Homework 6

Benjamin Noland

1. The following code generates 100 observations from the MA(1) model $Y_t = e_t + 0.7e_{t-1}$:

```
set.seed(1000)
y <- arima.sim(model = list(ma = c(0.7)), n = 100)
```

Let $Y_1, \dots, Y_{100}$ denote the observed time series, and let Σ denote the covariance matrix of $Y = (Y_1, \dots, Y_{100})$.

a. The covariance matrix Σ is given by $\Sigma_{ij} = (\gamma_{|i-j|})$. Explicitly,

$$\Sigma_{ij} = \begin{cases} (1 + \theta_1^2 + \theta_2^2)\sigma_e^2 & \text{if } i = j \\ (-\theta_1 + \theta_1\theta_2)\sigma_e^2 & \text{if } |i-j| = 1 \\ -\theta_2\sigma_e^2 & \text{if } |i-j| = 2 \\ 0 & \text{otherwise} \end{cases}.$$

b. Assume the elements of the white noise process $\{e_t\}$ are drawn independently from a $\mathrm{N}(0, \sigma_e^2)$ distribution (this is the case for the simulated data generated above). Then the sample $Y = (Y_1, \dots, Y_{100})$ has a mean zero multivariate normal distribution with covariance matrix Σ. Let $y = (y_1, \dots, y_{100})$ a denote realization of the sample Y. Then the likelihood function is given by

$$L(\theta_1, \theta_2 | y) = \frac{1}{\sqrt{(2\pi)^{100}|\Sigma|}} \exp\left(-\frac{1}{2} y^T \Sigma^{-1} y\right).$$

Maximizing $L(\theta_1, \theta_2|y)$ (or equivalently, the log-likelihood $\log L(\theta_1, \theta_2|y)$) with respect to (θ_1, θ_2) yields a maximum likelihood estimate (MLE) $(\hat{\theta}_1, \hat{\theta}_2)$ of (θ_1, θ_2). Note that, as computed in part (a), the covariance matrix Σ depends on θ_1 and θ_2.

c. The following code fits an MA(2) model to the data using maximum likelihood:

```
arima(y, order = c(0, 0, 2), method = "ML")
```

```
##
## Call:
## arima(x = y, order = c(0, 0, 2), method = "ML")
##
## Coefficients:
##          ma1      ma2  intercept
##        0.525  -0.184      0.011
## s.e.   0.107   0.110      0.134
##
## sigma^2 estimated as 0.995:  log likelihood = -142,  aic = 290
```

From the output, we see that the maximum likelihood estimates for θ_1 and θ_2 are $\hat{\theta}_1 = -0.5246$ and $\hat{\theta}_2 = 0.1835$, respectively (taking into account the differences in convention between R and the book with regard to the signs of the parameters).

d. From the output in part (c), we see that $\hat{\theta}_1$ and $\hat{\theta}_2$ have approximate standard errors of 0.1072 and 0.1101, respectively.

2. (Cryer & Chan, Exercise 7.1) We can compute method of moments estimates $\hat{\phi}_1$ and $\hat{\phi}_2$ of ϕ_1 and ϕ_2, respectively, by solving for $\hat{\phi}_1$ and $\hat{\phi}_2$ in the sample Yule-Walker equations:

$$\begin{aligned} r_1 &= \hat{\phi}_1 + r_1\hat{\phi}_2 \\ r_2 &= r_1\hat{\phi}_1 + \hat{\phi}_2 \end{aligned}.$$

We get

$$\begin{aligned} \hat{\phi}_1 &= \frac{r_1(1-r_2)}{1-r_1^2} = \\ \hat{\phi}_2 &= \frac{r_2 - r_1^2}{1-r_1^2} = \end{aligned}.$$

Using these estimates, we can get estimates $\hat{\theta}_0$ and $\hat{\sigma}_e^2$ of θ_0 and σ_e^2, respectively:

$$\begin{aligned} \hat{\theta}_0 &= \bar{Y}(1 - \hat{\phi}_1 - \hat{\phi}_2) = \\ \hat{\sigma}_e^2 &= (1 - \hat{\phi}_1 r_1 - \hat{\phi}_2 r_2)s^2 = \end{aligned}.$$

3. (Cryer & Chan, Exercise 7.11) The following code simulates the MA(1) process in question:

```
set.seed(1000)
n <- 48
theta <- -0.6
y <- arima.sim(model = list(ma = c(-theta)), n = n)
```

a. The following code fits an MA(1) model to the data simulated above using maximum likelihood:

```
arima(y, order = c(0, 0, 1), method = "ML")
```

```
##
## Call:
## arima(x = y, order = c(0, 0, 1), method = "ML")
##
## Coefficients:
##          ma1  intercept
##        0.542     -0.228
## s.e.   0.124      0.206
##
## sigma^2 estimated as 0.868:  log likelihood = -64.9,  aic = 134
```

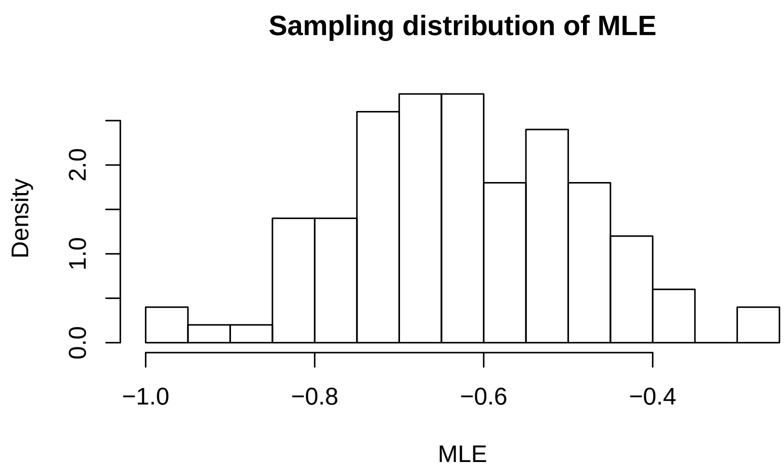
From the output we see that the maximum likelihood estimate of θ is $\hat{\theta} = -0.5422$.

b. The following code repeatedly simulates the same series and collects the maximum likelihood estimate of θ on each trial:

```
N <- 100
mle <- numeric(N)
for (i in 1:N) {
  y <- arima.sim(model = list(ma = c(-theta)), n = n)
  fit <- arima(y, order = c(0, 0, 1), method = "ML")
  mle[[i]] <- -fit$coef[["ma1"]]
}
```

c. The following displays the approximate sampling distribution of the MLE $\hat{\theta}$ based on the simulation in part (b):

```
hist(mle, freq = FALSE, breaks = 20,
     main = "Sampling distribution of MLE", xlab = "MLE")
```



d. The true parameter value is $\theta = -0.6$. The approximate mean of the MLE sampling distribution based on the simulation in part (b) is $\bar{\hat{\theta}} = -0.627$, with approximate variance $\widehat{\mathrm{Var}(\hat{\theta})} = 0.021$. Since the mean is close to the true value, and the variance is small, the estimates appear to be approximately unbiased (i.e., approximately centered around the true value θ).

e. The approximate variance of the sampling distribution is $\widehat{\mathrm{Var}(\hat{\theta})} = 0.021$. Large sample theory predicts that for large n, $\mathrm{Var}(\hat{\theta}) = (1 - \theta^2)/n = 0.013$. These two values are relatively close.

4. (Cryer & Chan, Exercise 7.31) The following code simulates the time series in question:

```
set.seed(100)
n <- 48
phi <- 0.7
y <- arima.sim(model = list(ar = c(phi)), n = n)
```

Next we fit an AR(1) model to this simulated data using maximum likelihood:

```r
fit <- arima(y, order = c(1, 0, 0), include.mean = TRUE, method = "ML")
fit
```

```
##
## Call:
## arima(x = y, order = c(1, 0, 0), include.mean = TRUE, method = "ML")
##
## Coefficients:
##          ar1  intercept
##        0.409      0.274
## s.e.   0.130      0.250
##
## sigma^2 estimated as 1.08:  log likelihood = -70,  aic = 144
```

Large sample theory predicts that for large n, the MLE $\theta\phi$ of ϕ is approximately unbiased and normally distributed with variance $\mathrm{Var}(\hat{\theta}) \approx (1-\phi^2)/n = blah$.

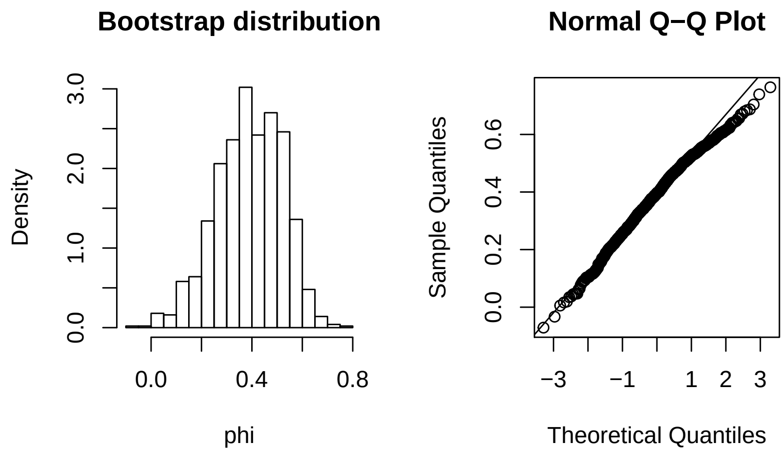
The following function produces a histogram showing the estimated distribution of ϕ based on given bootstrap estimates, as well as a normal Q-Q plot showing adherence to normality (of lack thereof):

```r
phi_dist_plots <- function(boot) {
  phi <- boot[,1]
  old_par <- par(mfrow = c(1, 2))
  hist(phi, prob = TRUE, breaks = 20,
     main = "Bootstrap distribution", xlab = "phi")
  qqnorm(phi)
  qqline(phi)
  par(old_par)
}
```

We now compute the estimated distribution of ϕ using four different bootstrapping techniques.

- **Method I:**

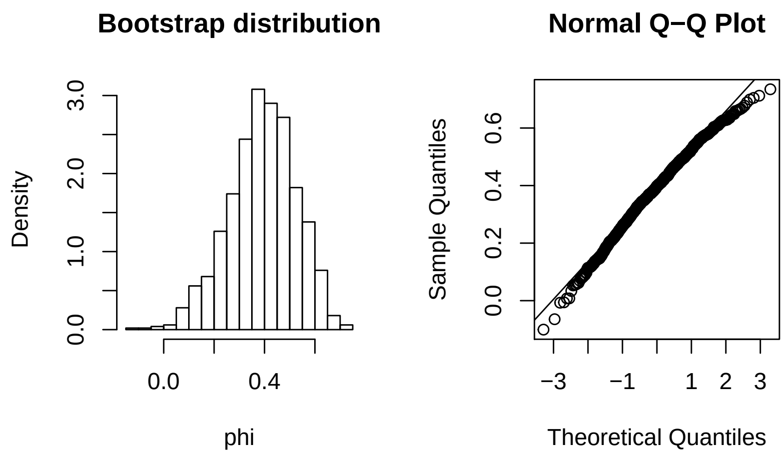
  ```r
  boot <- arima.boot(fit, cond.boot = TRUE, is.normal = TRUE,
                     B = 1000, init = y)
  phi_dist_plots(boot)
  ```



This distribution has approximate mean $\bar{\hat{\theta}} = 0.389$, and approximate variance $\widehat{\text{Var}}(\hat{\theta}) = 0.017$.

- **Method II:**

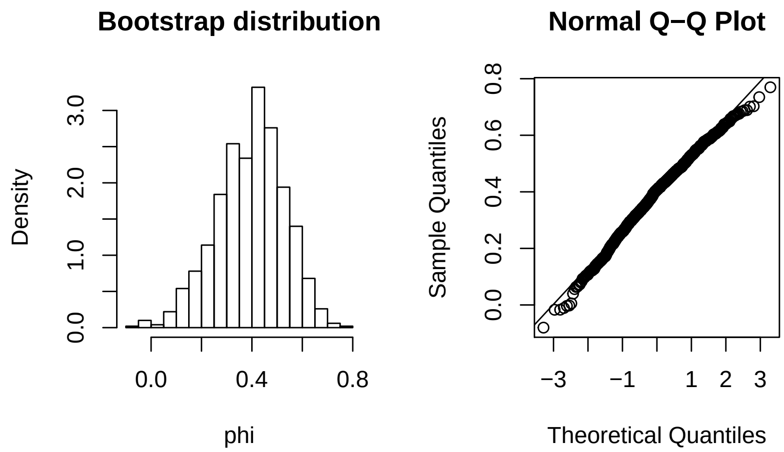
```r
boot <- arima.boot(fit, cond.boot = TRUE, is.normal = FALSE,
                   B = 1000, init = y)
phi_dist_plots(boot)
```



This distribution has approximate mean $\bar{\hat{\theta}} = 0.391$, and approximate variance $\widehat{\text{Var}}(\hat{\theta}) = 0.018$.

- **Method III:**

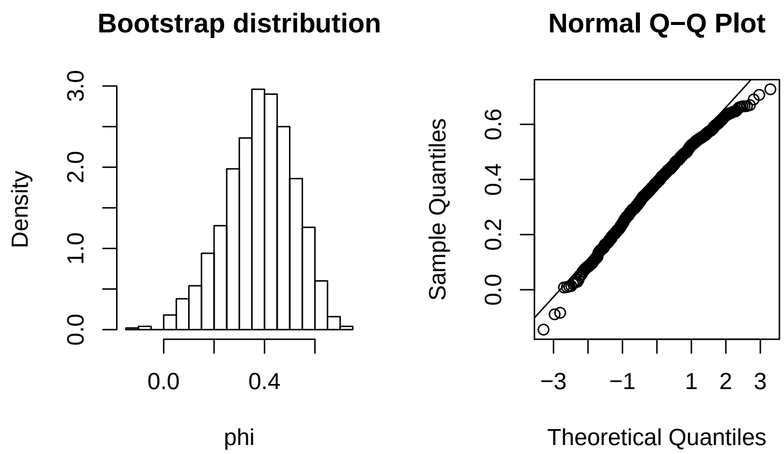
```r
boot <- arima.boot(fit, cond.boot = FALSE, is.normal = TRUE,
                   B = 1000, ntrans = 100, init = y)
phi_dist_plots(boot)
```



This distribution has approximate mean $\bar{\hat{\theta}} = 0.394$, and approximate variance $\widehat{\mathrm{Var}}(\hat{\theta}) = 0.018$.

- **Method IV:**

```
boot <- arima.boot(fit, cond.boot = FALSE, is.normal = FALSE,
                   B = 1000, ntrans = 100, init = y)
phi_dist_plots(boot)
```



This distribution has approximate mean $\bar{\hat{\theta}} = 0.381$, and approximate variance $\widehat{\mathrm{Var}}(\hat{\theta}) = 0.019$.

5. (Cryer & Chan, Exercise 8.9)