phi_1-\phi_2} < 0$, which means that indeed an increase in the GDP growth rate causes a decrease in the unemployment rate of about 4 percentage points. However, it also seems plausible there is an simultaneous relationship between GDP and Unemployment, or perhaps both of them are shaped by more fundamental parameters. In this case, the coefficients do not warrant a causal explanation, and there is only a negative *correlation* between GDP growth and unemployment.

Part 4

We know that the distribution of $Y_t|Y_{t-1} \sim N(\alpha + \phi_1 Y_{t-1} + \phi_3 Y_{t-3} + \gamma X_{t-1}, \sigma^2)$. Shifting everything by one period, we obtain that the distribution of Y_{t+1} conditional on the realized values of Y_t is:

$$Y_{t+1}|Y_t \sim N(\alpha + \phi_1 Y_t + \phi_3 Y_{t-2} + \gamma_1 X_t, \sigma^2)$$

Denote the mean of this distribution by μ. Then, we are looking for:

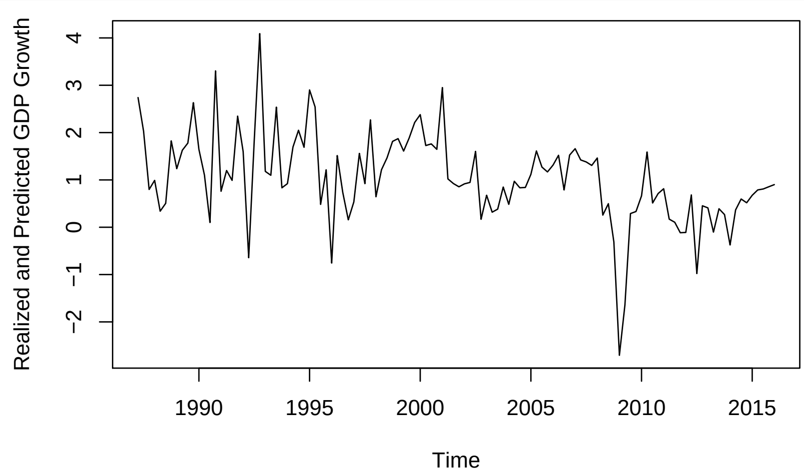
$$\mathbb{P}(Y_{t+1} > 0.078|Y_t) = \mathbb{P}\left(\frac{Y_{t+1} - \mu}{\sigma} > \frac{0.078 - \mu}{\sigma}\right) = 1 - \Phi\left(\frac{0.078 - \mu}{\sigma}\right)$$

Substituting $\mu = ...$ and $\sigma = \text{Var}(u_i) = ...$ gives:

Part 5

First, we produce an 8-quarter forecast for the AR-process underlying GDP growth:

```r
#input dataframe with one column
forecast_gdp <- function(no_periods, model, df){

  coefficients <- model$coefficients

  for(i in 1:no_periods){
    prediction <- coefficients[1] +
    coefficients[2] * df[length(df)] +
    coefficients[3] * df[length(df) - 2]

    df <- c(df, unname(prediction))

  }

  df

}

plot(ts(start = c(1987, 2),
        frequency = 4,
        forecast_gdp(8, ar, df2[,1])), ylab = "Realized and Predicted GDP Growth")
```




```r
# Predicted Values:
ts(start = c(1987, 2),
        frequency = 4,
        forecast_gdp(8, ar, df2[,1])
```

```r
  ) %>%
  .[(length(.)-8):length(.)]
```

```
## [1] -0.3744039  0.3660043  0.5967171  0.5170298  0.6747751  0.7884422  0.8085984
## [8]  0.8553275  0.9003790
```

Part 6

```r
# algorithm
irf_gdp <- function(model, lags_in_model, x, e, no_periods){

 coefficients <- model$coefficients

 # initial data frame
 irfs <- rep(x, lags_in_model)
 response <- x + e

 irfs <-  c(irfs, response)

 # all other predictions
  for(i in 2:no_periods){

    response <- coefficients[2] * irfs[length(irfs)] +
    coefficients[3] * irfs[length(irfs) - 2]

    irfs <- c(irfs, unname(response))

  }

 data.frame(time = seq(from = -lags_in_model, to = no_periods -1, by = 1),
            irfs = irfs)

}

irf_gdp(ar, 3, 0, 1, 10) %>%
  plot(type = "l")
```