Homework #1

Dustin Leatherman

1/11/2020

1.5

For a moving average process of the form $x_t = w_{t-1} + 2w_t + w_{t+1}$

where w_t are independent with zero means and variance σ_w^2 , determine the autocovariance and autocorrelation functions as a function of lag h and sketch the ACF as a function of h.

$$\gamma(\hat{h}) = cov(x_t, x_{t+h}) = E(x_t \times x_{t+h})$$
$$= E[(w_{t-1} + 2w_t + w_{t+1})(w_{t+h-1} + 2w_{t+h} + w_{t+h+1})]$$
$$= E[w_{t-1}w_{t+h-1} + 2w_{t-1}w_{t+h} + w_{t-1}w_{t+h+1} + 2w_tw_{t+h-1} + 4w_tw_{t+h} + 2w_tw_{t+h+1} + w_{t+1}w_{t+h-1} + 2w_{t+1}w_{t+h} + w_{t+}$$

$\rho(\hat{h}) = \frac{\gamma(\hat{h})}{\gamma(\hat{0})} = \frac{\hat{\gamma}(h)}{\sigma_w^2}$

Note the following definitions

$$E(w_t^2) = var(w_t) + E(w_t)^2$$
$$E(w_t) = 0$$
$$E(w_tw_s) = 0, \text{ where } s \neq t$$
$$var(w_t) = \sigma_w^2$$
$$E(w_t^2) = \sigma_w^2$$

In the interest of brevity and readability, the following Expectations only contain non-zero terms.

When |h| = 0:

$$E[x_t^2] = E[w_{t-1}^2 + 4w_t^2 + w_{t+1}^2] = \sigma_w^2 + 4\sigma_w^2 + \sigma_w^2 = 6\sigma_w^2$$

When |h| = 1:

$$E[x_tx_{t+1}] = E[2w_t^2 + 2w_{t+1}^2] = 4\sigma_w^2$$

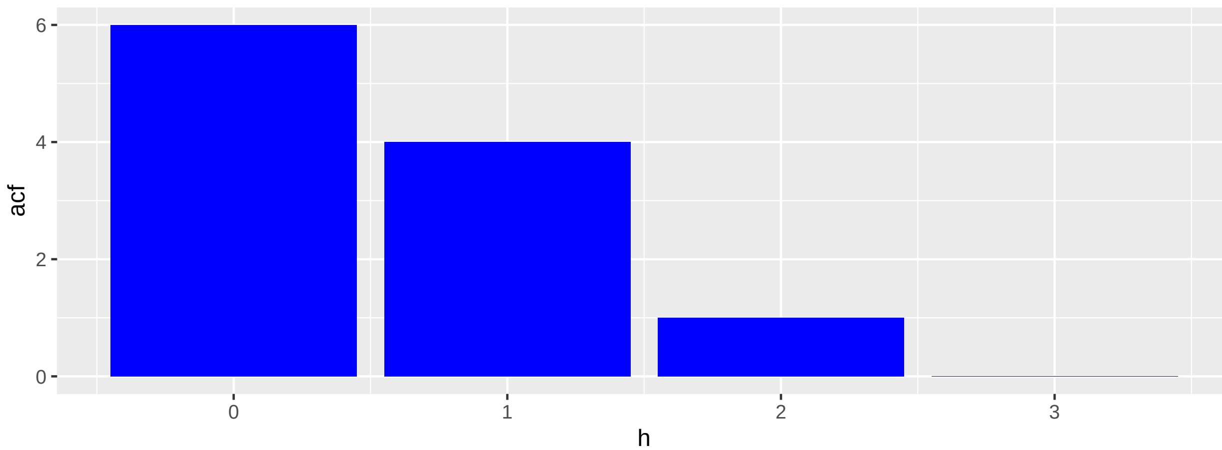
When |h| = 2:

$$E[x_tx_{t+2}] = E(w_{t+1}^2) = \sigma_w^2$$

When |h| = 3:

$$E[x_tx_{t+3}] = 0$$

```
data <- data.frame(h = c(0,1,2,3), acf = c(6,4,1,0))
ggplot(data, aes(x = h, y = acf)) +
  geom_bar(stat = "identity", fill = "blue")
```



1.12

> Let w_t , for $t = 0, \pm 1, \pm 2, \ldots$ \$ be a normal white noise process, and consider the series $x_t = w_t w_{t-1}$ Determine the mean and autocovariance function of x_t, and state whether it is stationary.

Given $w_t \sim ind\ N(0, \sigma_w^2)$.

$$\mu_{x_t} = E(x_t) = E(w_t w_{t-1}) = E(w_i) * E(w_{t-1}) = 0$$

$$\begin{aligned}\hat{\gamma}(h) &= E(x_t \times x_{t+h}) \\ &= E(w_t w_{t-1} w_{t+h} w_{t+h-1}) \\ &= E(w_t w_{t-1}) \times E(w_{t+h} w_{t+h-1}) \\ &= 0\end{aligned}$$

Since this time series is the product of two stationary time series, it is considered stationary. This is further proven by the mean and covariance functions being independent of time.

1.14

> (a) Simulate a series of n = 500 Gaussian white noise observations as in Example 1.6 and compute the sample ACF, $\hat{\rho}(h)$ to lag 20. Compare the sample ACF you obtain to the actual ACF, $\rho(h)$. [Recall Example 1.18.]
> (b) Repeat part (a) using only n = 50. How does changing n affect the results?

a

```
set.seed(123)

w500 <- rnorm(500)

w500.acf <-
  bind_cols(
  acf(w500, type="correlation", plot = FALSE) %>%
```

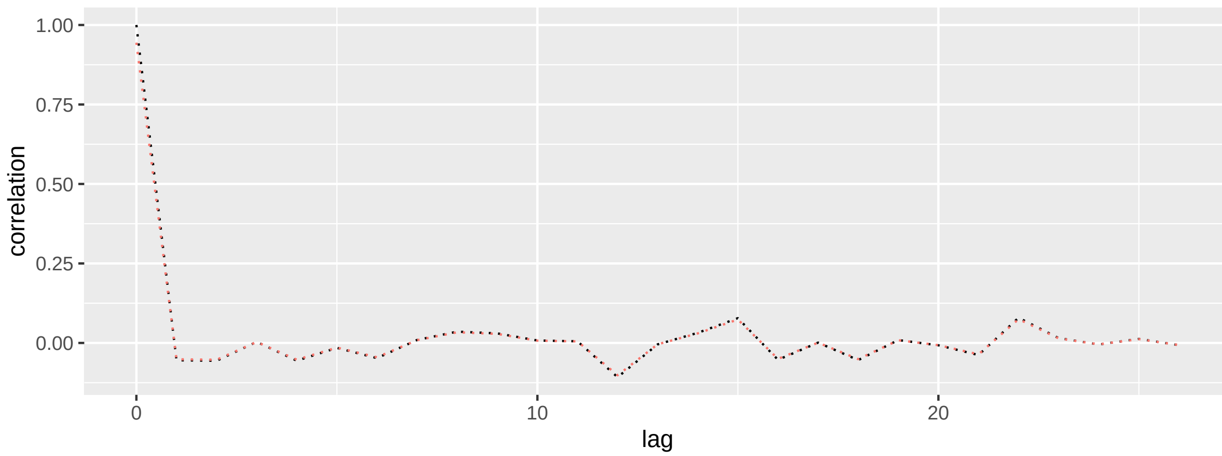
```r
    tidy %>%
    select(lag, correlation = acf),
  acf(w500, type="covariance", plot = FALSE) %>%
    tidy %>%
    select(covariance = acf)
)

w500.acf %>%
  filter(lag <= 20) %>%
  kable(
    caption = "First 20 lags of 500 random Normal observations"
  ) %>%
  kable_styling(full_width = FALSE, protect_latex = TRUE, latex_options = "hold_position")
```

Table 1: First 20 lags of 500 random Normal observations

lag	correlation	covariance
0	1.0000000	0.9443877
1	-0.0544080	-0.0513823
2	-0.0559941	-0.0528801
3	0.0034130	0.0032232
4	-0.0557653	-0.0526641
5	-0.0154724	-0.0146120
6	-0.0477022	-0.0450494
7	0.0094094	0.0088861
8	0.0355565	0.0335792
9	0.0301090	0.0284346
10	0.0076819	0.0072547
11	0.0052141	0.0049241
12	-0.1076564	-0.1016694
13	-0.0043968	-0.0041523
14	0.0318318	0.0300616
15	0.0779817	0.0736450
16	-0.0529327	-0.0499890
17	0.0009872	0.0009323
18	-0.0538619	-0.0508665
19	0.0090796	0.0085747
20	-0.0073696	-0.0069597

```r
ggplot(w500.acf, aes(x = lag)) +
  geom_line(aes(y = correlation), linetype = "dotted") +
  geom_line(aes(y = covariance, color = "red"), linetype = "dotted", show.legend = FALSE)
```



The actual correlation coefficient for Normal White Noise is equal to the covariance between a given time point and lag h. The values are very close as seen in the table and the graph.

b

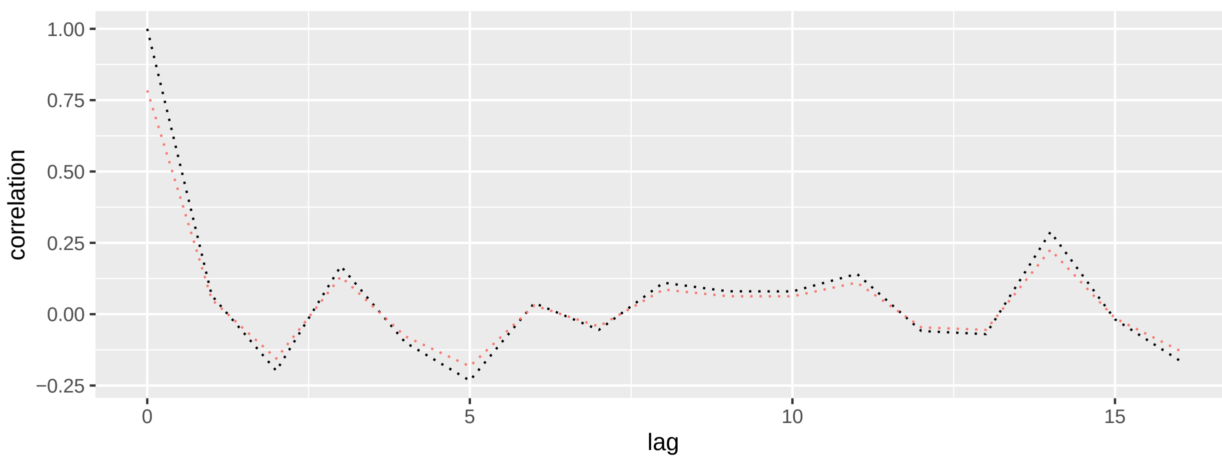
```
w50 <- rnorm(50)
w50.acf <-
  bind_cols(
  acf(w50, type="correlation", plot = FALSE) %>%
    tidy %>%
    select(lag, correlation = acf),
  acf(w50, type="covariance", plot = FALSE) %>%
    tidy %>%
    select(covariance = acf)
)

w50.acf %>%
  filter(lag <= 20) %>%
  kable(
    caption = "Lags of 50 random Normal observations"
  ) %>%
  kable_styling(full_width = FALSE, protect_latex = TRUE, latex_options = "hold_position")
```

```
ggplot(w50.acf, aes(x = lag)) +
  geom_line(aes(y = correlation), linetype = "dotted") +
  geom_line(aes(y = covariance, color = "red"), linetype = "dotted", show.legend = FALSE)
```

Table 2: Lags of 50 random Normal observations

lag	correlation	covariance
0	1.0000000	0.7843009
1	0.0654295	0.0513164
2	-0.1984148	-0.1556169
3	0.1683970	0.1320739
4	-0.0991071	-0.0777298
5	-0.2328667	-0.1826376
6	0.0390622	0.0306365
7	-0.0549713	-0.0431140
8	0.1102303	0.0864537
9	0.0801589	0.0628687
10	0.0797302	0.0625325
11	0.1412409	0.1107754
12	-0.0590652	-0.0463249
13	-0.0698747	-0.0548028
14	0.2882476	0.2260728
15	-0.0181276	-0.0142175
16	-0.1623468	-0.1273287



Changing n reduces the number of lags that are created and the sample correlation drifts further from the actual correlation. This is an expected effect when using smaller sample sizes.