Econometrics III
Assignment Part 3, 4, 5
Tinbergen Insitute

Stanislav Avdeev
590050sa
stnavdeev@gmail.com

Bas Machielsen
590049bm
590049bm@eur.nl

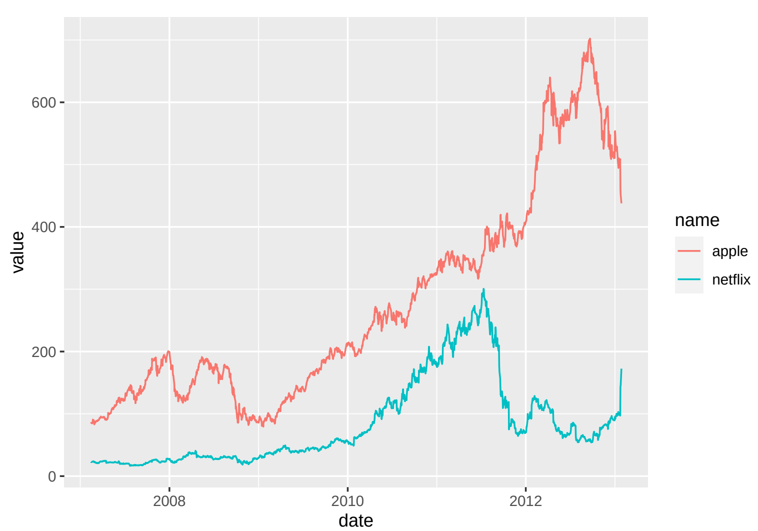
March 25, 2021

Question 1

Part 1:

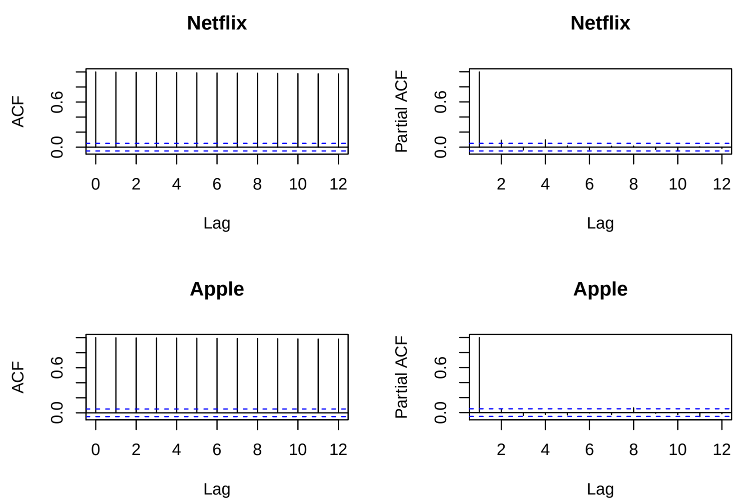
First, we plot the two time series:

```r
df3 <- readr::read_csv("./data/data_assign_p3.csv")

twostocks <- c("apple", "netflix")

df3_twostocks <- df3 %>%
    janitor::clean_names() %>%
    pivot_longer(-date) %>%
    filter(is.element(name, twostocks)) %>%
    mutate(date = lubridate::dmy(date))

df3_twostocks %>%
    ggplot(aes(x = date, y = value,
               group = name, color = name)) + geom_line()
```



Then, we show the acf and pacf-functions:

```
n1 <- df3$NETFLIX %>%
    acf(lag.max = 12, plot = F)

n2 <- df3$NETFLIX %>%
    pacf(lag.max = 12, plot = F)

a1 <- df3$APPLE %>%
    acf(lag.max = 12, plot = F)

a2 <- df3$APPLE %>%
    pacf(lag.max = 12, plot = F)

par(mfrow=c(2,2))

plot(n1, main = "Netflix");plot(n2, main = "Netflix")
plot(a1, main = "Apple"); plot(a2, main = "Apple")
```



The ACF's tell us that the stock price is highly dependent on the past stock price, and this dependence decays only very slowly: even the 50 or 100-period lag still shows significant autocorrelation.

Part 2

We now implement a general to specific unit root test function:

```
unit_root_test <- function(column, order){

  # Make the dataset
  series <- ts(column)
  first_differences <- diff(series, differences = 1)
  laggedvar <- stats::lag(series, -1)

  #other lagged first differences, delta x_{t-1}, .., delta x_{t-p+1}
  lagged_fds <- list()

  for(i in 1:(order-1)){

    lagged_fds[[i]] <- stats::lag(first_differences, k = -i)

  }

  df <- cbind(first_differences, laggedvar, purrr::reduce(lagged_fds, cbind)) %>%
    as_tibble()

  colnames(df) <- c("dxt", "xtm1", paste("dxtm", 1:(order-1), sep = ""))

  df <- df %>%
    na.omit()

  # run the stepwise regression - find the best model
  null = lm(data = df, formula = "dxt ~ xtm1")
  full = lm(data = df, formula = paste("dxt ~ xtm1 +",
                                       paste(paste("dxtm", 1:(order-1), sep = ""),
                                             collapse = ' + '),
                                       collapse = " ")
  )

  bestmodel <- step(full,
       scope = list(lower = null, upper = full),
       direction = "backward",
       criterion = "BIC",
       k = log(nrow(df)))

  # perform the unit root test (MacKinnon, 2010)
  b_critical <- -1.6156

  t_value <- bestmodel %>%
    summary() %>%
    .$coefficients %>%
    .[,3] %>%
    .["xtm1"]

  significant = abs(t_value) > abs(b_critical)

  data.frame(stock = deparse(substitute(column)),
             best_model = as.character(bestmodel$call[2]),
             t_value = t_value,
             sig = significant)
}
```

```r
summary_tests <- list()

for(i in 1:length(colnames(df3[,-1]))){
  df <- unit_root_test(df3[, i+1], 12)
  summary_tests[[i]] <- df %>%
    mutate(name = colnames(df3[,-1][i]))
}

summary_tests <- summary_tests %>%
  purrr::reduce(rbind) %>%
  select(-stock)

rownames(summary_tests) <- NULL
```

```r
knitr::kable(summary_tests)
```

best_model	t_value	sig	name
dxt ~ xtm1 + dxtm1 + dxtm3 + dxtm7	-0.6984454	FALSE	APPLE
dxt ~ xtm1 + dxtm1 + dxtm2	-2.0475011	TRUE	EXXON_MOBIL
dxt ~ xtm1	-1.1619768	FALSE	FORD
dxt ~ xtm1 + dxtm1 + dxtm3	-1.5455395	FALSE	GEN_ELECTRIC
dxt ~ xtm1	-2.2598273	TRUE	INTEL
dxt ~ xtm1	-2.3999971	TRUE	MICROSOFT
dxt ~ xtm1 + dxtm2	-1.0642387	FALSE	NETFLIX
dxt ~ xtm1 + dxtm2 + dxtm3	-0.6373731	FALSE	NOKIA
dxt ~ xtm1 + dxtm1	-1.3045653	FALSE	SP500
dxt ~ xtm1 + dxtm1 + dxtm6 + dxtm11	-2.6283893	TRUE	YAHOO

The test seems to be significant for a number of stocks, indicating that for these stocks, the null hypothesis of a unit root is rejected in favor of stationarity. With a 10% α-level, we expect to see a type I-error (rejecting the null while it is true) about one tenth of the time for every test. This means that the probability of having at least 1 type-I error is very large: $(1 - 0.9^{10}) = 0.6513216$. It would be better to use some kind of Bonferroni correction to correct for these compounding type I-errors, but alternatively, we could also lower the α-level.

Part 3

The forecast $\mathbb{E}[P_{t+1}|D_t] = \mathbb{E}[P_{t+1}|P_t] = \mathbb{E}[P_t + \epsilon_t] = P_t$. Similarly, the forecast $\mathbb{E}[P_{t+2}] = \mathbb{E}[P_{t+1} + \epsilon_{t+1}] = \mathbb{E}[P_{t+1}] = P_t$. Generalizing this pattern, the forecast for $P_{t+h} = P_t$. The variance of the forecast is derived using the distribution:

$$
\begin{aligned}
P_{t+1} &= P_t + \epsilon_t \Rightarrow P_{t+1}|P_t \sim N(P_t, \sigma^2) \\
P_{t+2} &= P_{t+1} + \epsilon_{t+1} = \\
&= P_t + \epsilon_t + \epsilon_{t+1} \Rightarrow P_{t+2}|P_t \sim N(P_t, 2\sigma^2)
\end{aligned}
$$

Hallo