Homework 6

Brian Detweiler

March 11, 2017

1. Consider three seperate AR(1) models: $\phi = 0.1, \phi = 0.5$, and $\phi = 0.8$.

(a) For each model, calculate ρ_1 and ρ_7.

$$Y_t = 0.1Y_{t-1} - e_t$$

We must find $\rho_k = \frac{\gamma_k}{\gamma_0}$. First we need γ_0 in each case.

$$
\begin{aligned}
\gamma_0 = Cov(Y_t, Y_t) &= Var(Y_t) \\
&= Var(0.1Y_{t-1} - e_t) \\
&= 0.01Var(Y_{t-1}) + Var(e_t) \\
&= 0.01\gamma_0 + \sigma_e^2 \\
\gamma_0 - 0.01\gamma_0 &= \sigma_e^2 \\
0.99\gamma_0 &= \sigma_e^2 \\
\gamma_0 &= \frac{\sigma_e^2}{0.99}
\end{aligned}
$$

More generally, $\gamma_0 = \frac{\sigma_e^2}{1 - \phi_1^2}$.

We will expand the Yule-Walker equations,

$$
\begin{aligned}
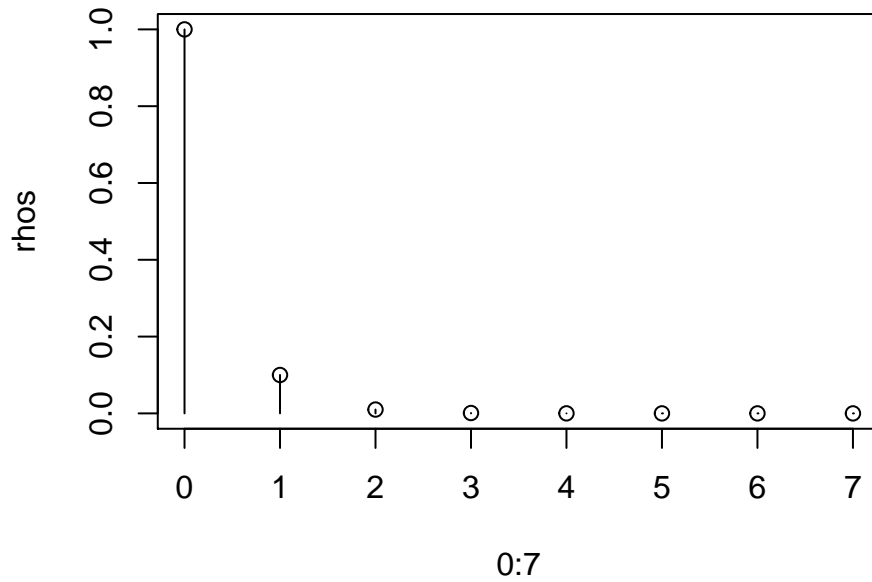\gamma_k &= \phi_1 \gamma_{k-1} \\
\rho_k &= \phi_1 \rho_{k-1}
\end{aligned}
$$

So we now have

$$\gamma_1 = \phi_1 \gamma_0$$
$$= 0.1\gamma_0$$
$$= \frac{0.1\sigma_e^2}{0.99}$$
$$= 0.1010101\sigma_e^2$$
$$\rho_1 = \frac{\gamma_1}{\gamma_0}$$
$$= \frac{0.1010101\sigma_e^2}{\frac{\sigma_e^2}{0.99}}$$
$$= 0.1010101(0.99)$$
$$= 0.1$$
$$= \phi_1$$
$$\rho_2 = \phi_1 \rho_1$$
$$\rho_2 = 0.1(0.1)$$
$$\rho_2 = \phi_1^2$$

Now we can see the pattern emerging. $\rho_k = \phi_1^k$, and hence $\rho_7 = \phi_1^7 = (0.1)^7$.

The autocorrelation function shows a decaying sequence.

ACF for first 7 lags for phi_1 = 0.1

Now, since we've generalized it, we can easily apply this to the other models.

$$Y_t = 0.5Y_{t-1} - e_t$$

$$
\begin{aligned}
\gamma_0 &= \frac{\sigma_e^2}{1 - \phi_1^2} \\
&= \frac{\sigma_e^2}{1 - 0.5} \\
&= \frac{\sigma_e^2}{0.5} \\
\gamma_1 &= \phi_1 \gamma_0 \\
&= 0.5\gamma_0 \\
&= \frac{0.5\sigma_e^2}{0.5} \\
&= \sigma_e^2 \\
\rho_1 &= \frac{\gamma_1}{\gamma_0} \\
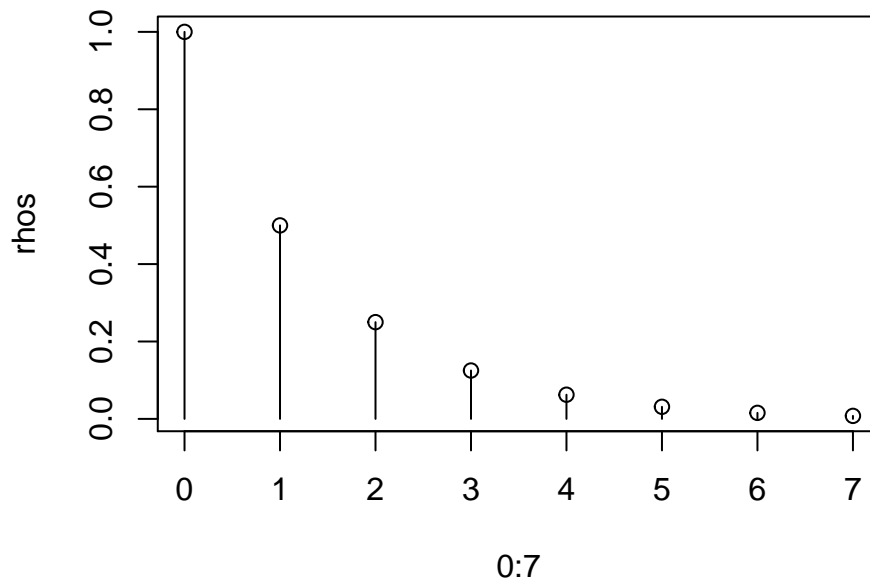&= \frac{\sigma_e^2}{\frac{\sigma_e^2}{0.5}} \\
&= 0.5 \\
&= \phi_1 \\
\rho_2 &= \phi_1 \rho_1 \\
\rho_2 &= 0.5(0.5) \\
\rho_2 &= \phi_1^2 \\
\rho_7 &= \phi_1^7 = 0.5^7
\end{aligned}
$$

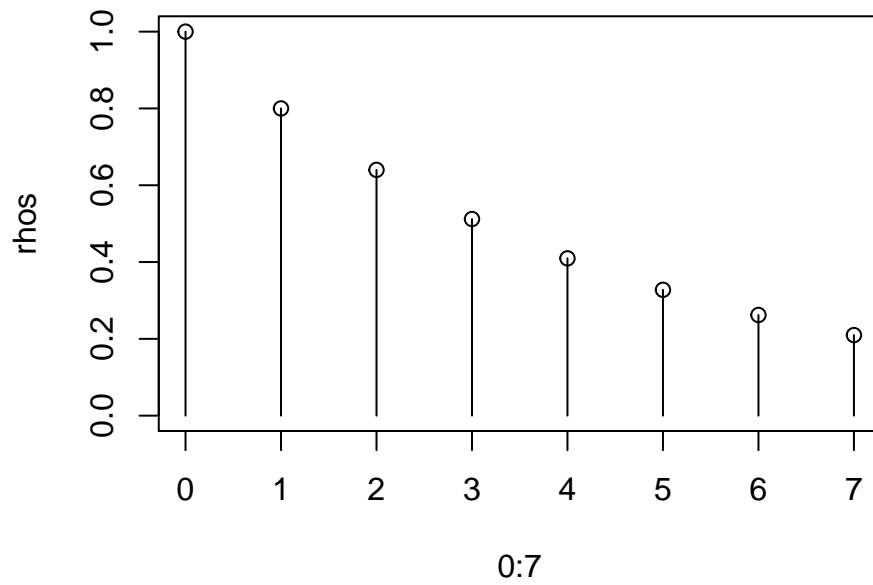## ACF for first 7 lags for phi_1 = 0.5

$$Y_t = 0.8Y_{t-1} - e_t$$

$$\rho_1 = \phi_1 = 0.8$$
$$\rho_7 = \phi_1^7 = 0.2097152$$

ACF for first 7 lags for phi_1 = 0.8

(b) For each model, calculate $Var(r_1)$ and $Var(r_7)$.

We define the sample ACF as

$$r_k = \frac{\sum_{t=k+1}^{n}(Y_t - \overline{Y})(Y_{t-k} - \overline{Y})}{\sum_{t=1}^{n}(Y_t - \overline{Y})^2}$$

For an AR(1) process with $\rho_k = \phi^k$ for $k > 0$,

$$Var(r_k) \approx \frac{1}{n}\left[\frac{(1+\phi^2)(1-\phi^{2k})}{1-\phi^2} - 2k\phi^{2k}\right]$$

We'll create a helper function to calculate the numerical parts of the variance in `R`:

```
partial.var.rk <- function(phi, k) {
  return(((1 + phi^2) * (1 - phi^(2 * k)) / (1 - phi^2) - 2 * k * phi^(2 * k)))
}
```

So for the given models and $k = 1, 7$, we have

Model 1, $\phi_1 = 0.1$

$$Var(r_1) \approx \frac{1}{n}\left[\frac{(1+(0.1)^2)(1-(0.1)^2)}{1-(0.1)^2} - 2(0.1)^2\right]$$
$$\approx \frac{0.99}{n}$$
$$Var(r_7) \approx \frac{1}{n}\left[\frac{(1+(0.1)^2)(1-(0.1)^{14})}{1-(0.1)^2} - 14(0.1)^{14}\right]$$
$$\approx \frac{1.020202}{n}$$

Model 2, $\phi_1 = 0.5$

$$Var(r_1) \approx \frac{1}{n}\left[\frac{(1+(0.5)^2)(1-(0.5)^2)}{1-(0.5)^2} - 2(0.5)^2\right]$$
$$\approx \frac{0.75}{n}$$
$$Var(r_7) \approx \frac{1}{n}\left[\frac{(1+(0.5)^2)(1-(0.5)^{14})}{1-(0.5)^2} - 14(0.5)^{14}\right]$$
$$\approx \frac{1.6657104}{n}$$

5

Model 3, $\phi_1 = 0.8$

$$Var(r_1) \approx \frac{1}{n}\left[\frac{(1+(0.8)^2)(1-(0.8)^2)}{1-(0.8)^2} - 2(0.8)^2\right]$$

$$\approx \frac{0.36}{n}$$

$$Var(r_7) \approx \frac{1}{n}\left[\frac{(1+(0.8)^2)(1-(0.8)^{14})}{1-(0.8)^2} - 14(0.8)^{14}\right]$$

$$\approx \frac{3.7394736}{n}$$

(c) For each model, use the `arima.sim` function to simulate a time series of length $n = 60$. Then use the `acf` function to calculate r_1 and r_7. Remember to set up a random seed for your simulation.
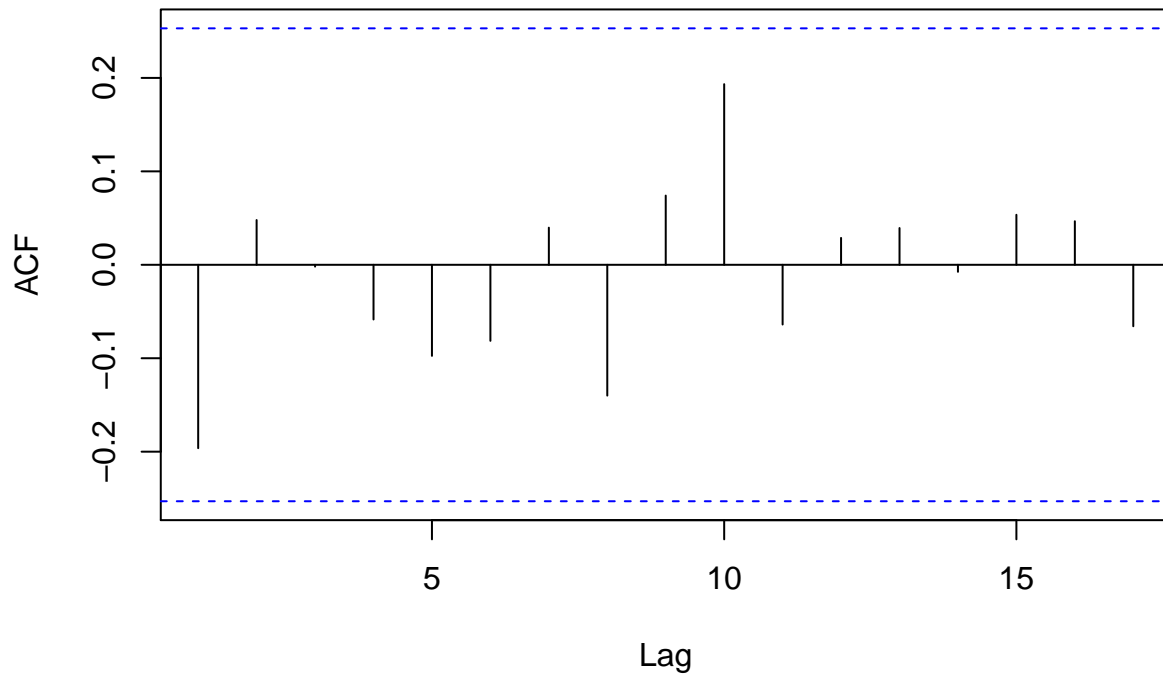
```
set.seed(0)

phi1 <- 0.1

model1.r1 <- c()
model1.r7 <- c()
for (i in 1:999) {
  sim <- arima.sim(n = 60, model = list(ar=(phi1)))
  r <- acf(sim, plot = FALSE)
  model1.r1 <- c(model1.r1, r[[1]][1])
  model1.r7 <- c(model1.r7, r[[1]][7])
}

r <- acf(sim)
```
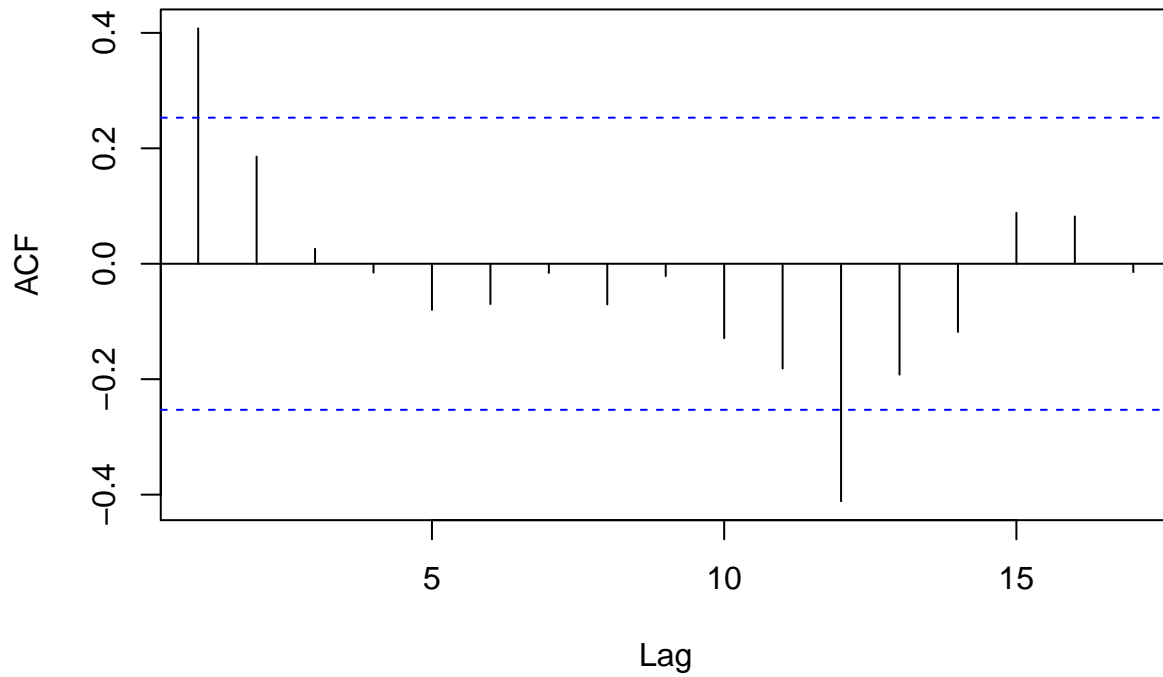
Series sim


```r
model1.r1 <- c(model1.r1, r[[1]][1])
model1.r7 <- c(model1.r7, r[[1]][7])

phi1 <- 0.5

model2.r1 <- c()
model2.r7 <- c()
for (i in 1:999) {
  sim <- arima.sim(n = 60, model = list(ar=(phi1)))
  r <- acf(sim, plot = FALSE)
  model2.r1 <- c(model2.r1, r[[1]][1])
  model2.r7 <- c(model2.r7, r[[1]][7])
}

r <- acf(sim)
```
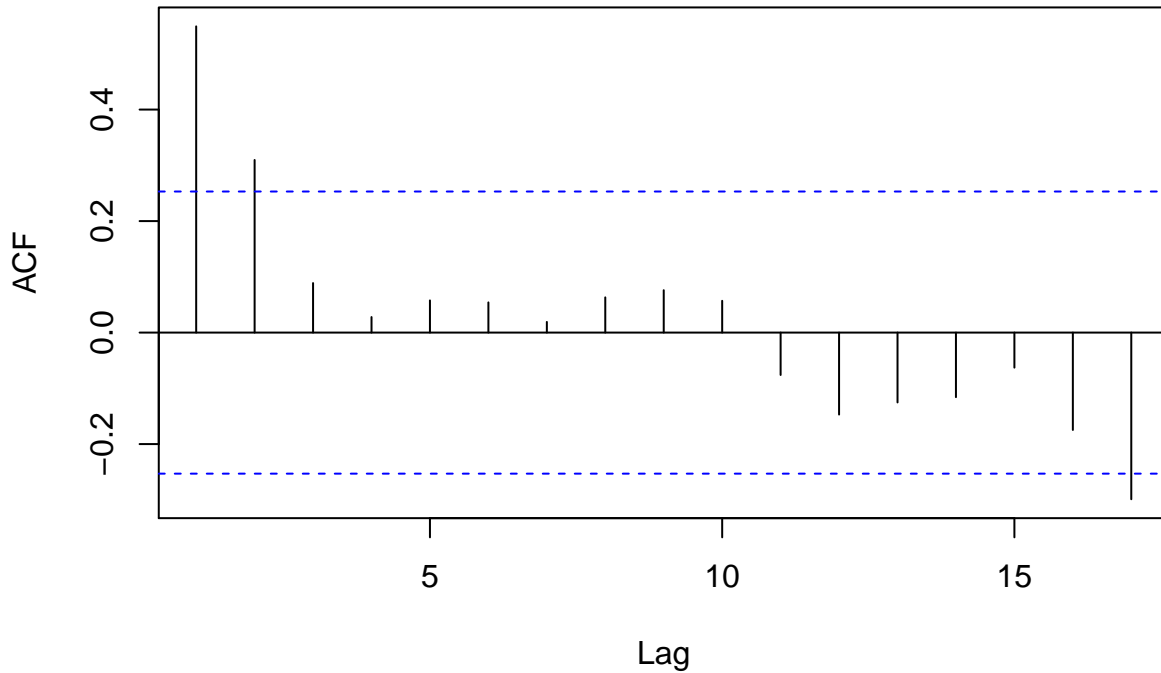
Series sim


```r
model2.r1 <- c(model2.r1, r[[1]][1])
model2.r7 <- c(model2.r7, r[[1]][7])

phi1 <- 0.8

model3.r1 <- c()
model3.r7 <- c()
for (i in 1:999) {
  sim <- arima.sim(n = 60, model = list(ar=(phi1)))
  r <- acf(sim, plot = FALSE)
  model3.r1 <- c(model3.r1, r[[1]][1])
  model3.r7 <- c(model3.r7, r[[1]][7])
}

r <- acf(sim)
```

Series sim


```
model3.r1 <- c(model3.r1, r[[1]][1])
model3.r7 <- c(model3.r7, r[[1]][7])
```

(d) Based on your results in parts (a) and (b), are r_1 and r_7 from part (c) within 2 standard deviations of ρ_1 and ρ_7 respectively?

$\phi = 0.1$

For an AR(1), we know that $c_{ii} = \frac{1}{n}\left[\frac{(1+\phi^2)(1-\phi^{2i})}{1-\phi^2} - 2i\phi^{2i}\right]$. Given the standard error of $r_k = \sqrt{\frac{c_{ii}}{n}}$. We've calculated these previously in part **(b)**, so we just need values for n.

We can investigate increasing values of n to to get an idea of how the variance behaves with more data. We'll use $n = 1, 100, 1000$.

Model 1, $\phi_1 = 0.1$

$$n = 1$$
$$SE(r_1) \approx 0.9949874$$
$$SE(r_7) \approx 1.0100505$$

$$n = 100$$
$$SE(r_1) \approx 0.0994987$$
$$SE(r_7) \approx 0.1010051$$

9

$$n = 1000$$
$$SE(r_1) \approx 0.0314643$$
$$SE(r_7) \approx 0.0319406$$

Model 2, $\phi_1 = 0.5$

$$n = 1$$
$$SE(r_1) \approx 0.8660254$$
$$SE(r_7) \approx 1.2906241$$

$$n = 100$$
$$SE(r_1) \approx 0.0866025$$
$$SE(r_7) \approx 0.1290624$$

$$n = 1000$$
$$SE(r_1) \approx 0.0273861$$
$$SE(r_7) \approx 0.0408131$$

Model 3, $\phi_1 = 0.8$

$$n = 1$$
$$SE(r_1) \approx 0.6$$
$$SE(r_7) \approx 1.9337719$$

$$n = 100$$
$$SE(r_1) \approx 0.06$$
$$SE(r_7) \approx 0.1933772$$

$$n = 1000$$
$$SE(r_1) \approx 0.0189737$$
$$SE(r_7) \approx 0.0611512$$

In all cases, the standard error falls within 2 standard deviations.

(e) Repeat part (c) for 1000 times. Draw histograms for r_1's and r_7's for each model. What proportion of r_1's and r_7's are within 2 standard deviations of ρ_1 and ρ_7?

Model 1, $\phi_1 = 0.1$
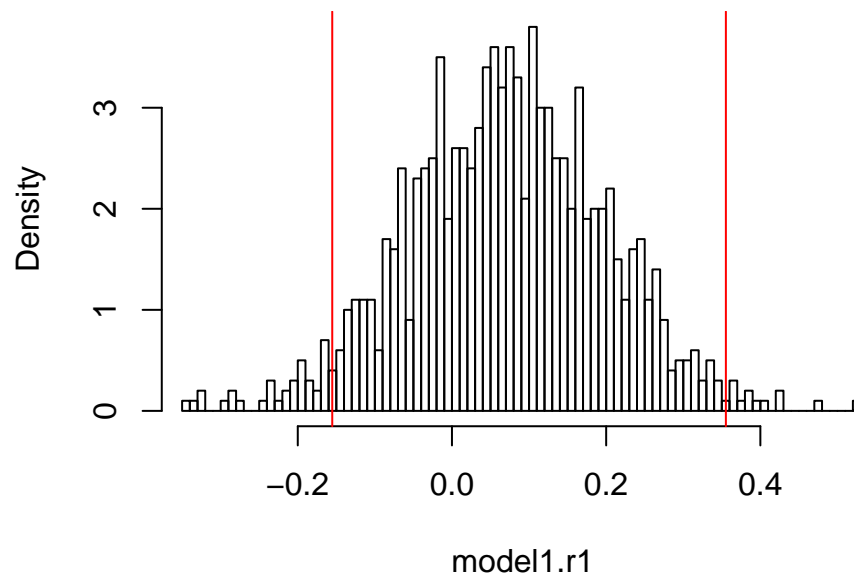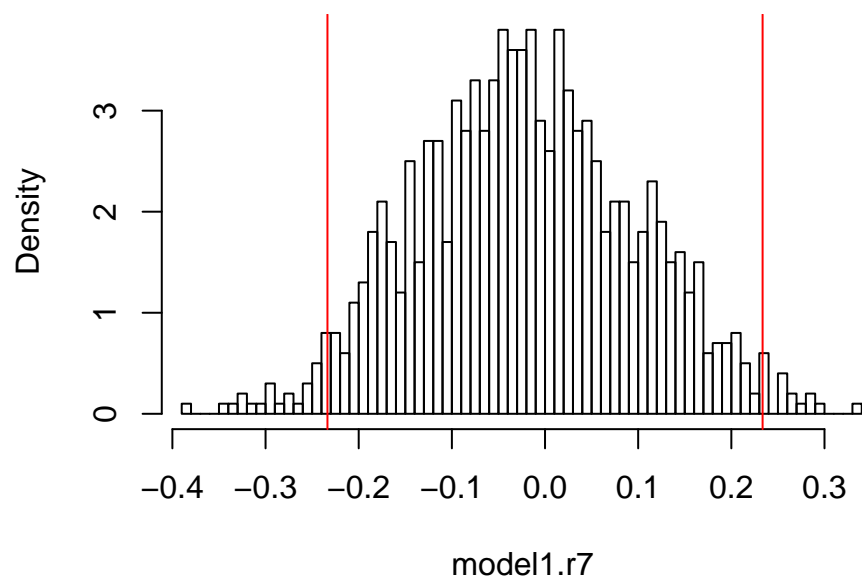
```
phi1 <- 0.1
lower <- phi1 - (2 * sd(model1.r1))
upper <- phi1 + (2 * sd(model1.r1))

within.sd <- (1 - sum((model1.r1 < lower) + (model1.r1 > upper)) / length(model1.r1)) * 100
hist(model1.r1, breaks=100, freq=F, xlim=c(min(lower, model1.r1), max(upper, model1.r1)))
abline(v = lower, col="red")
abline(v = upper, col="red")
```

Histogram of model1.r1

model1.r1

For r_1 in model 1, 95.2% fall within 2SE.

```
phi1 <- (0.1)^7
lower <- phi1 - (2 * sd(model1.r7))
upper <- phi1 + (2 * sd(model1.r7))

within.sd <- (1 - sum((model1.r7 < lower) + (model1.r7 > upper)) / length(model1.r7)) * 100
hist(model1.r7, breaks=100, freq=F, xlim=c(min(lower, model1.r7), max(upper, model1.r7)))
abline(v = lower, col="red")
abline(v = upper, col="red")
```
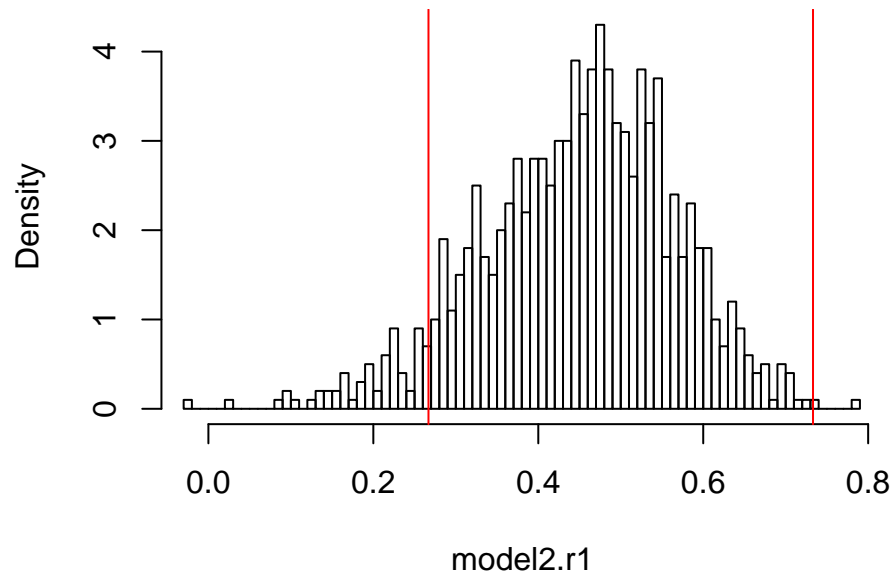
Histogram of model1.r7

For r_7 in model 1, 96.1% fall within 2SE.

Model 2, $\phi_1 = 0.5$

```r
phi1 <- 0.5
lower <- phi1 - (2 * sd(model2.r1))
upper <- phi1 + (2 * sd(model2.r1))

within.sd <- (1 - sum((model2.r1 < lower) + (model2.r1 > upper)) / length(model2.r1)) * 100
hist(model2.r1, breaks=100, freq=F, xlim=c(min(lower, model2.r1), max(upper, model2.r1)))
abline(v = lower, col="red")
abline(v = upper, col="red")
```
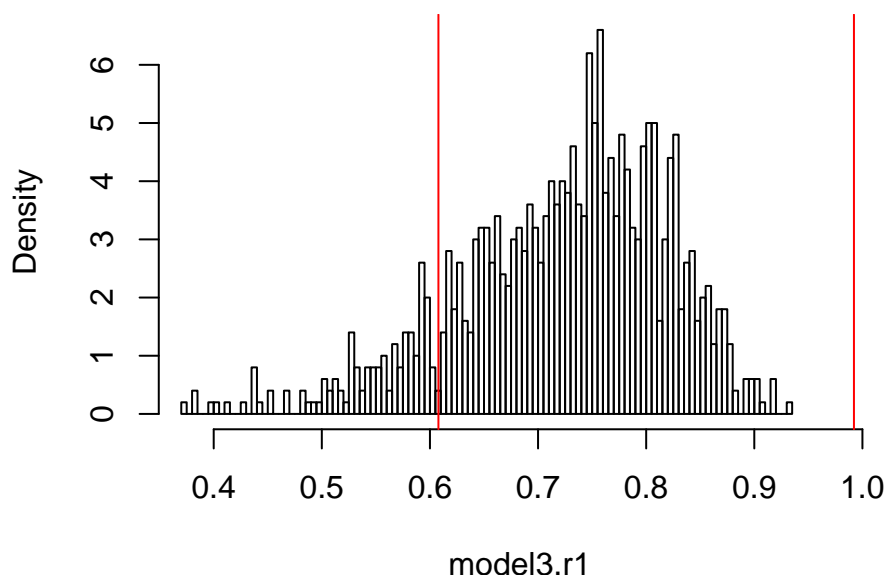
Histogram of model2.r1

For r_1 in model 2, 93.7% fall within 2SE.

```
phi1 <- (0.5)^7
lower <- phi1 - (2 * sd(model2.r7))
upper <- phi1 + (2 * sd(model2.r7))


within.sd <- (1 - sum((model2.r7 < lower) + (model2.r7 > upper)) / length(model2.r7)) * 100
hist(model2.r7, breaks=100, freq=F, xlim=c(min(lower, model2.r7), max(upper, model2.r7)))
abline(v = lower, col="red")
abline(v = upper, col="red")
```

Histogram of model2.r7
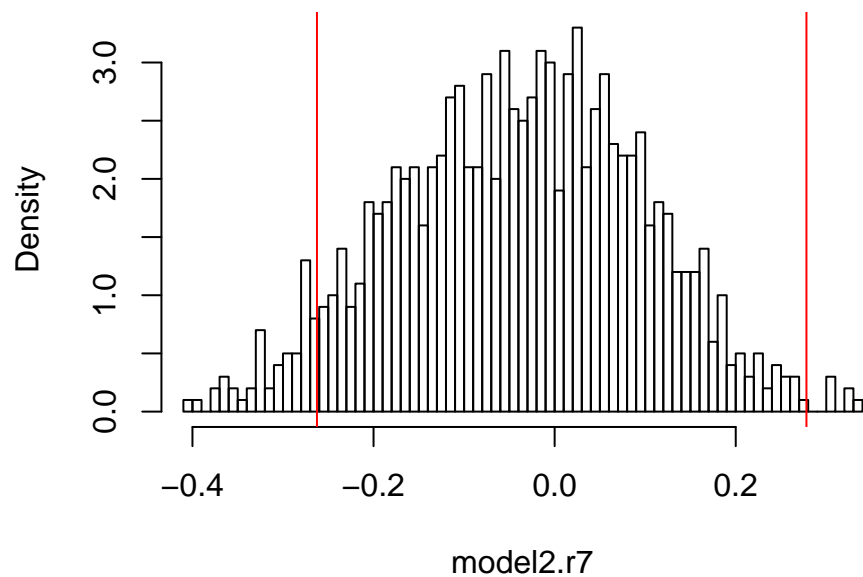
For r_7 in model 2, 93.9% fall within 2SE.

Model 3, $\phi_1 = 0.8$

```
phi1 <- 0.8
lower <- phi1 - (2 * sd(model3.r1))
upper <- phi1 + (2 * sd(model3.r1))

within.sd <- (1 - sum((model3.r1 < lower) + (model3.r1 > upper)) / length(model3.r1)) * 100
hist(model3.r1, breaks=100, freq=F, xlim=c(min(lower, model3.r1), max(upper, model3.r1)))
abline(v = lower, col="red")
abline(v = upper, col="red")
```

Histogram of model3.r1

For r_1 in model 3, 87.9% fall within 2SE.

```
phi1 <- (0.8)^7
lower <- phi1 - (2 * sd(model3.r7))
upper <- phi1 + (2 * sd(model3.r7))

within.sd <- (1 - sum((model3.r7 < lower) + (model3.r7 > upper)) / length(model3.r7)) * 100
hist(model3.r7, breaks=100, freq=F, xlim=c(min(lower, model3.r7), max(upper, model3.r7)))
abline(v = lower, col="red")
abline(v = upper, col="red")
```

Histogram of model3.r7
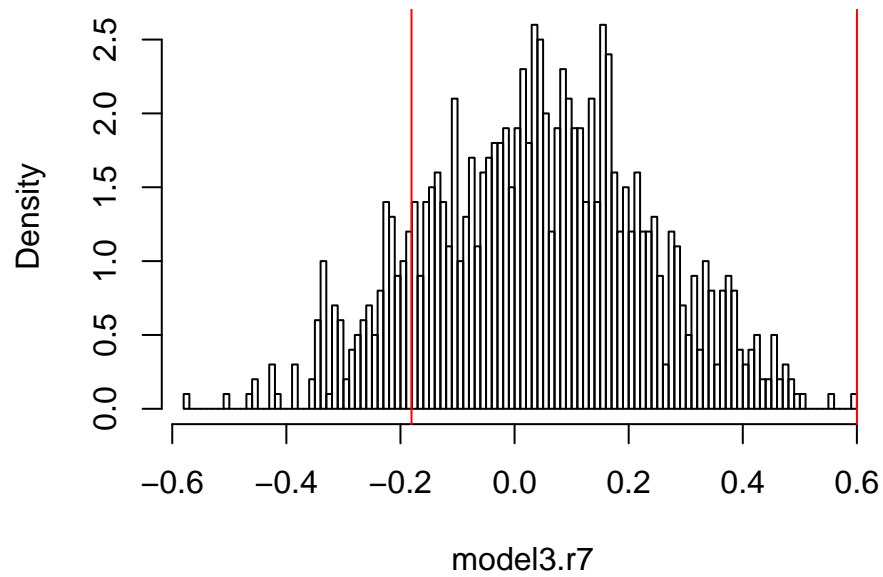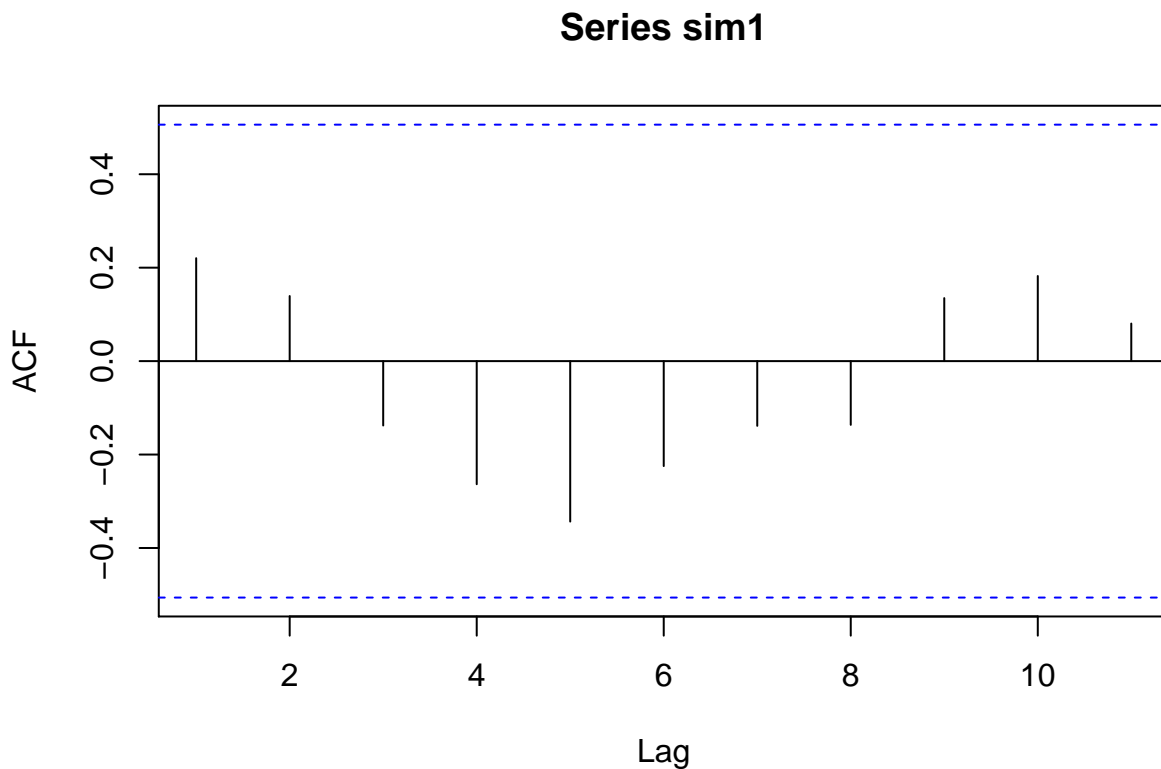

For r_7 in model 3, 86.2% fall within 2SE.

2. Consider an $AR(1)$ model with $\phi = 0.6$.

(a) Use the `arima.sim` function to simulate three time series of lengths $n = 15$, 75, and 100.
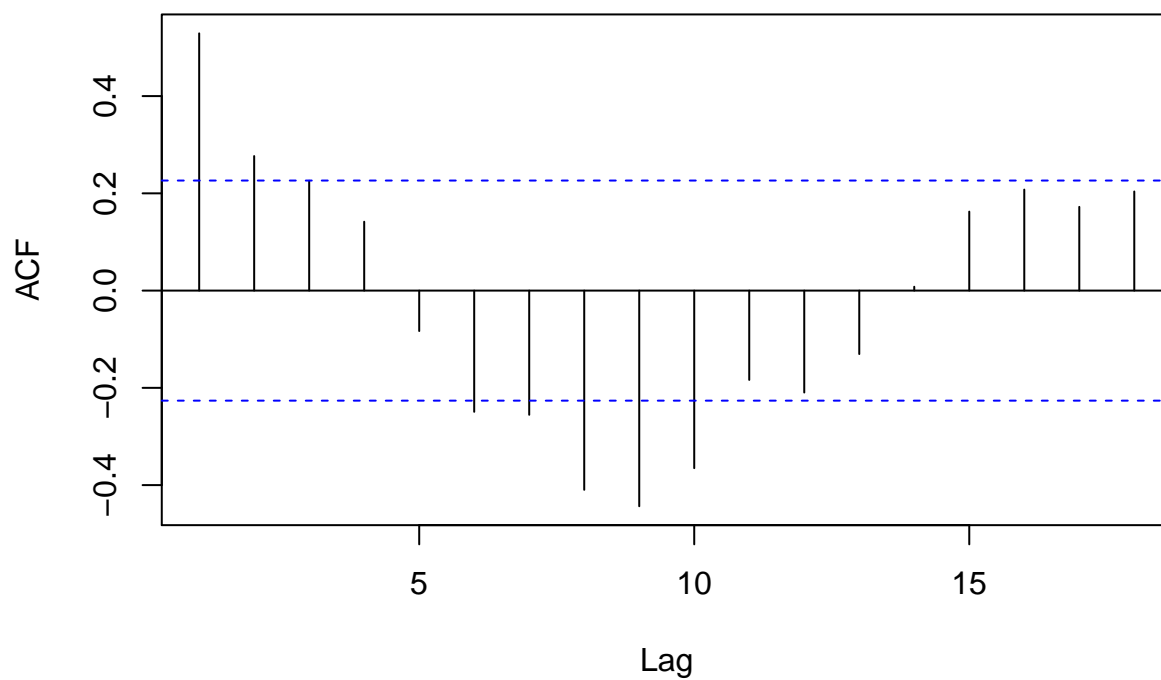
```
phi1 <- 0.6

sim1 <- arima.sim(n = 15, model = list(ar=(phi1)))
sim2 <- arima.sim(n = 75, model = list(ar=(phi1)))
sim3 <- arima.sim(n = 100, model = list(ar=(phi1)))

sim1.r <- acf(sim1)
```
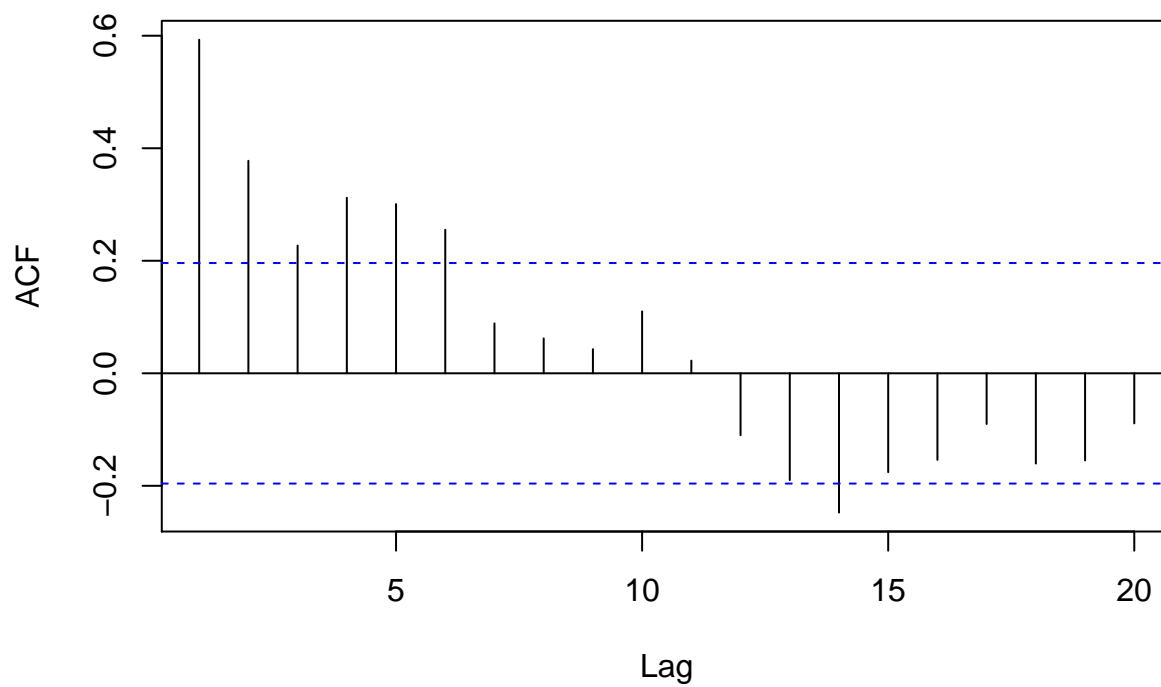
Series sim1


```
sim1.r1 <- sim1.r[[1]][1]
sim2.r <- acf(sim2)
```

Series sim2


```
sim2.r1 <- sim2.r[[1]][1]
sim3.r <- acf(sim3)
```

Series sim3


```
sim3.r1 <- sim3.r[[1]][1]
```

(b) For each set of simulated data, calculate r_1.

$n = 15$

```
sim1.r1
```

```
## [1] 0.2202908
```

$n = 75$

```
sim2.r1
```

```
## [1] 0.5290287
```

$n = 100$

```
sim3.r1
```

```
## [1] 0.5929305
```

(c) For each n, what is $Var(r_1)$? Is r_1 within 2 standard deviations of ρ_1 for each sample?

Using the function we defined earlier, `partial.var.rk`, we just need to substitute in our values for n.

$n = 15$

$$Var(r_1) = 0.0426667$$

$n = 75$

$$Var(r_1) = 0.0085333$$
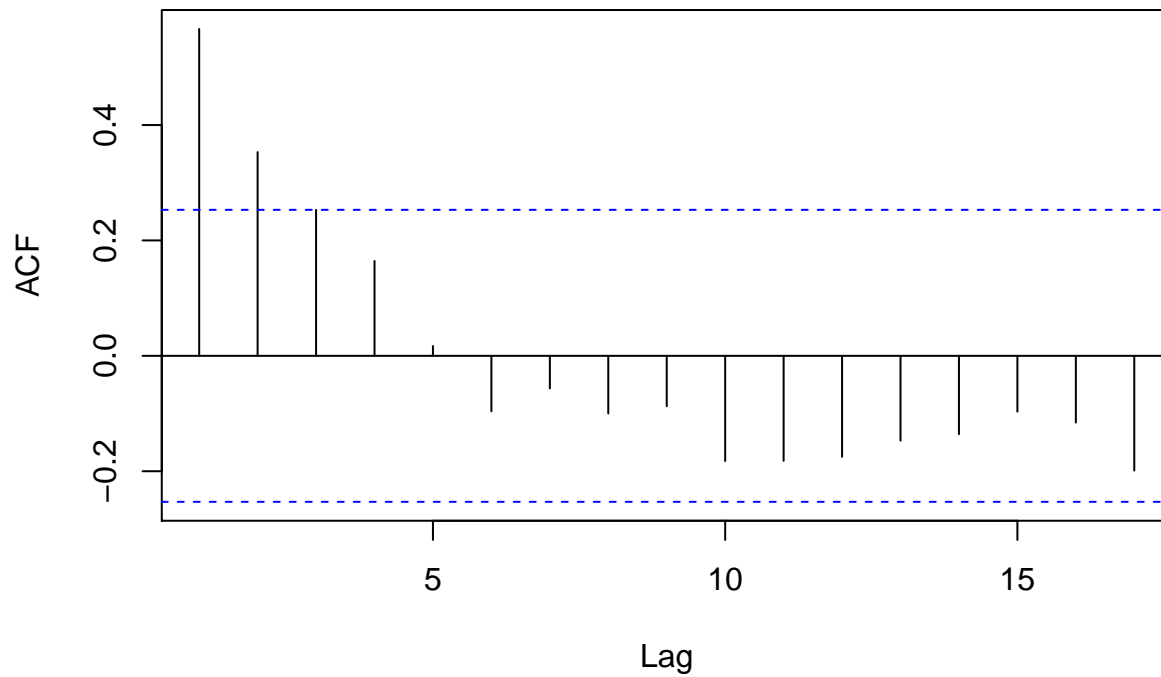
$n = 100$

$$Var(r_1) = 0.0064$$

(d) Repeat part (a) for 1000 times. For each n, draw a histogram of the 1000 r_1's, and find what proportion of r_1's are within 2 standard deviations of ρ_1.

```
phi1 <- 0.6

model4.r1 <- c()
for (i in 1:999) {
  sim <- arima.sim(n = 60, model = list(ar=(phi1)))
  r <- acf(sim, plot = FALSE)
  model4.r1 <- c(model4.r1, r[[1]][1])
```

```
}

r <- acf(sim)
```

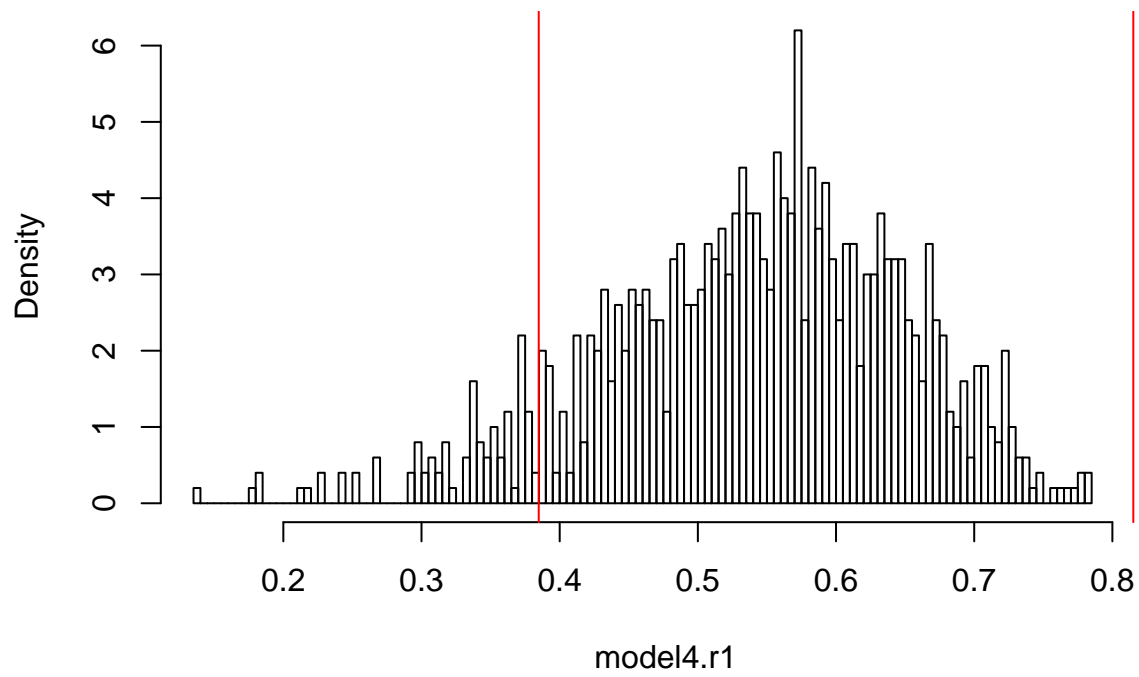## Series sim


```
model4.r1 <- c(model4.r1, r[[1]][1])

lower <- 0.6 - (2 * sd(model4.r1))
upper <- 0.6 + (2 * sd(model4.r1))

within.sd <- (1 - sum((model4.r1 < lower) + (model4.r1 > upper)) / length(model4.r1)) * 100

hist(model4.r1, breaks=100, freq=F, xlim=c(min(lower, model4.r1), max(upper, model4.r1)))
abline(v = lower, col="red")
abline(v = upper, col="red")
```

Histogram of model4.r1

91.5% of the observations fall within 2SE.

∎

3. Consider an MA(1) model with $\theta = 0.6$

(a) Use the `arima.sim` function to simulate three time series of lengths $n = 15$, 75, and 150. Note that **R** uses the negative of the MA coefficients.
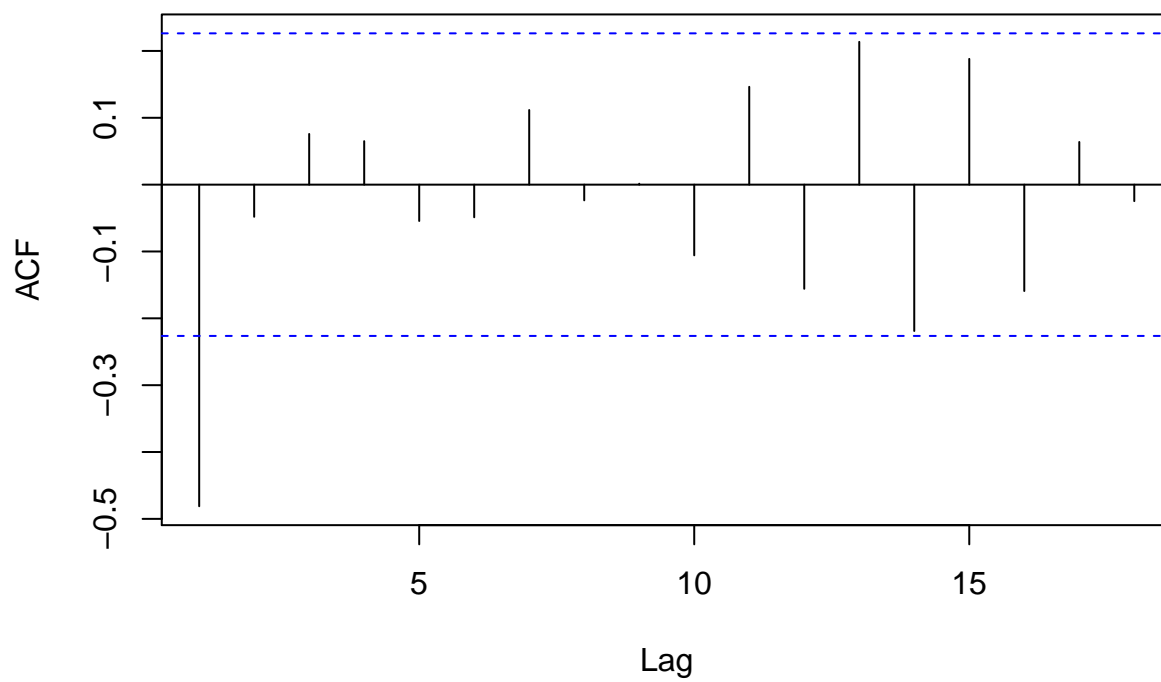
```
theta1 <- -0.6

sim1 <- arima.sim(n = 15, model = list(ma=(theta1)))
sim2 <- arima.sim(n = 75, model = list(ma=(theta1)))
sim3 <- arima.sim(n = 100, model = list(ma=(theta1)))

sim1.r <- acf(sim1)
```
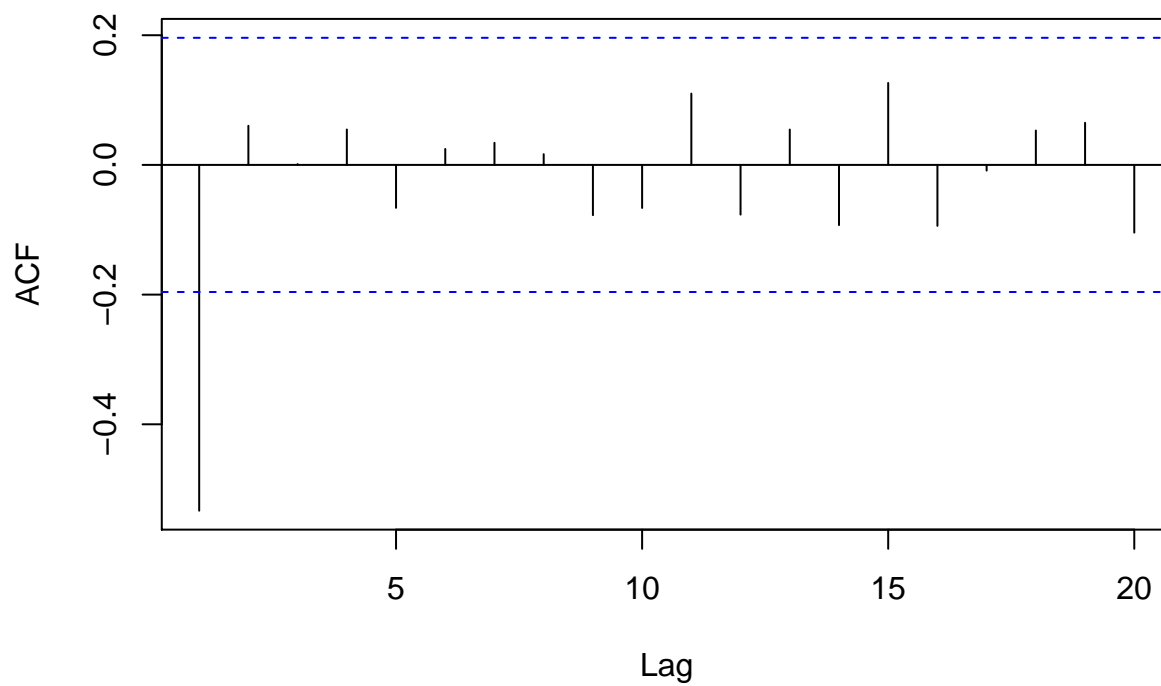
Series sim1


```
sim2.r <- acf(sim2)
```
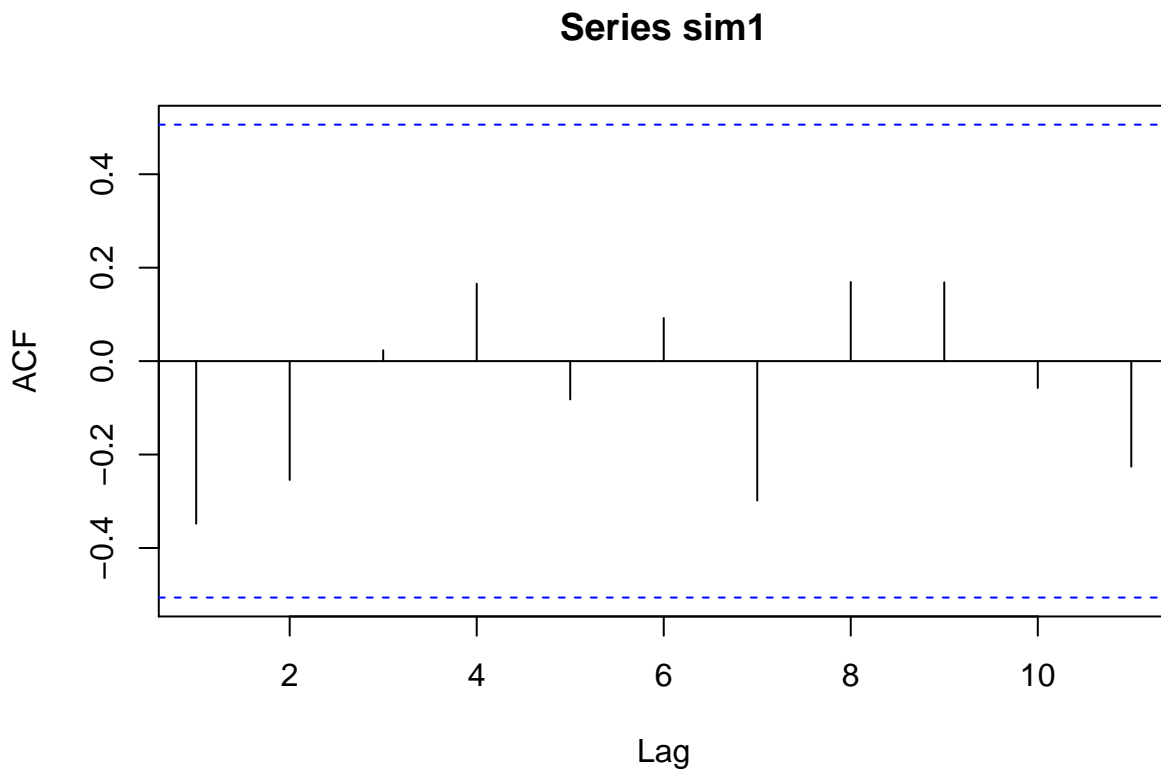
Series sim2


```r
sim3.r <- acf(sim3)
```

Series sim3

(b) For each set of simulated data, calculate r_1.

$n = 15$

```
sim1.r[[1]][1]
```

```
## [1] -0.3478818
```

$n = 75$

```
sim2.r[[1]][1]
```

```
## [1] -0.4811163
```

$n = 100$

```
sim3.r[[1]][1]
```

```
## [1] -0.5332678
```

(c) For each n, what is $Var(r_1)$? Is r_1 within 2 standard deviations of ρ_1 for each sample?

For an MA(1) model, $c_{11} = 1 - 3\rho_1^2 + 4\rho_1^4$.

Thus, $Var(r_1) = \frac{c_{11}}{n}$.

```
rho1 <- -0.6 / (1 + (0.6)^2)
```

For an MA(1) model, $\rho_1 = \frac{-\theta}{1+\theta^2}$. In this case, $\rho_1 = \frac{-0.6}{1+(0.6)^2} = -0.4411765$.

$$
\begin{aligned}
Var(r_1) = \frac{c_{11}}{n} &= \frac{1 - 3\rho_1^2 + 4\rho_1^4}{n} \\
&= \frac{1 - 3(-0.4411765)^2 + 4 - 0.4411765^4}{n} \\
&= \frac{0.5676237}{n}
\end{aligned}
$$

$n = 15$

$$
\begin{aligned}
Var(r_1) &= \frac{0.5676237}{15} \\
&= 0.0378416
\end{aligned}
$$

$n = 75$

$$
\begin{aligned}
Var(r_1) &= \frac{0.5676237}{75} \\
&= 0.0075683
\end{aligned}
$$

23

$n = 100$

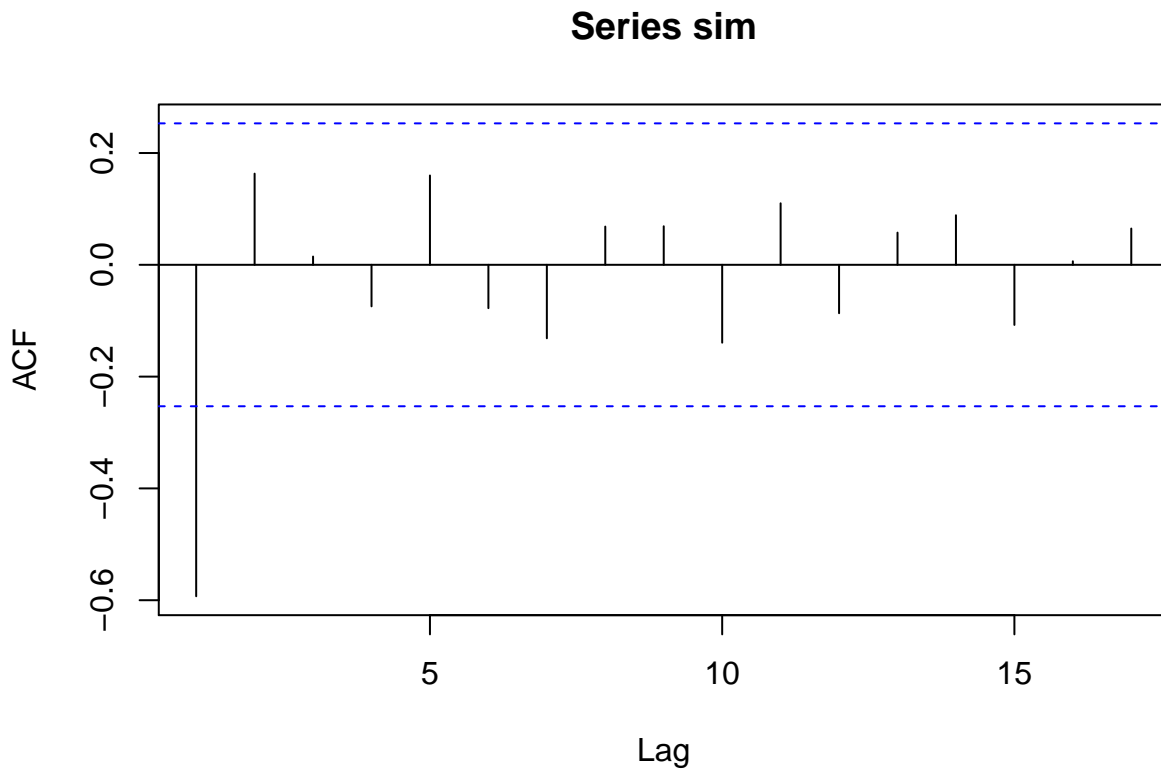$$Var(r_1) = \frac{0.5676237}{100}$$
$$= 0.0056762$$

(d) Repeat part (a) for 1000 times. For each n, draw a histogram of the 1000 r_1's, and find what proportion of r_1's are within 2 standard deviations of ρ_1.

```
theta1 <- -0.6

rho1 <- -0.6 / (1 + (0.6)^2)

model5.r1 <- c()
for (i in 1:999) {
  sim <- arima.sim(n = 60, model = list(ma=(theta1)))
  r <- acf(sim, plot = FALSE)
  model5.r1 <- c(model5.r1, r[[1]][1])
}

r <- acf(sim)
```

Series sim


```
model5.r1 <- c(model5.r1, r[[1]][1])

lower <- rho1 - (2 * sd(model5.r1))
upper <- rho1 + (2 * sd(model5.r1))

within.sd <- (1 - sum((model5.r1 < lower) + (model5.r1 > upper)) / length(model5.r1)) * 100
```
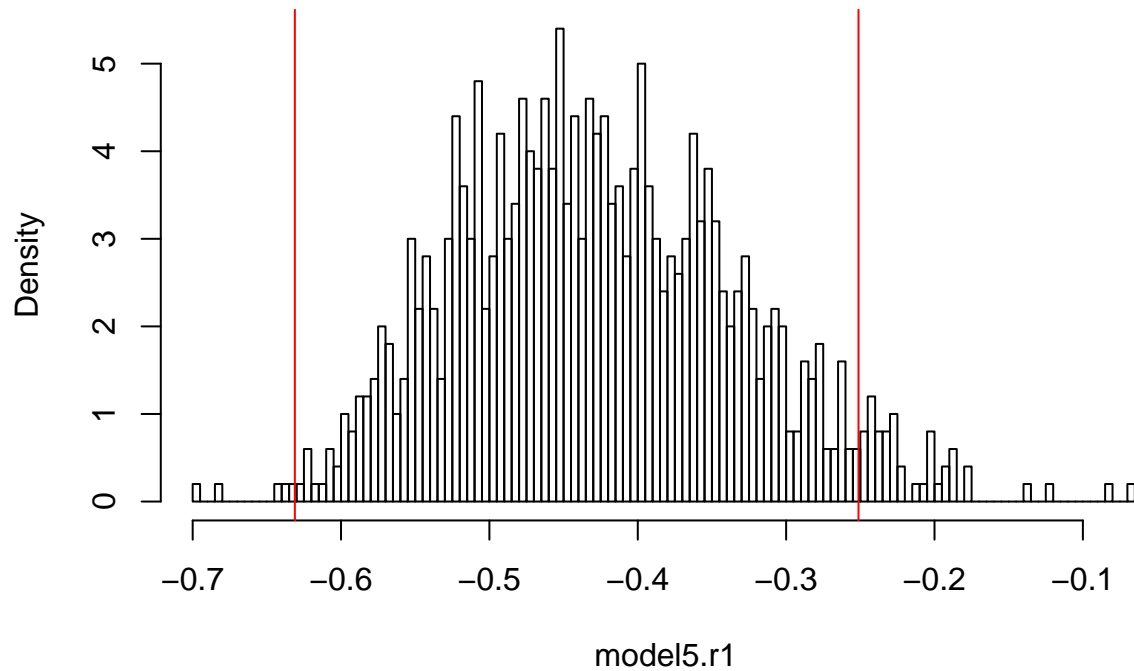
```
hist(model5.r1, breaks=100, freq=F, xlim=c(min(lower, model5.r1), max(upper, model5.r1)))
abline(v = lower, col="red")
abline(v = upper, col="red")
```

Histogram of model5.r1

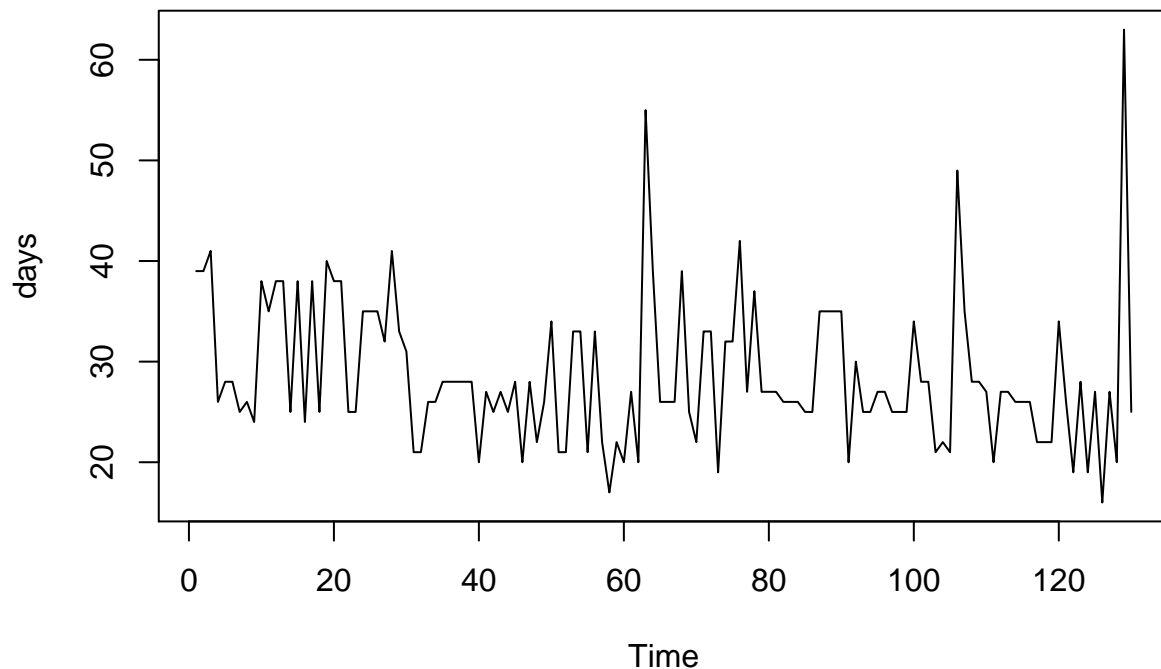95.1% of the observations fall within 2SE.

■

4. The dataset `days` contains accounting data. The data is the number of days it took to receive payment for 130 consecutive orders from a particular distributor.

```
data(days)
```

(a) Plot the times series. Are there any unusual values?

```
plot(days)
```
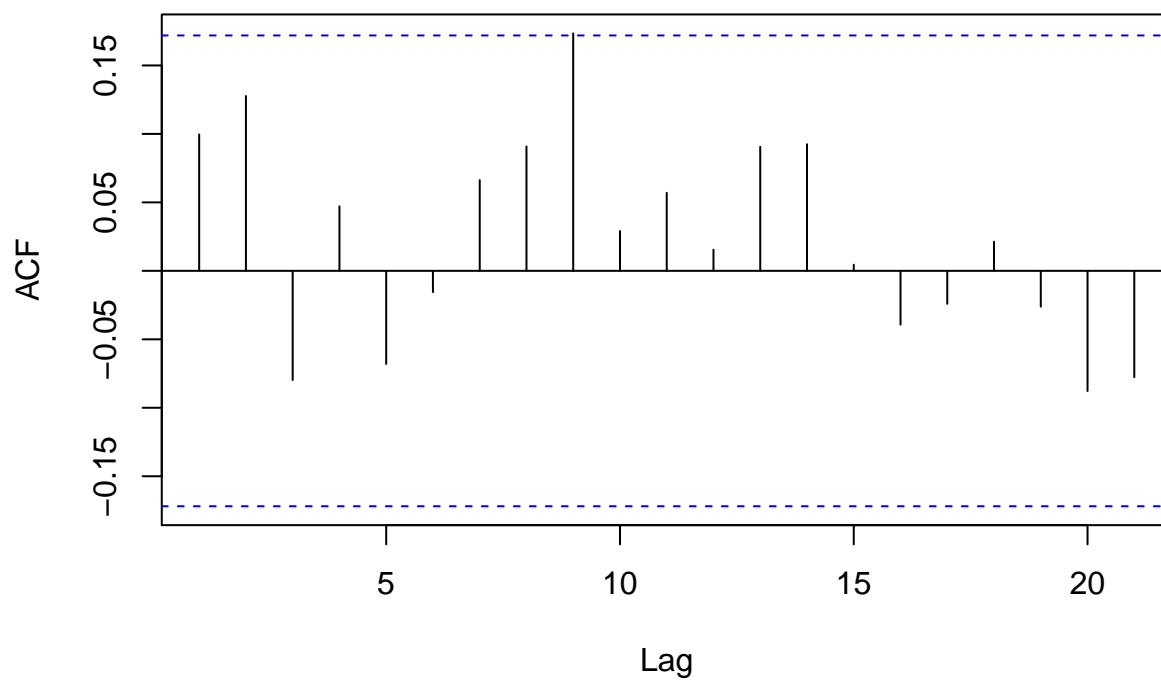


```
day1 <- which(days == 55)
day2 <- which(days == 49)
day3 <- which(days == 63)
```

There are three highly unusual days, at days 63, 106, and 129.

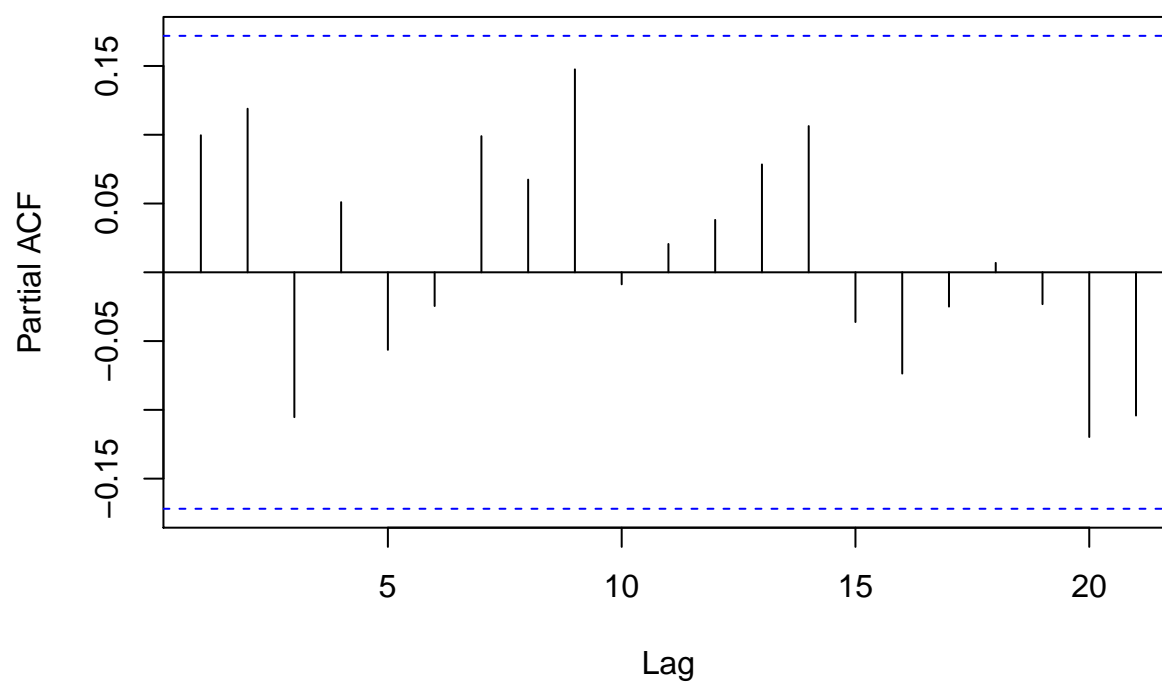## (b) Draw the sample ACF and sample PACF plots. What do you find?
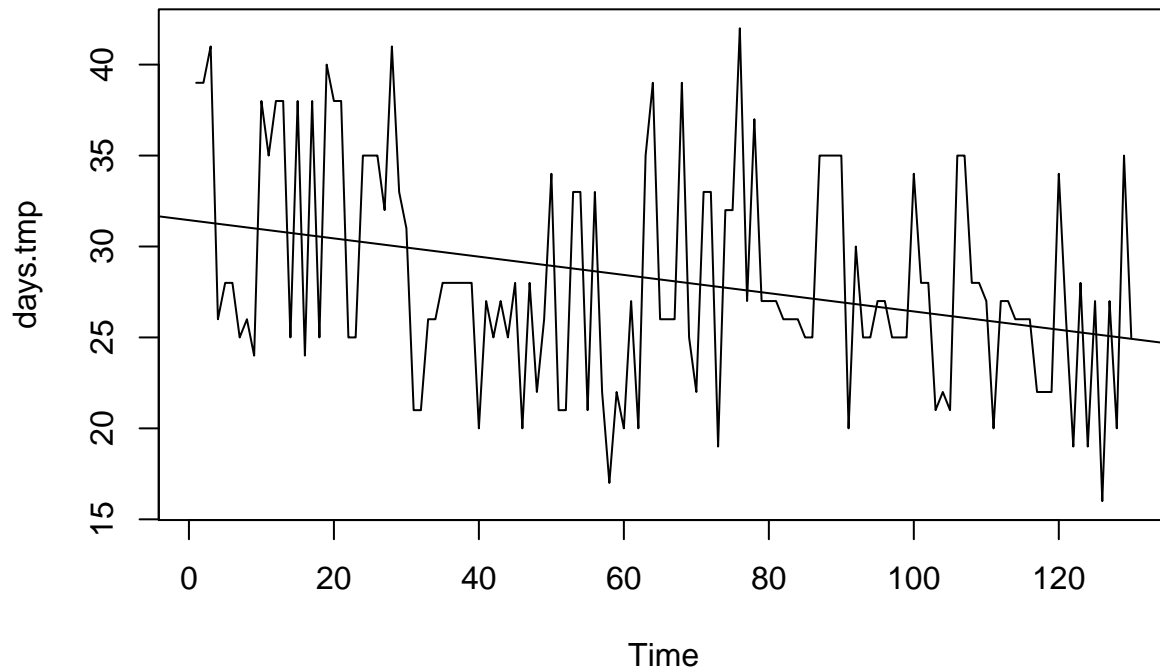
```
acf(days)
```

Series days


```
pacf(days)
```

Series days

(c) Replace the unusual values with a value of 35 days. Redraw the sample ACF and sample PACF plots. Are they different from part (b)?

```r
days.tmp <- days
days.tmp[which(days.tmp == 55)] <- 35
days.tmp[which(days.tmp == 49)] <- 35
days.tmp[which(days.tmp == 63)] <- 35
plot(days.tmp)
abline(reg=lm(days.tmp~time(days.tmp)))
```
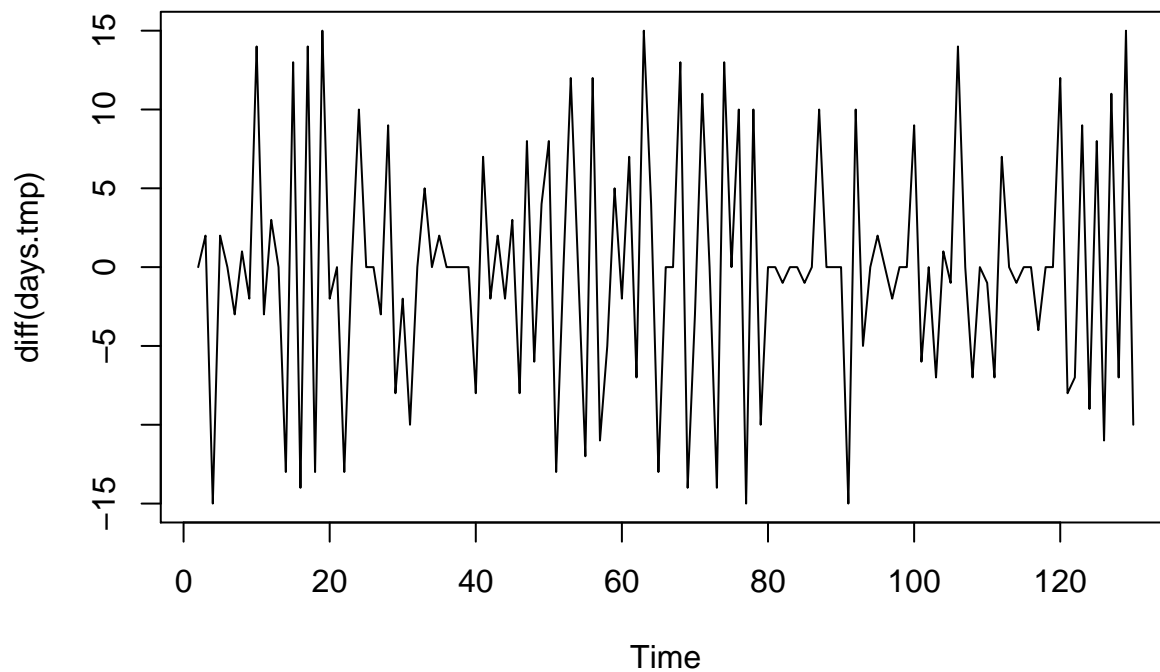

The time series appears to have a slightly decreasing trend, and although the variance appears to be constant, the mean is not, so it is not yet stationary.

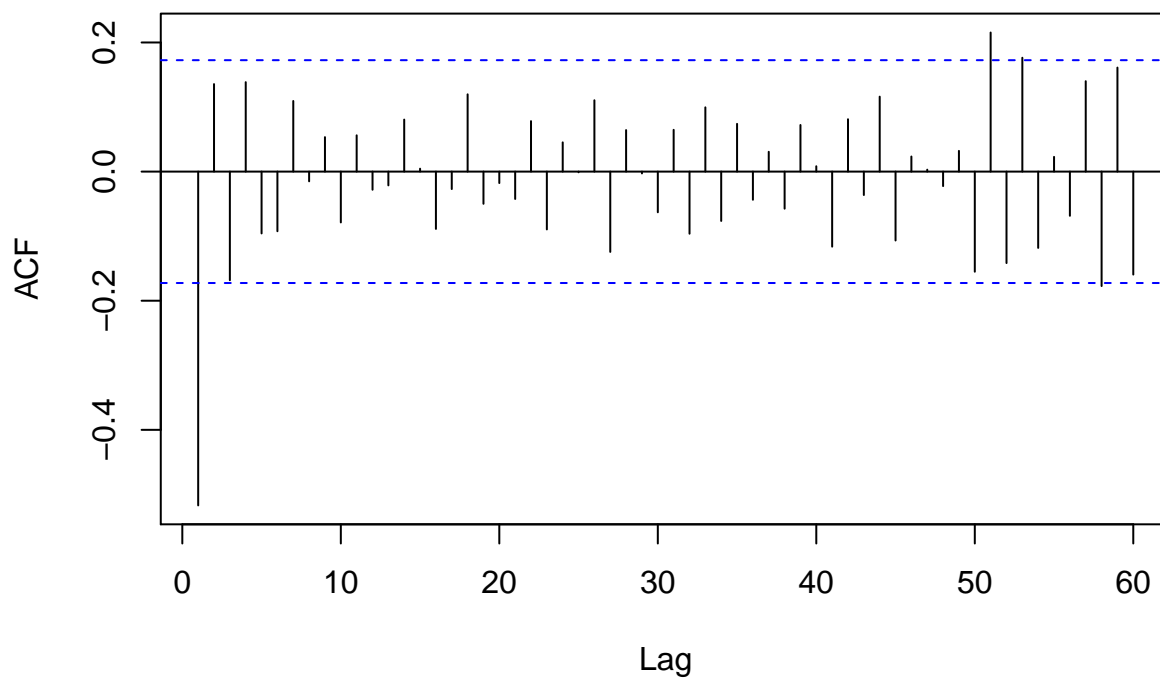**(d) What ARMA model would you specify for this series after removing the outliers? Explain.**

We'll take the first difference to get a stationary time series.
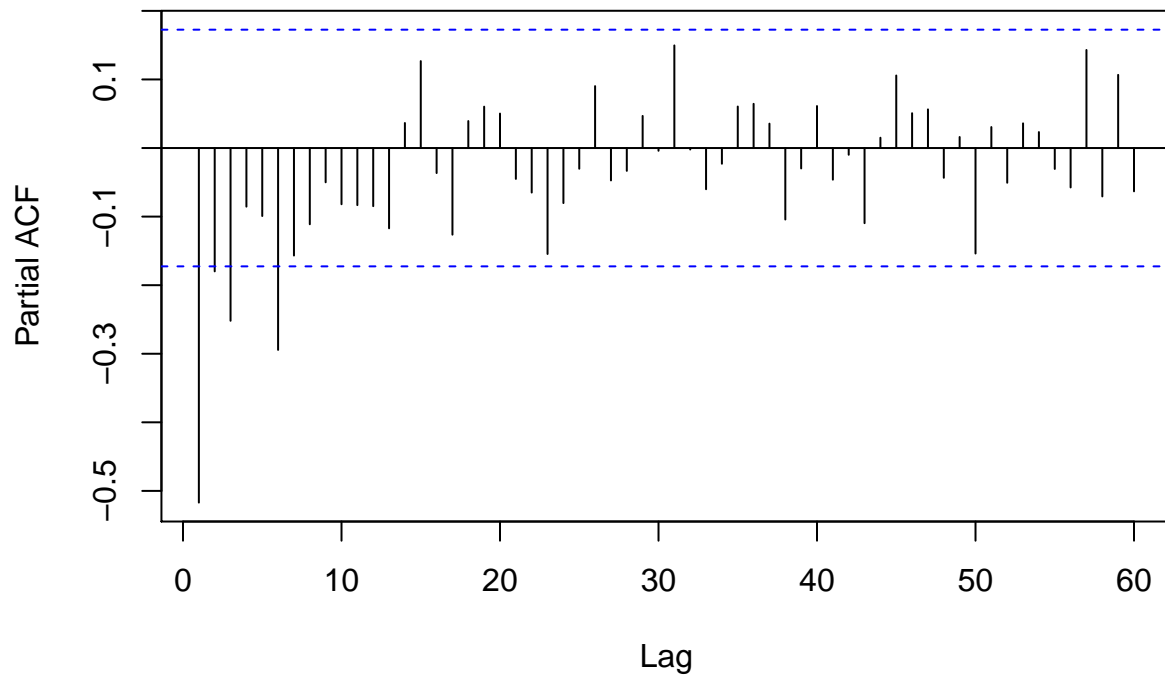
```r
plot(diff(days.tmp))
```

```r
acf.val <- acf(diff(days.tmp), lag.max = 60)
```

Series diff(days.tmp)


```r
phi1 <- acf.val[[1]][1]
pacf.val <- pacf(diff(days.tmp), lag.max = 60)
```

Series diff(days.tmp)


```
theta1 <- pacf.val[[1]][1]
```

The ACF cuts off after 1 lag, so the simplest model we could choose would be an MA(1). This is confirmed using `auto.arima`.

```
suggested.model <- auto.arima(days.tmp)
suggested.model
```
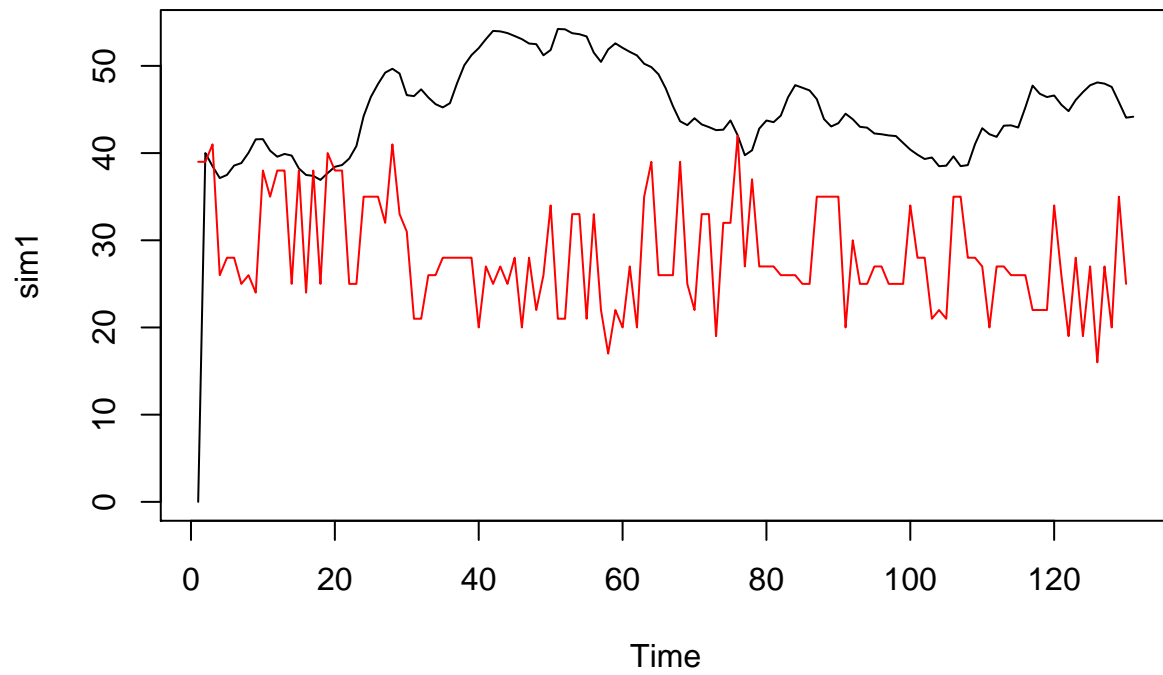
```
## Series: days.tmp
## ARIMA(0,1,1)
##
## Coefficients:
##           ma1
##        -0.8910
## s.e.    0.0432
##
## sigma^2 estimated as 33.78:  log likelihood=-410.36
## AIC=824.73   AICc=824.82   BIC=830.45
```

```
theta1 <- suggested.model$coef[[1]]
```

```
p <- 0
d <- 1
q <- 1

sim1 <- arima.sim(n = 130,
                  start.innov=(40/ -theta1),
                  n.start=1,
                  innov=c(0, rnorm(130 - 1)),
                  model = list(order=c(p, d, q),
                               sd=sd(days.tmp),
```

```
                                    ma=c(-theta1)))
plot(sim1)
lines(days.tmp, col="red")
```


This model does not fit very well though.

∎