

# Problem 2 (11 credits)

HW1

*Michael Brucek (mbrucek), Eduardo Chavez (echavezh), Kevin Grady (grady133), Danny Moncada (monca016), Tian Zhang (zhan7003)*

*February 17, 2020*

```
suppressWarnings(suppressPackageStartupMessages({  
  library(TSA)  
  library(forecast)  
  library(ggplot2)  
  library(dplyr)  
}))
```

## Moving average with the constant trend

Assume  $Y_t$  is a moving average process with the constant trend such as

$$Y_t = (e_t + e_{t-1} + e_{t-2})/3$$

where  $e_i \sim N(0, 1)$  i.i.d

### Question 1

**a) (1 credit)**

Please compute analytically the mean:

$$E[Y_3] =$$

```
E_Y3 <- 0
```

(please include only the numerical answer above not the computation. An acceptable format for your answer is such as `E_Y3 <- 5`)

**b) (1 credit)**

Please compute analytically the variance:

$$\text{Var}[Y_3] =$$

```
Var_Y3 <- 1/3
```

**c) (1 credit)**

Please compute analytically the autocovariance at lag 1:

$$\text{Cov}(Y_5, Y_4) =$$

```
Cov_Y5Y4 <- 2/9
```

**d) (1 credit)**

Please compute analytically the autocovariance at lag 2:

$\text{Cov}(Y_6, Y_4) =$

```
Cov_Y6Y4 <- 1/9
```

**e) (1 credit)**

Please compute analytically the autocovariance at lag 3:

$\text{Cov}(Y_7, Y_4) =$

```
Cov_Y7Y4 <- 0
```

**f) (1 credit)**

Please compute analytically the autocovariance at lag 4:

$\text{Cov}(Y_8, Y_4) =$

```
Cov_Y8Y4 <- 0
```

**g) (1 credit)**

Is the process stationary?

```
Stationarity <- TRUE #enter TRUE or FALSE
```

## Question 2

**a) (2 credits)**

Please generate  $N = 100$  sample paths of length  $T = 100$  for this stochastic process.

- Please save the results into a data.frame `df2b` where:
  - column `df2b$Y` has the values of the process
  - column `df2b$id` has the id of the sample path
  - column `df2b$t` has the time

```
set.seed(42) # Please do not change the seed

N <- 100L
T <- 100L

df2b <- data.frame(e = rnorm(T*N), id = rep(1:N, each = T), t = rep(1:T, N))

df2b <- df2b %>% group_by(id) %>%
  mutate(Y= 0.33*e + 0.33*zlag(e) + 0.33*zlag(e, 2)) %>% select(Y, id, t)

head(df2b)

## # A tibble: 6 x 3
## # Groups:   id [1]
##       Y     id     t
##   <dbl> <int> <int>
## 1 NA         1     1
## 2 NA         1     2
## 3 0.386       1     3
```

```
## 4  0.142    1    4
## 5  0.462    1    5
## 6  0.307    1    6
```

**b) (1 credit)**

Please plot the sample paths that you generated in the previous question

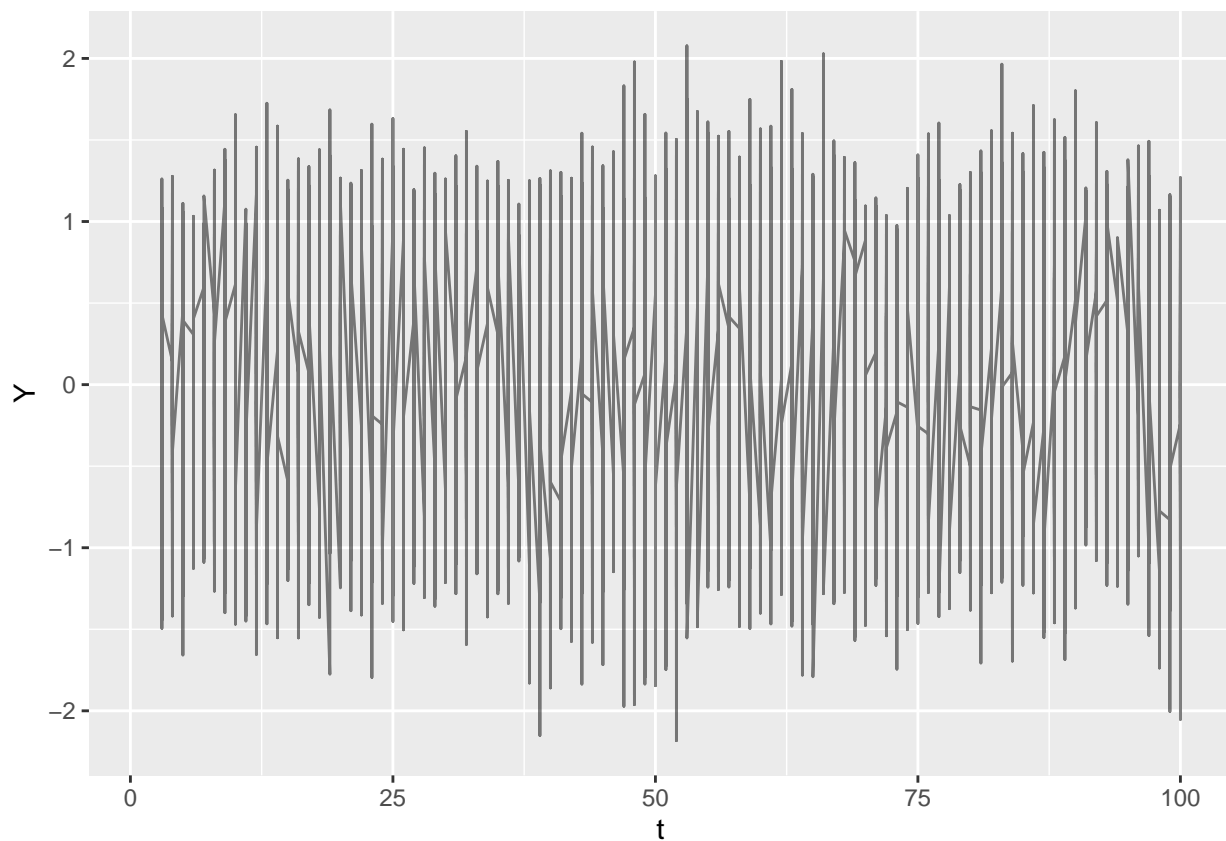
- Please save your plot into variable `p2c`

**Hints:**

- use `ggplot` and take advantage of the long format of the data
- please don't change the color (keep the lines black) but do put `alpha=0.5` into your `geom_line` to make sample paths somewhat transparent.
- do not use `geom_points` just `geom_line` is fine
- As you will see from your plot:
  - the fainter the line the less likely the stochastic process would reach this spot

```
p2c <- ggplot(data = df2b, aes(x = t, y = Y)) + geom_line(alpha = 0.5)
p2c
```

```
## Warning: Removed 200 rows containing missing values (geom_path).
```



- You should be able to clearly see that this process is indeed stationary.
  - The density of black color represents the probability of finding the process in this place.

c) (1 credits)

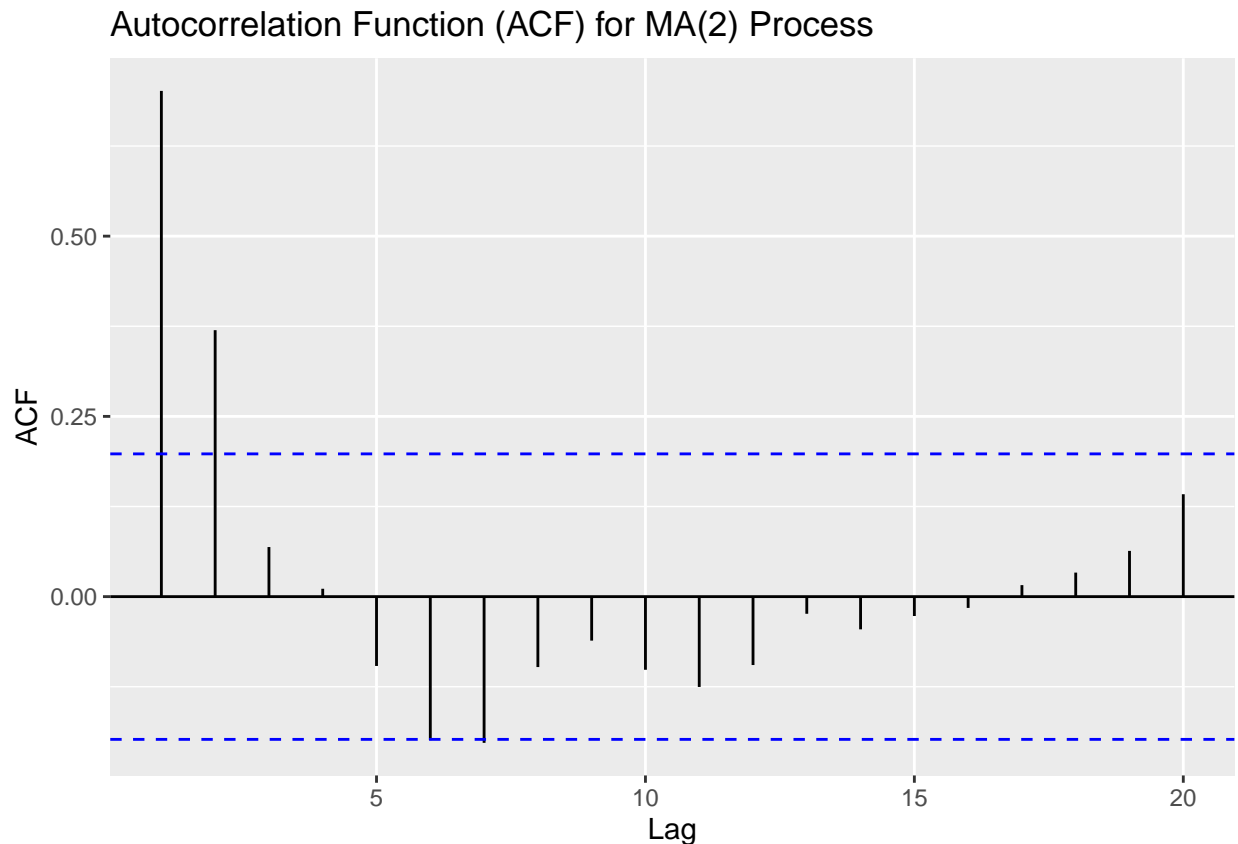
Please pick a sample path with `id=1` from `df2b` and plot an autocorrelation function (ACF).

- Autocorrelation function plot is a plot of approximately 20 different correlations estimated from one sample path. For example, point 1 on X-axis of that plot corresponds to your estimate of  $\text{Corr}(X_t, X_{t-1})$  (this is called *autocorrelation at lag 1*), point 2 on X-axis of that plot corresponds to your estimate of  $\text{Corr}(X_t, X_{t-2})$  (*autocorrelation at lag 2*) and so on.
- Autocorrelation plots also contain the (dotted blue) lines that corresponds to the noise threshold. All autocorrelations within the band are indistinguishable from noise (statistically insignificant). However, autocorrelations at some lags will clearly stick outside of that band.
- The work that you did in class over the previous sessions should start paying off now. This autocorrelation plot will clearly reveal something interesting: you have a strong auto-correlation at lags 1 and 2 and nothing beyond that. This is a clear indication of the moving average process of an order 2.

Hints:

- use `ggAcf` function from `forecast` package

```
ggAcf(df2b[df2b$id == 1, ]$Y) + ggtitle("Autocorrelation Function (ACF) for MA(2) Process")
```



- The analytical work that you did in class over the previous sessions should start paying off now. Please look back all the way to problem 2 Q1 and compare to your answers to Q1 e) to Q1 f). Autocorrelation is standardized autocovariance. You will find that a zero autocovariance leads to a very small autocorrelation in the ACF.
- This autocorrelation plot will clearly reveal something interesting: you do indeed have a strong auto-correlation at lags 1 and 2 and almost zero beyond that. In other words, when you work with real

data and you happen to get this ACF with two peaks at lag 1 and lag 2 – you should immediately understand what type of process tends to generate such ACF plot – moving average!

- (no need to report anything about your conclusions here)