

Taller 3 - Análisis espectral

Adaptado del taller 2022 de Federico Molina

Series Cronológicas 2024

Abril 2024

1. Práctico 4 - Ejercicio 2

```
# Definimos los parámetros de los modelos AR(1)

# Parte a
phi_a <- 0.5

# Parte b
phi_b <- 0.99

# Parte c
phi_c <- -0.5

# Parte a

# Obtenemos el espectro teórico (escalado entre 0 y 0,5)
espectro_a <- ARMAspec(model = list(ar = phi_a),
                        plot = FALSE,
                        freq = seq(from = 0,
                                   to = 0.5,
                                   length.out = 500))

# Guardamos los datos para poder graficar en ggplot()
espectro_a <- data.frame(Frecuencia = espectro_a$freq,
                        Espectro = espectro_a$spec)

# Graficamos el espectro teórico
ggplot(espectro_a) +
  geom_line(aes(x = Frecuencia, y = Espectro)) +
  ggtitle(expression(phi ~ "= 0,5"))
```

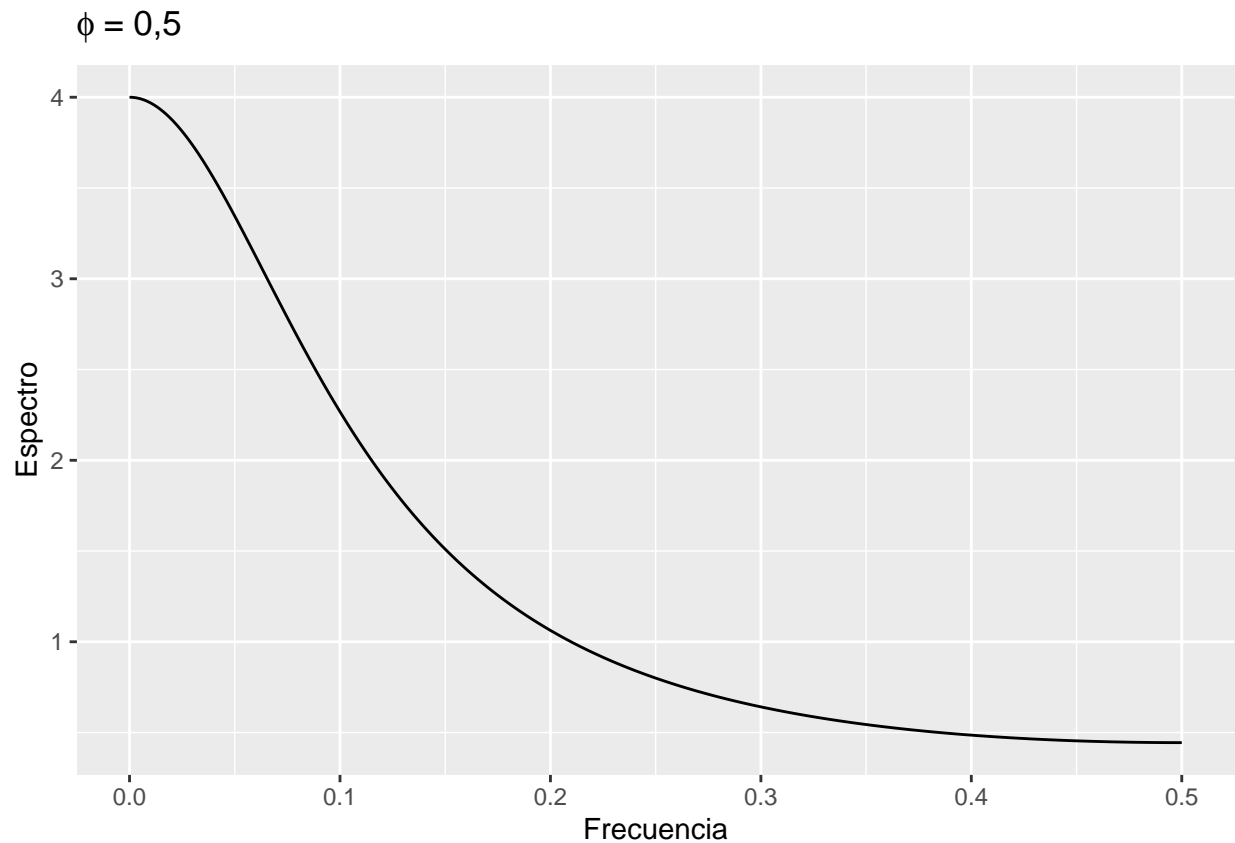


Figura 1: Espectro teórico de un proceso AR(1) con coeficiente igual a 0,5.

```
# Otra forma: graficar directamente como gráfico de Rbase
# ARMAspec(model = list(ar = phi_a),
#           plot = TRUE,
#           freq = seq(from = 0,
#                       to = 0.5,
#                       length.out = 500))
```

```
# Parte b

# Obtenemos el espectro teórico (escalado entre 0 y 0,5)
espectro_b <- ARMAspec(model = list(ar = phi_b),
                        plot = FALSE,
                        freq = seq(from = 0,
                                    to = 0.5,
                                    length.out = 500))

# Guardamos los datos para poder graficar en ggplot()
espectro_b <- data.frame(Frecuencia = espectro_b$freq,
                        Espectro = espectro_b$spec)

# Graficamos el espectro teórico
ggplot(espectro_b) +
  geom_line(aes(x = Frecuencia, y = Espectro)) +
```

```
ggtitle(expression(phi ~ "=" 0,99"))
```

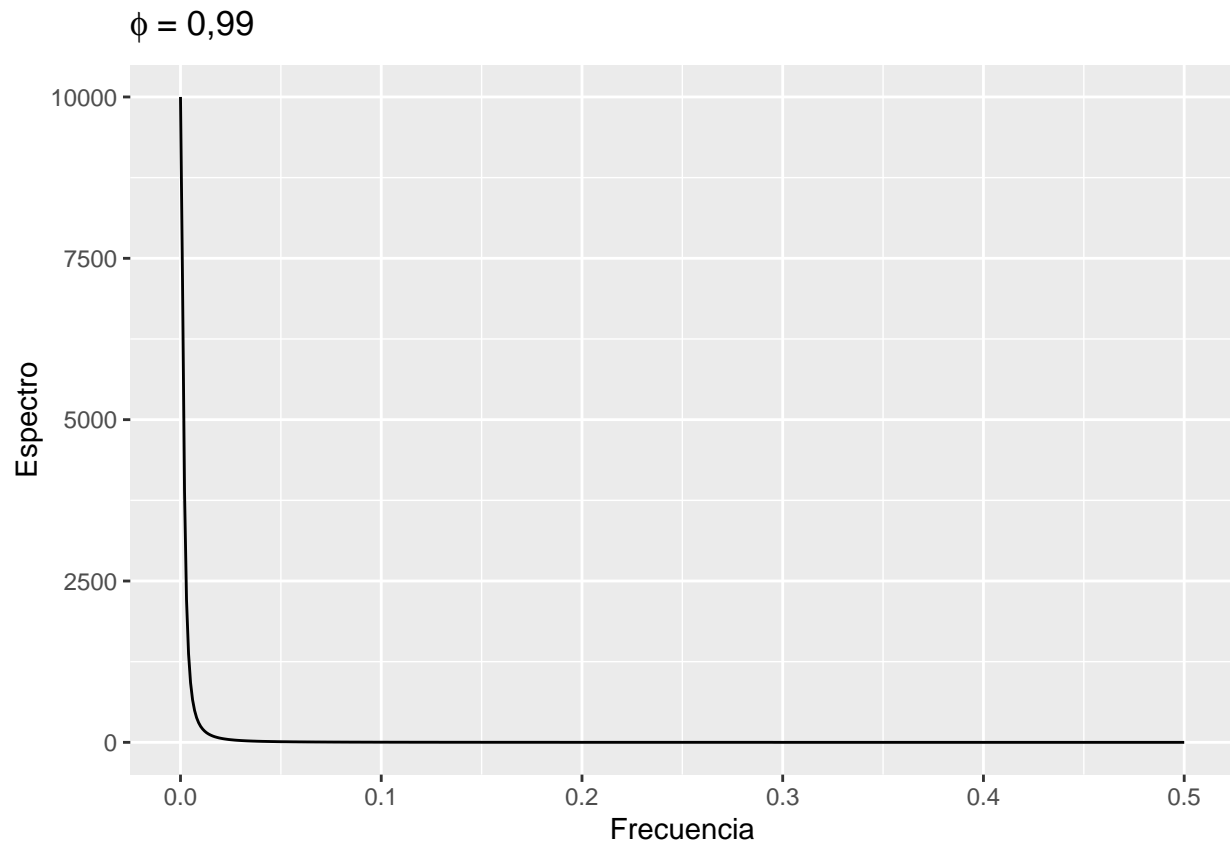


Figura 2: Espectro teórico de un proceso AR(1) con coeficiente igual a 0,99.

```
# Parte c

# Obtenemos el espectro teórico (escalado entre 0 y 0,5)
espectro_c <- ARMAspec(model = list(ar = phi_c),
  plot = FALSE,
  freq = seq(from = 0,
    to = 0.5,
    length.out = 500))

# Guardamos los datos para poder graficar en ggplot()
espectro_c <- data.frame(Frecuencia = espectro_c$freq,
  Espectro = espectro_c$spec)

# Graficamos el espectro teórico
ggplot(espectro_c) +
  geom_line(aes(x = Frecuencia, y = Espectro)) +
  ggtitle(expression(phi ~ "=" -0,5"))
```

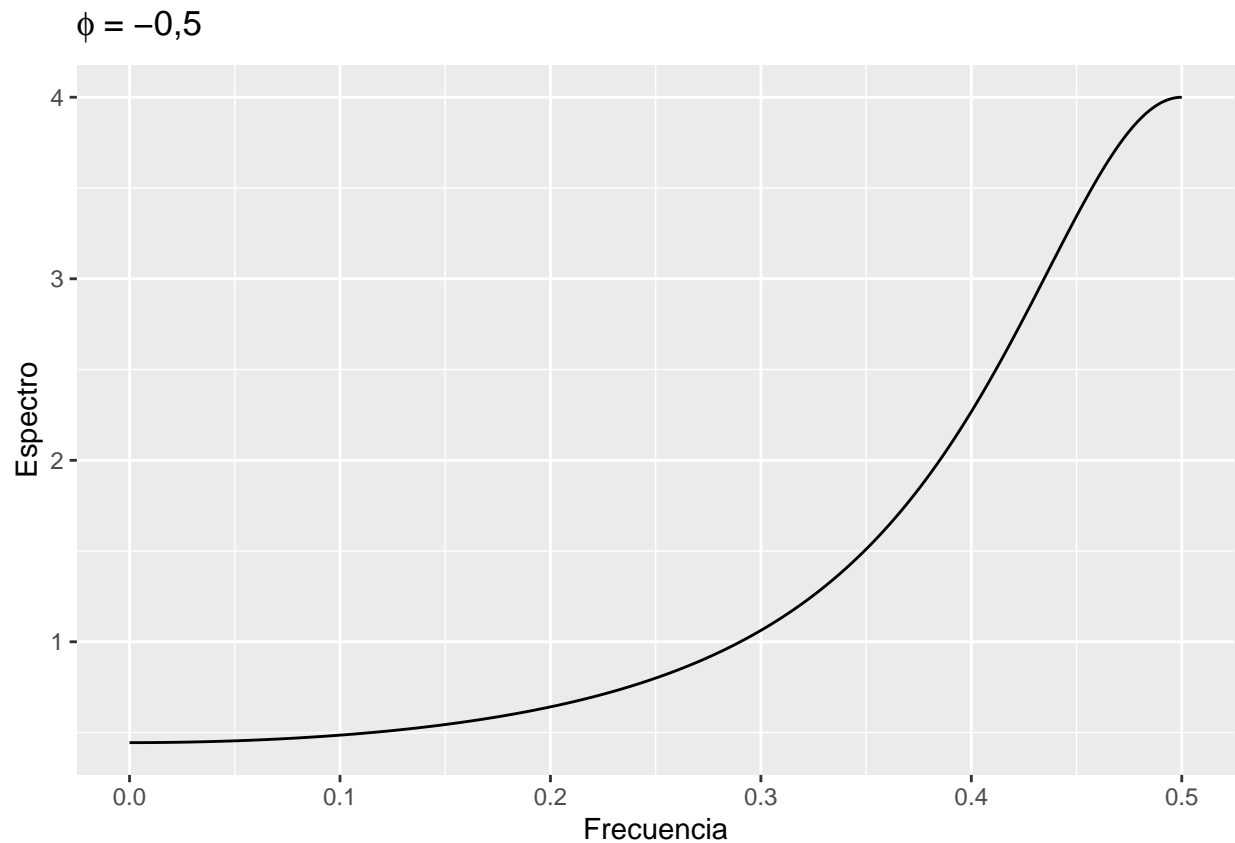


Figura 3: Espectro teórico de un proceso AR(1) con coeficiente igual a -0,5.

2. Práctico 4 - Ejercicio 3

```
# Definimos los parámetros de los modelos MA(1)
# Recordar que los parámetros tienen signo opuesto a la notación que vimos en clase

# Proceso 1
theta1 <- 1

# Proceso 2
theta2 <- 0.2

# Proceso 3
theta3 <- 0.6

# Obtenemos el espectro teórico (escalado entre 0 y 0,5)
espectro1_teorico <- ARMAspec(model = list(ma = theta1,
                                           sd = sqrt(2)),
                              plot = FALSE,
                              freq = seq(from = 0.01,
                                         to = 0.5,
                                         by = 0.01))
```

```

# Simulamos 100 observaciones MA(1)
proceso1 <- arima.sim(model = list(ma = theta1,
                                   sd = sqrt(2)),
                     n = 100)

# Estimamos el espectro del proceso mediante un periodograma
espectro1_est <- spectrum(proceso1,
                          # Guardamos los valores sin graficarlos
                          plot = FALSE,
                          # No aplicamos logaritmo al espectro
                          log = "no")

espectro1_est_suavizado <- spectrum(proceso1,
                                    # Suavizamos el periodograma
                                    spans = 50,
                                    plot = FALSE,
                                    log = "no")

# Guardamos toda la información en un dataframe
proceso1_teorico_est <- data.frame(Frecuencia = espectro1_teorico$freq,
                                   Espectro_teo = espectro1_teorico$spec,
                                   Espectro_est = espectro1_est$spec,
                                   Espectro_est_suavizado = espectro1_est_suavizado$spec)

proceso1_teorico_est <- proceso1_teorico_est %>%
  pivot_longer(cols = c(Espectro_teo,
                        Espectro_est,
                        Espectro_est_suavizado),
               values_to = "valores") %>%
  mutate(name = recode(name,
                       "Espectro_teo" = "Espectro teórico",
                       "Espectro_est" = "Periodograma",
                       "Espectro_est_suavizado" = "Periodograma suavizado"))

# Graficamos las tres curvas juntas
ggplot(proceso1_teorico_est) +
  geom_line(aes(x = Frecuencia,
                y = valores,
                color = name)) +
  labs(x = "Frecuencia",
       y = "Espectro",
       color = "Espectro") +
  ggtitle(expression(theta ~ "= 1"))

```

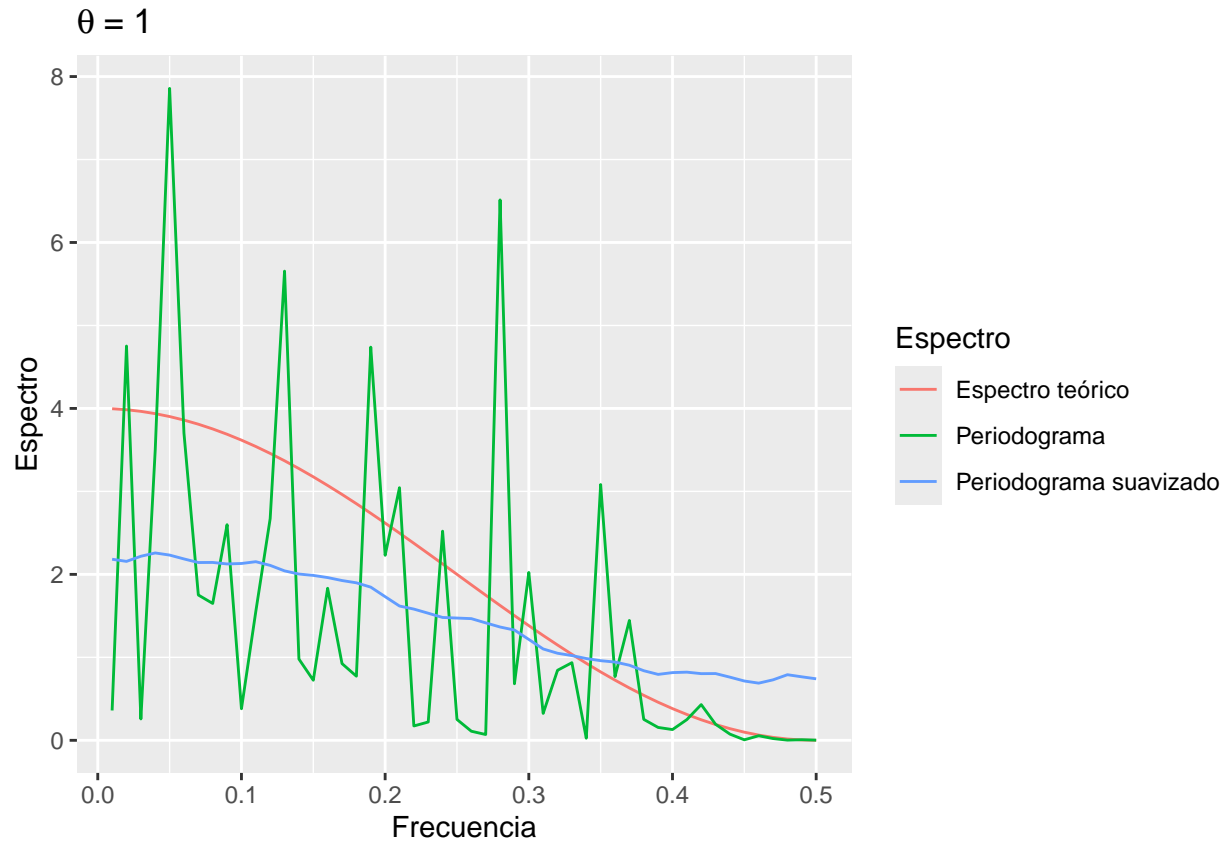


Figura 4: Espectro teórico y periodogramas de un proceso MA(1) con coeficiente igual a 1.

```
# Obtenemos el espectro teórico (escalado entre 0 y 0,5)
espectro2_teorico <- ARMAspec(model = list(ma = theta2,
                                           sd = sqrt(2)),
                             plot = FALSE,
                             freq = seq(from = 0.01,
                                         to = 0.5,
                                         by = 0.01))

# Simulamos 100 observaciones MA(1)
proceso2 <- arima.sim(model = list(ma = theta2,
                                   sd = sqrt(2)),
                     n = 100)

# Estimamos el espectro del proceso mediante un periodograma
espectro2_est <- spectrum(proceso2,
                          # Guardamos los valores sin graficarlos
                          plot = FALSE,
                          # No aplicamos logaritmo al espectro
                          log = "no")

espectro2_est_suavizado <- spectrum(proceso2,
                                   # Suavizamos el periodograma
                                   spans = 50,
```

```

        plot = FALSE,
        log = "no")

# Guardamos toda la información en un dataframe
proceso2_teorico_est <- data.frame(Frecuencia = espectro2_teorico$freq,
                                   Espectro_teo = espectro2_teorico$spec,
                                   Espectro_est = espectro2_est$spec,
                                   Espectro_est_suavizado = espectro2_est_suavizado$spec)

proceso2_teorico_est <- proceso2_teorico_est %>%
  pivot_longer(cols = c(Espectro_teo,
                        Espectro_est,
                        Espectro_est_suavizado),
               values_to = "valores") %>%
  mutate(name = recode(name,
                       "Espectro_teo" = "Espectro teórico",
                       "Espectro_est" = "Periodograma",
                       "Espectro_est_suavizado" = "Periodograma suavizado"))

# Graficamos las tres curvas juntas
ggplot(proceso2_teorico_est) +
  geom_line(aes(x = Frecuencia,
                y = valores,
                color = name)) +
  labs(x = "Frecuencia",
       y = "Espectro",
       color = "Espectro") +
  ggtitle(expression(theta ~ "= 0,2"))

```

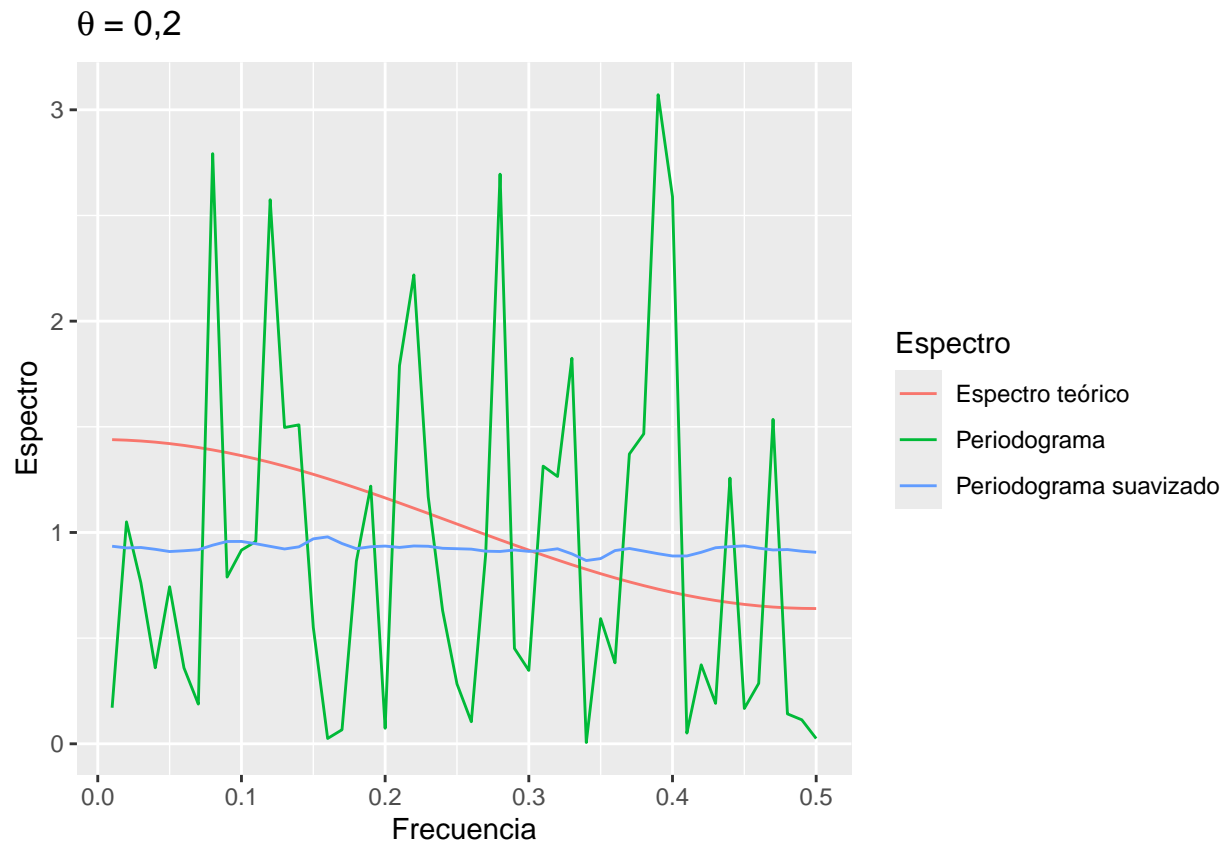


Figura 5: Espectro teórico y periodogramas de un proceso MA(1) con coeficiente igual a 0,2.

```
# Obtenemos el espectro teórico (escalado entre 0 y 0,5)
espectro3_teorico <- ARMAspec(model = list(ma = theta3,
                                           sd = sqrt(2)),
                             plot = FALSE,
                             freq = seq(from = 0.01,
                                         to = 0.5,
                                         by = 0.01))

# Simulamos 100 observaciones MA(1)
proceso3 <- arima.sim(model = list(ma = theta3,
                                   sd = sqrt(2)),
                     n = 100)

# Estimamos el espectro del proceso mediante un periodograma
espectro3_est <- spectrum(proceso3,
                          # Guardamos los valores sin graficarlos
                          plot = FALSE,
                          # No aplicamos logaritmo al espectro
                          log = "no")

espectro3_est_suavizado <- spectrum(proceso3,
                                   # Suavizamos el periodograma
                                   spans = 50,
```



```

        plot = FALSE,
        log = "no")

# Guardamos toda la información en un dataframe
proceso3_teorico_est <- data.frame(Frecuencia = espectro3_teorico$freq,
                                   Espectro_teo = espectro3_teorico$spec,
                                   Espectro_est = espectro3_est$spec,
                                   Espectro_est_suavizado = espectro3_est_suavizado$spec)

proceso3_teorico_est <- proceso3_teorico_est %>%
  pivot_longer(cols = c(Espectro_teo,
                        Espectro_est,
                        Espectro_est_suavizado),
               values_to = "valores") %>%
  mutate(name = recode(name,
                       "Espectro_teo" = "Espectro teórico",
                       "Espectro_est" = "Periodograma",
                       "Espectro_est_suavizado" = "Periodograma suavizado"))

# Graficamos las tres curvas juntas
ggplot(proceso3_teorico_est) +
  geom_line(aes(x = Frecuencia, y = valores, color = name)) +
  labs(x = "Frecuencia",
       y = "Espectro",
       color = "Espectro") +
  ggtitle(expression(theta ~ "= 0,6"))

```

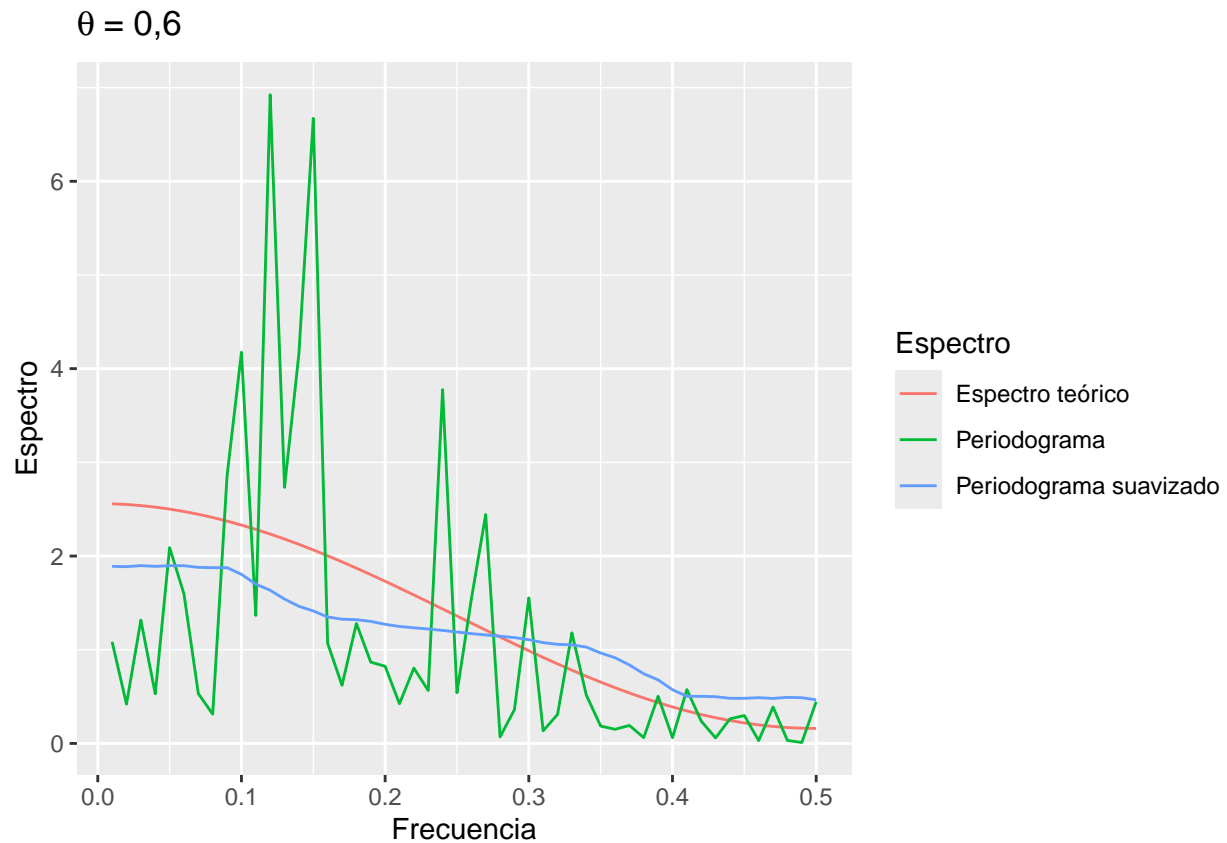


Figura 6: Espectro teórico y periodogramas de un proceso MA(1) con coeficiente igual a 0,6.

3. Práctico 4 - Ejercicio 4

```
# Definimos el parámetro del proceso
phi3 <- 0.99

# Obtenemos el espectro teórico (escalado entre 0 y 0,5)
espectro_ar3_teorico <- ARMAspec(model = list(ar = phi3),
  plot = FALSE,
  freq = seq(from = 0.01,
    to = 0.5,
    length.out = 60))

# Simulamos 100 observaciones AR(3)
proceso_ar3 <- arima.sim(model = list(ar = phi3), n = 60)

# Estimamos el espectro del proceso mediante un periodograma
espectro_ar3_est <- spectrum(proceso_ar3,
  # Guardamos los valores sin graficarlos
  plot = FALSE,
  # No aplicamos logaritmo al espectro
  log = "no")
```

```

espectro_ar3_est_suavizado <- spectrum(proceso_ar3,
                                     # Suavizamos el periodograma
                                     spans = 50,
                                     plot = FALSE,
                                     log = "no")

# Guardamos toda la información en un dataframe
proceso_ar3_teorico_est <- data.frame(Frecuencia = espectro_ar3_teorico$freq,
                                     Espectro_teo = espectro_ar3_teorico$spec,
                                     Espectro_est = espectro_ar3_est$spec,
                                     Espectro_est_suavizado = espectro_ar3_est_suavizado$spec)

proceso_ar3_teorico_est <- proceso_ar3_teorico_est %>%
  pivot_longer(cols = c(Espectro_teo,
                        Espectro_est,
                        Espectro_est_suavizado),
               values_to = "valores") %>%
  mutate(name = recode(name,
                       "Espectro_teo" = "Espectro teórico",
                       "Espectro_est" = "Periodograma",
                       "Espectro_est_suavizado" = "Periodograma suavizado"))

# Graficamos las tres curvas juntas
ggplot(proceso_ar3_teorico_est) +
  geom_line(aes(x = Frecuencia,
                y = valores,
                color = name)) +
  labs(x = "Frecuencia",
       y = "Espectro",
       color = "Espectro") +
  ggtitle(expression(phi ~ "= 0,99"))

```

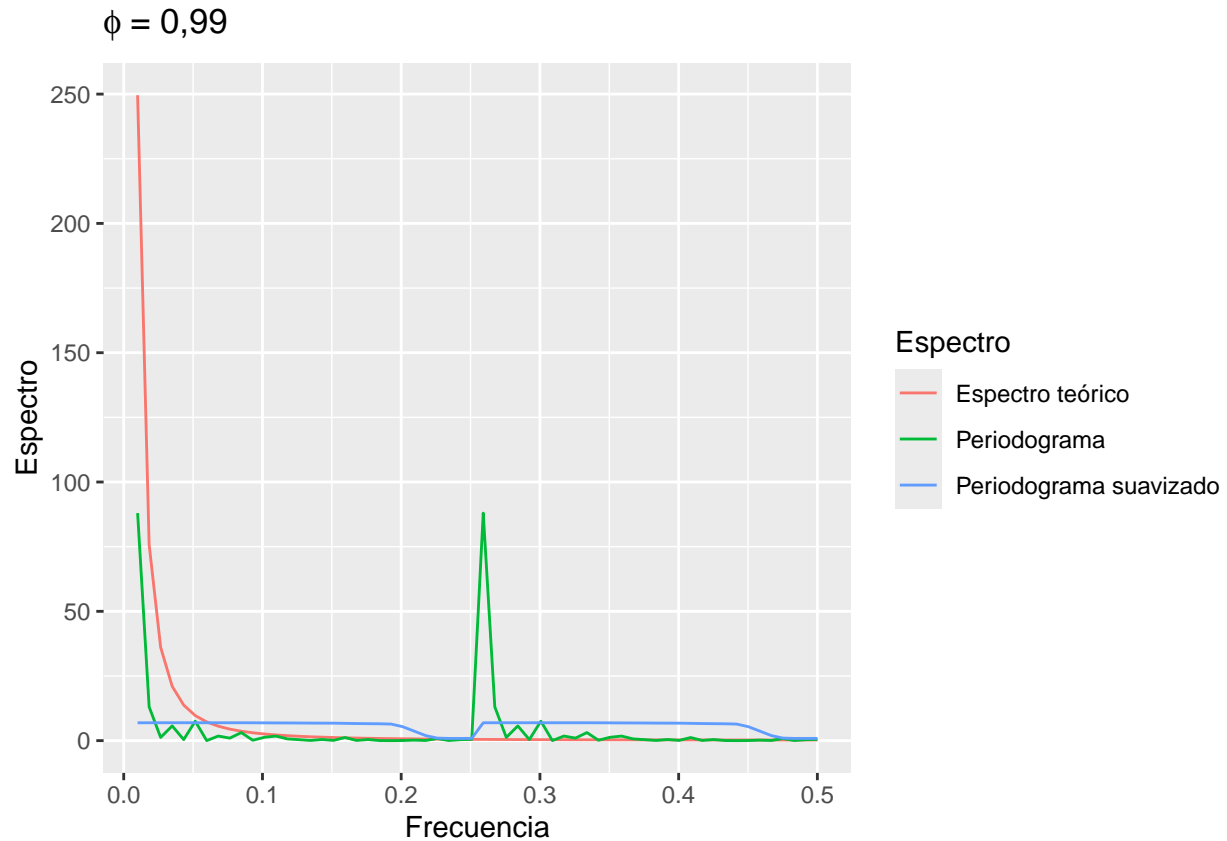


Figura 7: Espectro teórico y periodogramas de un proceso AR(3) con coeficiente igual a 0,99.

4. Práctico 4 - Ejercicio 5

```
# Definimos el valor del parámetro del proceso
theta <- -0.5 # Notación inversa a la del curso

# Obtenemos el espectro teórico del proceso MA(1) (escalado entre 0 y 0,5)
# Cada valor del espectro se multiplica por 2*pi
espectro_MA <- ARMAspec(model = list(ma = theta),
  plot = FALSE,
  freq = seq(from = 0,
    to = 0.5,
    length.out = 500))

# Obtenemos el espectro teórico del Ruido Blanco (escalado entre 0 y 0,5)
espectro_RB <- ARMAspec(model = list(order = c(0,0,0)),
  plot = FALSE,
  freq = seq(from = 0,
    to = 0.5,
    length.out = 500))

# Guardamos los datos para poder graficar en ggplot()
```

```

espectro <- data.frame(Frecuencia = espectro_MA$freq,
                      Espectro_MA = espectro_MA$spec,
                      Espectro_RB = espectro_RB$spec) %>%
  pivot_longer(cols = c(Espectro_MA, Espectro_RB),
               names_to = "Espectro",
               values_to = "valores") %>%
  mutate(Espectro = recode(Espectro,
                           "Espectro_MA" = "Espectro MA(1)",
                           "Espectro_RB" = "Espectro Ruido Blanco"))

# Graficamos el espectro teórico
ggplot(espectro) +
  geom_line(aes(x = Frecuencia, y = valores, color = Espectro)) +
  ggtitle(expression(theta ~ "= 0,5"))

```

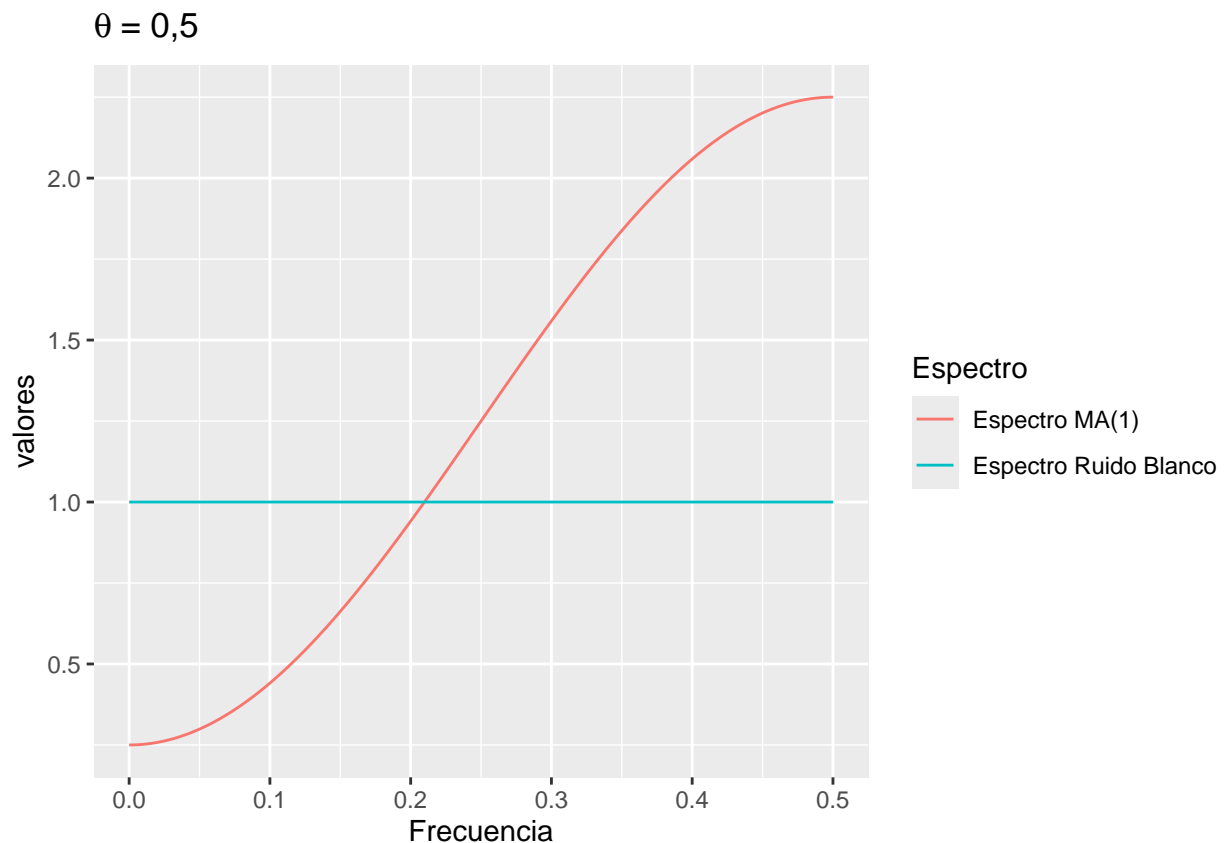


Figura 8: Espectro teórico de un proceso MA(1) con coeficiente igual a 0,5 y de un Ruido Blanco.

5. Práctico 4 - Ejercicio 6

```

# Función de transferencia de la diferencia estacional
transferencia <- data.frame(w = seq(0, pi, length.out = 500))
transferencia <- transferencia %>%

```

```
mutate(cos = cos(12*w),  
       f = 2*(1 - cos))
```

```
# Graficamos la función de transferencia  
ggplot(transferencia) +  
  geom_line(aes(x = w, y = f)) +  
  labs(y = "f(w)") +  
  geom_hline(yintercept = 1, color = "red")
```

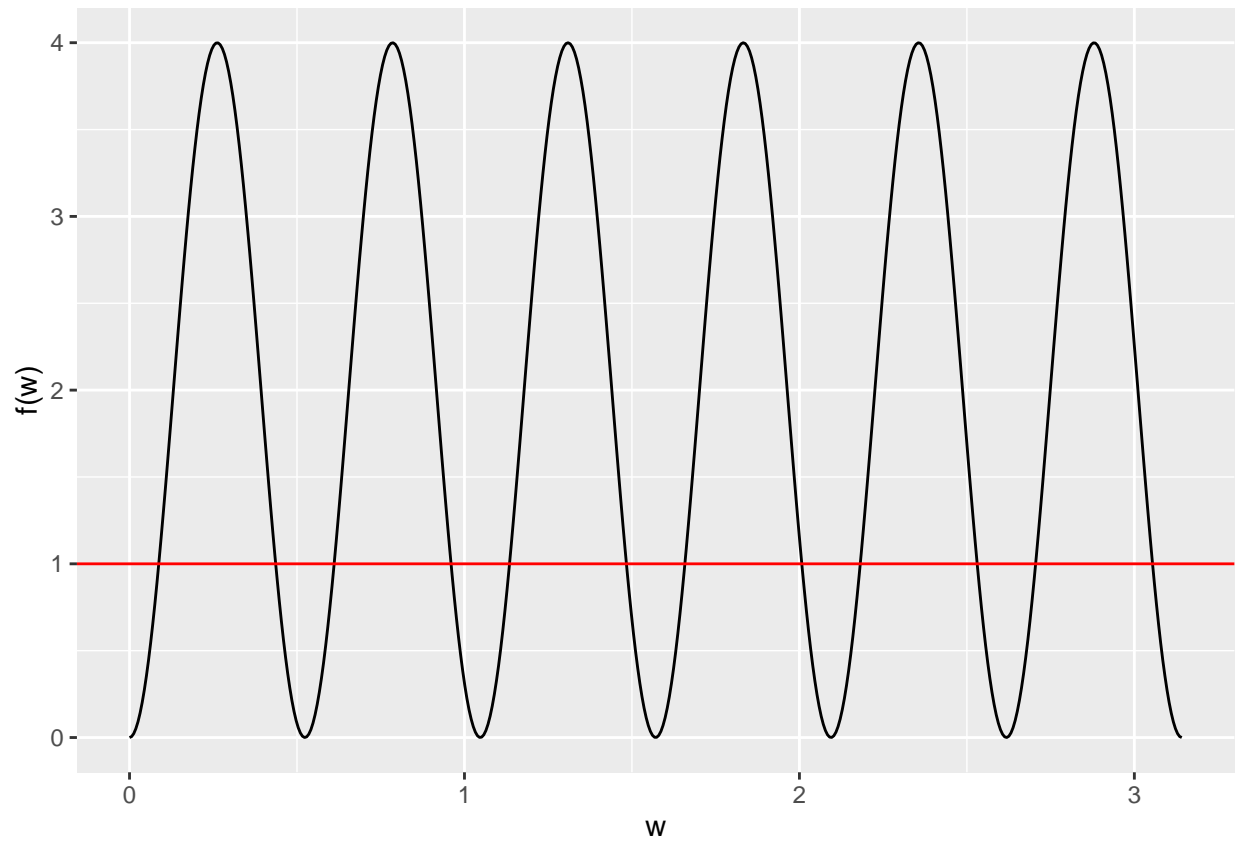


Figura 9: Función de transferencia para la diferencia estacional de una serie mensual.