HOMEWORK ASSIGNMENT 7

Zachary Lazerick

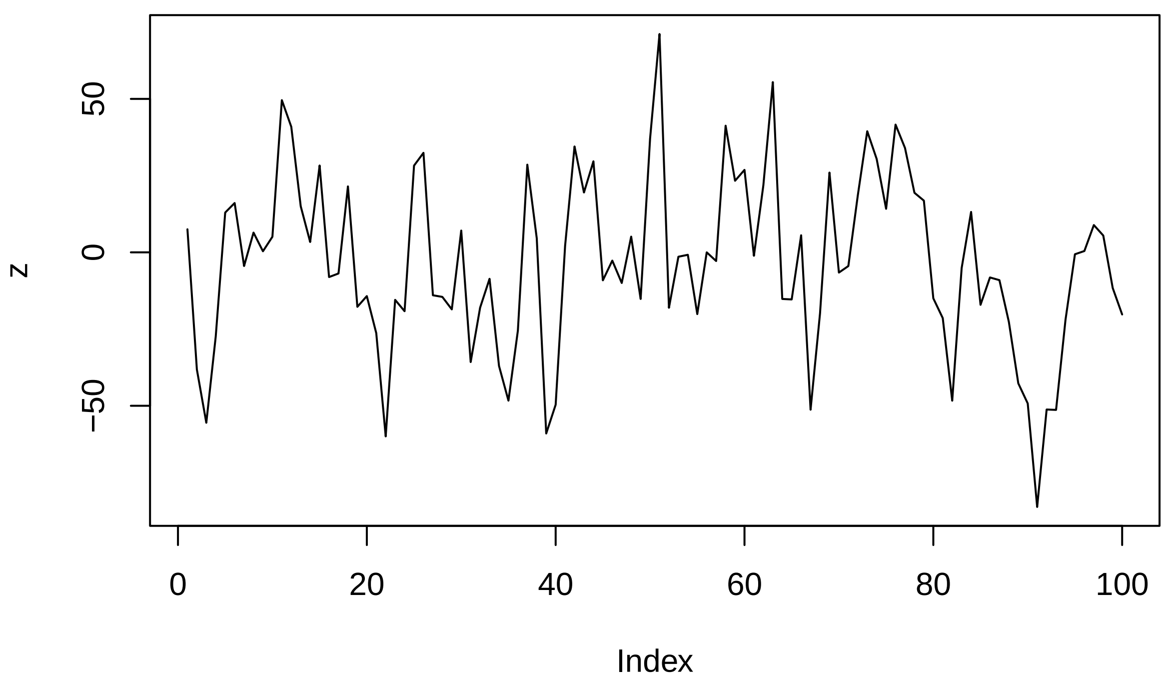
30 March 2023

PROBLEM 1

(a) Produce a time plot for $\{x_t : t = 1, \ldots, 100\}$, where $x_t = 70 + 2t - 3t^2 + z_t$, $\{z_t\}$ is the AR(1) process $z_t = 0.5z_{t-1} + wt$, and $\{w_t\}$ is white noise with standard deviation 25.

```
## Create Z
set.seed(1124)
z <- w <- rnorm(100, mean = 0, sd = 25); alpha = 0.5
for (i in 2:100) {
  z[i] <- alpha * z[i-1] + w[i]
}
```

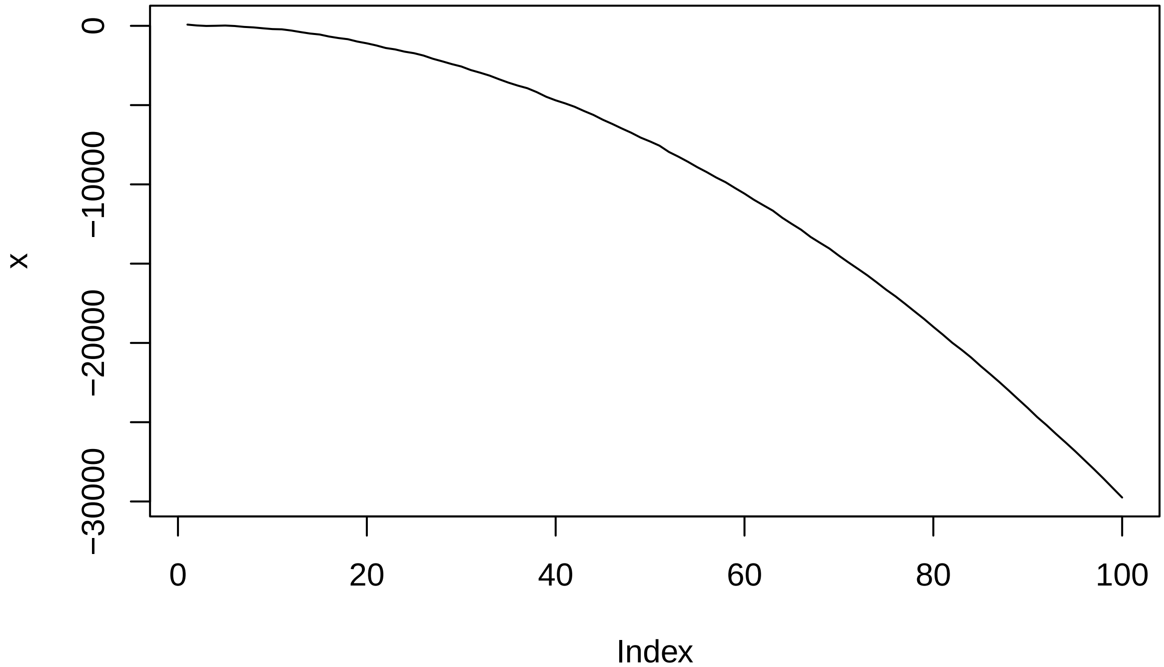
```
plot(z, type = 'l')
```




```
## Create X
x <- rep(0, 100)
for (t in 1:100) {
```

```
  x[t] <- (70 + (2 * t) - (3 * t^2) + z[t])
}
```

```
plot(x, type = 'l')
```



(b) Fit a quadratic trend to the series $\{x_t\}$. Give the coefficients of the fitted model. Higher order models can be fit using syntax such as `lm(x ~ Time + I(Time^2))`.

```
time <- 1:100
x.ols <- lm(x ~ time + I(time^2))
summary.ols <- summary(x.ols)
```

The model parameters are:

```
## Parameter Estimates
x.ols$coefficients
```

```
## (Intercept)        time   I(time^2)
##   57.686611    2.676763   -3.007484
```

```
## Parameter Estimate Std. Error
sqrt(vcov(x.ols)[1, 1]); sqrt(vcov(x.ols)[2, 2]); sqrt(vcov(x.ols)[3, 3])
```

```
## [1] 8.613023

## [1] 0.3936386

## [1] 0.003776025
```

(c) Find a 95% confidence interval for the parameters of the quadratic model, and comment.

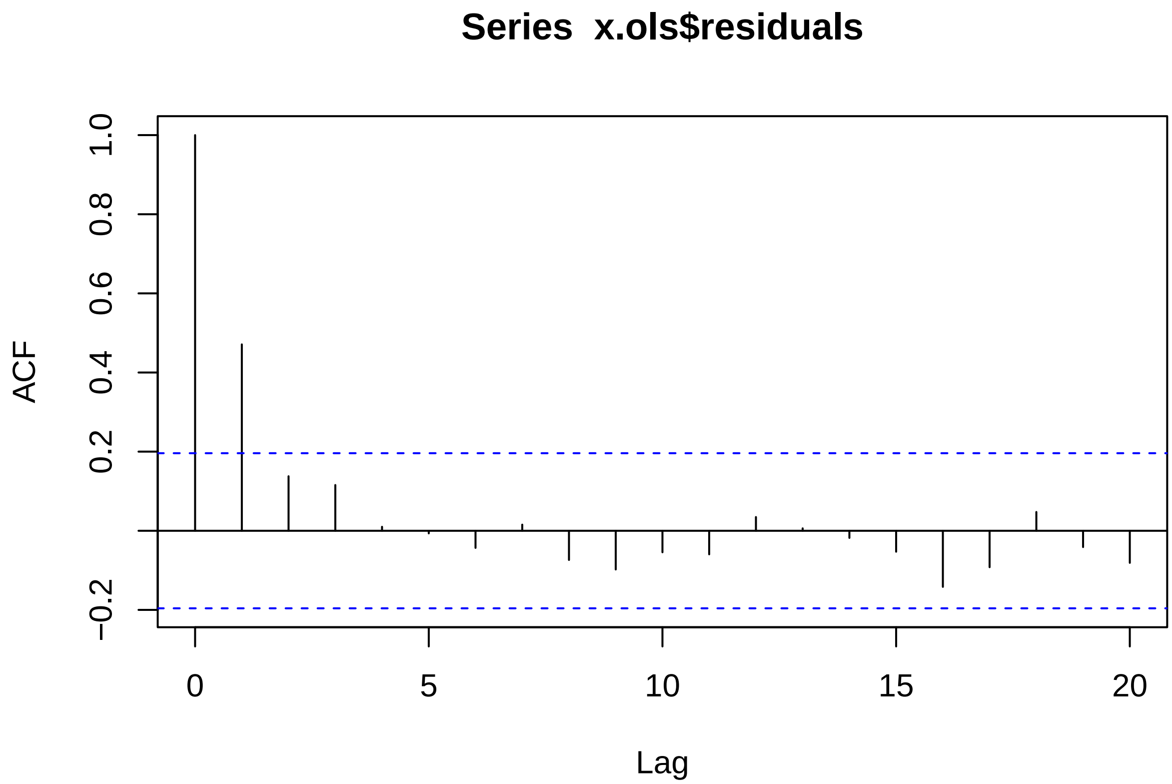
```
confint(x.ols)
```

```
##                 2.5 %    97.5 %
## (Intercept) 40.592145 74.781078
## time         1.895499  3.458027
## I(time^2)   -3.014978 -2.999990
```

Each confidence interval for the parameters of the model contain the true parameters used to generate x. Thus, the model is likely a good fit.

(d) Plot the correlogram of the residuals and comment.

```
acf(x.ols$residuals)
```



From the correlogram, we see that for lags 0 and 1, a significant autocorrelation. For everything after, the autocorrelation is within the confidence band, and thus not significant.

(e) Refit the model using GLS. Give the standard errors of the parameter estimates, and comment.

```
x.gls <- gls(x ~ time + I(time^2), cor = corAR1(0.5))

## Parameter Estimates; Parameter Estimate Std. Error (Intercept; Time; I(Time^2))
x.gls$coefficients
```

```
## (Intercept)        time   I(time^2)
##   59.450392    2.607325   -3.006918
```

```
sqrt(vcov(x.gls)[1, 1]); sqrt(vcov(x.gls)[2, 2]); sqrt(vcov(x.gls)[3, 3])
```

```
## [1] 14.87174
```

```
## [1] 0.6801079
```

```
## [1] 0.006515474
```

The parameters estimated via GLS are largely similar to those of OLS. The difference in the parameters of the model are rather negligible. Thus, OLS was not so much of an underestimate of the model parameters like we are typically worried about. The difference comes in the standard error. The GLS estimates for the standard error are nearly double there OLS counterparts.

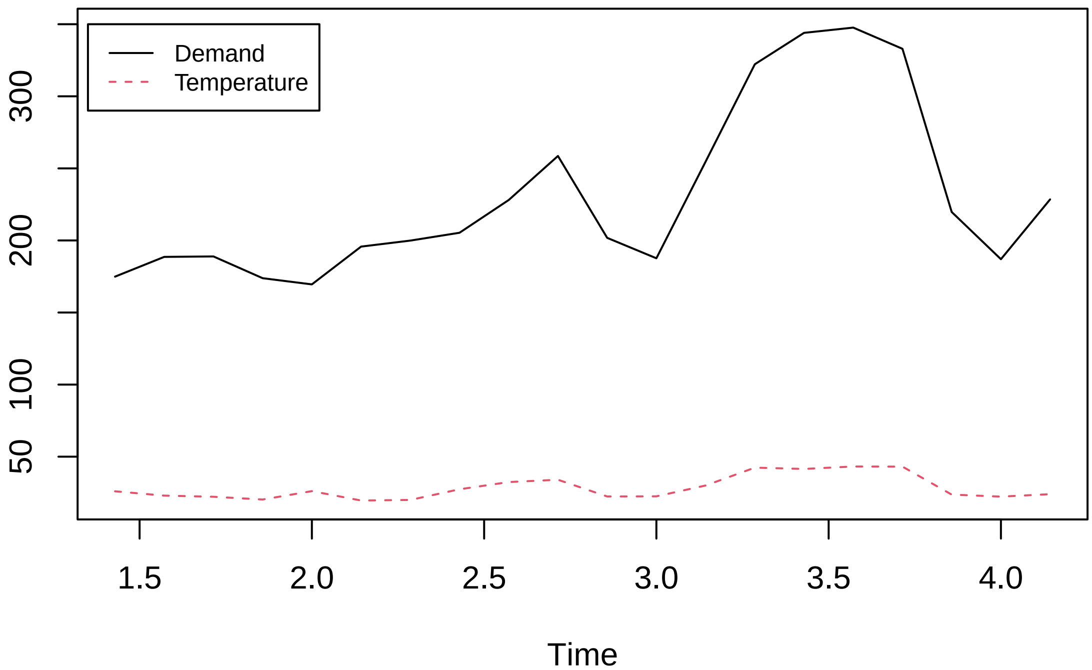
PROBLEM 2

(a) Daily electricity demand for Victoria, Australia, during 2014 is contained in `elecdaily`. The data for the first 20 days can be obtained as follows. Be sure the `fpp2` package is first installed.

```
library(fpp2)
daily20 <- head(elecdaily,20)
```

(b) The time series `daily20` contains three separate time series: `Demand`, `Workday`, and `Temperature`. Define each of these variables from `daily20`, and then use `cbind()` and `plot()` to plot the `Demand` and `Temperature` together in one plot.

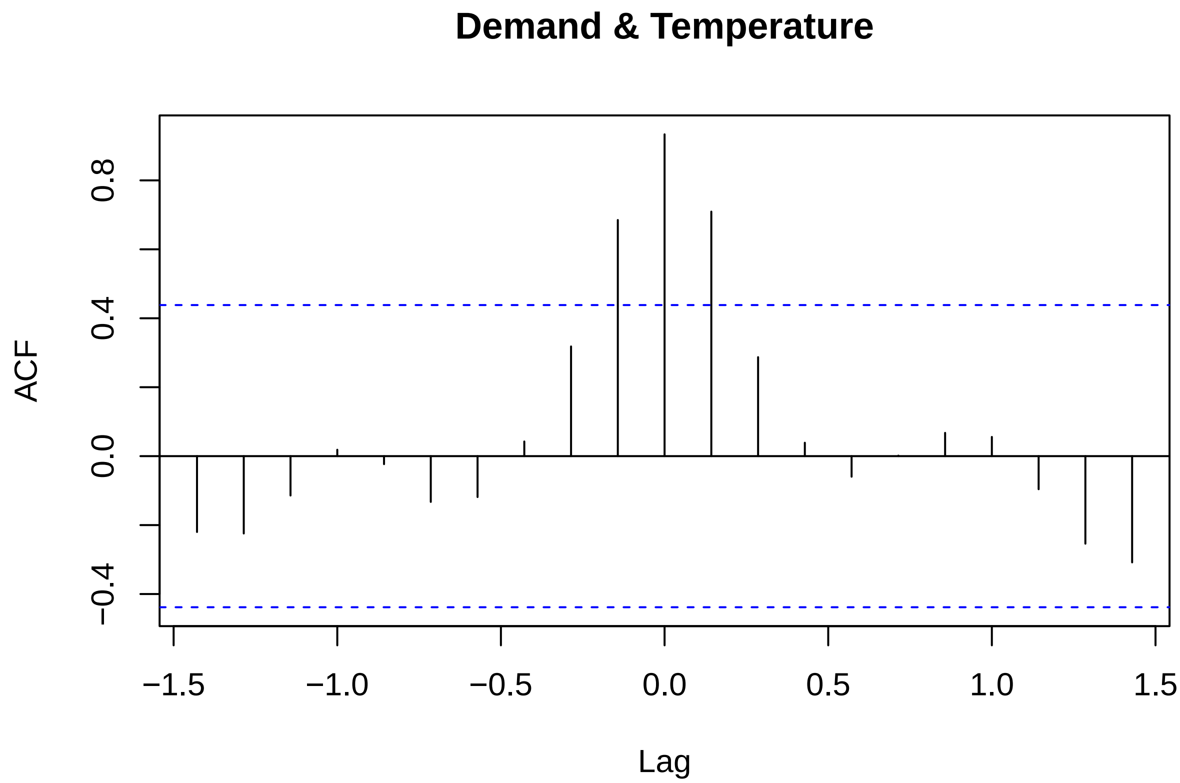
```
Demand <- daily20[, "Demand"]
Workday <- daily20[, "WorkDay"]
Temperature <- daily20[, "Temperature"]

ts.plot(cbind(Demand, Temperature), type = 'l', lty = 1:2, col = 1:2)
legend(1.35, 350, legend = c("Demand", "Temperature"), cex = .75, lty = 1:2, col = 1:2)
```



(c) Examining the plots, at what lag would you expect the cross-correlation between `Demand` and `Temperature` to be a maximum? Confirm your answer by computing the cross-correlation between `Demand` and `Temperature` using `ccf()`.

From the plots, it appears that neither variable is leading the other. Both peaks and troughs appear to occur at roughly the same time. However, these changes are more pronounced in the 'Demand' plot. Thus, cross-correlation should be a maximum at lag 0.

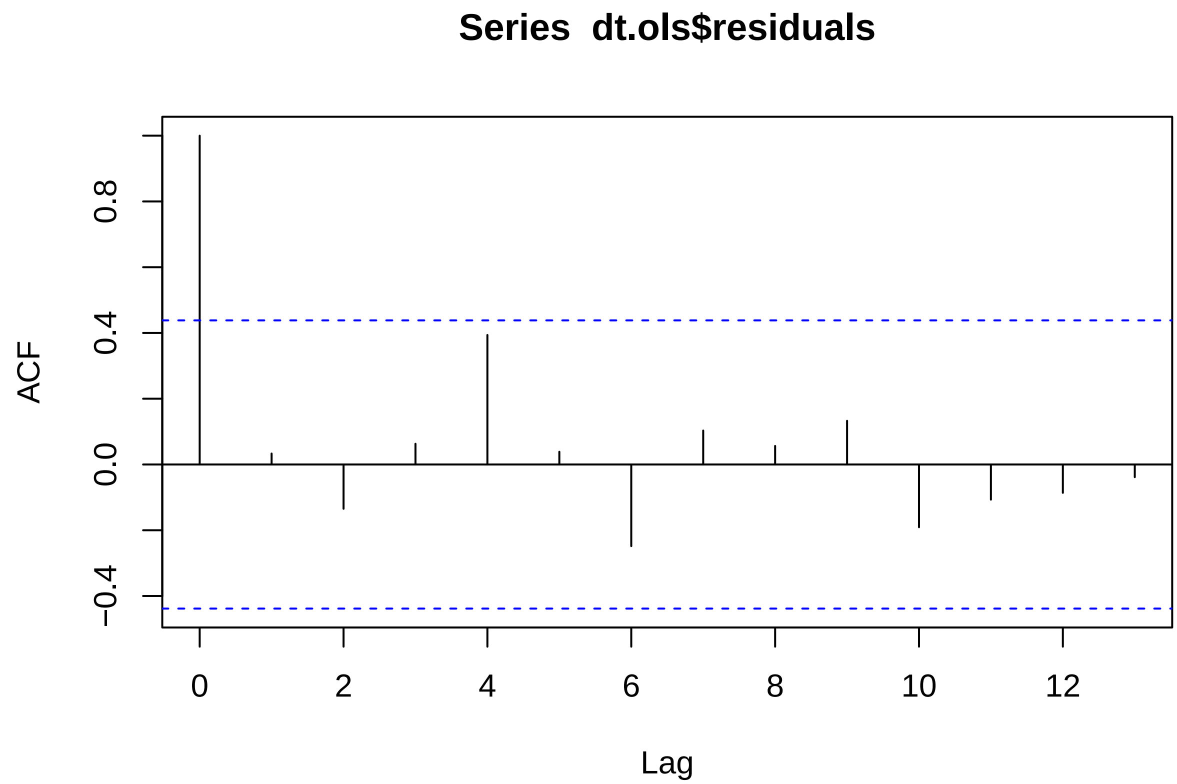
```
ccf(Demand, Temperature)
```



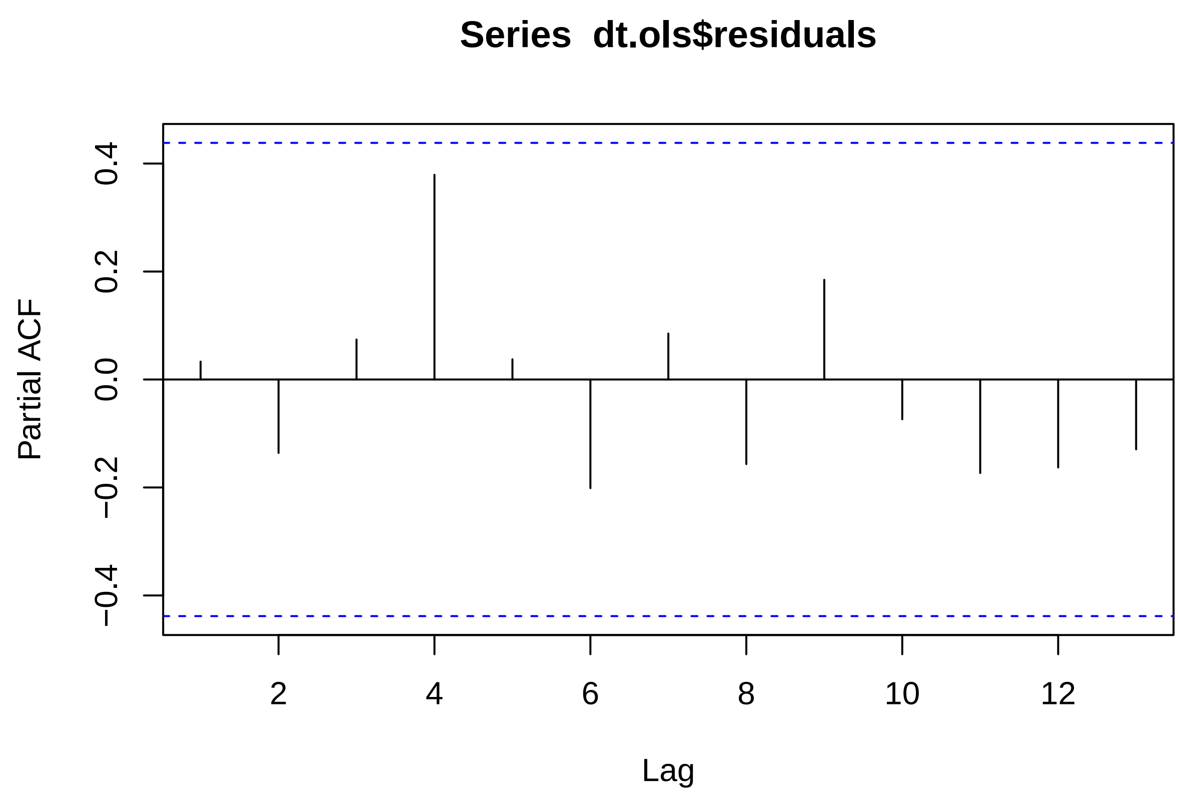
After checking the cross-correlation plot, this is confirmed. Cross-correlation does indeed peak at lag 0.

(d) Create a linear regression model using `Demand` as the response and `Temperature` as the predictor using `lm()`. Then compute the autocorrelation and partial autocorrelation of the residual series using `acf()` and the `pacf()`. Based on the results, do you think the residual series is AR(1) or white noise? Explain your reasoning.

```
dt.ols <- lm(Demand ~ Temperature)

acf(dt.ols$residuals)
```




```
pacf(dt.ols$residuals)
```



Based on the acf and pacf plots, I think that the residual series is white noise. This is because in the pacf, there are no significant correlations, thus at each lag, the autocorrelation is essentially zero. This is confirmed from the acf plot, since the autocorrelation is significant only at lag 0.

(e) Create a generalized least squares regression model using `Demand` as the response and `Temperature` as the predictor using `gls()`. Remember you will have to estimate the `corAR1` value from the acf of the residual series generated by `lm`.

```
## From the acf plot, correlation at lag 1 is about 0.05
dt.gls <- gls(Demand ~ Temperature, cor = corAR1(0.05))
```

(f) Compare the estimates of the standard errors of the regression parameters of the `lm` and the `gls`. Confirm that the estimates obtained using `gls` are higher than those using `lm`.

```
dt.gls$coefficients; dt.ols$coefficients
```

```
## (Intercept) Temperature
##   76.169327    5.406853

## (Intercept) Temperature
##   39.211720    6.757225
```

```
sqrt(vcov(dt.gls)[1, 1]) > sqrt(vcov(dt.ols)[1, 1]) ## Intercept
```

```
## [1] TRUE
```

```
sqrt(vcov(dt.gls)[2, 2]) > sqrt(vcov(dt.ols)[2, 2]) ## Temperature
```

```
## [1] TRUE
```

(g) Compute confidence intervals for the regression parameters obtained using `gls`. Based on this information, determine if there is a significant association between `Demand` and `Temperature`.

```
confint(dt.gls)
```

```
##                 2.5 %     97.5 %
## (Intercept) 26.766060 125.572593
## Temperature  3.811352   7.002353
```

There is a significant association between 'Demand' and 'Temperature' because $0 \notin C.I.$