10 Time Series

Immanuel Klein

```r
library("tidyverse")
library("ggplot2")
```

The task is about analyzing the weekly cardiovascular mortality data from Los Angeles County (1970-1979) using time series models. First, we fit an AR(2) model using linear regression, and use the coefficients to forecast the next four weeks along with a 95% confidence interval. We then apply The Yule-Walker method to estimate the AR(2) model, and compare the results, including coefficient estimates and standard errors, to those from linear regression. We alse forecast using the Yule-Walker estimates. Additionally, we compare the standard errors from linear regression with their asymptotic approximations. Finally, we fit an ARMA(2,2) model to determine if the added complexity improves the fit compared to the simpler AR(2) model.

Exercise (a)

```r
library(astsa)

# Load the cmort dataset
data(cmort)

# Prepare data for AR(2) model
# Create a data frame with original series and its lags
n <- length(cmort)
df.cmort <- data.frame(
  y = cmort[3:n],
  y1 = cmort[2:(n-1)],
  y2 = cmort[1:(n-2)]
)

# Fit  AR(2) model with linear regression
```

```r
ar2.model <- lm(y ~ y1 + y2, data = df.cmort)
summary(ar2.model)
```

```

Call:
lm(formula = y ~ y1 + y2, data = df.cmort)

Residuals:
     Min       1Q   Median       3Q      Max 
-17.8192  -4.0339  -0.2112   3.4219  22.1840 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 11.45061    2.40080   4.769 2.42e-06 ***
y1           0.42859    0.03991  10.738  < 2e-16 ***
y2           0.44179    0.03988  11.078  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.702 on 503 degrees of freedom
Multiple R-squared:  0.6752,	Adjusted R-squared:  0.6739 
F-statistic: 522.8 on 2 and 503 DF,  p-value: < 2.2e-16
```

Exercise (b)

```r
# Extract coefficients from AR(2) model
coefficients <- coef(ar2.model)

# Initialize a vector to store forecasted values
forecast <- numeric(4)

# Use last two observed values of cmort
x1 <- cmort[length(cmort) - 1]
x2 <- cmort[length(cmort)]

# Forecast the next 4 weeks
for (i in 1:4) {
    forecast[i] <-
      coefficients[1] +
      coefficients[2] * x1 +
```

```
      coefficients[3] * x2
    x2 <- x1
    x1 <- forecast[i]
}

# Get residual standard error from the model
residual.se <- summary(ar2.model)$sigma

# Calculate 95% confidence intervals
alpha <- 0.05
z <- qnorm(1 - alpha/2)
lower.bound <- forecast - z * residual.se
upper.bound <- forecast + z * residual.se

data.frame(
    Forecast = forecast,
    Lower95.CI = lower.bound,
    Upper95.CI = upper.bound
)
```

```
  Forecast Lower95.CI Upper95.CI
1 87.54787   76.37259   98.72314
2 88.48185   77.30657   99.65712
3 88.05064   76.87537   99.22591
4 88.27845   77.10317   99.45372
```

Exercise (c)

```
# Estimate AR(2) model withbYule-Walker method
yw.model <- ar.yw(cmort, order.max = 2)

# Extract coefficients and their standard errors from YW
yw.coefficients <- yw.model$ar
yw.se <- sqrt(diag(yw.model$asy.var.coef))

# Extract coefficients and standard errors from linear regr
lr.coefficients <- coef(ar2.model)[2:3]
lr.se <- summary(ar2.model)$coefficients[2:3, 2]

# Comparison dataframe
```

```r
data.frame(
    Method = c("Linear Regression",
               "Linear Regression",
               "Yule-Walker",
               "Yule-Walker"),
    Coefficient = rep(c("x1", "x2"), 2),
    Estimate = c(lr.coefficients, yw.coefficients),
    Std.Error = c(lr.se, yw.se)
)
```

```
             Method Coefficient  Estimate  Std.Error
1 Linear Regression          x1 0.4285906 0.03991283
2 Linear Regression          x2 0.4417874 0.03988003
3       Yule-Walker          x1 0.4339481 0.04001303
4       Yule-Walker          x2 0.4375768 0.04001303
```

- Regarding the coefficients, the two methods provide only slightly different estimates for the AR coefficients. These differences could be because of the different underlying assumptions and estimation techniques used.
- The standard errors from the Yule-Walker method are marginally larger, suggesting there might be more uncertainty in the coefficient estimates compared to the linear regression.
- All in all, the difference between the two estimations is only very small.

Exercise (d)

```r
# Basically repition of (b) with some small adjustments.

# Initialize a vector to store forecasted values
forecast.yw <- numeric(4)

# Use last two observed values of cmort
x1 <- cmort[length(cmort) - 1]
x2 <- cmort[length(cmort)]

# Forecast the next 4 weeks
for (i in 1:4) {
    forecast.yw[i] <-
      yw.coefficients[1] * x1 + yw.coefficients[2] * x2
    x2 <- x1
```

```r
    x1 <- forecast.yw[i]
}

# Get residual standard error from Yule-Walker model
residual.se.yw <- sqrt(yw.model$var.pred)

# Calculate 95% confidence intervals
alpha <- 0.05
z <- qnorm(1 - alpha/2)
lower.bound.yw <- forecast.yw - z * residual.se.yw
upper.bound.yw <- forecast.yw + z * residual.se.yw

data.frame(
    Forecast = forecast.yw,
    Lower.95.CI = lower.bound.yw,
    Upper.95.CI = upper.bound.yw
)
```

```
  Forecast Lower.95.CI Upper.95.CI
1 76.21642    64.98452    87.44832
2 72.20646    60.97456    83.43837
3 64.68440    53.45249    75.91630
4 59.66554    48.43364    70.89745
```

Exercise (e)

$$SE(\hat{\phi}_1) = \sqrt{\frac{1}{n(1-\hat{\phi}_1^2)}} \qquad SE(\hat{\phi}_2) = \sqrt{\frac{1}{n(1-\hat{\phi}_2)}}$$

$$\hat{\phi}_1 = 0{,}4285906 \qquad \hat{\phi}_2 = 0{,}4417874 \qquad n = 508$$

$$SE(\hat{\phi}_1) = \sqrt{\frac{1}{508 \cdot (1 - 0{,}4285906^2)}} \qquad SE(\hat{\phi}_2) = \sqrt{\frac{1}{508 \cdot (1 - 0{,}4417874^2)}}$$

$$SE(\hat{\phi}_1) \approx 0{,}0491 \qquad SE(\hat{\phi}_2) = 0{,}0455$$

The standard errors from the linear regression are smaller compared to the asymptotic standard errors. This suggests that the linear regression estimates are more precise in this sample than what the asymptotic theory would predict.

Exercise (f)

```
# Fit ARMA(2,2) model
arma22.model <- arima(cmort, order = c(2, 0, 2))

# Extract AIC and BIC for both models
aic.ar2 <- AIC(ar2.model)
bic.ar2 <- BIC(ar2.model)

aic.arma22 <- AIC(arma22.model)
bic.arma22 <- BIC(arma22.model)

# Comparison dataframe
data.frame(
    Model = c("AR(2)", "ARMA(2,2)"),
    AIC = c(aic.ar2, aic.arma22),
    BIC = c(bic.ar2, bic.arma22)
)
```

```
      Model      AIC      BIC
1     AR(2) 3202.624 3219.530
2 ARMA(2,2) 3220.770 3246.153
```

Both the AIC and BIC values being lower for the simpler model indicates that the AR(2) model provides a better fit compared to the ARMA(2,2) model. The AR(2) model provides a better fit without the additional complexity that the ARMA(2,2) model introduces.