Tutorial 10 Answers

1. Simulation:

```
library(forecast)
set.seed(30004)
reps <- 1000

# Estimation sample size
n <- 200
# Forecast horizon
hmax <- 12

# AR(1) true coefficient
ar <- 0.8

# Storage for inclusion indicators
# 1 : if Y_n+h included in (L_n+h, U_n+h)
# 0 : otherwise
Included <- matrix(nrow=reps, ncol=hmax)

for (r in 1:reps){

  # Standard normal prediction errors
  U <- rnorm(n+hmax)
  # Simulating from AR(1)
  Y <- arima.sim(n=n+hmax, model=list(ar=ar), innov=U)
  # Forecast sample
  Yf <- Y[(n+1):(n+hmax)]
  # Estimation sample
  Y <- Y[1:n]

  # Estimate AR(1) model
  eq <- Arima(Y, order=c(1,0,0))
  # Forecast up to hmax steps ahead
  eqf <- forecast(eq, h=hmax)
  # 95% prediction intervals
  Lower <- eqf$lower[,"95%"]
  Upper <- eqf$upper[,"95%"]
  # Check inclusion for each h=1,...,hmax
  Included[r,] <- 1*(Lower<Yf & Upper>Yf)
}

# Coverage: proportion of inclusion at each h
Coverage <- apply(Included, 2, mean)*100
names(Coverage) <- paste0("h=",1:hmax)
print(Coverage)
```

```
 h=1  h=2  h=3  h=4  h=5  h=6  h=7  h=8  h=9 h=10 h=11 h=12
94.4 93.3 94.7 93.5 92.9 92.3 94.4 93.9 94.7 93.9 93.5 93.2
```

This illustrates the "95%" concept. The term *coverage* is used in confidence interval analysis to refer to the same idea — the probability in repeated samples that an interval contains ("covers") the true value. These simulated coverages are not exactly 95%, which arises partly because of simulation error (only 1000 replications) and partly because the arises partly because of simulation error (only 1000 replications) and partly because the forecast intervals are based on model estimates from $n = 200$ sample sizes which do not yield the exact model coefficients.

2. Including length calculations:

```
library(forecast)
set.seed(30004)
reps <- 1000

# Estimation sample size
n <- 200
# Forecast horizon
hmax <- 12

# AR(1) true coefficient
ar <- 0.8

# Storage for inclusion indicators
# 1 : if Y_n+h included in (L_n+h, U_n+h)
# 0 : otherwise
Included <- matrix(nrow=reps, ncol=hmax)

# Storage for length of each interval
Length <- matrix(nrow=reps, ncol=hmax)

for (r in 1:reps){

  # Standard normal prediction errors
  U <- rnorm(n+hmax)
  # Simulating from AR(1)
  Y <- arima.sim(n=n+hmax, model=list(ar=ar), innov=U)
  # Forecast sample
  Yf <- Y[(n+1):(n+hmax)]
  # Estimation sample
  Y <- Y[1:n]

  # Estimate AR(1) model
  eq <- Arima(Y, order=c(1,0,0))
  # Forecast up to hmax steps ahead
  eqf <- forecast(eq, h=hmax)
  # 95% prediction intervals
  Lower <- eqf$lower[,"95%"]
  Upper <- eqf$upper[,"95%"]
  # Check inclusion for each h=1,...,hmax
  Included[r,] <- 1*(Lower<Yf & Upper>Yf)
  # Length of each interval:
  Length[r,] <- Upper-Lower
}

# Coverage: proportion of inclusion at each h
Coverage <- apply(Included, 2, mean)*100
names(Coverage) <- paste0("h=",1:hmax)
print(Coverage)
```

```
 h=1  h=2  h=3  h=4  h=5  h=6  h=7  h=8  h=9 h=10 h=11 h=12
94.4 93.3 94.7 93.5 92.9 92.3 94.4 93.9 94.7 93.9 93.5 93.2
```

```
# Average interval lengths:
AvgLength <- round(apply(Length, 2, mean),2)
names(AvgLength) <- paste0("h=",1:hmax)
print(AvgLength)
```

```
 h=1  h=2  h=3  h=4  h=5  h=6  h=7  h=8  h=9 h=10 h=11 h=12
3.92 4.98 5.54 5.87 6.06 6.19 6.26 6.31 6.35 6.37 6.38 6.39
```

Notice the average length of the interval is increasing in h, which reflects the formula we have seen in lectures for the variances of the h-step-ahead forecasts, and the intuition that it is *more difficult* to forecast further into the future.

3. The only change is to the estimation command, replacing AR(1) with MA(1).

```
library(forecast)
set.seed(30004)
reps <- 1000

# Estimation sample size
n <- 200
# Forecast horizon
hmax <- 12

# AR(1) true coefficient
ar <- 0.8

# Storage for inclusion indicators
# 1 : if Y_n+h included in (L_n+h, U_n+h)
# 0 : otherwise
Included <- matrix(nrow=reps, ncol=hmax)

# Storage for length of each interval
Length <- matrix(nrow=reps, ncol=hmax)

for (r in 1:reps){

  # Standard normal prediction errors
  U <- rnorm(n+hmax)
  # Simulating from AR(1)
  Y <- arima.sim(n=n+hmax, model=list(ar=ar), innov=U)
  # Forecast sample
  Yf <- Y[(n+1):(n+hmax)]
  # Estimation sample
  Y <- Y[1:n]

  # Estimate MA(1) model - misspecified!!!
  eq <- Arima(Y, order=c(0,0,1))
  # Forecast up to hmax steps ahead
  eqf <- forecast(eq, h=hmax)
  # 95% prediction intervals
```

```
  Lower <- eqf$lower[,"95%"]
  Upper <- eqf$upper[,"95%"]
  # Check inclusion for each h=1,...,hmax
  Included[r,] <- 1*(Lower<Yf & Upper>Yf)
  # Length of each interval:
  Length[r,] <- Upper-Lower
}

# Coverage: proportion of inclusion at each h
Coverage <- apply(Included, 2, mean)*100
names(Coverage) <- paste0("h=",1:hmax)
print(Coverage)
```

```
 h=1  h=2  h=3  h=4  h=5  h=6  h=7  h=8  h=9 h=10 h=11 h=12
93.0 89.6 89.8 89.6 89.1 89.2 90.9 89.6 90.3 89.9 89.5 89.2
```

The coverage rates of the prediction intervals decrease below 90% as h increases. The true value is included within the interval less frequently than the desired 90%.

```
# Average interval lengths:
AvgLength <- round(apply(Length, 2, mean),2)
names(AvgLength) <- paste0("h=",1:hmax)
print(AvgLength)
```

```
 h=1  h=2  h=3  h=4  h=5  h=6  h=7  h=8  h=9 h=10 h=11 h=12
4.76 5.65 5.65 5.65 5.65 5.65 5.65 5.65 5.65 5.65 5.65 5.65
```

The prediction formulae for the MA(1) model can be derived from the representation

$$Y_t = U_t + \theta_1 U_{t-1},$$

where

$$E(\, U_t \,|\, \mathscr{Y}_{t-1} \,) = 0.$$

It follows that for any $h >= 2$ the h-step-ahead forecast is

$$E(\, Y_{n+h} \,|\, \mathscr{Y}_n \,) = E(\, U_{n+h} \,|\, \mathscr{Y}_n \,) + \theta_1 E(\, U_{n+h-1} \,|\, \mathscr{Y}_n \,) = 0,$$

and the h-step-ahead forecast error is

$$U_{n+h}^{(h)} = Y_{n+h} - E(\, Y_{n+h} \,|\, \mathscr{Y}_n \,) = Y_{n+h},$$

which has variance

$$\text{var}(U_{n+h}^{(h)}) = \text{var}(Y_{n+h}) = \sigma(1+\theta_1^2),$$

assuming $\text{var}(U_t) = \sigma^2$ for all t. For each $h > 1$, the prediction error is computed from the formula

$$\begin{aligned}&\widehat{E}(\, Y_{n+h} \,|\, \mathscr{Y}_n \,) \pm 1.96 \times \widehat{\text{sd}}(\, Y_{n+h} \,|\, \mathscr{Y}_n \,)\\ =&0 \pm 1.96 \times \sqrt{\hat{\sigma}^2(1+\hat{\theta}_1^2)}\end{aligned}$$

which implies the length

$$2 \times 1.96 \times \sqrt{\widehat{\sigma}^2(1 + \hat{\theta}_1^2)}.$$

Applying an MA(1) model produces prediction intervals of equal length for every $h > 1$. (The fact that the MA(1) model is misspecified in this case does not change its formula for calcualtion.)

4. **a.** Simulations are presented for three true values of the AR(1) coefficient: $\phi_1 = 0.3, 0.6, 0.9$.

Coverage: The coverage results are very similar across the board. Remember that these are simulation estimates from only 1,000 replications, i.e. don't over-interpret small differences. Nevertheless there might be a slight deficiency of about 1% in the coverage of the bootstrap intervals relative to the normal intervals when $\phi = 0.9$. Otherwise there is little to choose between the two intervals on this criterion, which is quite interesting because theoretically the bootstrap intervals are valid in this case but the normal bootstrap intervals are not. (Bear in mind the central limit theorem does *not* provide an approximation for non-normal data when computing prediction intervals.)

Length: The bootstrap intervals can be as much as 10% shorter for small h, although this advantage reduces to 5% or less as h increases.

Overall there are not major differences between the two intervals and the choice would be subjective. The advantage in length may suggest some weak preference for the bootstrap intervals in this particular model.

b. Again simulations are presented for $\phi_1 = 0.3, 0.6, 0.9$. The shape of the non-normality in this case is quite different. In part (a) we consider a strongly skewed distribution, while in this part we consider a symmetric distribution with fatter tails (more outliers) than the normal distribution.

Overall there is no advantage to be found for the bootstrap intervals in this model. If anything they are very slightly longer. Despite the non-normality, the normal prediction intervals work quite well in this case.

One finding to note is that the coverage properties deteriorate for larger h when $\phi_1 = 0.9$. The finite sample statistical properties of prediction with strong autocorrelation (i.e. large ϕ_1) can require larger samples for the asymptotic properties (i.e. 95% coverage) to apply. Along the same lines, the average lengths of the prediction intervals increases with h and ϕ_1 (the same applied in part a). Long horizon prediction is more difficult when time series are more strongly autocorrelated. In general the fact that the value of ϕ_1 is influencing the results implies that the estimation of the model is a relevant factor to the properties of the prediction intervals. (This is a topic of some research, beyond our scope!)