

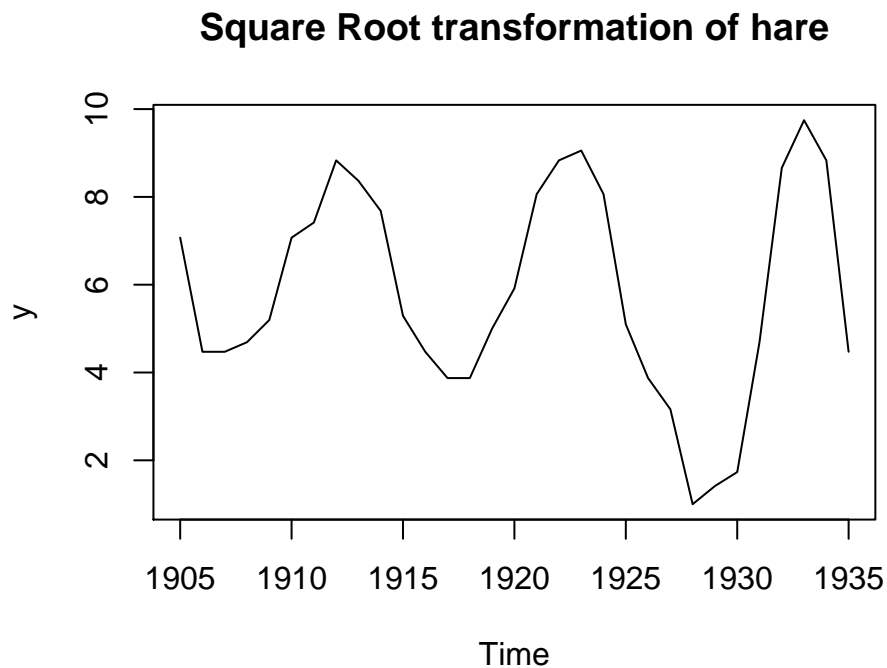
# Homework 8

Brian Detweiler

April 12, 2017

1. Consider the hare data in the TSA package. Fit an AR(3) model to the square root of the data using maximum likelihood estimation.

```
set.seed(48548493)
data(hare)
y <- sqrt(hare)
plot(y, main="Square Root transformation of hare")
```



```
fit <- arima(y, order=c(3, 0, 0), method = "ML")
fit$coef
```

```
##          ar1          ar2          ar3  intercept
##  1.0518982 -0.2292459 -0.3930406  5.6922688
```

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \phi_3 Y_{t-3} + e_t$$

$$\phi_1 = 1.0518982$$

$$\phi_2 = -0.2292459$$

$$\phi_3 = -0.3930406$$

(a) Use the four different Bootstrap resampling methods to get the 95% confidence intervals for the five parameters:  $\phi_1, \phi_2, \phi_3, \mu, \sigma_e^2$ .

#### Bootstrap Method 1

```
b1.phi1.sim <- rep(NA, 1000)
b1.phi2.sim <- rep(NA, 1000)
b1.phi3.sim <- rep(NA, 1000)
b1.mu.sim <- rep(NA, 1000)
b1.sigmae2.sim <- rep(NA, 1000)

b1 <- y
f <- arima(b1, order=c(3,0,0))
phi <- f$coef

for (i in 1:1000) {

  b1 <- y

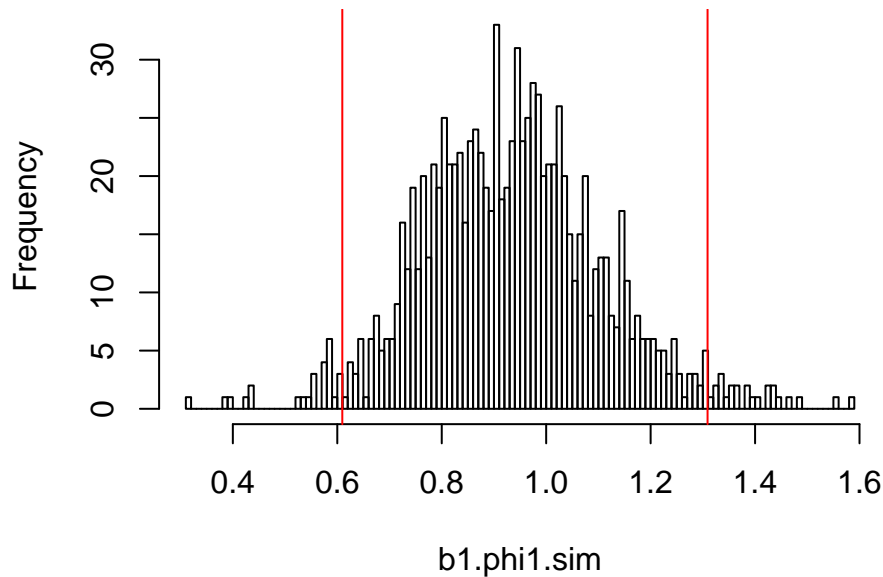
  e <- rep(NA, length(y) - 3)
  for (j in 4:length(y)) {
    e[j - 3] <- rnorm(1)
    b1[j] <- sum(phi[1:3] * (b1[(j - 1):(j - 3)] - phi[4])) + phi[4] + e[j - 3]
  }

  f1 <- arima(b1, order=c(3, 0 ,0), method="ML")
  b1.phi1.sim[i] <- f1$coef[[1]]
  b1.phi2.sim[i] <- f1$coef[[2]]
  b1.phi3.sim[i] <- f1$coef[[3]]
  b1.mu.sim[i] <- mean(b1)
  b1.sigmae2.sim[i] <- var(e)

}

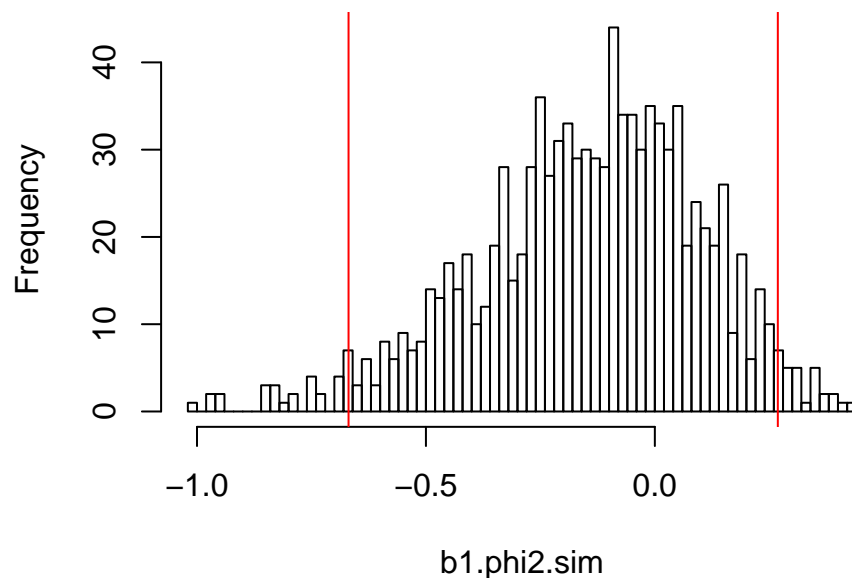
hist(b1.phi1.sim, main="Bootstrap Method 1, Phi 1", breaks=100)
phi1.ci.lower <- quantile(b1.phi1.sim, c(0.025,0.975))[[1]]
phi1.ci.upper <- quantile(b1.phi1.sim, c(0.025,0.975))[[2]]
abline(v=phi1.ci.lower, col="red")
abline(v=phi1.ci.upper, col="red")
```

### Bootstrap Method 1, Phi 1



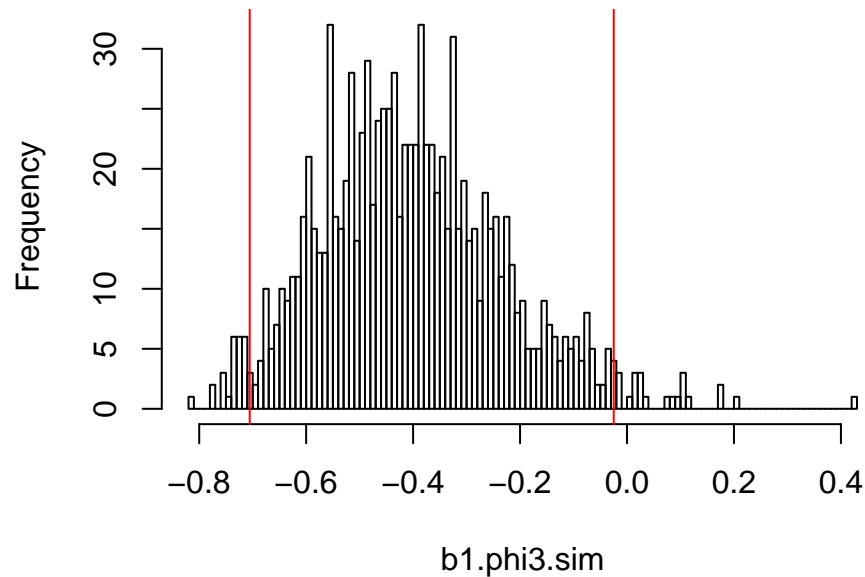
```
hist(b1.phi2.sim, main="Bootstrap Method 1, Phi 2", breaks=100)
phi2.ci.lower <- quantile(b1.phi2.sim, c(0.025,0.975))[[1]]
phi2.ci.upper <- quantile(b1.phi2.sim, c(0.025,0.975))[[2]]
abline(v=phi2.ci.lower, col="red")
abline(v=phi2.ci.upper, col="red")
```

### Bootstrap Method 1, Phi 2



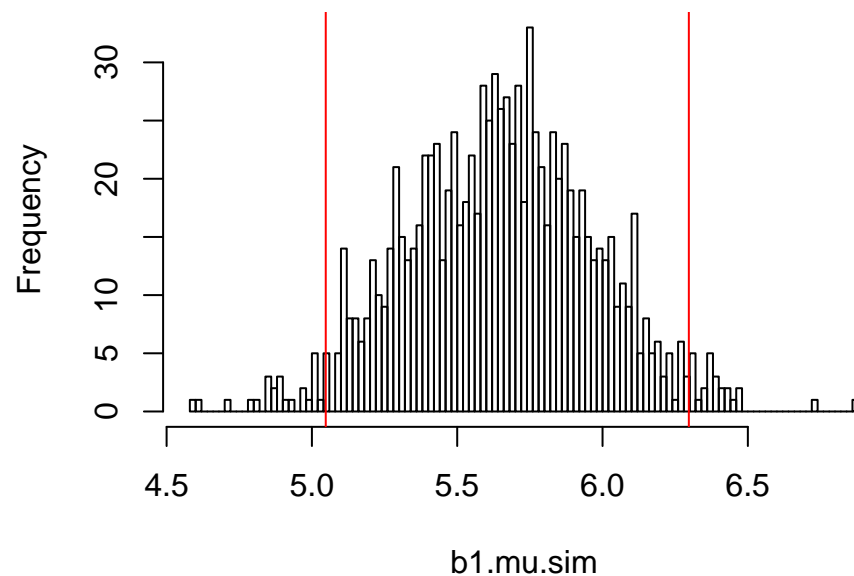
```
hist(b1.phi3.sim, main="Bootstrap Method 1, Phi 3", breaks=100)
phi3.ci.lower <- quantile(b1.phi3.sim, c(0.025,0.975))[[1]]
phi3.ci.upper <- quantile(b1.phi3.sim, c(0.025,0.975))[[2]]
abline(v=phi3.ci.lower, col="red")
abline(v=phi3.ci.upper, col="red")
```

### Bootstrap Method 1, Phi 3



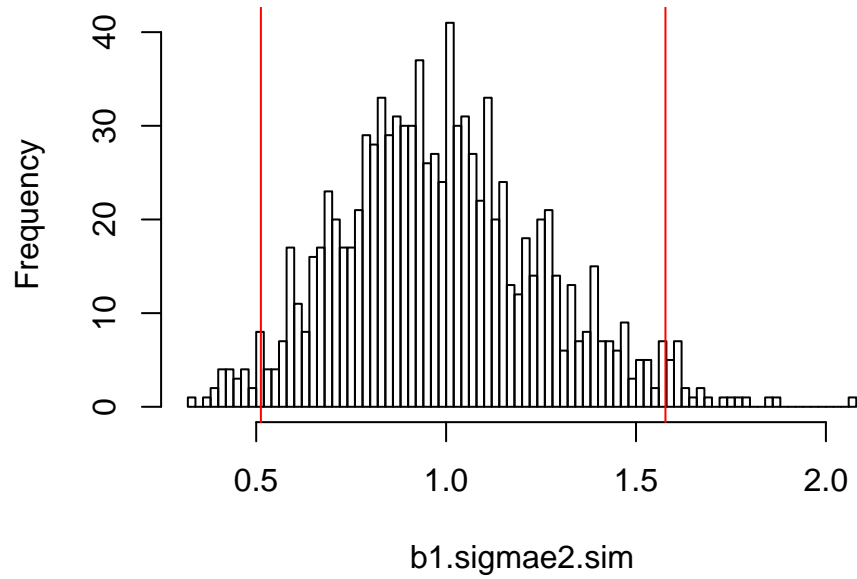
```
hist(b1.mu.sim, main="Bootstrap Method 1, Mu", breaks=100)
mu.ci.lower <- quantile(b1.mu.sim, c(0.025,0.975))[[1]]
mu.ci.upper <- quantile(b1.mu.sim, c(0.025,0.975))[[2]]
abline(v=mu.ci.lower, col="red")
abline(v=mu.ci.upper, col="red")
```

### Bootstrap Method 1, Mu



```
hist(b1.sigmae2.sim, main="Bootstrap Method 1, Sigma_e^2", breaks=100)
sigmae2.ci.lower <- quantile(b1.sigmae2.sim, c(0.025,0.975))[[1]]
sigmae2.ci.upper <- quantile(b1.sigmae2.sim, c(0.025,0.975))[[2]]
abline(v=sigmae2.ci.lower, col="red")
abline(v=sigmae2.ci.upper, col="red")
```

## Bootstrap Method 1, Sigma\_e^2



We have the following 95% confidence intervals:

$$\phi_1 : (0.6095839, 1.3090067)$$

$$\phi_2 : (-0.6690048, 0.2686062)$$

$$\phi_3 : (-0.7055698, -0.0248301)$$

$$\mu : (5.0476376, 6.2969115)$$

$$\sigma_e^2 : (0.5123336, 1.5775382)$$

## Bootstrap Method 2

```
b2.phi1.sim <- rep(NA, 1000)
b2.phi2.sim <- rep(NA, 1000)
b2.phi3.sim <- rep(NA, 1000)
b2.mu.sim <- rep(NA, 1000)
b2.sigmae2.sim <- rep(NA, 1000)

b2 <- y
f <- arima(b2, order=c(3,0,0))
phi <- f$coef

for (j in 1:1000) {

  b2 <- y[1:3]
  for (i in 4:length(y)){
    b2[i] = sum(phi[1:3] * (b2[(i - 1):(i - 3)] - phi[4])) + phi[4] + sample(f$residuals, 1)
  }

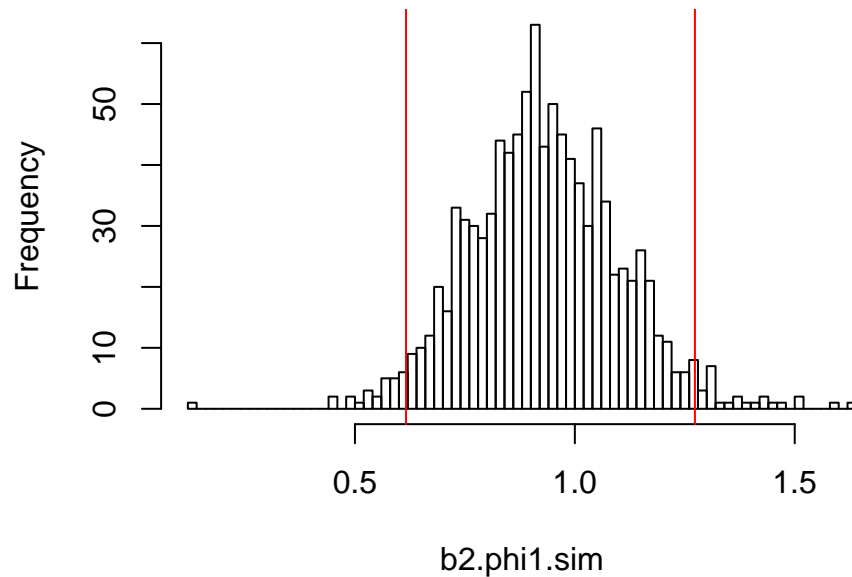
  e <- rep(NA, length(y) - 3)
  for (i in 4:length(y)) {
    e[i - 3] <- rnorm(1)
    b2[i] <- sum(phi[1:3] * (b2[(i - 1):(i - 3)] - phi[4])) + phi[4] + e[i - 3]
  }

  f1 <- arima(b2, order=c(3, 0 ,0), method="ML")
  b2.phi1.sim[j] <- f1$coef[[1]]
  b2.phi2.sim[j] <- f1$coef[[2]]
  b2.phi3.sim[j] <- f1$coef[[3]]
  b2.mu.sim[j] <- mean(b2)
  b2.sigmae2.sim[j] <- var(e)

}

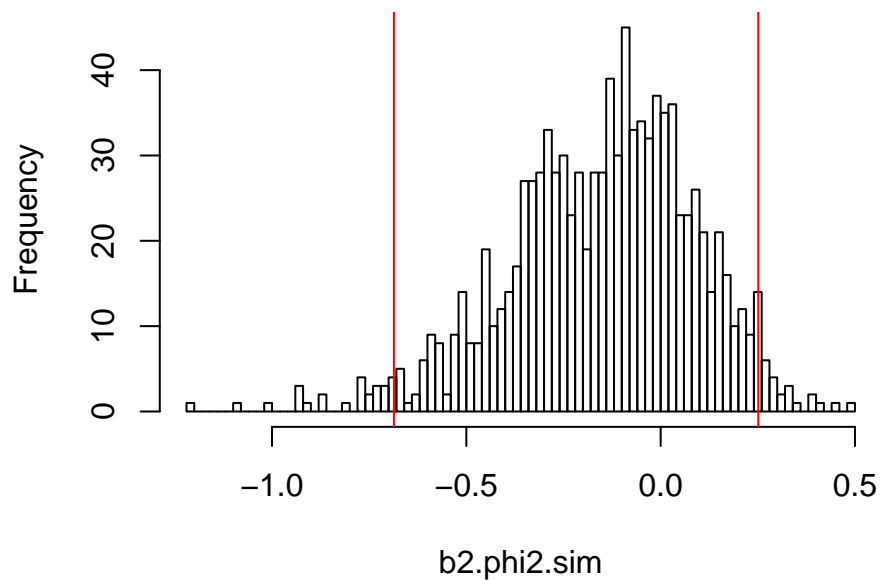
hist(b2.phi1.sim, main="Bootstrap Method 2, Phi 1", breaks=100)
phi1.ci.lower <- quantile(b2.phi1.sim, c(0.025,0.975))[[1]]
phi1.ci.upper <- quantile(b2.phi1.sim, c(0.025,0.975))[[2]]
abline(v=phi1.ci.lower, col="red")
abline(v=phi1.ci.upper, col="red")
```

## Bootstrap Method 2, Phi 1



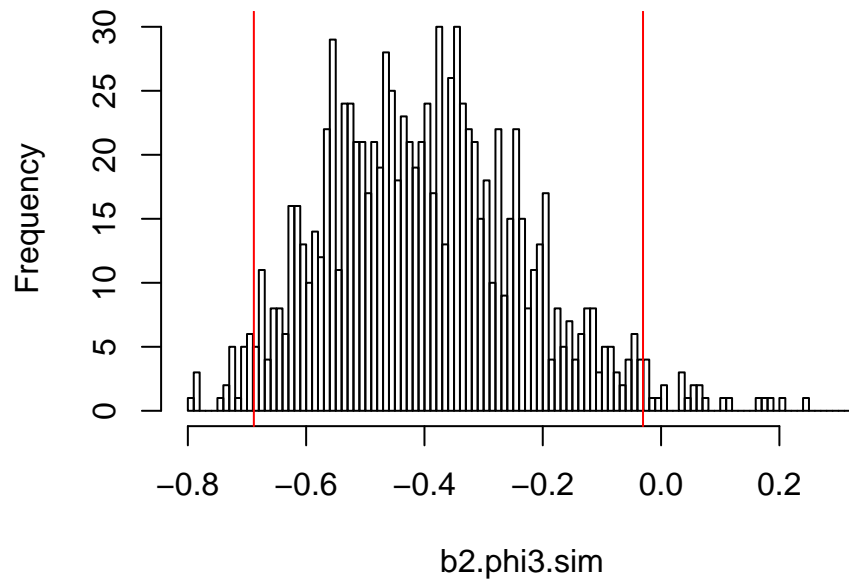
```
hist(b2.phi2.sim, main="Bootstrap Method 2, Phi 2", breaks=100)
phi2.ci.lower <- quantile(b2.phi2.sim, c(0.025,0.975))[[1]]
phi2.ci.upper <- quantile(b2.phi2.sim, c(0.025,0.975))[[2]]
abline(v=phi2.ci.lower, col="red")
abline(v=phi2.ci.upper, col="red")
```

## Bootstrap Method 2, Phi 2



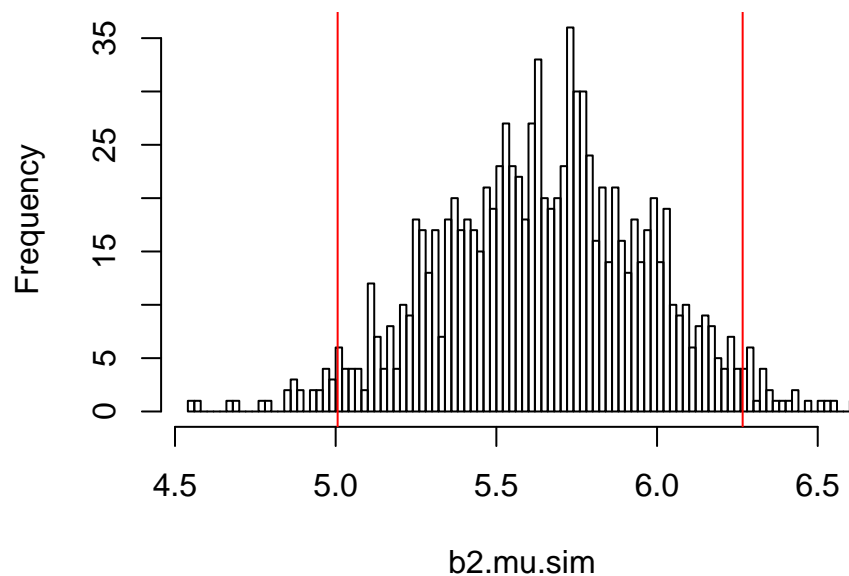
```
hist(b2.phi3.sim, main="Bootstrap Method 2, Phi 3", breaks=100)
phi3.ci.lower <- quantile(b2.phi3.sim, c(0.025,0.975))[[1]]
phi3.ci.upper <- quantile(b2.phi3.sim, c(0.025,0.975))[[2]]
abline(v=phi3.ci.lower, col="red")
abline(v=phi3.ci.upper, col="red")
```

### Bootstrap Method 2, Phi 3



```
hist(b2.mu.sim, main="Bootstrap Method 2, Mu", breaks=100)
mu.ci.lower <- quantile(b2.mu.sim, c(0.025,0.975))[[1]]
mu.ci.upper <- quantile(b2.mu.sim, c(0.025,0.975))[[2]]
abline(v=mu.ci.lower, col="red")
abline(v=mu.ci.upper, col="red")
```

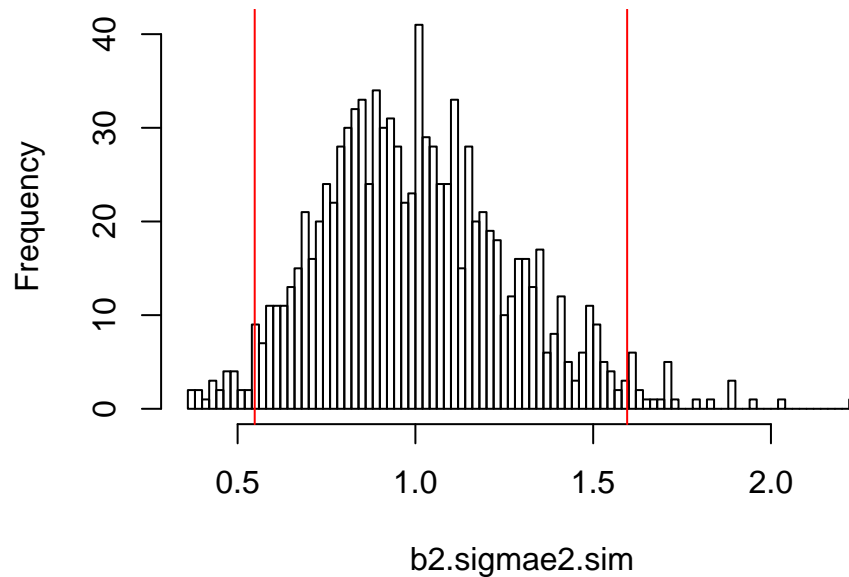
### Bootstrap Method 2, Mu



```
hist(b2.sigmae2.sim, main="Bootstrap Method 2, Sigma_e^2", breaks=100)
sigmae2.ci.lower <- quantile(b2.sigmae2.sim, c(0.025,0.975))[[1]]
sigmae2.ci.upper <- quantile(b2.sigmae2.sim, c(0.025,0.975))[[2]]
abline(v=sigmae2.ci.lower, col="red")
abline(v=sigmae2.ci.upper, col="red")
```



## Bootstrap Method 2, Sigma\_e^2



We have the following 95% confidence intervals:

$$\begin{aligned}\phi_1 &: (0.6161653, 1.2730788) \\ \phi_2 &: (-0.6863655, 0.2511593) \\ \phi_3 &: (-0.6884145, -0.0305322) \\ \mu &: (5.006153, 6.2664254) \\ \sigma_e^2 &: (0.548046, 1.5955376)\end{aligned}$$

## Bootstrap Method 3

```
b3.phi1.sim <- rep(NA, 1000)
b3.phi2.sim <- rep(NA, 1000)
b3.phi3.sim <- rep(NA, 1000)
b3.mu.sim <- rep(NA, 1000)
b3.sigmae2.sim <- rep(NA, 1000)

b3 <- y
f <- arima(b3, order=c(3,0,0))
phi <- f$coef

for (j in 1:1000) {
  b3 <- y[1:3]
  for (i in 4:length(y)){
    b3 <- arima.sim(n=100, model=list(ar=phi[1:3]), sd=sqrt(f$sigma2)) + phi[4]
    b3 <- b3[-(1:(100-31))]
  }
}
```

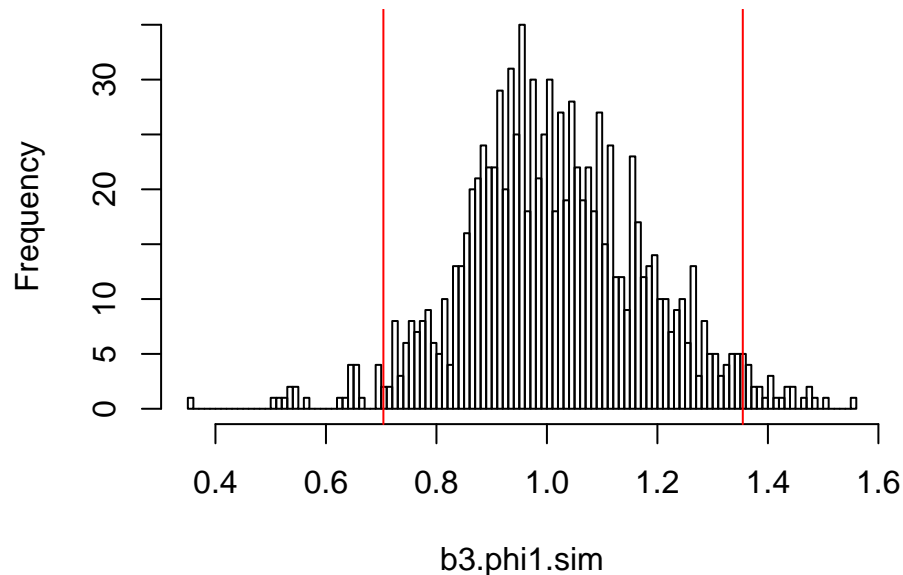
```

f1 <- arima(b3, order=c(3, 0 ,0), method="ML")
b3.phi1.sim[j] <- f1$coef[[1]]
b3.phi2.sim[j] <- f1$coef[[2]]
b3.phi3.sim[j] <- f1$coef[[3]]
b3.mu.sim[j] <- mean(b3)
b3.sigmae2.sim[j] <- var(resid(f1))
}

hist(b3.phi1.sim, main="Bootstrap Method 3, Phi 1", breaks=100)
phi1.ci.lower <- quantile(b3.phi1.sim, c(0.025,0.975))[[1]]
phi1.ci.upper <- quantile(b3.phi1.sim, c(0.025,0.975))[[2]]
abline(v=phi1.ci.lower, col="red")
abline(v=phi1.ci.upper, col="red")

```

### Bootstrap Method 3, Phi 1

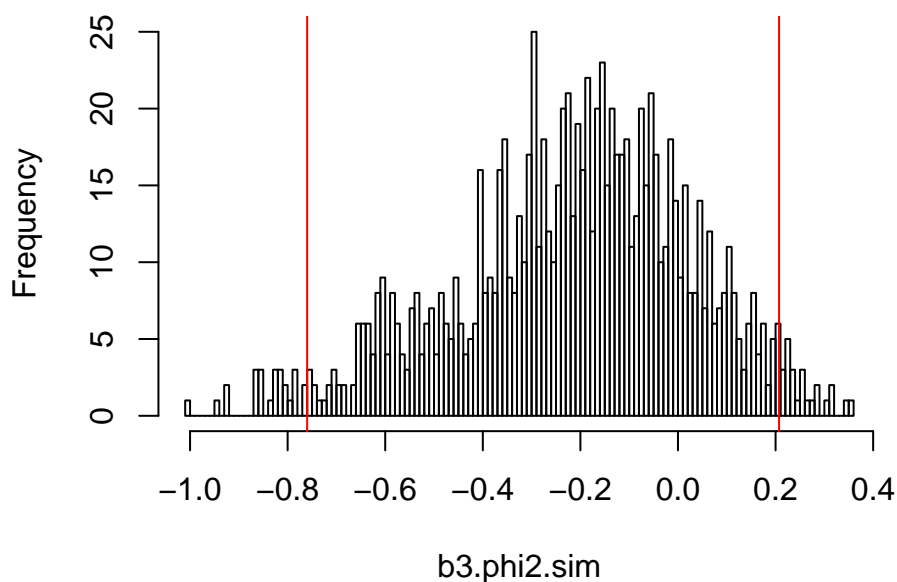


```

hist(b3.phi2.sim, main="Bootstrap Method 3, Phi 2", breaks=100)
phi2.ci.lower <- quantile(b3.phi2.sim, c(0.025,0.975))[[1]]
phi2.ci.upper <- quantile(b3.phi2.sim, c(0.025,0.975))[[2]]
abline(v=phi2.ci.lower, col="red")
abline(v=phi2.ci.upper, col="red")

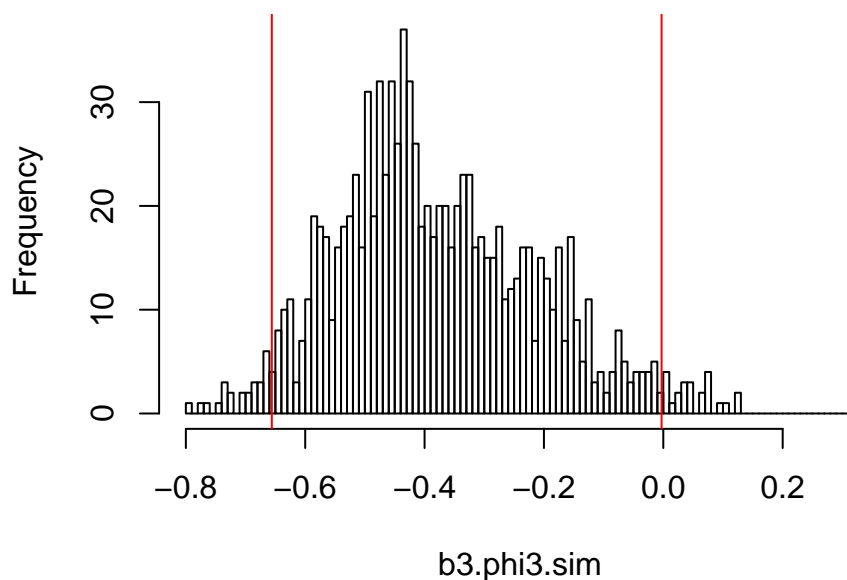
```

### Bootstrap Method 3, Phi 2



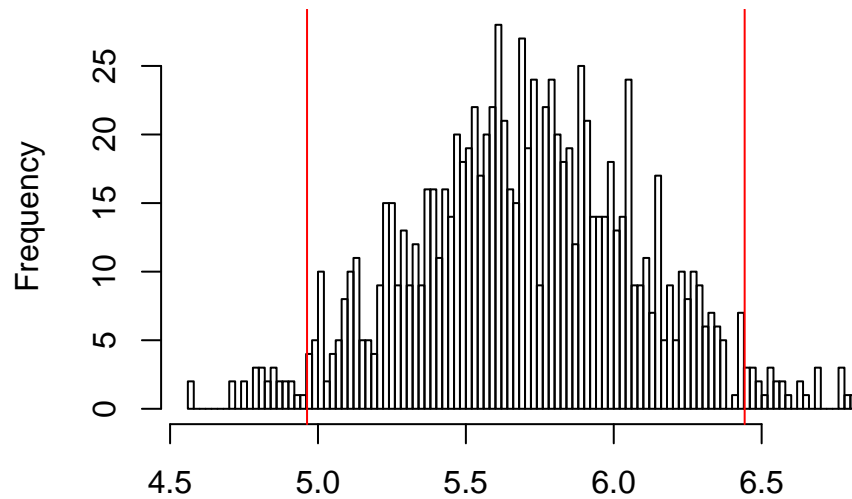
```
hist(b3.phi3.sim, main="Bootstrap Method 3, Phi 3", breaks=100)
phi3.ci.lower <- quantile(b3.phi3.sim, c(0.025,0.975))[[1]]
phi3.ci.upper <- quantile(b3.phi3.sim, c(0.025,0.975))[[2]]
abline(v=phi3.ci.lower, col="red")
abline(v=phi3.ci.upper, col="red")
```

### Bootstrap Method 3, Phi 3



```
hist(b3.mu.sim, main="Bootstrap Method 3, Mu", breaks=100)
mu.ci.lower <- quantile(b3.mu.sim, c(0.025,0.975))[[1]]
mu.ci.upper <- quantile(b3.mu.sim, c(0.025,0.975))[[2]]
abline(v=mu.ci.lower, col="red")
abline(v=mu.ci.upper, col="red")
```

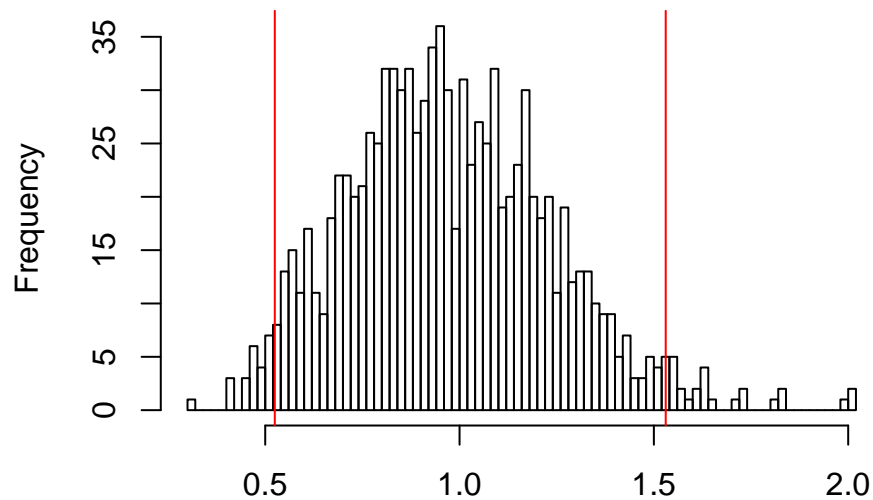
### Bootstrap Method 3, Mu



b3.mu.sim

```
hist(b3.sigmae2.sim, main="Bootstrap Method 3, Sigma_e^2", breaks=100)
sigmae2.ci.lower <- quantile(b3.sigmae2.sim, c(0.025,0.975))[[1]]
sigmae2.ci.upper <- quantile(b3.sigmae2.sim, c(0.025,0.975))[[2]]
abline(v=sigmae2.ci.lower, col="red")
abline(v=sigmae2.ci.upper, col="red")
```

### Bootstrap Method 3, Sigma\_e^2



b3.sigmae2.sim

We have the following 95% confidence intervals:

$$\begin{aligned}
\phi_1 &: (0.7040089, 1.3545191) \\
\phi_2 &: (-0.7600312, 0.2070895) \\
\phi_3 &: (-0.6559341, -0.0028681) \\
\mu &: (4.963064, 6.442584) \\
\sigma_e^2 &: (0.5246057, 1.5304712)
\end{aligned}$$

## Bootstrap Method 4

```
b4.phi1.sim <- rep(NA, 1000)
b4.phi2.sim <- rep(NA, 1000)
b4.phi3.sim <- rep(NA, 1000)
b4.mu.sim <- rep(NA, 1000)
b4.sigmae2.sim <- rep(NA, 1000)

b4 <- y
f <- arima(b4, order=c(3,0,0))
phi <- f$coef

for (j in 1:1000) {

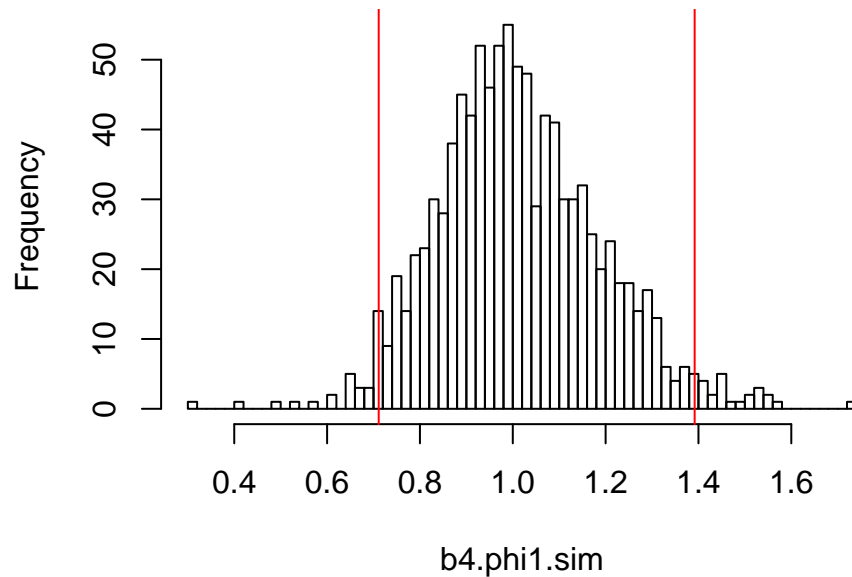
  b4 <- y[1:3]
  for (i in 4:length(y)){
    b4 <- arima.sim(n=100, model=list(ar=phi[1:3]), rand.gen=function(n, r=f$residuals){ sample(r, n, r)})
    b4 <- b4[-(1:(100 - 31))]
  }

  f1 <- arima(b4, order=c(3, 0 ,0), method="ML")
  b4.phi1.sim[j] <- f1$coef[[1]]
  b4.phi2.sim[j] <- f1$coef[[2]]
  b4.phi3.sim[j] <- f1$coef[[3]]
  b4.mu.sim[j] <- mean(b4)
  b4.sigmae2.sim[j] <- var(resid(f1))

}

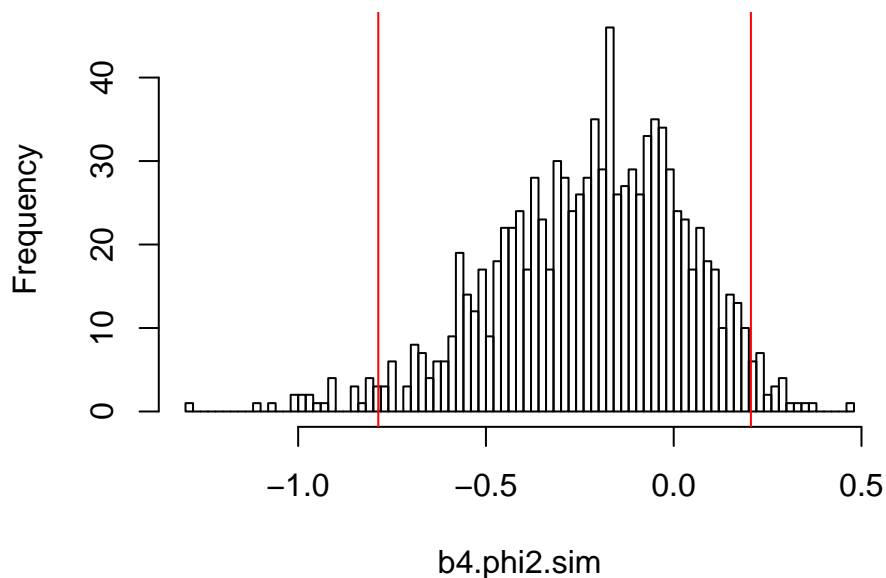
hist(b4.phi1.sim, main="Bootstrap Method 4, Phi 1", breaks=100)
phi1.ci.lower <- quantile(b4.phi1.sim, c(0.025,0.975))[[1]]
phi1.ci.upper <- quantile(b4.phi1.sim, c(0.025,0.975))[[2]]
abline(v=phi1.ci.lower, col="red")
abline(v=phi1.ci.upper, col="red")
```

### Bootstrap Method 4, Phi 1



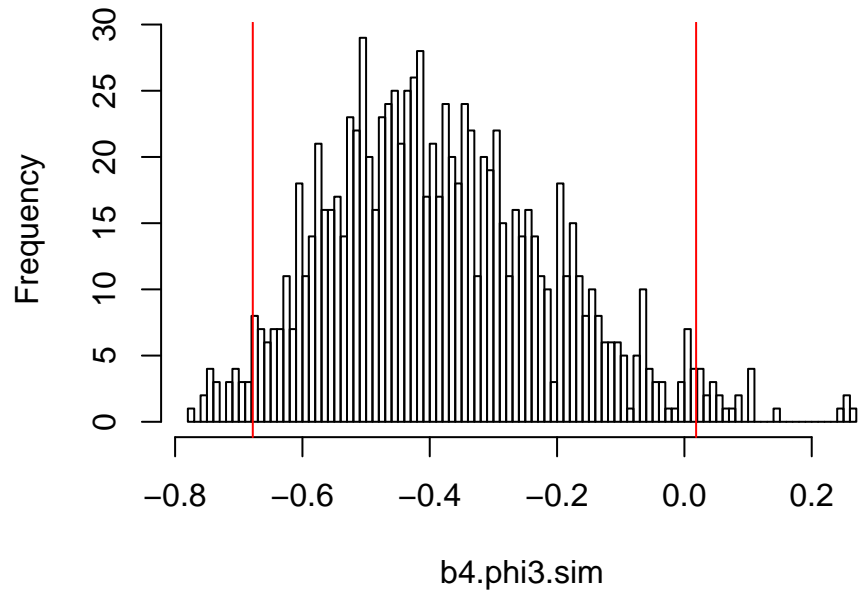
```
hist(b4.phi2.sim, main="Bootstrap Method 4, Phi 2", breaks=100)
phi2.ci.lower <- quantile(b4.phi2.sim, c(0.025,0.975))[[1]]
phi2.ci.upper <- quantile(b4.phi2.sim, c(0.025,0.975))[[2]]
abline(v=phi2.ci.lower, col="red")
abline(v=phi2.ci.upper, col="red")
```

### Bootstrap Method 4, Phi 2



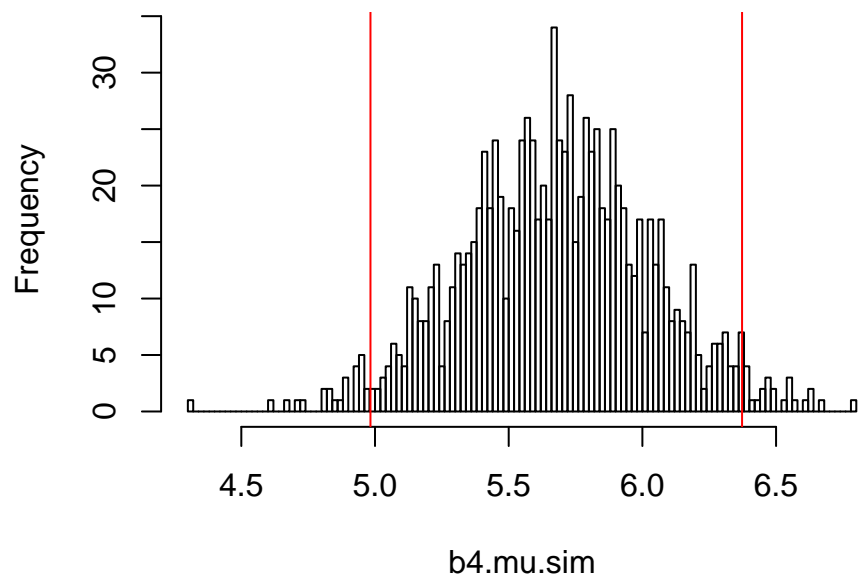
```
hist(b4.phi3.sim, main="Bootstrap Method 4, Phi 3", breaks=100)
phi3.ci.lower <- quantile(b4.phi3.sim, c(0.025,0.975))[[1]]
phi3.ci.upper <- quantile(b4.phi3.sim, c(0.025,0.975))[[2]]
abline(v=phi3.ci.lower, col="red")
abline(v=phi3.ci.upper, col="red")
```

### Bootstrap Method 4, Phi 3



```
hist(b4.mu.sim, main="Bootstrap Method 4, Mu", breaks=100)
mu.ci.lower <- quantile(b4.mu.sim, c(0.025,0.975))[[1]]
mu.ci.upper <- quantile(b4.mu.sim, c(0.025,0.975))[[2]]
abline(v=mu.ci.lower, col="red")
abline(v=mu.ci.upper, col="red")
```

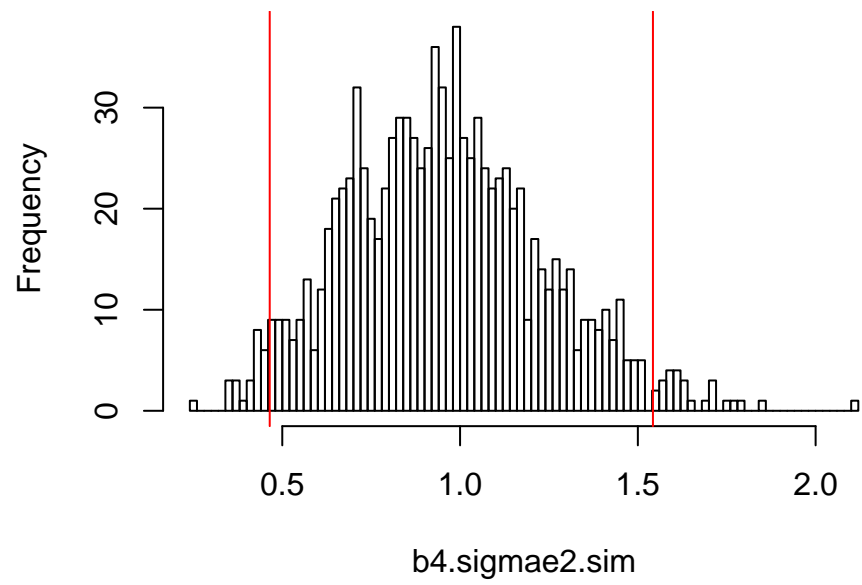
### Bootstrap Method 4, Mu



```
hist(b4.sigmae2.sim, main="Bootstrap Method 4, Sigma_e^2", breaks=100)
sigmae2.ci.lower <- quantile(b4.sigmae2.sim, c(0.025,0.975))[[1]]
sigmae2.ci.upper <- quantile(b4.sigmae2.sim, c(0.025,0.975))[[2]]
abline(v=sigmae2.ci.lower, col="red")
abline(v=sigmae2.ci.upper, col="red")
```

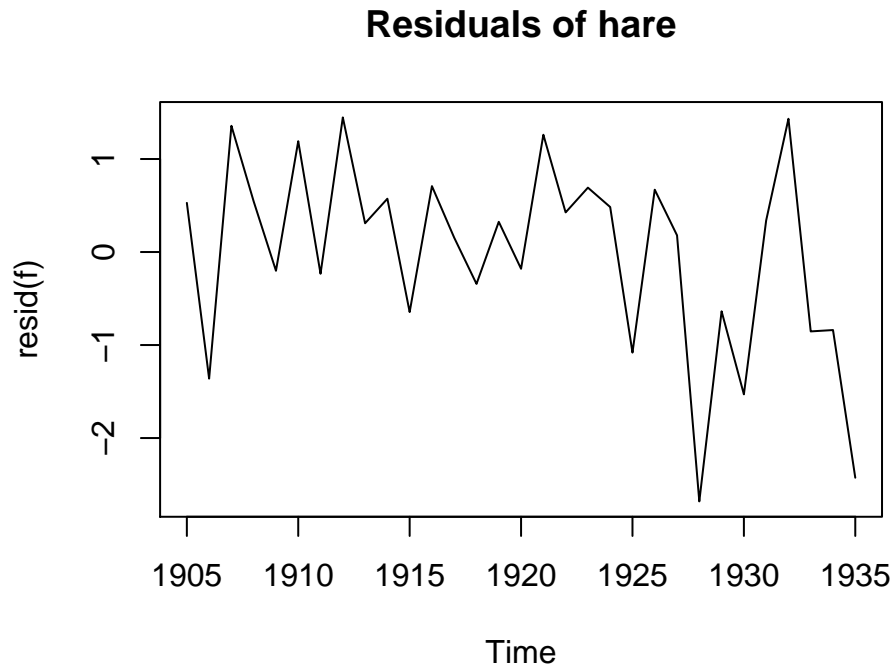


### Bootstrap Method 4, Sigma\_e^2

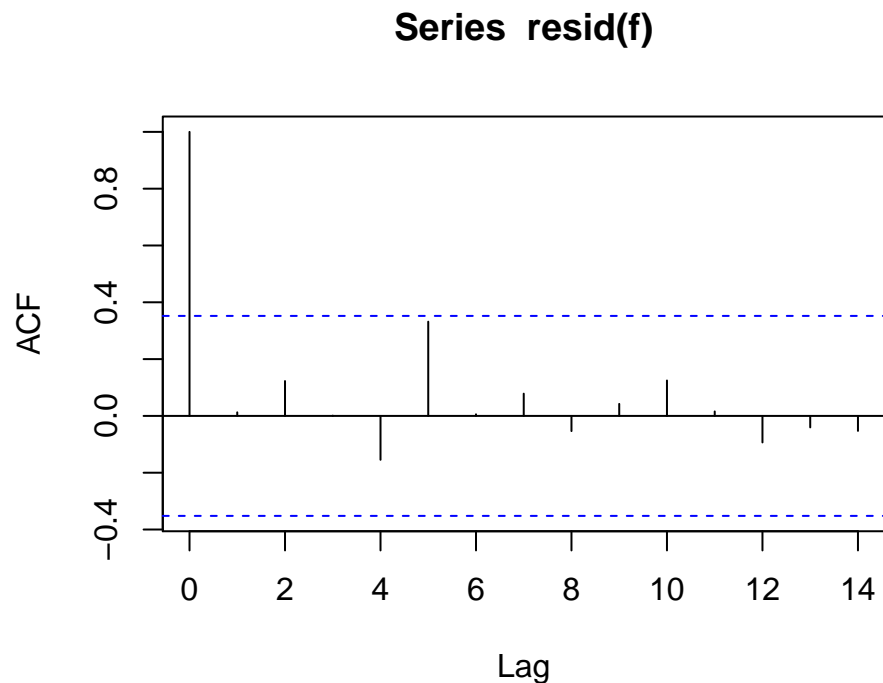


(b) Construct a time series plot and a sample ACF plot of the residuals. Perform the Ljung-Box test for all values of  $K$  between 4 and 20 inclusive. Based on these 3 things does it seem like an AR(3) is a suitable model?

```
f <- arima(y, order=c(3,0,0))  
plot(resid(f), main="Residuals of hare")
```



```
stats::acf(resid(f))
```



```
for (k in 4:20) {  
  bt <- Box.test(y, lag=k, type="Ljung-Box")  
}
```

```

print(paste0("K = ", k))
print(paste0("p-value = ", bt$p.value))
}

```

```

## [1] "K = 4"
## [1] "p-value = 1.79277963363589e-06"
## [1] "K = 5"
## [1] "p-value = 6.01616878537214e-09"
## [1] "K = 6"
## [1] "p-value = 3.04929415051447e-11"
## [1] "K = 7"
## [1] "p-value = 8.37963032296329e-12"
## [1] "K = 8"
## [1] "p-value = 2.52862175642576e-11"
## [1] "K = 9"
## [1] "p-value = 1.8373857990639e-11"
## [1] "K = 10"
## [1] "p-value = 6.76014799694258e-13"
## [1] "K = 11"
## [1] "p-value = 2.54241072639161e-14"
## [1] "K = 12"
## [1] "p-value = 1.15463194561016e-14"
## [1] "K = 13"
## [1] "p-value = 3.10862446895044e-14"
## [1] "K = 14"
## [1] "p-value = 3.80806497446429e-14"
## [1] "K = 15"
## [1] "p-value = 5.77315972805081e-15"
## [1] "K = 16"
## [1] "p-value = 3.33066907387547e-16"
## [1] "K = 17"
## [1] "p-value = 1.11022302462516e-16"
## [1] "K = 18"
## [1] "p-value = 1.11022302462516e-16"
## [1] "K = 19"
## [1] "p-value = 3.33066907387547e-16"
## [1] "K = 20"
## [1] "p-value = 2.22044604925031e-16"

```

While the ACF appears to be white noise, the residuals plot appears to have decreased variance in the middle and increased variance toward the end. The Ljung-Box tests have p-values  $\ll 0.05$  for all values of  $K$ , and thus we reject the null hypothesis that these are i.i.d., so we must conclude that an AR(3) may not be the best fit model for this series.

**(c) Perform a runs test on the standardized residuals from this model. Comment on your findings.**

```

runs.test(factor(residuals(f) > median(residuals(f))))

##
## Runs Test
##
## data: factor(residuals(f) > median(residuals(f)))

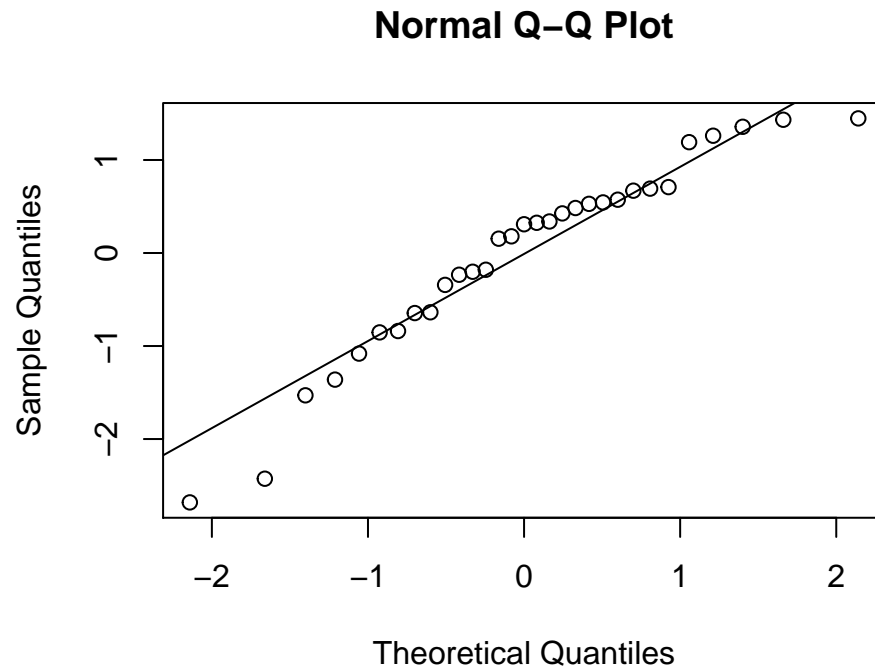
```

```
## Standard Normal = 1.286, p-value = 0.1984
## alternative hypothesis: two.sided
```

With a p-value  $\gg 0.05$ , there do not appear to be any patterns in the residuals.

(d) Do a Normal Quantile plot of the residuals and perform the Shapiro-Wilk test for Normality. Comment on your findings.

```
qqn <- qqnorm(residuals(f))
qqline(residuals(f))
```



```
r.squared <- cor(qqn$x, qqn$y)

shapiro.wilk.test <- shapiro.test(x = residuals(f))
```

The qq-plot strongly suggests a normal distribution, with  $R^2 = 0.9687552$ , however the Shapiro-Wilk test results in a p-value of 0.0604299, which is not statistically significant, and therefore we should not say the residuals are i.i.d.

(e) Look at the model fit, are there any parameters that are not significantly different from zero? If so, refit the model with those parameters fixed to be zero. In terms of AIC, does this model fit better? Check the model diagnostics, and comment on your findings.

```
fit <- arima(y, order=c(3,0,0))
fit

##
## Call:
## arima(x = y, order = c(3, 0, 0))
##
```

```
## Coefficients:
##          ar1      ar2      ar3  intercept
##      1.0519 -0.2292 -0.3931    5.6923
## s.e.  0.1877  0.2942  0.1915    0.3371
##
## sigma^2 estimated as 1.066:  log likelihood = -46.54,  aic = 101.08
```

$\phi_2$  is -0.2291852, but has a standard error of 0.2942, and therefore overlaps with 0. We can fix this parameter and refit the model.

```
refit <- arima(y, order = c(3, 0, 0), fixed = c(NA, 0, NA, NA), transform.pars = FALSE)
refit
```

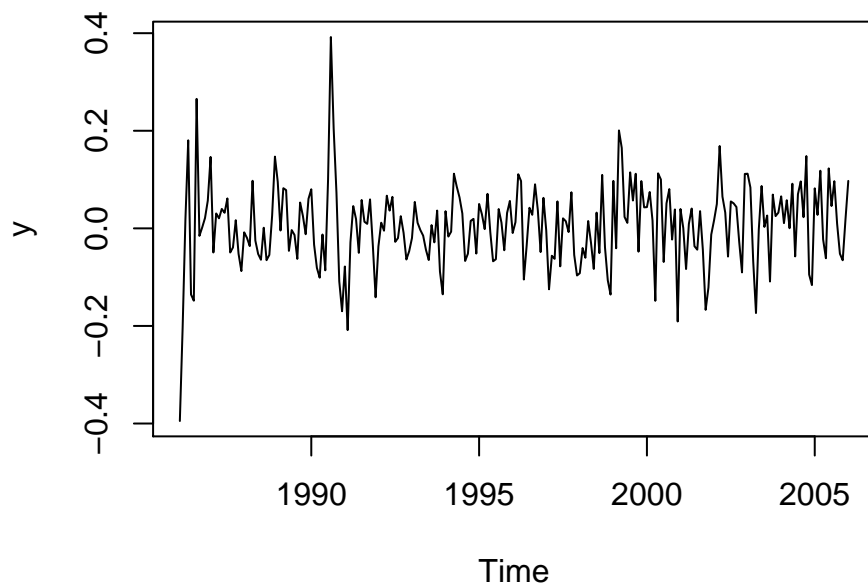
```
##
## Call:
## arima(x = y, order = c(3, 0, 0), transform.pars = FALSE, fixed = c(NA, 0, NA,
##      NA))
##
## Coefficients:
##          ar1  ar2      ar3  intercept
##      0.9190   0 -0.5313    5.6889
## s.e.  0.0791   0  0.0697    0.3179
##
## sigma^2 estimated as 1.088:  log likelihood = -46.85,  aic = 99.69
```

We see that the AIC does improve on the refitted model.

■

2. At the end of Chapter 6 we looked at fitting an ARIMA model to the log of the oil.price data. It was suggested that three possible models for the first difference of the log of the oil price would be an AR(1), an AR(4), and a MA(1).

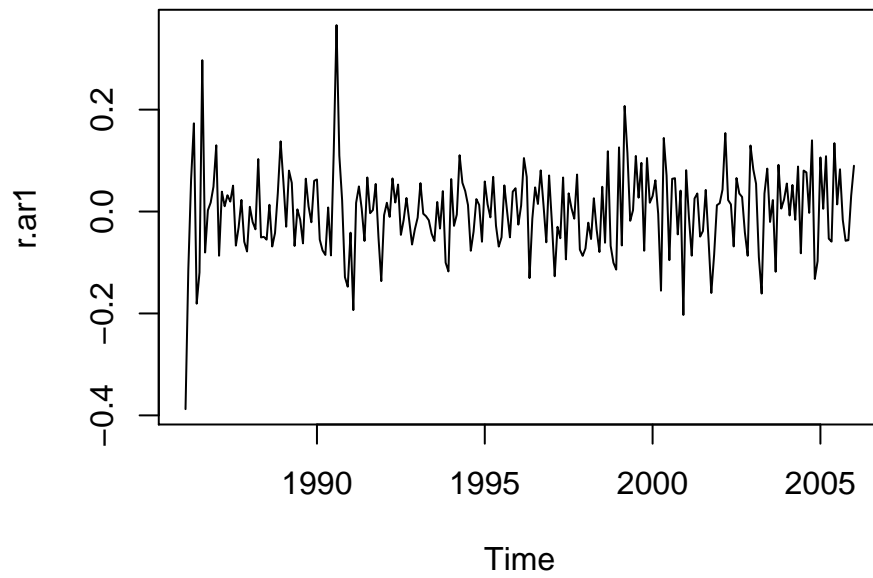
```
data(oil.price)
y <- diff(log(oil.price))
plot(y)
```



### AR(1) Model

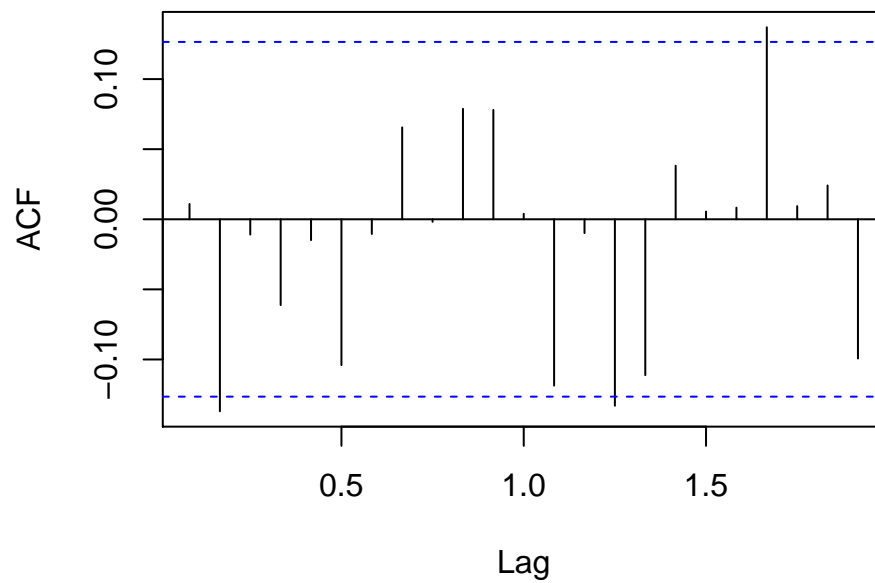
```
fit.ar1 <- arima(y, order = c(1, 0, 0), method="ML")
fit.ar1
```

```
##
## Call:
## arima(x = y, order = c(1, 0, 0), method = "ML")
##
## Coefficients:
##          ar1  intercept
##       0.2337    0.0040
## s.e.  0.0661    0.0069
##
## sigma^2 estimated as 0.006778:  log likelihood = 258.72,  aic = -513.43
r.ar1 <- residuals(fit.ar1)
plot(r.ar1)
```



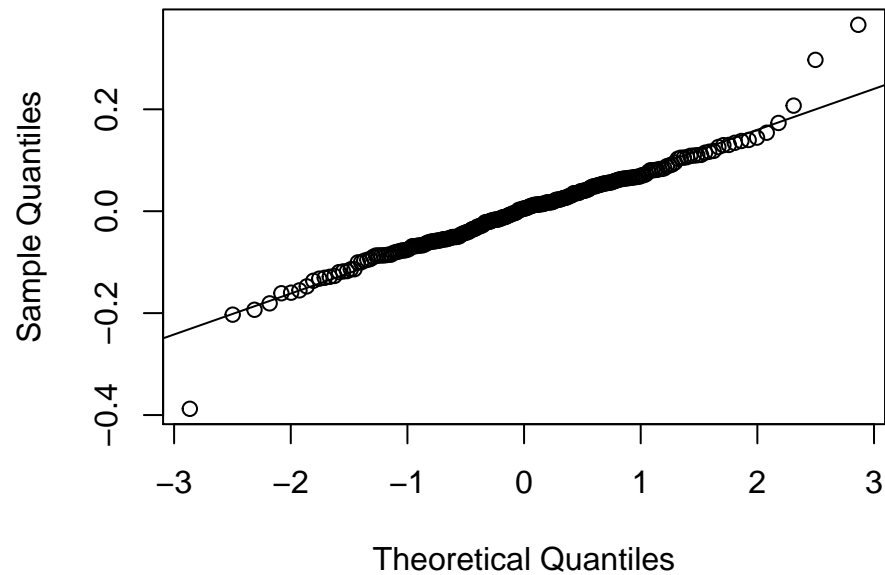
```
acf(r.ar1)
```

### Series r.ar1



```
qqn <- qqnorm(r.ar1)
qqline(r.ar1)
```

## Normal Q-Q Plot



```
r.squared.ar1 <- cor(qqn$x, qqn$y)
r.squared.ar1
```

```
## [1] 0.9810401
```

```
shapiro.test(r.ar1)
```

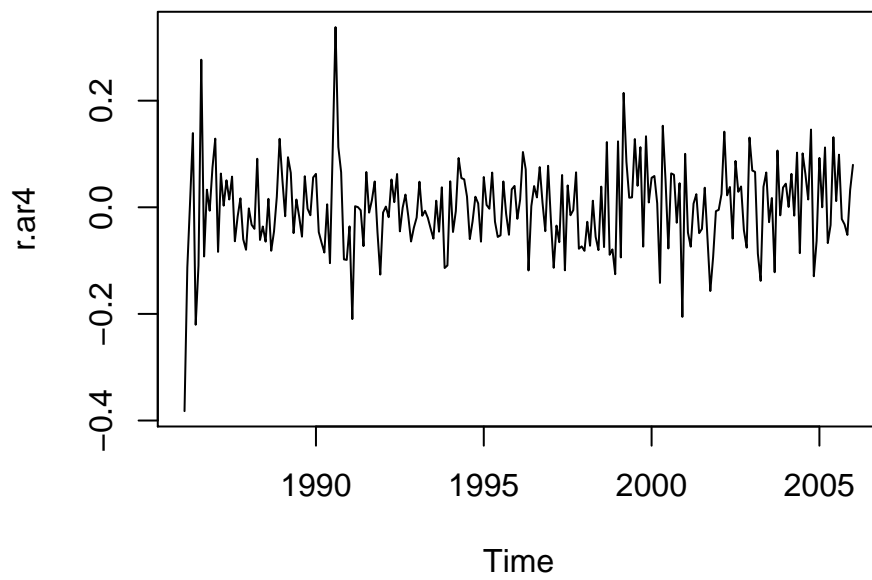
```
##
##  Shapiro-Wilk normality test
##
## data:  r.ar1
## W = 0.96683, p-value = 2.221e-05
```

## AR(4) Model

```
fit.ar4 <- arima(y, order = c(4, 0, 0), method="ML")
fit.ar4
```

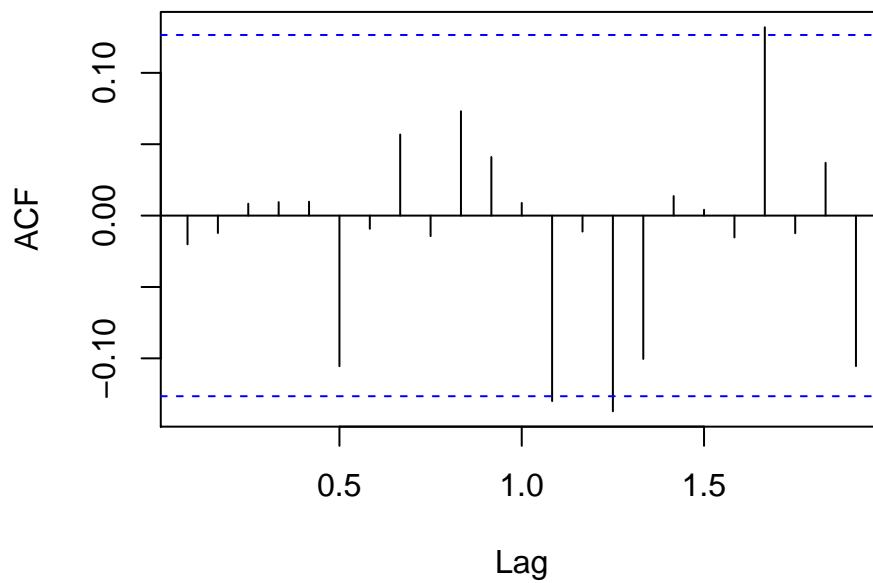
```
##
## Call:
## arima(x = y, order = c(4, 0, 0), method = "ML")
##
## Coefficients:
##      ar1      ar2      ar3      ar4  intercept
##    0.2638 -0.1583  0.0203 -0.1008    0.0045
## s.e.  0.0669  0.0691  0.0692  0.0681    0.0054
##
## sigma^2 estimated as 0.006584:  log likelihood = 262.16,  aic = -514.32
r.ar4 <- residuals(fit.ar4)
plot(r.ar4)
```





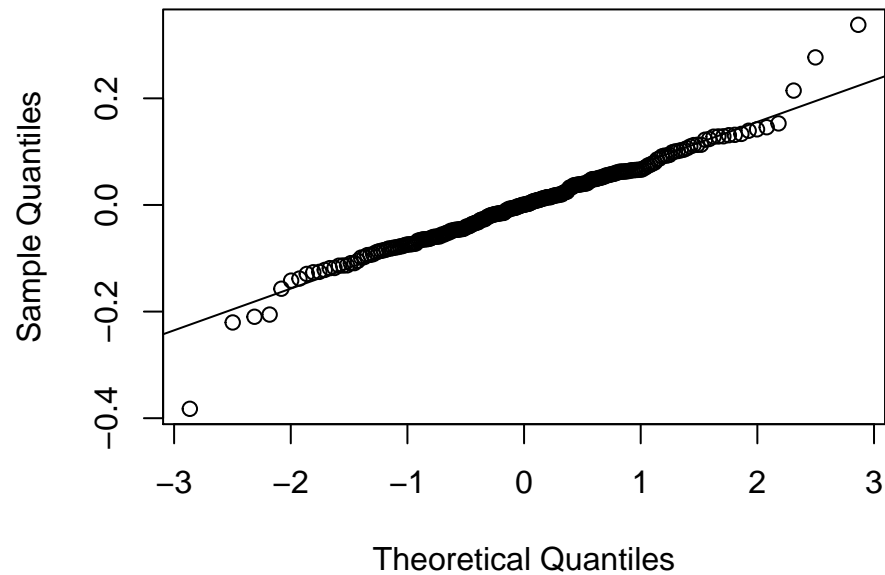
```
acf(r.ar4)
```

### Series r.ar4



```
qqn <- qqnorm(r.ar4)
qqline(r.ar4)
```

## Normal Q-Q Plot



```
r.squared.ar4 <- cor(qqn$x, qqn$y)
r.squared.ar4
```

```
## [1] 0.9827563
```

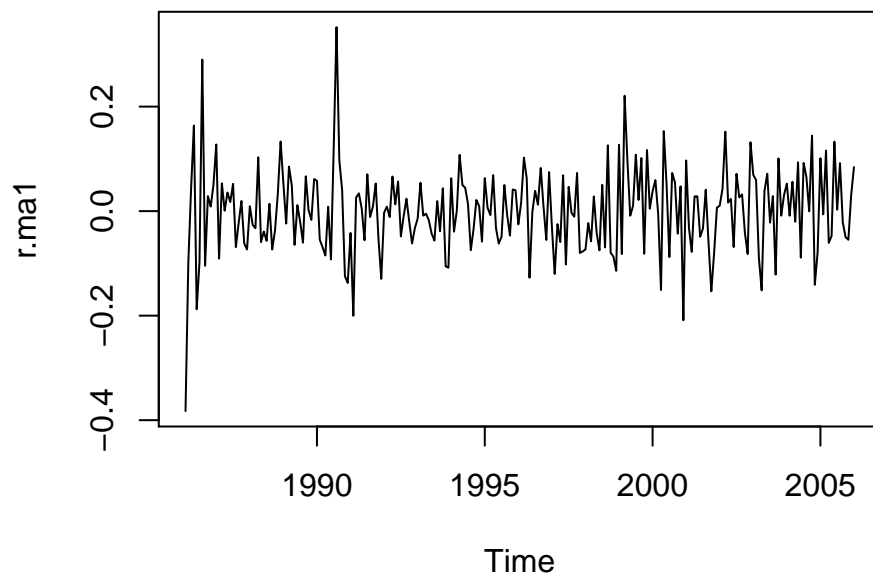
```
shapiro.test(r.ar4)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  r.ar4
## W = 0.96983, p-value = 5.597e-05
```

## MA(1) Model

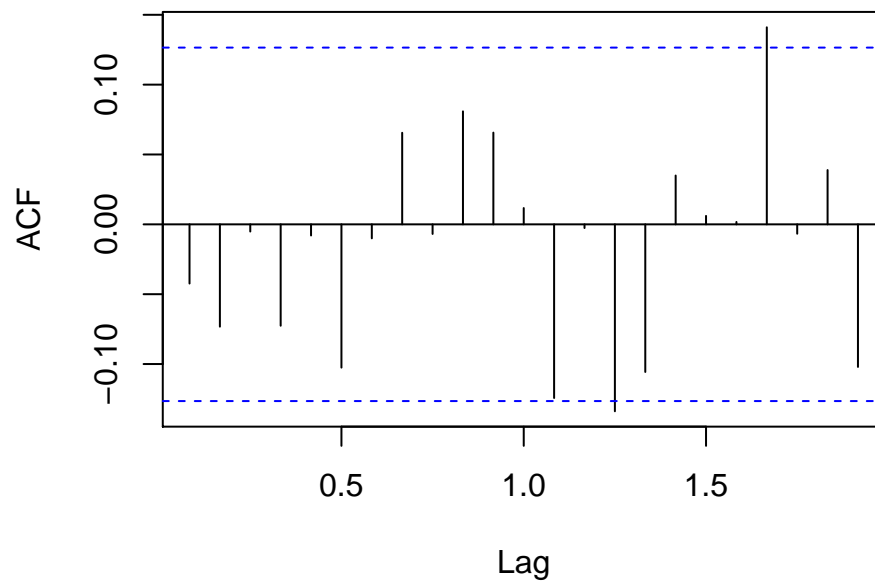
```
fit.ma1 <- arima(y, order = c(0, 0, 1), method="ML")
fit.ma1
```

```
##
## Call:
## arima(x = y, order = c(0, 0, 1), method = "ML")
##
## Coefficients:
##          ma1  intercept
##      0.2939    0.0041
## s.e.  0.0696    0.0068
##
## sigma^2 estimated as 0.006679:  log likelihood = 260.47,  aic = -516.94
r.ma1 <- residuals(fit.ma1)
plot(r.ma1)
```



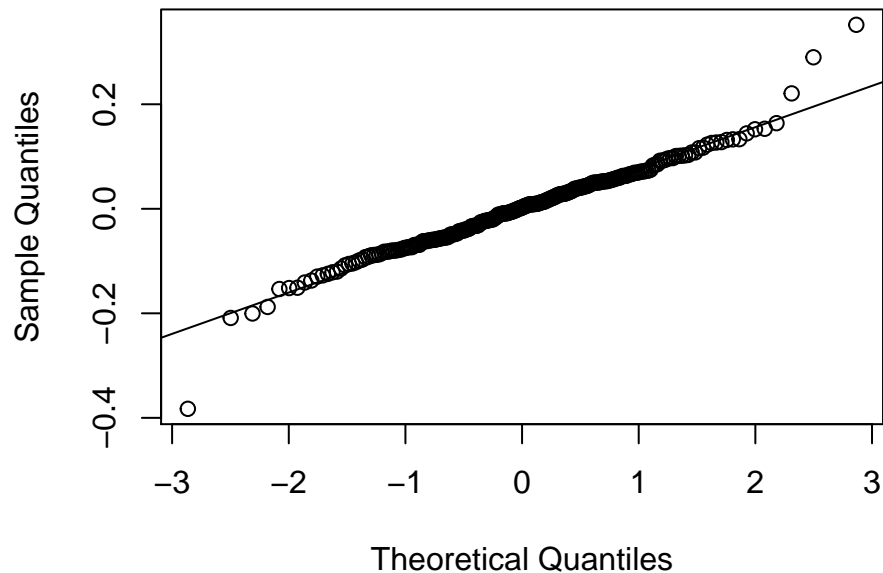
```
acf(r.ma1)
```

### Series r.ma1



```
qqn <- qqnorm(r.ma1)  
qqline(r.ma1)
```

## Normal Q-Q Plot



```
r.squared.ma1 <- cor(qqn$x, qqn$y)
r.squared.ma1
```

```
## [1] 0.9821782
```

```
shapiro.test(r.ma1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  r.ma1
## W = 0.96883, p-value = 4.095e-05
```

(a) Estimate all of these models using maximum likelihood and compare them using AIC.

We have the following AICs:

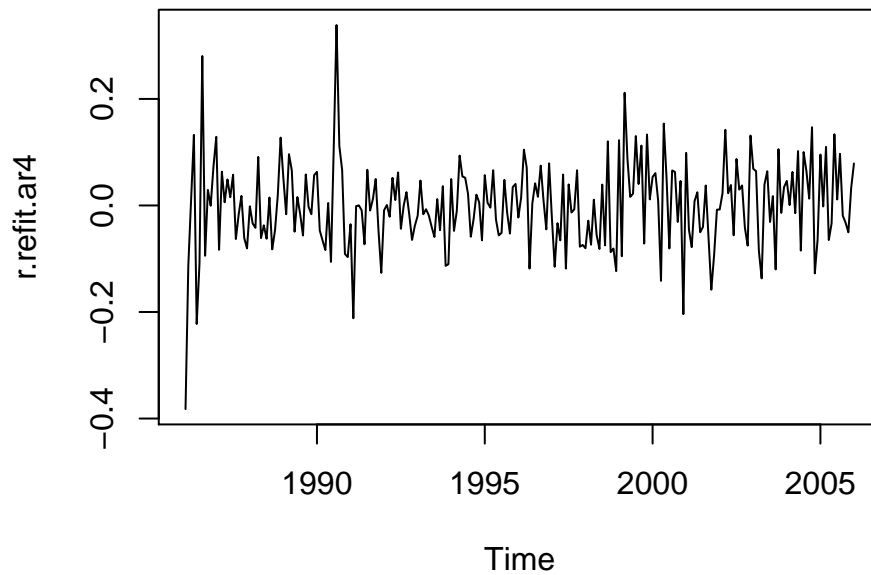
- AR(1): -513.4341766
- AR(4): -514.3213814
- MA(1): -516.9358391

(b) Are all parameters in each model significantly different from zero? (An easy check, are they at least 2 standard deviations away from 0)? If there are parameters that are not significantly different from zero, fit a new model with those parameters removed. Compare to the three models in part (a).

In the AR(4) model,  $\phi_3$  overlaps with 0, so we can fix it.

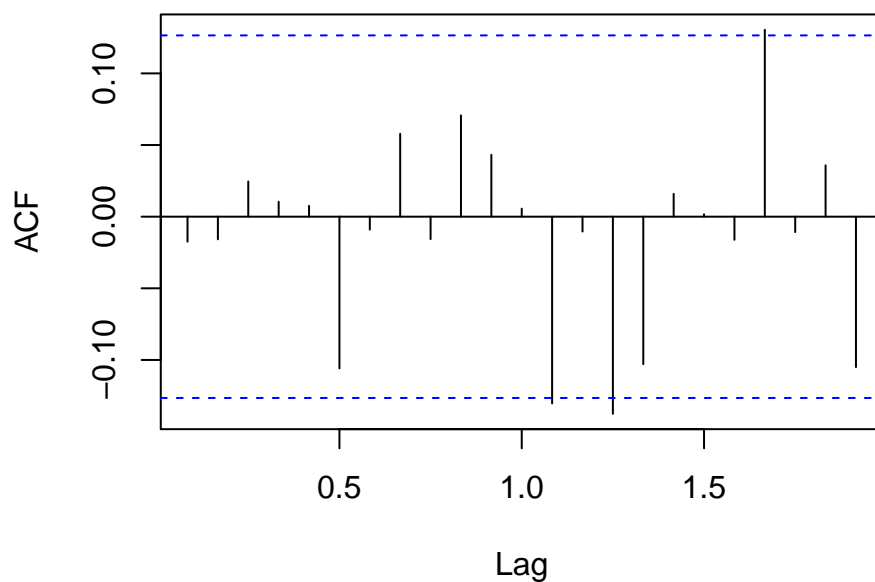
```
refit.ar4 <- arima(y, order = c(4, 0, 0), method="ML", fixed=c(NA, NA, 0, NA, NA), transform.pars=FALSE)
refit.ar4
```

```
##
## Call:
## arima(x = y, order = c(4, 0, 0), transform.pars = FALSE, fixed = c(NA, NA, 0,
##      NA, NA), method = "ML")
##
## Coefficients:
##          ar1      ar2  ar3      ar4  intercept
##      0.2611 -0.1537   0 -0.0966   0.0045
## s.e. 0.0663  0.0673   0  0.0666   0.0053
##
## sigma^2 estimated as 0.006586:  log likelihood = 262.12,  aic = -516.24
r.refit.ar4 <- residuals(refit.ar4)
plot(r.refit.ar4)
```



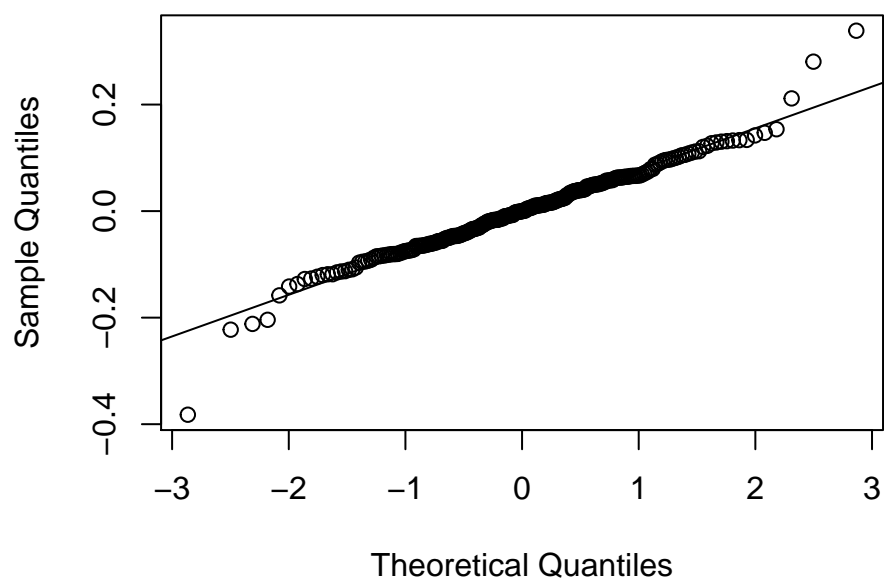
```
acf(r.refit.ar4)
```

### Series r.refit.ar4



```
qqn <- qqnorm(r.refit.ar4)
qqline(r.refit.ar4)
```

### Normal Q-Q Plot



```
r.squared.refit.ar4 <- cor(qqn$x, qqn$y)
r.squared.refit.ar4
```

```
## [1] 0.9824507
```

```
shapiro.test(r.refit.ar4)
```

```
##
## Shapiro-Wilk normality test
```

```
##  
## data:  r.refit.ar4  
## W = 0.96925, p-value = 4.663e-05
```

We can see that the refitted model has a smaller AIC than the original AR(4), and thus we would choose this model over the original.

**(c) Does it seem like the AR(4) is an overfit? If so, fit a more appropriate model instead.**

Yes. The better choice, according to AIC, is the simpler MA(1), which we have already fitted previously.

**(d) Perform the diagnostics on the AR(1), AR(4), MA(1), and any additional model fit in part (c).**

See above.

**(e) Which of the models you considered in (d) would you prefer?**

All of these models have a slight amount of autocorrelation in the residuals, but they are also normally distributed. Based on the AIC and the principle of parsimony, we would select the MA(1) model.

■

3. Recall the `days` dataset from homework assignment 6, and again replace the 63rd, 106th, and 129th observations with 35.

```
data(days)
days[63] <- 35
days[106] <- 35
days[129] <- 35
```

```
y <- days
```

*# Hey. I got drunk and deleted Snapchat the other night. It was draining my phone's battery and becoming*

(a) Fit an AR(2) and an MA(2) model. Compare them using AIC.

```
fit.ar2 <- arima(y, order = c(2, 0, 0), method="ML")
fit.ar2
```

```
##
## Call:
## arima(x = y, order = c(2, 0, 0), method = "ML")
##
## Coefficients:
##          ar1      ar2  intercept
##          0.1732  0.2055      28.2414
## s.e.    0.0863  0.0877      0.8060
##
## sigma^2 estimated as 32.99:  log likelihood = -411.79,  aic = 829.57
```

```
fit.ma2 <- arima(y, order = c(0, 0, 2), method="ML")
fit.ma2
```

```
##
## Call:
## arima(x = y, order = c(0, 0, 2), method = "ML")
##
## Coefficients:
##          ma1      ma2  intercept
##          0.1893  0.1958      28.1957
## s.e.    0.0894  0.0740      0.6980
##
## sigma^2 estimated as 33.22:  log likelihood = -412.23,  aic = 830.45
```

The AR(2) has a slightly smaller AIC than the MA(1).

(b) Use the four different Bootstrap resampling methods to get the 95% confidence intervals for  $\theta_1, \theta_2, \mu, \sigma_e^2$  in the MA(2) model.

Bootstrap Method 1

```
b1.theta1.sim <- rep(NA, 1000)
b1.theta2.sim <- rep(NA, 1000)
```



```

b1.mu.sim <- rep(NA, 1000)
b1.sigmae2.sim <- rep(NA, 1000)

b1 <- y
f <- arima(b1, order=c(0, 0, 2))
theta <- f$coef

for (i in 1:1000) {

  b1 <- y

  e <- rep(NA, length(y) - 2)
  for (j in 3:length(y)) {
    e[j - 2] <- rnorm(1)
    b1[j] <- sum(theta[1:2] * (b1[(j - 1):(j - 2)] - theta[3])) + theta[3] + e[j - 2]
  }

  f1 <- arima(b1, order=c(3, 0, 0), method="ML")
  b1.theta1.sim[i] <- f1$coef[[1]]
  b1.theta2.sim[i] <- f1$coef[[2]]
  b1.mu.sim[i] <- mean(b1)
  b1.sigmae2.sim[i] <- var(e)
}

```

```

## Warning in stats:::arima(x = x, order = order, seasonal = seasonal, xreg =
## xreg, : possible convergence problem: optim gave code = 1

```

```

## Warning in stats:::arima(x = x, order = order, seasonal = seasonal, xreg =
## xreg, : possible convergence problem: optim gave code = 1

```

```

## Warning in stats:::arima(x = x, order = order, seasonal = seasonal, xreg =
## xreg, : possible convergence problem: optim gave code = 1

```

```

## Warning in stats:::arima(x = x, order = order, seasonal = seasonal, xreg =
## xreg, : possible convergence problem: optim gave code = 1

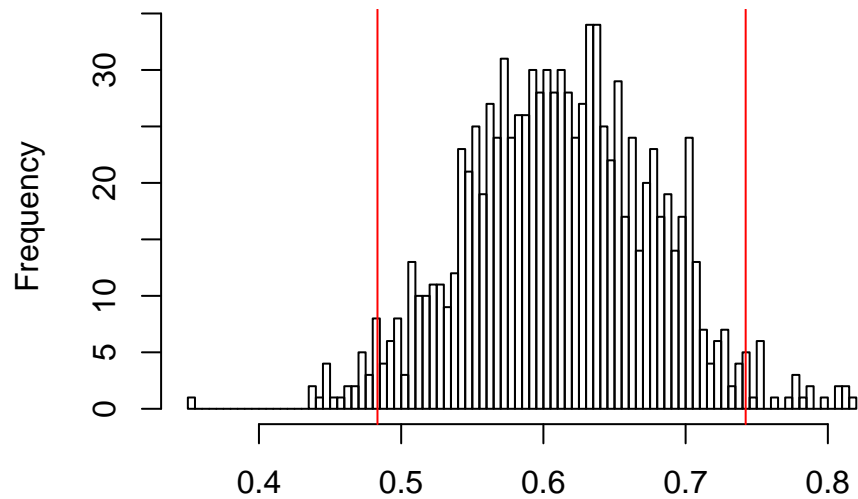
```

```

hist(b1.theta1.sim, main="Bootstrap Method 1, Theta 1", breaks=100)
theta1.ci.lower <- quantile(b1.theta1.sim, c(0.025,0.975))[[1]]
theta1.ci.upper <- quantile(b1.theta1.sim, c(0.025,0.975))[[2]]
abline(v=theta1.ci.lower, col="red")
abline(v=theta1.ci.upper, col="red")

```

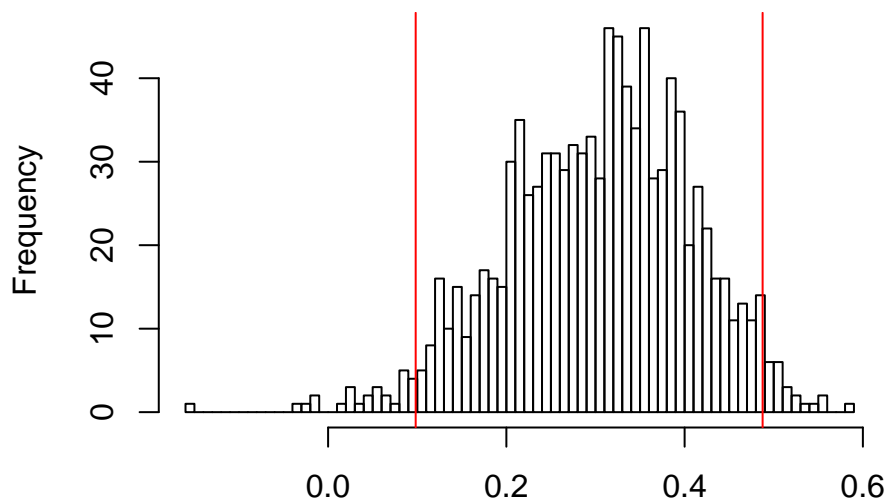
## Bootstrap Method 1, Theta 1



b1.theta1.sim

```
hist(b1.theta2.sim, main="Bootstrap Method 1, Theta 2", breaks=100)
theta2.ci.lower <- quantile(b1.theta2.sim, c(0.025,0.975))[[1]]
theta2.ci.upper <- quantile(b1.theta2.sim, c(0.025,0.975))[[2]]
abline(v=theta2.ci.lower, col="red")
abline(v=theta2.ci.upper, col="red")
```

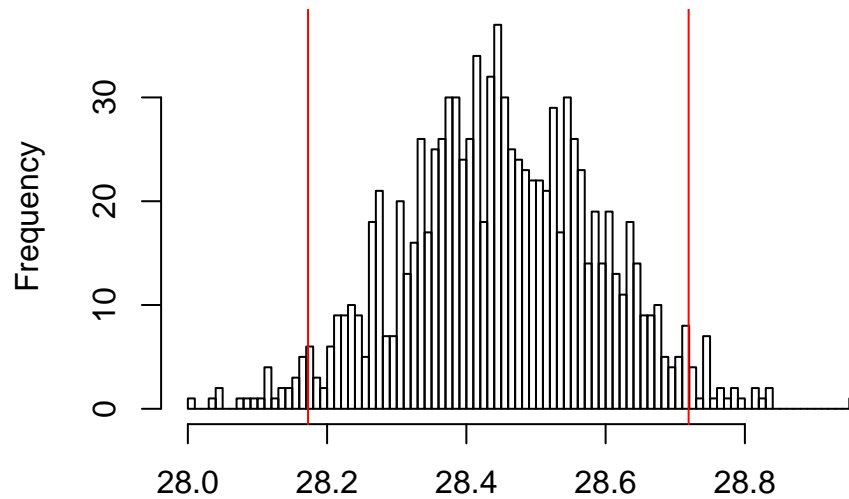
## Bootstrap Method 1, Theta 2



b1.theta2.sim

```
hist(b1.mu.sim, main="Bootstrap Method 1, Mu", breaks=100)
mu.ci.lower <- quantile(b1.mu.sim, c(0.025,0.975))[[1]]
mu.ci.upper <- quantile(b1.mu.sim, c(0.025,0.975))[[2]]
abline(v=mu.ci.lower, col="red")
abline(v=mu.ci.upper, col="red")
```

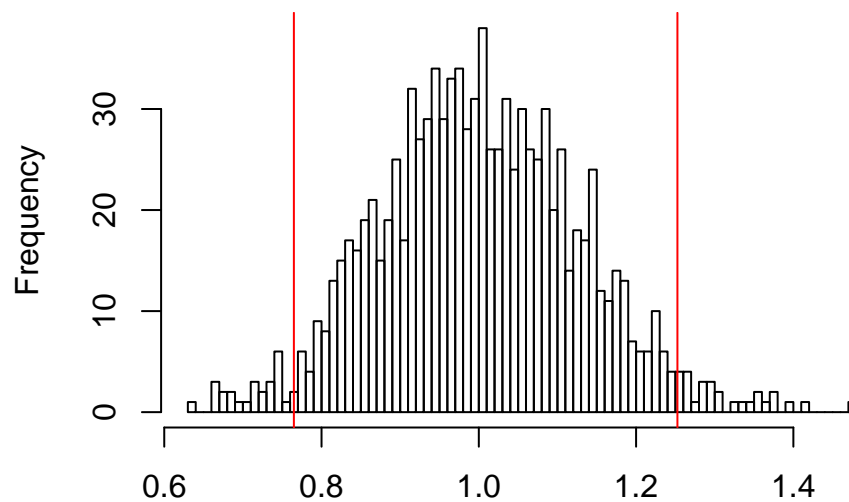
### Bootstrap Method 1, Mu



b1.mu.sim

```
hist(b1.sigmae2.sim, main="Bootstrap Method 1, Sigma_e^2", breaks=100)
sigmae2.ci.lower <- quantile(b1.sigmae2.sim, c(0.025,0.975))[[1]]
sigmae2.ci.upper <- quantile(b1.sigmae2.sim, c(0.025,0.975))[[2]]
abline(v=sigmae2.ci.lower, col="red")
abline(v=sigmae2.ci.upper, col="red")
```

### Bootstrap Method 1, Sigma\_e^2



b1.sigmae2.sim

### Bootstrap Method 2

```
b2.theta1.sim <- rep(NA, 1000)
b2.theta2.sim <- rep(NA, 1000)
```

```

b2.mu.sim <- rep(NA, 1000)
b2.sigmae2.sim <- rep(NA, 1000)

b2 <- y
f <- arima(b2, order=c(0, 0, 2))
theta <- f$coef

for (j in 1:1000) {

  b2 <- y[1:3]
  for (i in 4:length(y)){
    b2[i] = sum(theta[1:2] * (b2[(i - 1):(i - 2)] - theta[3])) + theta[3] + sample(f$residuals, 1)
  }

  e <- rep(NA, length(y) - 2)
  for (i in 3:length(y)) {
    e[i - 2] <- rnorm(1)
    b2[i] <- sum(theta[1:2] * (b2[(i - 1):(i - 2)] - theta[3])) + theta[3] + e[i - 2]
  }

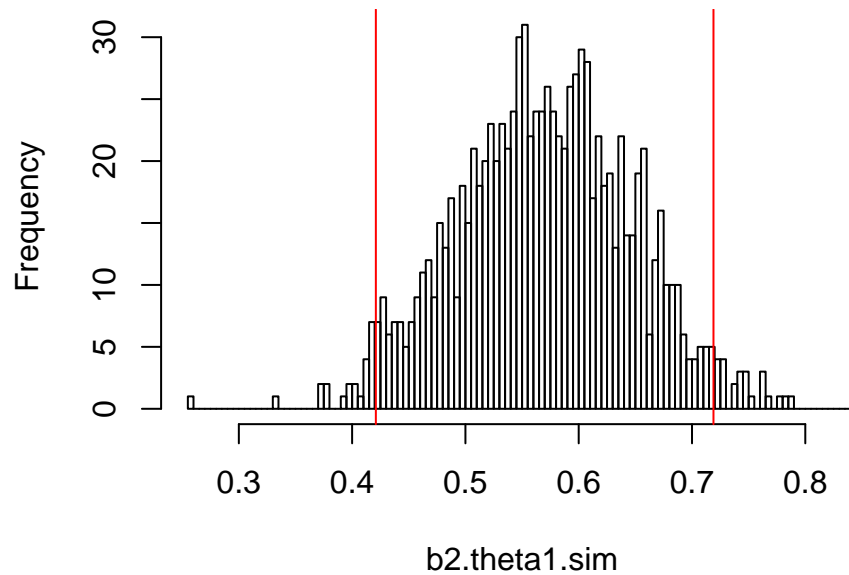
  f1 <- arima(b2, order=c(0, 0, 2), method="ML")
  b2.theta1.sim[j] <- f1$coef[[1]]
  b2.theta2.sim[j] <- f1$coef[[2]]
  b2.mu.sim[j] <- mean(b2)
  b2.sigmae2.sim[j] <- var(e)

}

hist(b2.theta1.sim, main="Bootstrap Method 2, Theta 1", breaks=100)
theta1.ci.lower <- quantile(b2.theta1.sim, c(0.025,0.975))[[1]]
theta1.ci.upper <- quantile(b2.theta1.sim, c(0.025,0.975))[[2]]
abline(v=theta1.ci.lower, col="red")
abline(v=theta1.ci.upper, col="red")

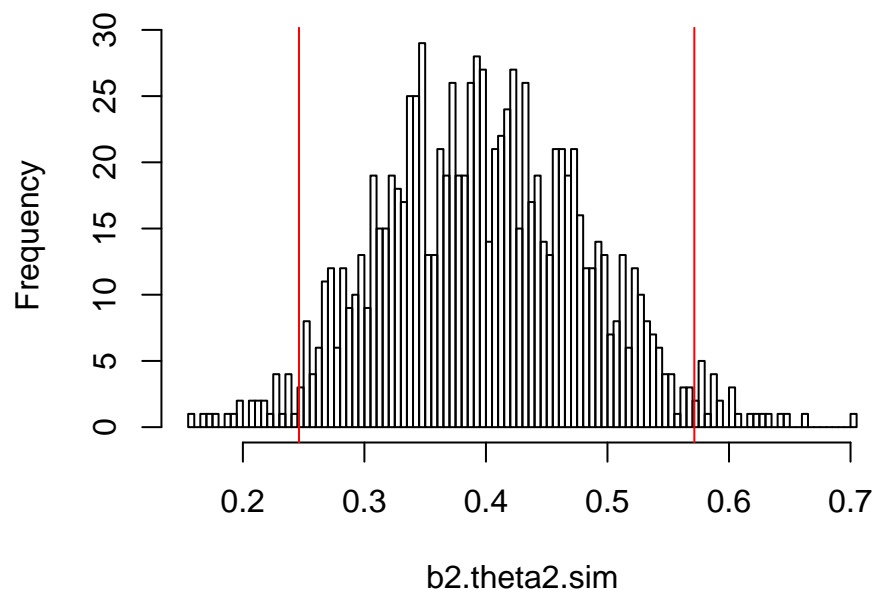
```

## Bootstrap Method 2, Theta 1



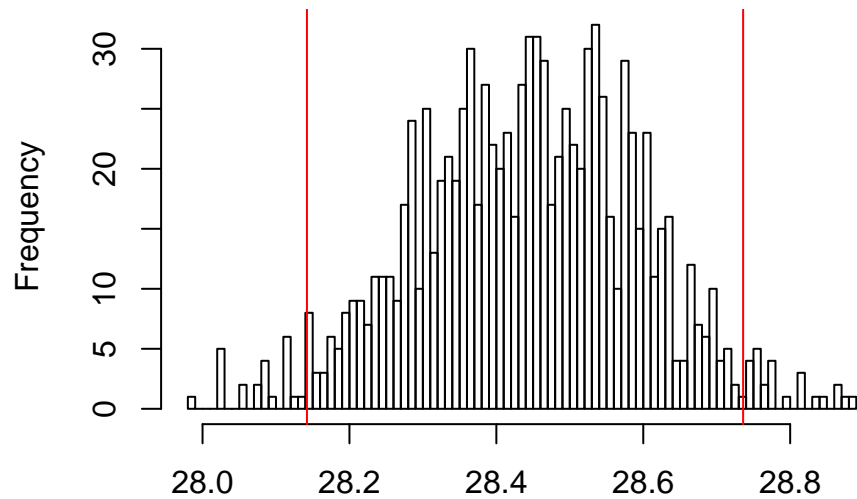
```
hist(b2.theta2.sim, main="Bootstrap Method 2, Theta 2", breaks=100)
theta2.ci.lower <- quantile(b2.theta2.sim, c(0.025,0.975))[[1]]
theta2.ci.upper <- quantile(b2.theta2.sim, c(0.025,0.975))[[2]]
abline(v=theta2.ci.lower, col="red")
abline(v=theta2.ci.upper, col="red")
```

## Bootstrap Method 2, Theta 2



```
hist(b2.mu.sim, main="Bootstrap Method 2, Mu", breaks=100)
mu.ci.lower <- quantile(b2.mu.sim, c(0.025,0.975))[[1]]
mu.ci.upper <- quantile(b2.mu.sim, c(0.025,0.975))[[2]]
abline(v=mu.ci.lower, col="red")
abline(v=mu.ci.upper, col="red")
```

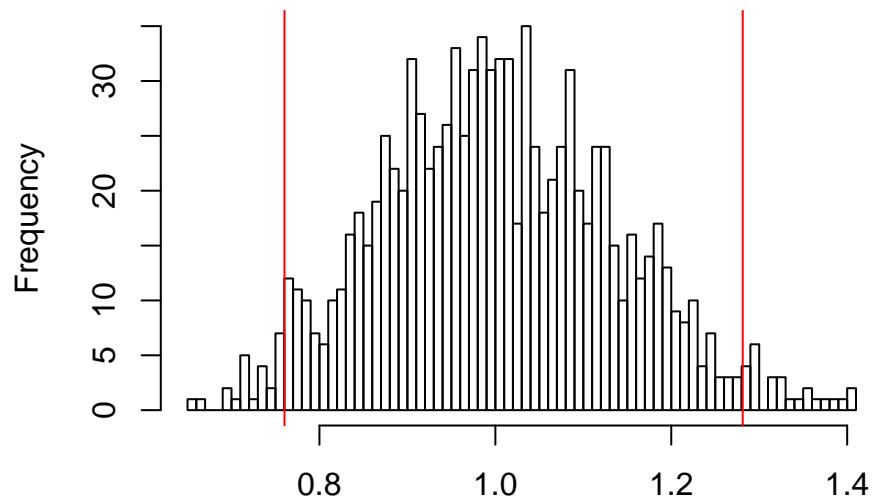
### Bootstrap Method 2, Mu



b2.mu.sim

```
hist(b2.sigmae2.sim, main="Bootstrap Method 2, Sigma_e^2", breaks=100)
sigmae2.ci.lower <- quantile(b2.sigmae2.sim, c(0.025,0.975))[[1]]
sigmae2.ci.upper <- quantile(b2.sigmae2.sim, c(0.025,0.975))[[2]]
abline(v=sigmae2.ci.lower, col="red")
abline(v=sigmae2.ci.upper, col="red")
```

### Bootstrap Method 2, Sigma\_e^2



b2.sigmae2.sim

### Bootstrap Method 3

```
b3.theta1.sim <- rep(NA, 1000)
b3.theta2.sim <- rep(NA, 1000)
```

```

b3.mu.sim <- rep(NA, 1000)
b3.sigmae2.sim <- rep(NA, 1000)

b3 <- y
f <- arima(b3, order=c(0, 0, 2))
theta <- f$coef

for (j in 1:1000) {

  b3 <- y[1:2]
  for (i in 3:length(y)){
    b3 <- arima.sim(n=100, model=list(ma=theta[1:2]), sd=sqrt(f$sigma2)) + theta[3]
    b3 <- b3[-(1:(100-31))]
  }

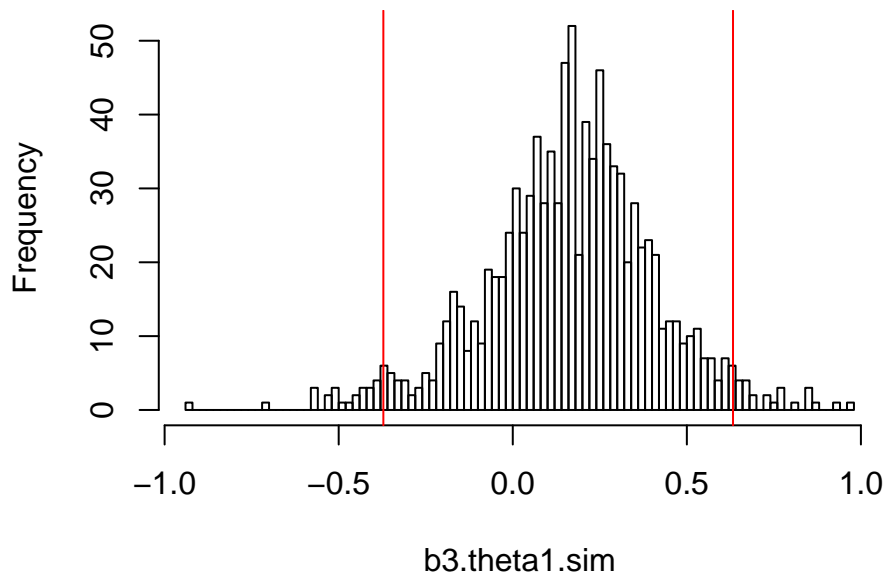
  f1 <- arima(b3, order=c(0, 0, 2), method="ML")
  b3.theta1.sim[j] <- f1$coef[[1]]
  b3.theta2.sim[j] <- f1$coef[[2]]
  b3.mu.sim[j] <- mean(b3)
  b3.sigmae2.sim[j] <- var(resid(f1))

}

hist(b3.theta1.sim, main="Bootstrap Method 3, Theta 1", breaks=100)
theta1.ci.lower <- quantile(b3.theta1.sim, c(0.025,0.975))[[1]]
theta1.ci.upper <- quantile(b3.theta1.sim, c(0.025,0.975))[[2]]
abline(v=theta1.ci.lower, col="red")
abline(v=theta1.ci.upper, col="red")

```

### Bootstrap Method 3, Theta 1



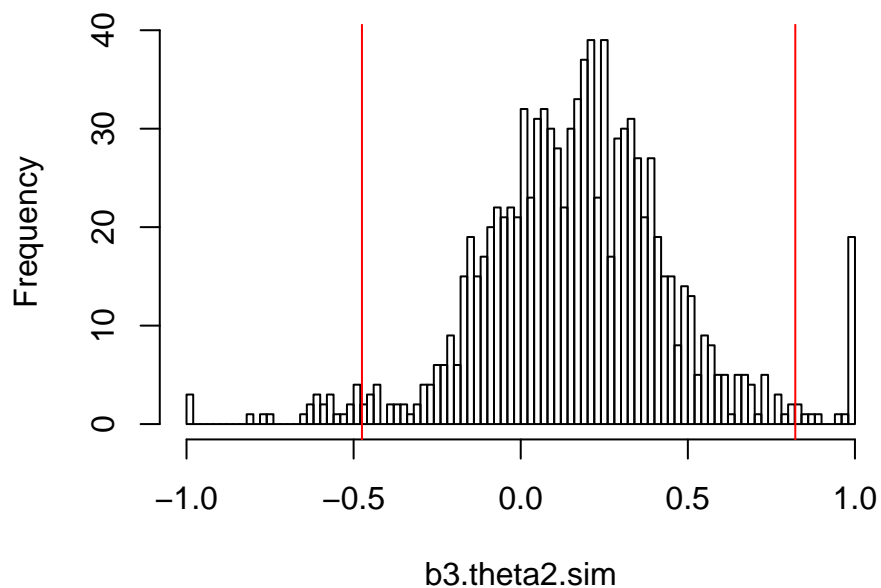
```

hist(b3.theta2.sim, main="Bootstrap Method 3, Theta 2", breaks=100)
theta2.ci.lower <- quantile(b3.theta2.sim, c(0.025,0.975))[[1]]

```

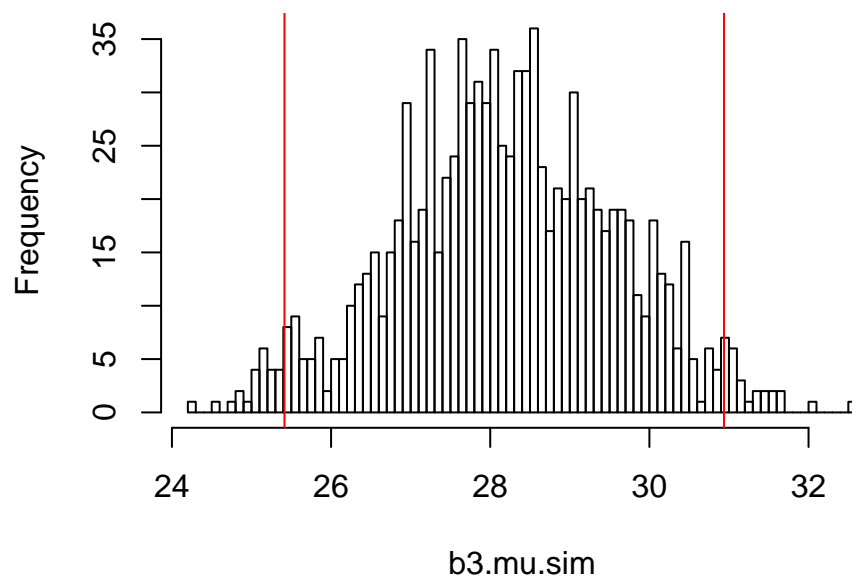
```
theta2.ci.upper <- quantile(b3.theta2.sim, c(0.025,0.975))[[2]]
abline(v=theta2.ci.lower, col="red")
abline(v=theta2.ci.upper, col="red")
```

### Bootstrap Method 3, Theta 2



```
hist(b3.mu.sim, main="Bootstrap Method 3, Mu", breaks=100)
mu.ci.lower <- quantile(b3.mu.sim, c(0.025,0.975))[[1]]
mu.ci.upper <- quantile(b3.mu.sim, c(0.025,0.975))[[2]]
abline(v=mu.ci.lower, col="red")
abline(v=mu.ci.upper, col="red")
```

### Bootstrap Method 3, Mu



```
hist(b3.sigmae2.sim, main="Bootstrap Method 3, Sigma_e^2", breaks=100)
```

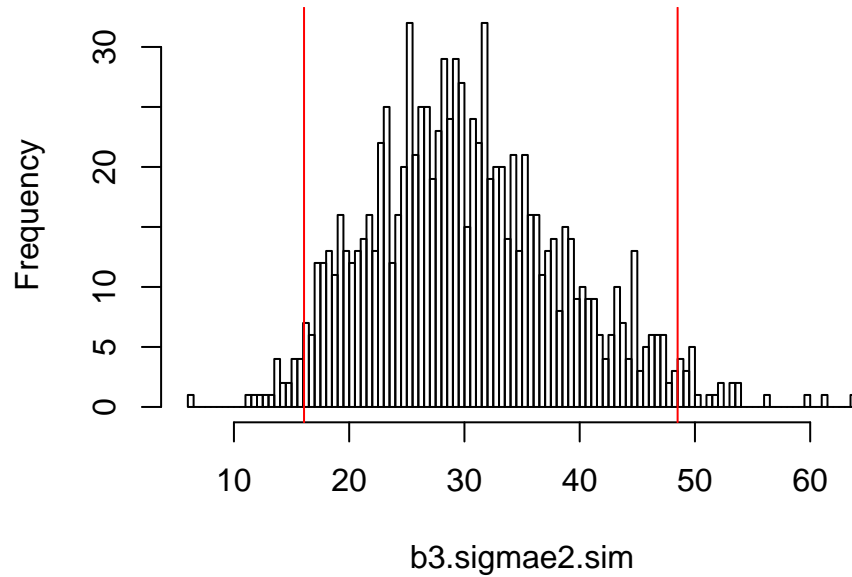


```

sigmae2.ci.lower <- quantile(b3.sigmae2.sim, c(0.025,0.975))[[1]]
sigmae2.ci.upper <- quantile(b3.sigmae2.sim, c(0.025,0.975))[[2]]
abline(v=sigmae2.ci.lower, col="red")
abline(v=sigmae2.ci.upper, col="red")

```

### Bootstrap Method 3, Sigma\_e^2



We have the following 95% confidence intervals:

$$\theta_1 : (-0.3716978, 0.6325969)$$

$$\theta_2 : (-0.4750161, 0.8215227)$$

$$\mu : (25.414245, 30.9379105)$$

$$\sigma_e^2 : (16.0832109, 48.4955137)$$

## Bootstrap Method 4

```
b4.theta1.sim <- rep(NA, 1000)
b4.theta2.sim <- rep(NA, 1000)
b4.mu.sim <- rep(NA, 1000)
b4.sigmae2.sim <- rep(NA, 1000)

b4 <- y
f <- arima(b4, order=c(0, 0, 2))
theta <- f$coef

for (j in 1:1000) {

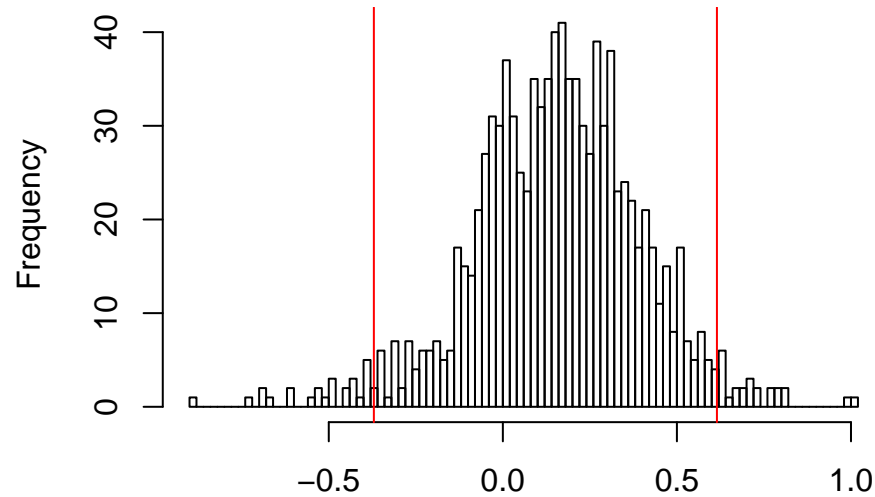
  b4 <- y[1:3]
  for (i in 4:length(y)){
    b4 <- arima.sim(n=100, model=list(ma=theta[1:2]), rand.gen=function(n, r=f$residuals){ sample(r, n,
    b4 <- b4[-(1:(100 - 31))])
  }

  f1 <- arima(b4, order=c(0, 0, 2), method="ML")
  b4.theta1.sim[j] <- f1$coef[[1]]
  b4.theta2.sim[j] <- f1$coef[[2]]
  b4.mu.sim[j] <- mean(b4)
  b4.sigmae2.sim[j] <- var(resid(f1))

}

hist(b4.theta1.sim, main="Bootstrap Method 4, Theta 1", breaks=100)
theta1.ci.lower <- quantile(b4.theta1.sim, c(0.025,0.975))[[1]]
theta1.ci.upper <- quantile(b4.theta1.sim, c(0.025,0.975))[[2]]
abline(v=theta1.ci.lower, col="red")
abline(v=theta1.ci.upper, col="red")
```

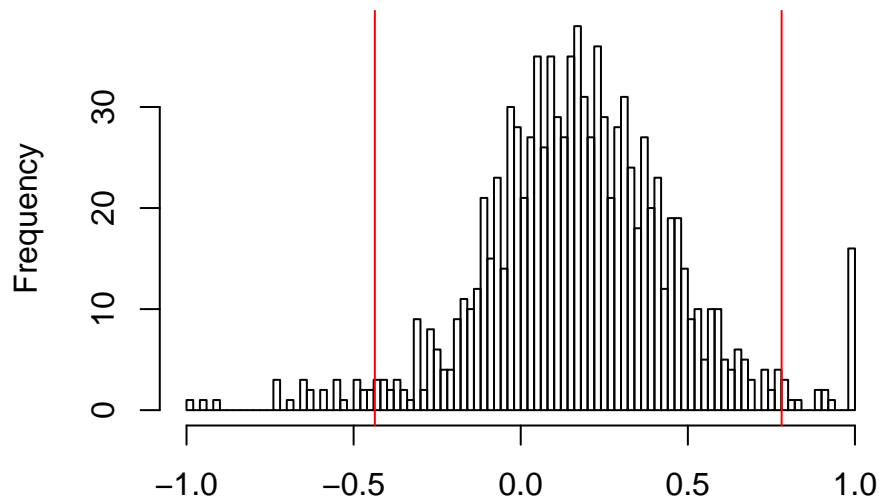
## Bootstrap Method 4, Theta 1



b4.theta1.sim

```
hist(b4.theta2.sim, main="Bootstrap Method 4, Theta 2", breaks=100)
theta2.ci.lower <- quantile(b4.theta2.sim, c(0.025,0.975))[[1]]
theta2.ci.upper <- quantile(b4.theta2.sim, c(0.025,0.975))[[2]]
abline(v=theta2.ci.lower, col="red")
abline(v=theta2.ci.upper, col="red")
```

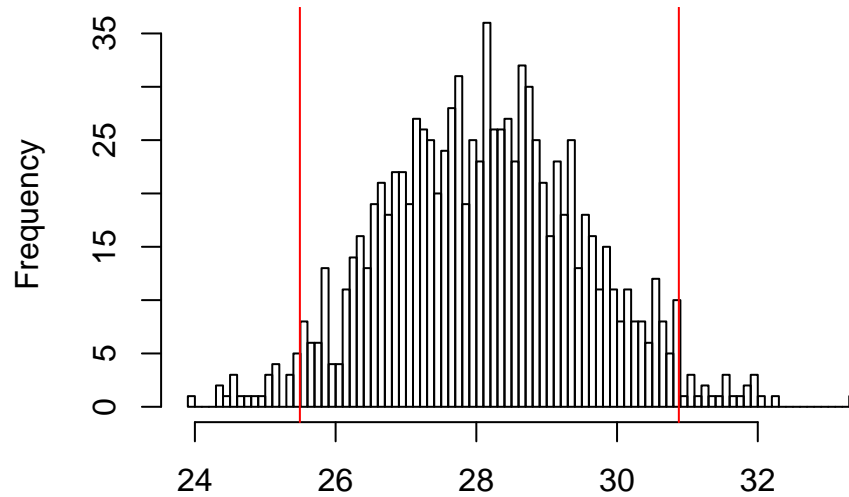
## Bootstrap Method 4, Theta 2



b4.theta2.sim

```
hist(b4.mu.sim, main="Bootstrap Method 4, Mu", breaks=100)
mu.ci.lower <- quantile(b4.mu.sim, c(0.025,0.975))[[1]]
mu.ci.upper <- quantile(b4.mu.sim, c(0.025,0.975))[[2]]
abline(v=mu.ci.lower, col="red")
abline(v=mu.ci.upper, col="red")
```

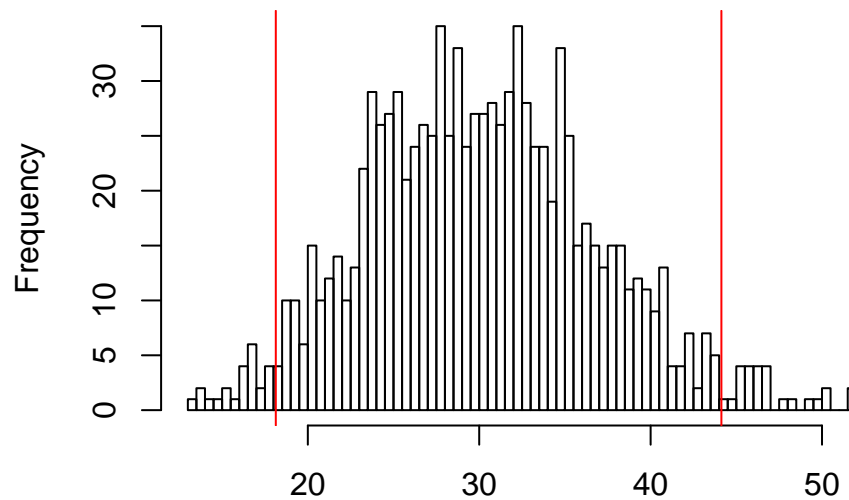
### Bootstrap Method 4, Mu



b4.mu.sim

```
hist(b4.sigmae2.sim, main="Bootstrap Method 4, Sigma_e^2", breaks=100)
sigmae2.ci.lower <- quantile(b4.sigmae2.sim, c(0.025,0.975))[[1]]
sigmae2.ci.upper <- quantile(b4.sigmae2.sim, c(0.025,0.975))[[2]]
abline(v=sigmae2.ci.lower, col="red")
abline(v=sigmae2.ci.upper, col="red")
```

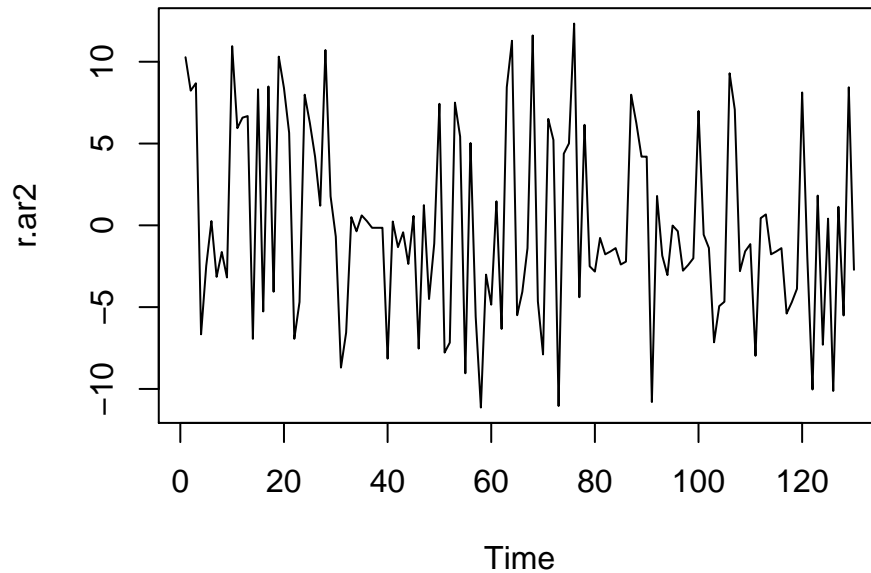
### Bootstrap Method 4, Sigma\_e^2



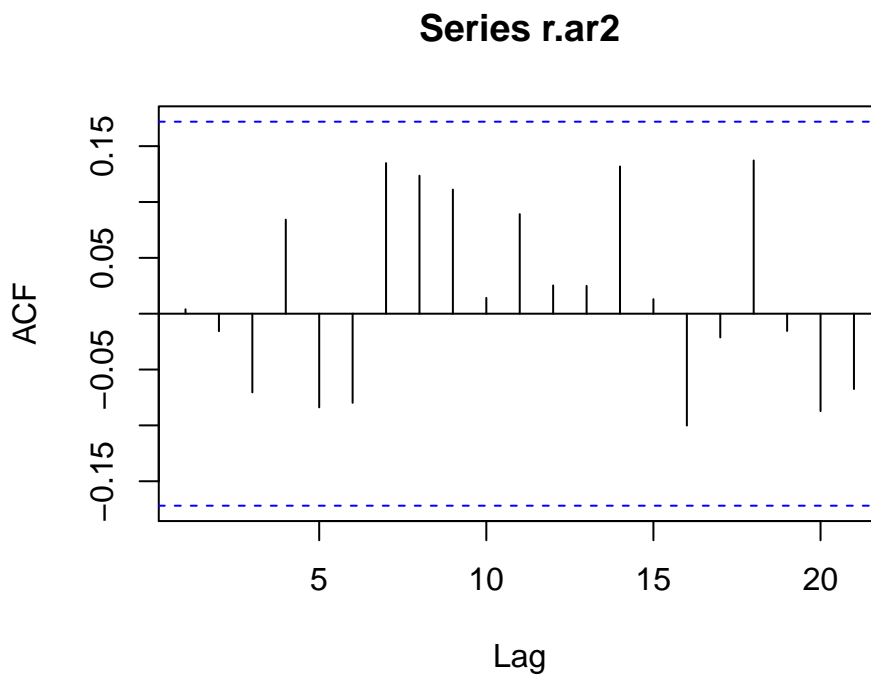
b4.sigmae2.sim

(c) Use the diagnostics discussed in Chapter 8 to analyze the suitability of the AR(2) and MA(2) models.

```
r.ar2 <- residuals(fit.ar2)
plot(r.ar2)
```

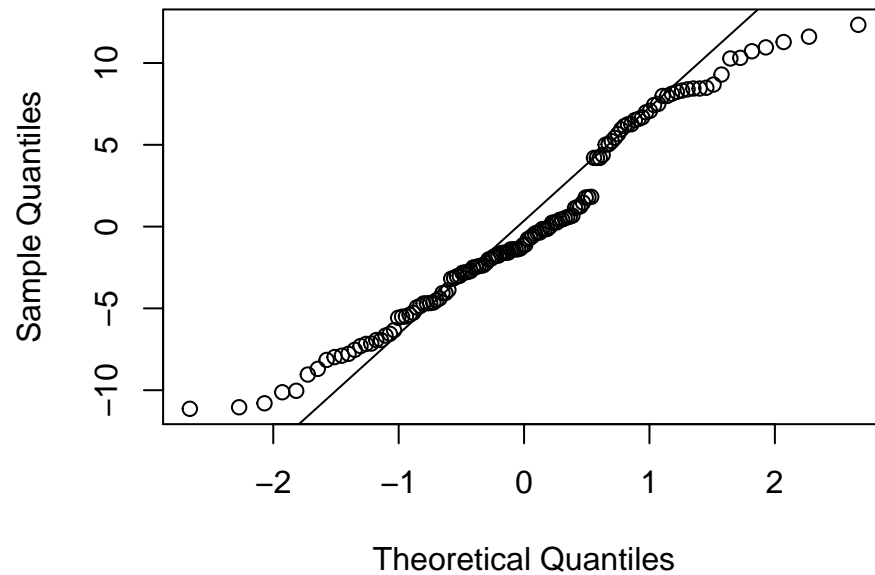


```
acf(r.ar2)
```



```
qqn <- qqnorm(r.ar2)
qqline(r.ar2)
```

## Normal Q-Q Plot



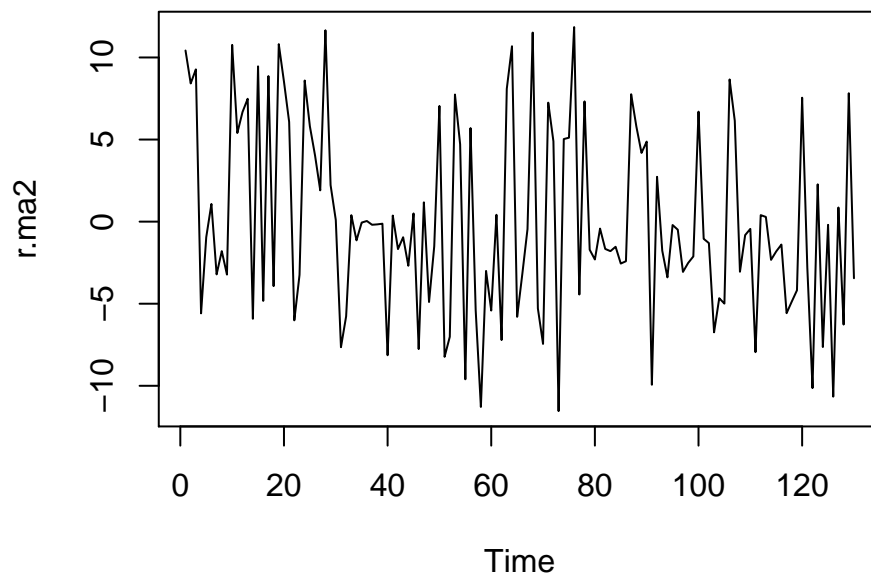
```
r.squared.r.ar2 <- cor(qqn$x, qqn$y)
r.squared.r.ar2
```

```
## [1] 0.9845098
```

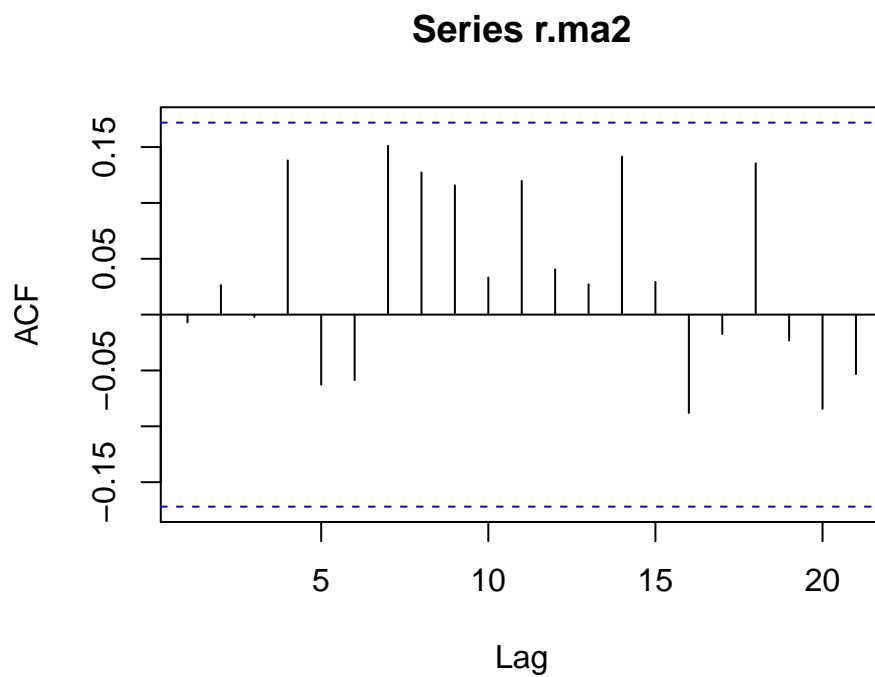
```
shapiro.test(r.ar2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  r.ar2
## W = 0.96617, p-value = 0.002488
```

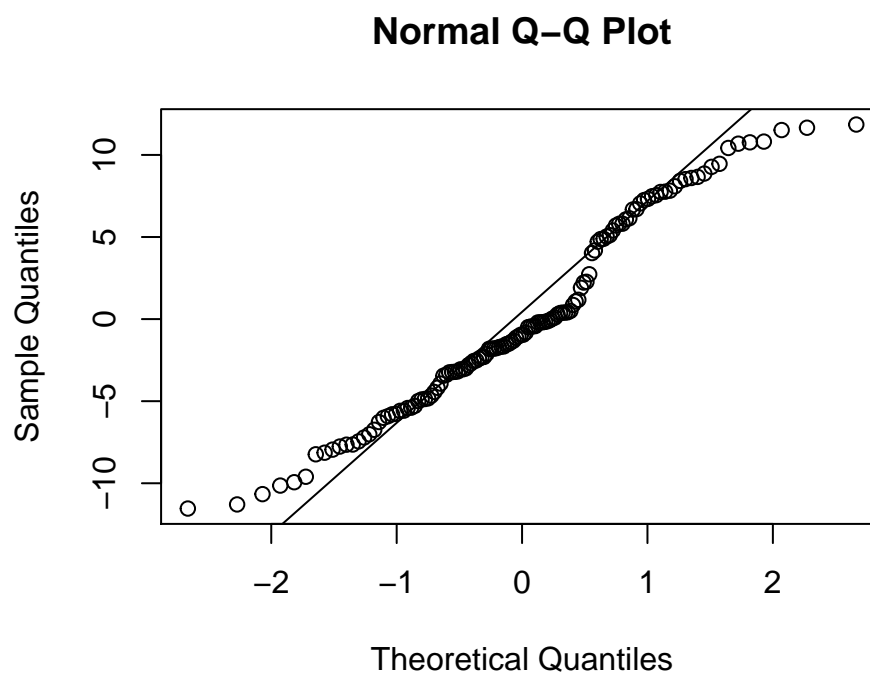
```
r.ma2 <- residuals(fit.ma2)
plot(r.ma2)
```



```
acf(r.ma2)
```



```
qqn <- qqnorm(r.ma2)  
qqline(r.ma2)
```



```
r.squared.r.ma2 <- cor(qqn$x, qqn$y)  
r.squared.r.ma2
```

```
## [1] 0.984269
```

```
shapiro.test(r.ma2)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data:  r.ma2  
## W = 0.96563, p-value = 0.002228
```

Both of the residual plots look good with very little autocorrelation and Shapiro-Wilk test p-values  $< 0.05$ , so they are normally distributed. We would choose the AR(2) based off of the AIC.

