Statistics 435/535/711 21 February 2022
P. Shaman

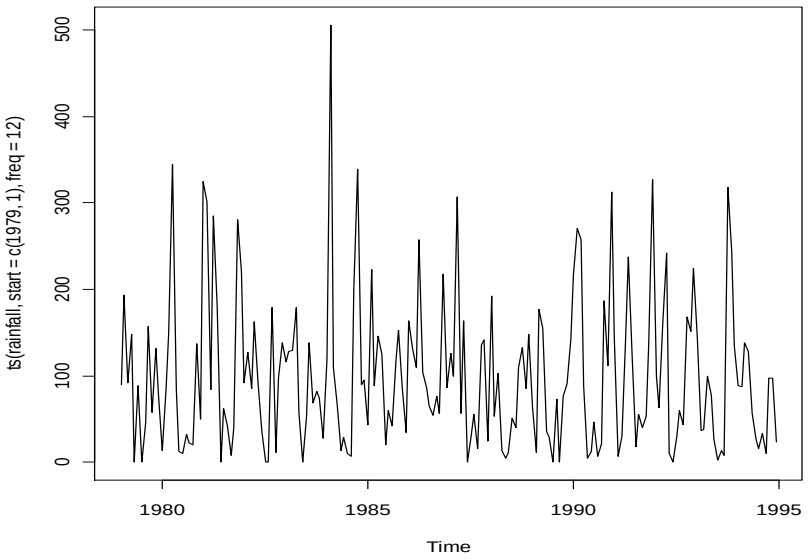
Rainfall

The file rainfall,txt gives monthly rainfall, in millimeters, on a farm in Argentina for 1979-1994.

```
> rain<-read.csv("G:/Stat71122Spring/rainfall.txt")
> attach(rain)
> head(rain)

  year month monthlength time rainfall
1 1979     1          31    1       90
2 1979     2          28    2      193
3 1979     3          31    3       92
4 1979     4          30    4      148
5 1979     5          31    5        0
6 1979     6          30    6       88
```

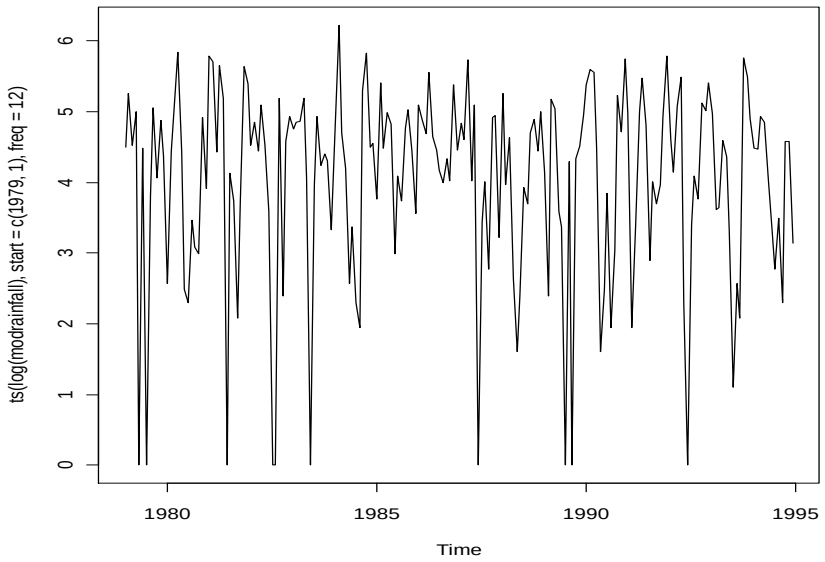
Here is a plot of the data:

```
> plot(ts(rainfall,start=c(1979,1),freq=12))
```



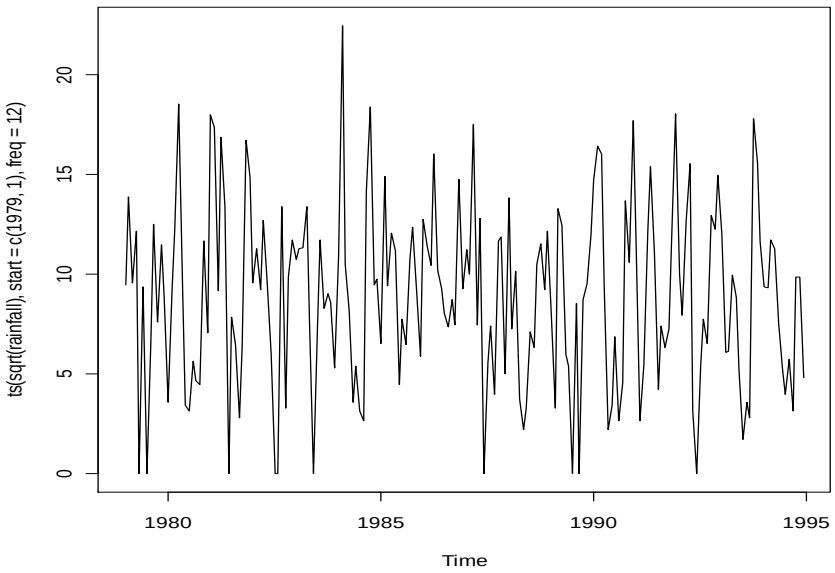
The plot displays evidence of nonconstant variance. To try to fix this, let's consider the logged data. There are, however, ten months during which rainfall is 0, and we cannot obtain the logged data for these months. To fix this, change the data values for these ten months from 0 to 1 millimeter. Here is the plot of these logged modified data (the logarithm with base e is used):

```
> modrainfall<-rainfall
> for(i in 1:length(rainfall)){
+ if(rainfall[i]==0)modrainfall[i]<-rainfall[i]+1
+ }

> plot(ts(log(modrainfall),start=c(1979,1),freq=12))
```



The goal of the log transformation is to adjust the data in order to stabilize the variance. But clearly it is an overadjustment. Let's compromise by using the square root of rainfall. This is a transformation that is less severe than is the log transformation. In fact, it is essentially intermediate between no transformation and the log transformation. (I'll discuss this in class.) Here is the plot of the square root of rainfall:

```
> plot(ts(sqrt(rainfall),start=c(1979,1),freq=12))
```



This looks quite good. So let's analyze the square root of the data.

First we fit a model including a trend component (not shown) and determined that the trend is not significant. The fit with cosines and sines to model the seasonal follows.

```
> time<-as.numeric(1:length(rainfall))
> cosm<-matrix(nrow=length(rainfall),ncol=6)
> sinm<-matrix(nrow=length(rainfall),ncol=5)
> for(i in 1:5){
+ cosm[,i]<-cos(2*pi*i*time/12)
+ sinm[,i]<-sin(2*pi*i*time/12)
+ }
> cosm[,6]<-cos(pi*time)

> model1<-
lm(sqrt(rainfall)~cosm[,1]+sinm[,1]+cosm[,2]+sinm[,2]+cosm[,3]+sinm[,3
]+cosm[,4]+sinm[,4]+cosm[,5]+sinm[,5]+cosm[,6]);summary(model1)

Call:
lm(formula = sqrt(rainfall) ~ cosm[, 1] + sinm[, 1] + cosm[,
    2] + sinm[, 2] + cosm[, 3] + sinm[, 3] + cosm[, 4] + sinm[,
    4] + cosm[, 5] + sinm[, 5] + cosm[, 6])

Residuals:
    Min      1Q  Median      3Q     Max
-8.2879 -2.4459  0.0804  2.0217 11.5607
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   8.8849     0.2732  32.518  < 2e-16 ***
cosm[, 1]     2.7150     0.3864   7.026 4.22e-11 ***
sinm[, 1]     1.8291     0.3864   4.734 4.45e-06 ***
cosm[, 2]    -0.8887     0.3864  -2.300 0.022600 *
sinm[, 2]    -1.2941     0.3864  -3.349 0.000988 ***
cosm[, 3]     0.5053     0.3864   1.308 0.192623
sinm[, 3]    -0.2137     0.3864  -0.553 0.580835
cosm[, 4]    -0.6180     0.3864  -1.599 0.111514
sinm[, 4]     0.2062     0.3864   0.534 0.594282
cosm[, 5]    -0.1749     0.3864  -0.453 0.651444
sinm[, 5]    -0.1179     0.3864  -0.305 0.760609
cosm[, 6]     0.1439     0.2732   0.526 0.599196
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.786 on 180 degrees of freedom
Multiple R-squared: 0.3424,     Adjusted R-squared: 0.3022
F-statistic:  8.52 on 11 and 180 DF,  p-value: 5.002e-12
```

There is perfect lack of collinearity (this results because the explanatory variables are cosines and sines that are harmonically related, and the sample size is an integer multiple of 12, the fundamental period).

The third through sixth harmonics are not significant. The partial *F* test that all are simultaneously insignificant follows. First we give the reduced model.

```
> model2<-
lm(sqrt(rainfall)~cosm[,1]+sinm[,1]+cosm[,2]+sinm[,2]);summary(model2)

Call:
lm(formula = sqrt(rainfall) ~ cosm[, 1] + sinm[, 1] + cosm[,
    2] + sinm[, 2])

Residuals:
   Min     1Q Median     3Q    Max
-8.504 -2.386  0.132  2.157 11.344

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   8.8849     0.2721  32.655  < 2e-16 ***
cosm[, 1]     2.7150     0.3848   7.056 3.24e-11 ***
sinm[, 1]     1.8291     0.3848   4.753 3.98e-06 ***
cosm[, 2]    -0.8887     0.3848  -2.310 0.022001 *
sinm[, 2]    -1.2941     0.3848  -3.363 0.000935 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.77 on 187 degrees of freedom
Multiple R-squared: 0.3225,     Adjusted R-squared: 0.308
F-statistic: 22.26 on 4 and 187 DF,  p-value: 4.806e-15
```

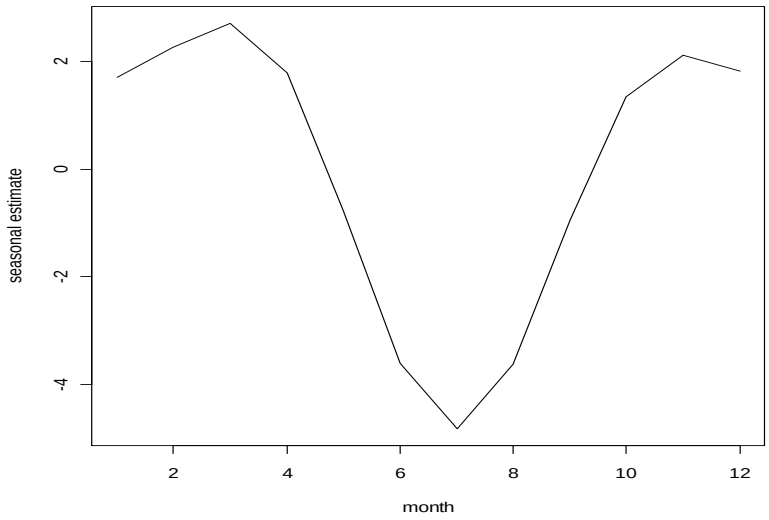
And here is the partial *F* test:

```
> anova(model2,model1)
Analysis of Variance Table

  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1    187 2658.0
2    180 2580.1  7    77.885 0.7762  0.608
```

Thus, only the first two harmonics need to be retained. Because of the lack of any collinearity, the estimates and sums of squares for the parameters in the reduced model fit are the same as in the previous analysis. The standard errors of the estimates and the *t* ratios and accompanying *p* values have changed, however, because the root mean square error has changed. The changes are small, though. Removal of the seven insignificant trigonometric variables has reduced *R* square by two per cent.

The reduced model consists only of two trigonometric pairs. Here is a plot of the seasonal pattern (on the square root scale) estimated by the model:

```
> seas2<-(predict(model2)-coef(model2)[1])[1:12]
> plot(ts(seas2),xlab="month",ylab="seasonal estimate")
```



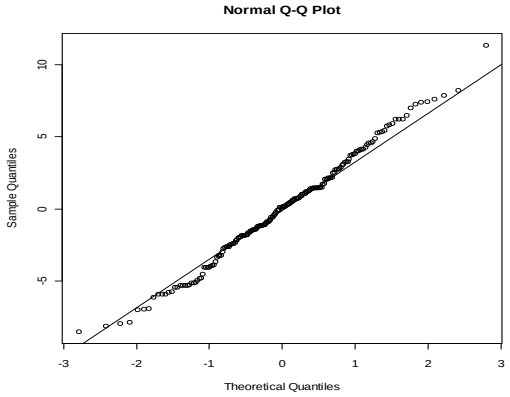
The rainy season spans October-April, and June-August are dry months.

Let's tabulate the estimated seasonal indices (for the square root of rainfall; the model is additive).

```
cbind(1:12,seas2)

   Month Seasonal
     1  1.7007794
     2  2.2651957
     3  2.7177618
     4  1.7915266
     5 -0.7604333
     6 -3.6037428
     7 -4.8308531
     8 -3.6178537
     9 -0.9403461
    10  1.3385472
    11  2.1130913
    12  1.8263272
```

The normal quantile plot of the residuals shows decent agreement with normality (there is one modest outlier and the upper tail is slightly long).

```
> qqnorm(resid(model2))
> qqline(resid(model2))
```

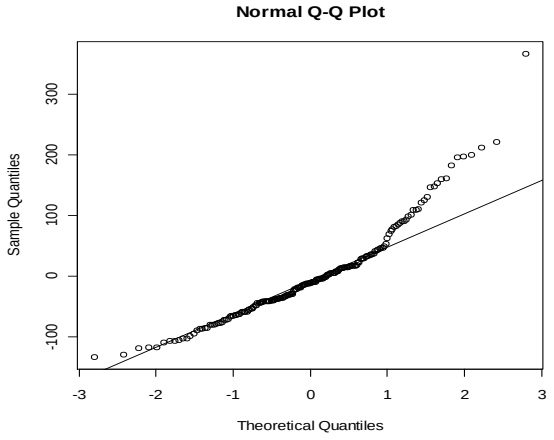



```
> shapiro.test(resid(model2))

        Shapiro-Wilk normality test

data:  resid(model2)
W = 0.99356, p-value = 0.5691
```

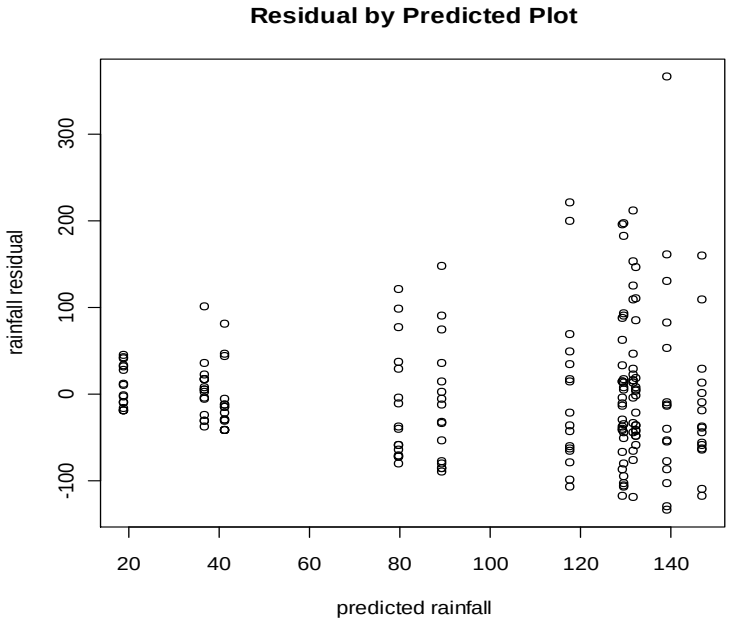
For comparison, let's look at the normal quantile plot of the residuals from a fit to the original rainfall data. This fit has no trend, and, as with the reduced model above, only the first two trigonometric pairs are retained. We show the fit first, and then the normal quantile plot of the residuals and the residual vs. predicted values plot.

```
> model3<-
lm(rainfall~cosm[,1]+sinm[,1]+cosm[,2]+sinm[,2]);summary(model3)

Call:
lm(formula = rainfall ~ cosm[, 1] + sinm[, 1] + cosm[, 2] + sinm[,
    2])
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-132.28  -43.80  -10.95   30.08  366.72

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   99.375      5.487  18.112  < 2e-16 ***
cosm[, 1]     44.334      7.759   5.714 4.30e-08 ***
sinm[, 1]     33.646      7.759   4.336 2.37e-05 ***
cosm[, 2]    -13.964      7.759  -1.800  0.07353 .
sinm[, 2]    -21.227      7.759  -2.736  0.00682 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 76.02 on 187 degrees of freedom
Multiple R-squared: 0.2495,     Adjusted R-squared: 0.2335
F-statistic: 15.54 on 4 and 187 DF,  p-value: 5.379e-11

> qqnorm(resid(model3))
> qqline(resid(model3))
```

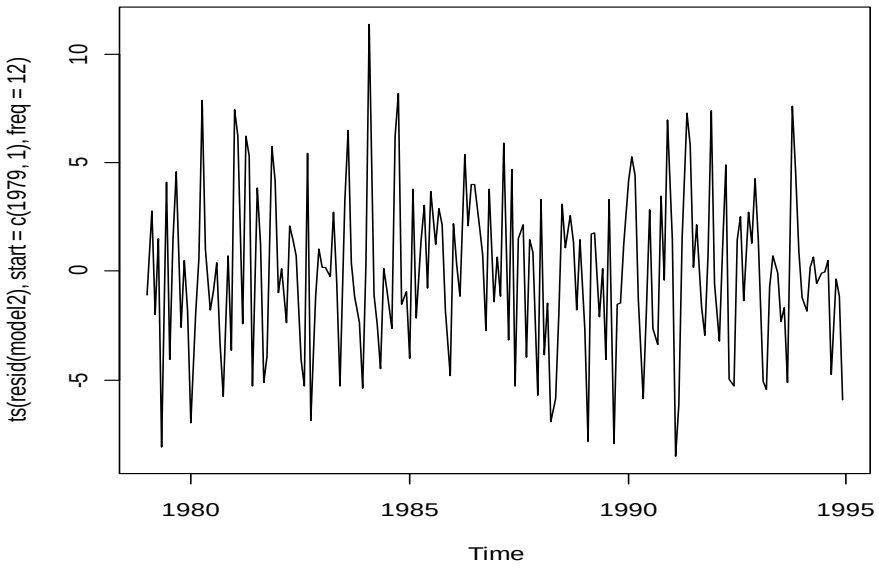



```
> plot(predict(model3),resid(model3),xlab="predicted
rainfall",ylab="rainfall residual",main="Residual by Predicted Plot")
```



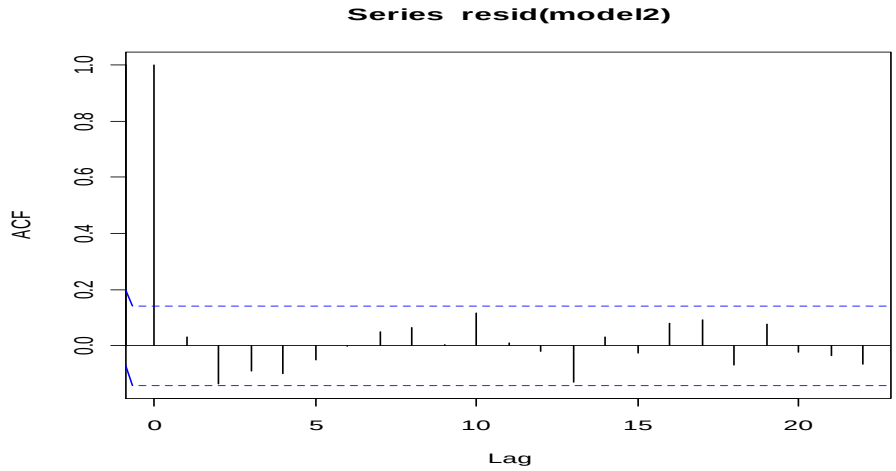
Thus, using the rainfall data without transformation does not address the heteroscedasticity. Let's return to the model fit to the square root of rainfall with two trigonometric pairs. The plot of the residual time series follows. One can see a bit of uncaptured trend structure. However, it is too weak to model with statistical significance.

```
> plot(ts(resid(model2),start=c(1979,1),freq=12))
```



The autocorrelations of the residuals suggest adequate reduction to white noise.

```
> acf(resid(model2))
```



Let's interpret the seasonal component.

```
> ampltd<-c(rep(0,times=2))
> b2<-coef(model2)[2:5]
> for(i in 1:2){
+ i1<-2*i-1
+ i2<-i1+1
+ ampltd[i]<-sqrt(b2[i1]^2+b2[i2]^2)
+ }
> ampltd
[1] 3.273661 1.569833

> phase<-c(rep(0,times=2))
> for(i in 1:2){
+ i1<-2*i-1
+ i2<-i1+1
+ phase[i]<-atan(-b2[i2]/b2[i1])
+ if(b2[i1]<0)phase[i]<-phase[i]+pi
+ if((b2[i1]>0)&(b2[i2]>0))phase[i]<-phase[i]+2*pi
+ }
> phase
[1] 5.690346 2.172583

> phase*180/pi
[1] 326.0328 124.4798

> peak<-c(rep(0,times=2))
> for(i in 1:2){
+ peak[i]<-(12/i)-6*phase[i]/(pi*i)
+ }
> peak
[1] 1.132240 3.925336
```

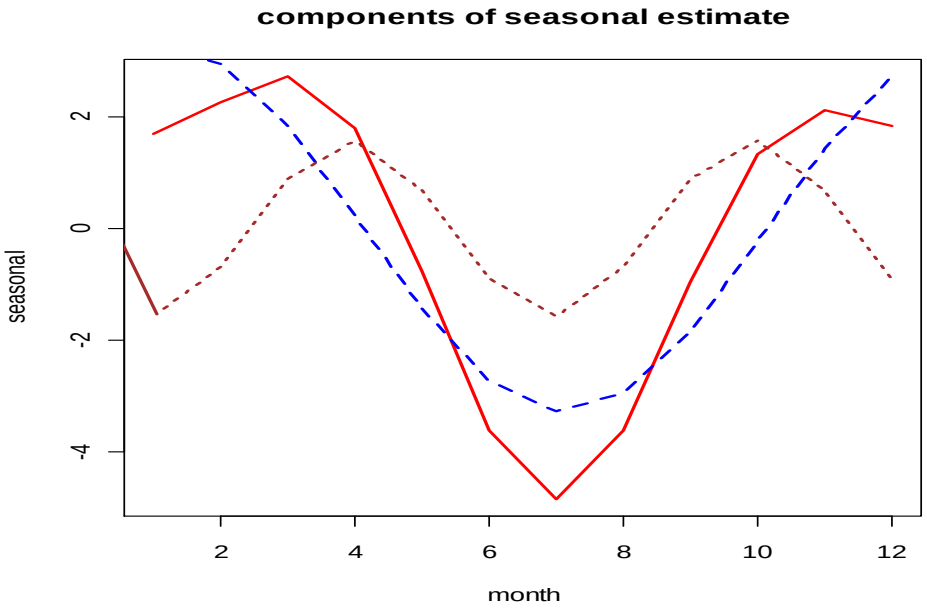
Amplitude and Phase Estimates

	Amplitude	Phase		Peak *t* (period)
		Degrees	Radians	
Fundamental	3.274	326.03	5.690	1.133 (12)
Second harmonic	1.570	124.48	2.173	3.925, 9.925 (6)

A plot of the seasonal and the two components forming it follows:

```
> b<-coef(model2)
> harm1<-(b[2]*cosm[,1]+b[3]*sinm[,1])[1:12]
> harm2<-(b[4]*cosm[,2]+b[5]*sinm[,2])[1:12]

>
plot(ts(seas2),lty=1,lwd=2,xlab="month",ylab="seasonal",main="component
s of seasonal estimate",col="red")
> lines(ts(harm1),lty=2,lwd=2,col="blue")
> lines(ts(harm2),lty=3,lwd=2,col="brown")
```

The solid red line is the plot of the seasonal index estimates, and the two dashed lines show the two estimated harmonic components combining to form the seasonal estimate given by the solid line.



At this point, we make two comments.

1. If we want to give seasonal indices for rainfall itself, the square root transformation makes interpretation awkward.

2. The signal-to-noise ratio for the rainfall data is rather low. *R* square in the reduced model is only 32 per cent. By contrast, the U.S. beer data has a very high signal-to-noise ratio, with *R* square equal to 92 per cent. For the Australian beer data, a strong part of the signal is the trend. When the trend is removed by differencing, the model fit with the seasonal component has *R* square equal to 53 per cent, still a much higher signal-to-noise ratio than for the rainfall data.

Let's adjust the rainfall data to equalize the lengths of the months, and then repeat some of the above analysis. Note that monthly rainfall is a flow variable. Clearly it is affected by the lengths of the months, but not by any of the other calendar factors. To make the adjustment, we multiply each monthly reading by 365.25/(12 times the month length in days). Note that the calendar trigonometric pairs with frequencies 0.220, 0.348, and 0.432 should not be used for the rainfall data.

```
> adjrainfall<-rainfall*365.25/(12*monthlength)
```

We continue to use the square root transformation. Only the first two seasonal trigonometric pairs are significant.

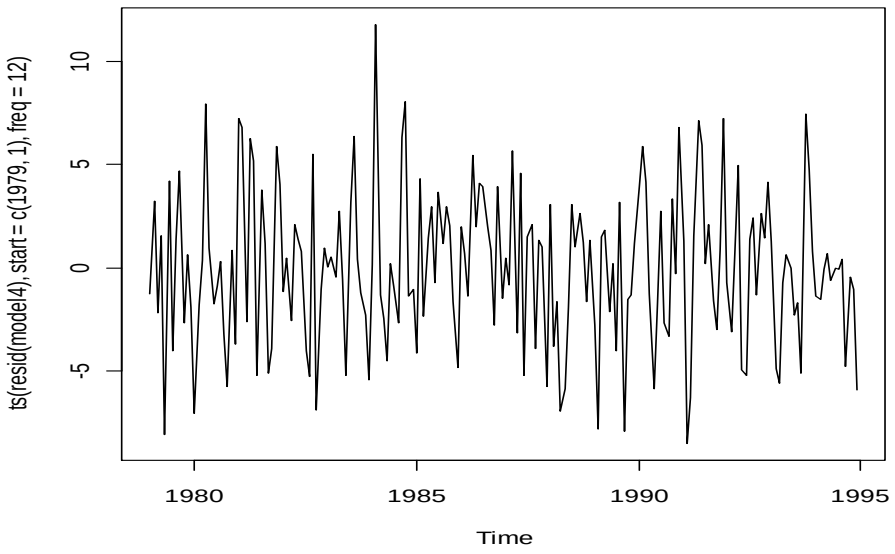
```
> model4<-
lm(sqrt(adjrainfall)~cosm[,1]+sinm[,1]+cosm[,2]+sinm[,2]);summary(model
4)

Call:
lm(formula = sqrt(adjrainfall) ~ cosm[, 1] + sinm[, 1] + cosm[,
    2] + sinm[, 2])

Residuals:
    Min      1Q  Median      3Q     Max
-8.5007 -2.3860  0.1723  2.0834 11.7860

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   8.8943     0.2730  32.581  < 2e-16 ***
cosm[, 1]     2.7314     0.3861   7.075 2.91e-11 ***
sinm[, 1]     1.8798     0.3861   4.869 2.38e-06 ***
cosm[, 2]    -0.9312     0.3861  -2.412  0.01683 *
sinm[, 2]    -1.2636     0.3861  -3.273  0.00127 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.783 on 187 degrees of freedom
Multiple R-squared: 0.3256,     Adjusted R-squared: 0.3112
F-statistic: 22.57 on 4 and 187 DF,  p-value: 3.162e-15
```

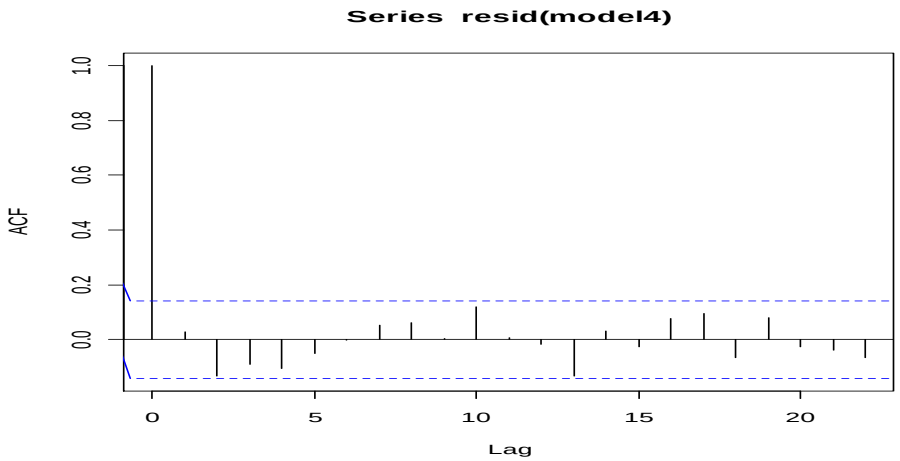
The following table shows the seasonal index estimates for both the square root of the unadjusted data (model 2), calculated previously, and the square root of the adjusted data (model 4), calculated directly above.

Month	Seasonal index est unadjusted data	Seasonal index est adjusted data
J	1.701	1.745
F	2.265	2.365
M	2.718	2.811
A	1.792	1.822
M	-0.760	-0.797
J	-3.604	-3.663
J	-4.831	-4.865
A	-3.618	-3.622
S	-0.940	-0.949
O	1.339	1.298
N	2.113	2.054
D	1.826	1.800

We see that the differences between the two model results are minor.

Let's next examine the residuals from the model fit to the square root of the adjusted data.





The results here are very similar to those found for the square root of the unadjusted data. Finally, we give the amplitude and phase analysis for the square root of the adjusted data.

Amplitude and Phase Estimates

	Amplitude	Phase		Peak *t* (period)
		Degrees	Radians	
Fundamental	3.316	325.46	5.680	1.152 (12)
Second harmonic	1.570	126.39	2.206	3.893, 9.893 (6)

It remains to discuss whether we can infer seasonal index estimates for the data without the square root transformation from the above estimations. The models we have fit include only an intercept and a seasonal component. That is, there is no required trend estimation.

Let y_t denote the month-length adjusted data. Write the model *without the square root transformation* as

(1) $$y_t = \alpha + S_t + \varepsilon_t.$$

When we fit this model, the predicted values are

(2) $$\hat{y}_t = \hat{\alpha} + \hat{S}_t.$$

For the model *with the square root transformation*, we write

(3) $$\sqrt{y_t} = \alpha^* + S_t^* + \varepsilon_t^*.$$

When we fit this model, the predicted values are

$$\text{pred}\left(\sqrt{y_t}\right) = \hat{\alpha}^* + \hat{S}_t^*. \tag{4}$$

From (2) and (4), we write

$$\hat{\alpha} + \hat{S}_t = \left(\hat{\alpha}^* + \hat{S}_t^*\right)^2 = \hat{\alpha}^{*2} + 2\hat{\alpha}^*\hat{S}_t^* + \hat{S}_t^{*2}. \tag{5}$$

The values $\hat{S}_t^*$ are the estimated seasonal indices obtained from the model (3) fit to the square root of the month-length adjusted data, and the values $\hat{S}_t$ denote the estimated seasonal indices obtained from the model (1) fit to the month-length adjusted data. As we have noted, the fit for the model (1) is suspect, and thus we want to infer the values $\hat{S}_t$ from the fit to the model (3). From (5), it is evident that we can estimate by using

$$\hat{S}_t = 2\hat{\alpha}^*\hat{S}_t^* + \hat{S}_t^{*2}. \tag{6}$$

One further step is required, though, because seasonal indices in an additive model must add to zero.

The table below shows the result of the estimation. Both model 2 on page 4 (square root of rainfall) and model 4 on page 10 (square root of adjusted rainfall) are used.

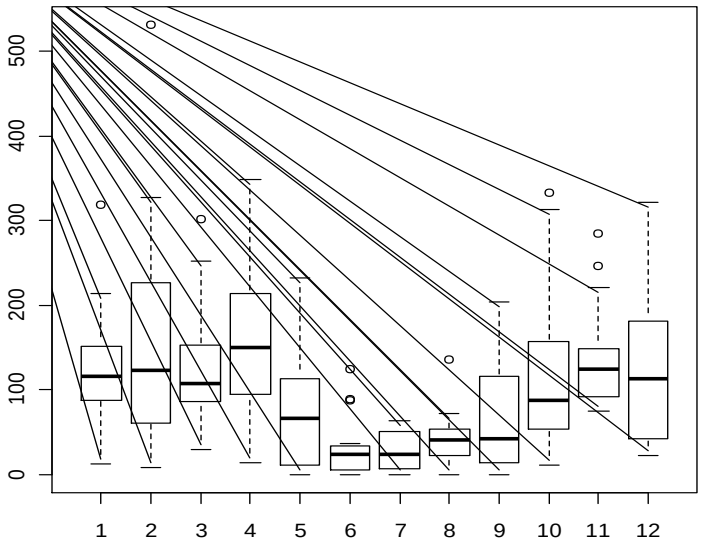
```
> seas22<-2*seas2*coef(model2)[1]+seas2*seas2
> seas22<-seas22-mean(seas22)
> seas22

 [1]  26.52443  38.79242  49.08952  28.45390 -19.52505 -57.64120 -
69.09644
 [8] -57.79004 -22.41606  18.98672  35.42361  29.19820

> seas4<-(predict(model4)-coef(model4)[1])[1:12]
> seas42<-2*seas4*coef(model4)[1]+seas4*seas4
> seas42<-seas42-mean(seas42)
> seas42

 [1]  27.36490  40.93254  51.17789  29.00696 -20.26817 -58.46723 -
69.60506
 [8] -58.04449 -22.70282  18.03920  34.03326  28.53301
```

Month	Seasonal index est from sqrt estimation	Seasonal index est from adjusted sqrt estmtn
Jan	26.52	27.36
Feb	38.79	40.93
Mar	49.09	51.18
Apr	28.45	29.01
May	-19.53	-20.27
Jun	-57.64	-58.47
Jul	-69.10	-69.61
Aug	-57.79	-58.04
Sep	-22.42	-22.70
Oct	18.99	18.04
Nov	35.42	34.03
Dec	29.20	28.53

Let's try still a different approach to estimation of seasonal structure. Instead of using the square root transformation, we apply a weighted least squares estimation. We construct weights from the data and work with the adjusted rainfall values.

The following output shows the adjusted rainfall values plotted vs. month with boxplots appended, and below the plot a table of the monthly means and standard deviations is given.

```
> boxplot(adjrainfall~month)
```



```
> means<-tapply(adjrainfall,month,mean)
> stddevs<-tapply(adjrainfall,month,sd)
> cbind(1:12,means,stddevs)

 month  means    stddevs
1    1 125.00239  72.88222
2    2 156.10230 137.80490
3    3 126.29108  70.94935
4    4 159.67005  89.63917
5    5  76.58468  69.14656
6    6  31.70573  36.94440
7    7  27.98286  23.97281
8    8  42.71069  32.26306
9    9  68.29414  67.18730
10 10 118.25214  94.88598
11 11 137.66628  61.98290
12 12 127.02747  96.22035
```

There is an outlier observation in February 1984.

Note that the dispersion of rainfall during June, July, and August, which are winter months, is very low. Of course, the February standard deviation is inflated by the outlier.

We perform a weighted least squares analysis, with the weights given by the reciprocals of the monthly standard deviations. The months for which the rainfall data are more (less) variable are given less (more) weight in the analysis. The purpose of the weighted analysis is to attempt to compensate for the heteroscedasticity.

```
> wts<-rep(1/stddevs,times=16)

> modelw1<-
lm(adjrainfall~cosm[,1]+sinm[,1]+cosm[,2]+sinm[,2]+cosm[,3]+sinm[,3]+co
sm[,4]+sinm[,4]+cosm[,5]+sinm[,5]+cosm[,6],weights=wts);summary(modelw1
)

Call:
lm(formula = adjrainfall ~ cosm[, 1] + sinm[, 1] + cosm[, 2] +
    sinm[, 2] + cosm[, 3] + sinm[, 3] + cosm[, 4] + sinm[, 4] +
    cosm[, 5] + sinm[, 5] + cosm[, 6], weights = wts)

Residuals:
    Min      1Q  Median      3Q     Max
-15.364  -5.702  -1.373   3.389  31.943
```

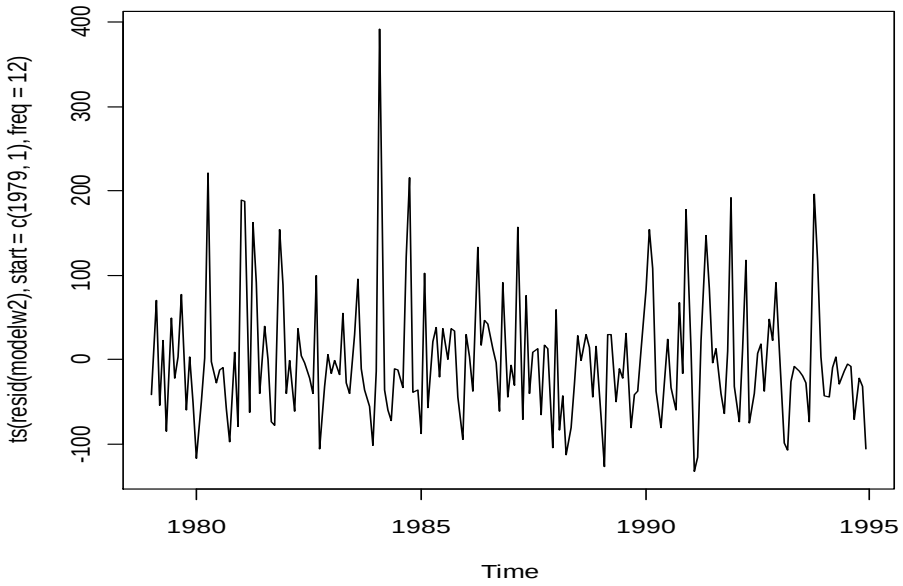
```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  99.7742     5.1353  19.429  < 2e-16 ***
cosm[, 1]    44.7047     6.9660   6.418 1.19e-09 ***
sinm[, 1]    35.0058     7.5472   4.638 6.74e-06 ***
cosm[, 2]   -15.1003     7.1769  -2.104  0.03677 *
sinm[, 2]   -20.2614     7.3469  -2.758  0.00642 **
cosm[, 3]     3.8913     7.7624   0.501  0.61677
sinm[, 3]    -3.6765     6.7253  -0.547  0.58528
cosm[, 4]   -11.4445     7.1769  -1.595  0.11255
sinm[, 4]     2.5755     7.3469   0.351  0.72633
cosm[, 5]    -0.9352     6.9660  -0.134  0.89336
sinm[, 5]    -9.6838     7.5472  -1.283  0.20110
cosm[, 6]     6.1372     5.1353   1.195  0.23362
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.435 on 180 degrees of freedom
Multiple R-squared: 0.3767,     Adjusted R-squared: 0.3386
F-statistic: 9.887 on 11 and 180 DF,  p-value: 6.135e-14
```

Only the first two trigonometric pairs are significant. Here is the revised fit with just the two pairs:

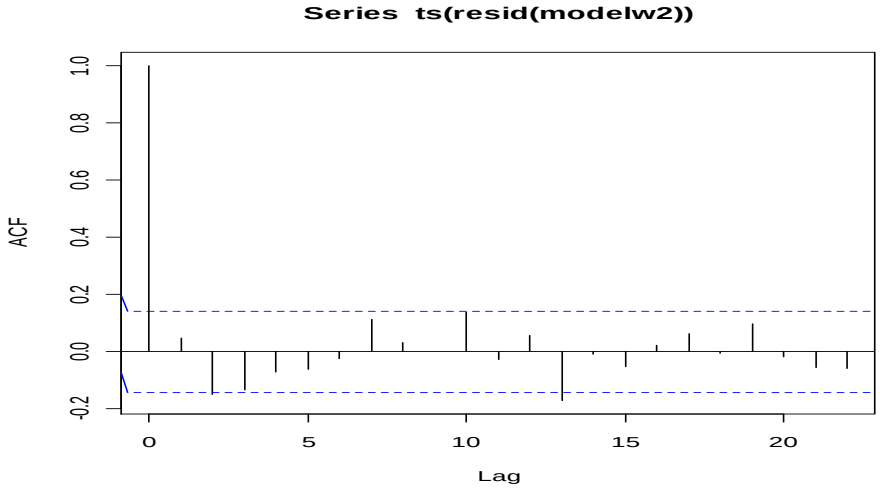
```
> modelw2<-
lm(adjrainfall~cosm[,1]+sinm[,1]+cosm[,2]+sinm[,2],weights=wts);summary
(modelw2)

Call:
lm(formula = adjrainfall ~ cosm[, 1] + sinm[, 1] + cosm[, 2] +
    sinm[, 2], weights = wts)

Residuals:
    Min      1Q  Median      3Q     Max
-13.767  -5.951  -1.603   3.447  33.334

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   99.100      4.998  19.826  < 2e-16 ***
cosm[, 1]     44.326      6.792   6.526 6.16e-10 ***
sinm[, 1]     31.625      7.290   4.338 2.35e-05 ***
cosm[, 2]    -14.129      6.748  -2.094  0.03763 *
sinm[, 2]    -18.403      6.905  -2.665  0.00837 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.416 on 187 degrees of freedom
Multiple R-squared: 0.3554,     Adjusted R-squared: 0.3416
F-statistic: 25.77 on 4 and 187 DF,  p-value: < 2.2e-16
```

Compare the result of model 3 (page 7), which is the same analysis without weighting. The estimation here is similar, with the standard errors being somewhat smaller. Some residual analysis of the present model follows.

```
> plot(ts(resid(modelw2),start=c(1979,1),freq=12))
```




```
> acf(ts(resid(modelw2)))
```



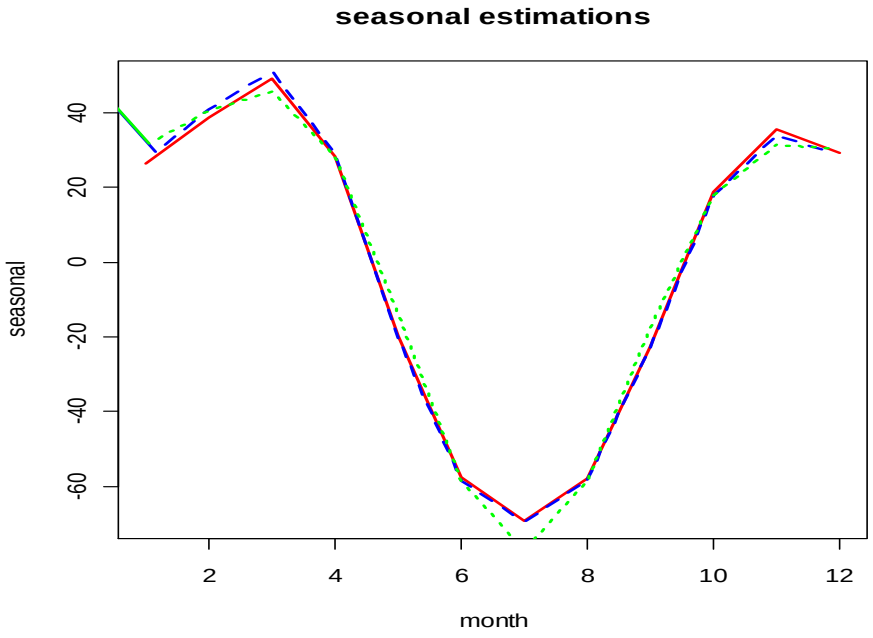
The February 1984 residual is very large. The residual autocorrelations suggest only a slight departure from white noise. These plots resemble the corresponding results for model 2 and model 4.

The seasonal index estimates from this fitted regression are shown next alongside the estimates on pages 14 and 15, and all three sets of estimates are plotted.

```
> seasw2<-(predict(modelw2)-coef(modelw2)[1])[1:12]
> seasw2
        1         2         3         4         5         6         7         8
 31.19822  40.67765  45.75326  28.22648 -13.70213 -58.45500 -77.20206 -58.42431
        9        10        11        12
-17.49609  17.77735  31.44880  30.19783
```

Month	Seasonal index estimate from sqrt	Seasonal index estimate from adj sqrt	Seasonal index estimate from wted ls
Jan	26.52	27.36	31.20
Feb	38.79	40.93	40.68
Mar	49.09	51.18	45.75
Apr	28.45	29.01	28.23
May	-19.53	-20.27	-13.70
Jun	-57.64	-58.47	-58.46
Jul	-69.10	-69.61	-77.20
Aug	-57.79	-58.04	-58.42
Sep	-22.42	-22.7	-17.50
Oct	18.99	18.04	17.78
Nov	35.42	34.03	31.45
Dec	29.20	28.53	30.20

```
>
plot(ts(seas22),lty=1,lwd=2,xlab="month",ylab="seasonal",main="seasonal
estimations",col="red")
> lines(ts(seas42),lty=2,lwd=2,col="blue")
> lines(ts(seasw2),lty=3,lwd=2,col="green")
```



The red line is the estimation from model 2, the blue line the estimation from model 4, and the green line is the estimation from the weighted least squares fit.

The differences among the three sets of estimates are not great.

Summary and additional remarks

1. The monthly rainfall data series is heteroscedastic. In particular, there is more variability in the months with heavy rainfall than in the months with light rainfall. When a log transformation is applied to the data (after adding an offset of one millimeter to the months with zero rainfall), we see that it represents too severe an adjustment. A compromise is the square root transformation, and this is seen to produce a reasonably homoscedastic time series.

2. There is no visible trend during the 16 years covered by the time series. A model is fit with only an intercept and seasonal structure, and the trigonometric basis is used to describe the seasonal. Only the fundamental and the first overtone pairs are significant. This model indicates adequate reduction to white noise. The rainy season begins in October and extends through April. The driest months are those during the winter, June, July, and August

3. The signal-to-noise ratio for the rainfall data is rather low, with *R* square in the model at only 32 per cent. A major reason for this low *R* square value is the lack of trending in the data series.

4. The monthly rainfall is a flow variable. Clearly there are no trading day effects. Variation due to the calendar is determined only by the differing lengths of the months, and thus the length of the calendar cycle is four years. One can take this calendar variation into account by adjusting the rainfall data to equalize the lengths of the months. To make the adjustment, we multiply each monthly reading by 365.25/(12 times the month length in days). This transformation is applied to the square root data. The fit from this second model is similar to that of the first model, with an intercept and the same two trigonometric pairs to describe seasonal variation. Estimates of the seasonal indices are slightly different from those in the first model, but display the same pattern.

5. The description provided by the first two model fits is awkward, because the monthly seasonal index estimates are given in terms of the square root of millimeters. Methodology is presented to allow conversion to the millimeter scale for the seasonal index estimates from the first two models.

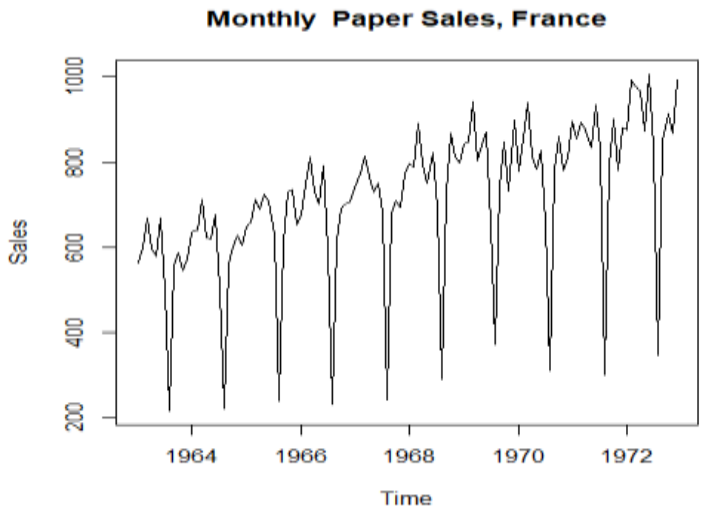
6. Weighted least squares is an alternative approach to use of the square root transformation to take into account the heteroscedasticity. Such an approach is convenient for the rainfall data, as the differing variances are essentially determined by the months, and we have multiple measurements for each month. Weights are constructed for all of the months and used to build a model via weighted least squares. The weights are given by the reciprocals of estimates of the monthly standard deviations. Thus, the weighted least squares methodology downweights the data from months with high variability. The model is fit to the calendar-adjusted data (on the millimeter scale).

Finally, the seasonal index estimates from the three models are compared using the millimeter scale.

Printing and Writing Paper, France

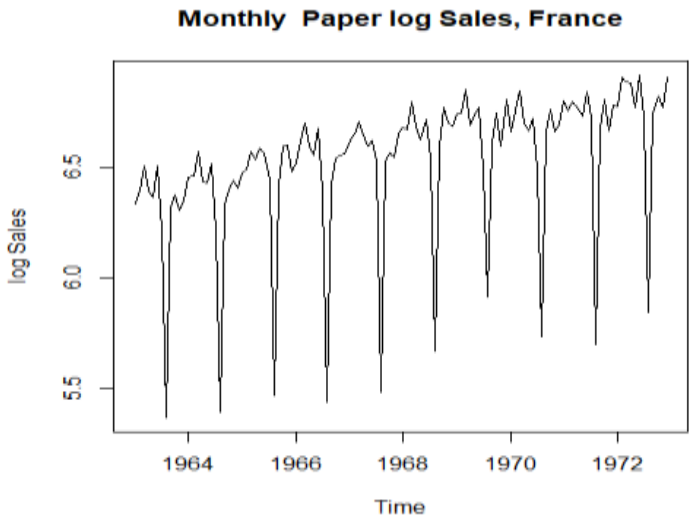
The file PaperSalesFrance.txt gives monthly industry sales for printing and writing paper, in thousands of French francs, for 1963–1972.

```
> paper<-read.csv("F:/Stat71122Spring/PaperSalesFrance.txt")
> attach(paper)
> head(paper)
  Sales logSales Month Time Obs80
1 562.7 6.332747     1    1     0
2 599.0 6.395262     2    2     0
3 668.5 6.505036     3    3     0
4 597.8 6.393256     4    4     0
5 579.9 6.362856     5    5     0
6 668.2 6.504588     6    6     0

>
plot(ts(Sales,start=c(1963,1),freq=12),xlab="Time",ylab="Sales",main="M
onthly  Paper Sales, France")
```



The plot of sales vs. time shows some increasing volatility as the level rises. Thus, let's consider the logged sales data.

```
> plot(ts(logSales,start=c(1963,1),freq=12),xlab="Time",ylab="log
Sales",main="Monthly  Paper log Sales, France")
```



The log transformation has approximately equalized the volatility.

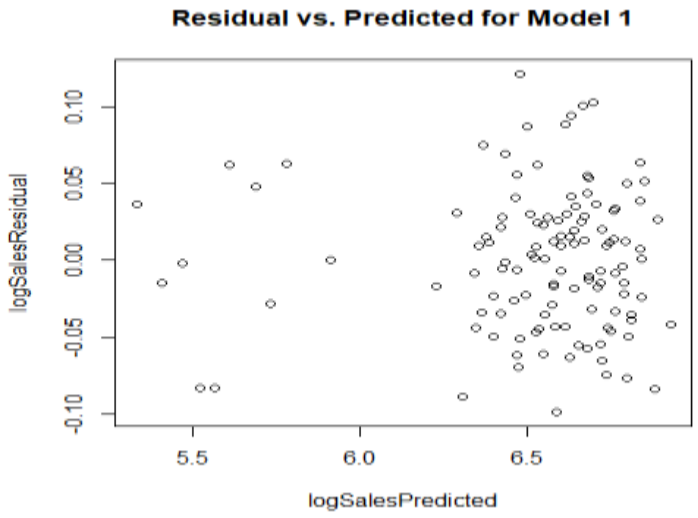
The 80th observation (August 1969) appears to be unusually high. The August readings are 215.2, 220.3, 237.5, 230.7, 241.4, 290.7, *370.5*, 310.0, 300.0 and 345.6. Let's attach a dummy variable to this observation. We fit a model to the log of sales with a third degree polynomial in time for the trend, month dummies to capture seasonal structure, and the outlier dummy.

```
> Time<-as.numeric(Time);fMonth<-as.factor(Month)
> Time2<-Time*Time;Time3<-Time*Time2
> model1<-lm(logSales~Time+Time2+Time3+fMonth+Obs80);summary(model1)

Call:
lm(formula = logSales ~ Time + Time2 + Time3 + fMonth + Obs80)

Residuals:
      Min        1Q    Median        3Q       Max
-0.098858 -0.034024  0.000393  0.029333  0.121768
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.333e+00  2.284e-02 277.290  < 2e-16 ***
Time         7.784e-03  1.325e-03   5.877 5.09e-08 ***
Time2       -6.055e-05  2.544e-05  -2.380  0.01913 *
Time3        2.680e-07  1.385e-07   1.935  0.05565 .
fMonth2      3.511e-02  2.176e-02   1.614  0.10964
fMonth3      1.081e-01  2.176e-02   4.966 2.69e-06 ***
fMonth4      1.464e-02  2.177e-02   0.673  0.50265
fMonth5     -1.733e-02  2.178e-02  -0.796  0.42802
fMonth6      5.713e-02  2.179e-02   2.622  0.01005 *
fMonth7     -1.553e-01  2.180e-02  -7.122 1.43e-10 ***
fMonth8     -1.057e+00  2.245e-02 -47.101  < 2e-16 ***
fMonth9     -1.095e-01  2.184e-02  -5.013 2.21e-06 ***
fMonth10    -7.307e-03  2.186e-02  -0.334  0.73885
fMonth11    -6.515e-02  2.188e-02  -2.977  0.00362 **
fMonth12    -2.126e-02  2.191e-02  -0.970  0.33424
Obs80        2.664e-01  5.179e-02   5.143 1.28e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04865 on 104 degrees of freedom
Multiple R-squared:  0.9802,    Adjusted R-squared:  0.9773
F-statistic: 342.8 on 15 and 104 DF,  p-value: < 2.2e-16
```

A plot of residuals vs. predicted values for this model can be used to judge whether the log transformation has controlled the heteroscedasticity.

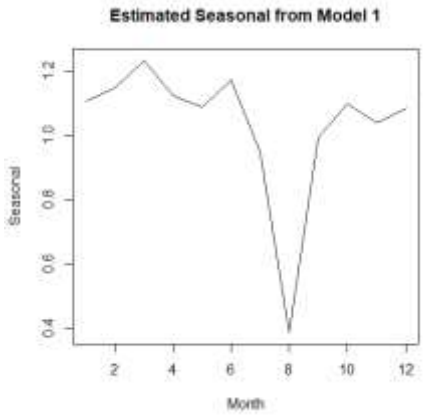
```
>
plot(predict(model1),resid(model1),xlab="logSalesPredicted",ylab="logSa
lesResidual",main="Residual vs. Predicted for Model 1")
```



The result is not perfect, but adequate for our purposes.

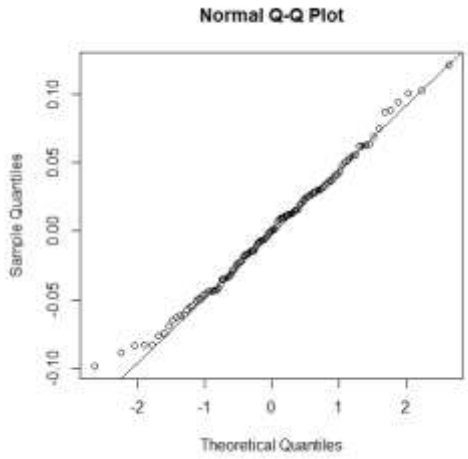
Next, let's calculate and plot the estimated seasonal indices.

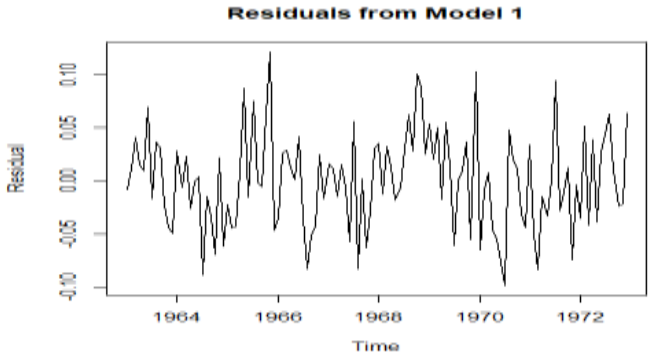
```
> b1<-coef(model1)[1]
> b2<-coef(model1)[5:15]+b1
> b3<-c(b1,b2)
> seas<-exp(b3-mean(b3))
> cbind(seas)
                  seas
(Intercept) 1.1068424
fMonth2     1.1463905
fMonth3     1.2331601
fMonth4     1.1231683
fMonth5     1.0878288
fMonth6     1.1719218
fMonth7     0.9476661
fMonth8     0.3845053
fMonth9     0.9920705
fMonth10    1.0987842
fMonth11    1.0370305
fMonth12    1.0835625
```

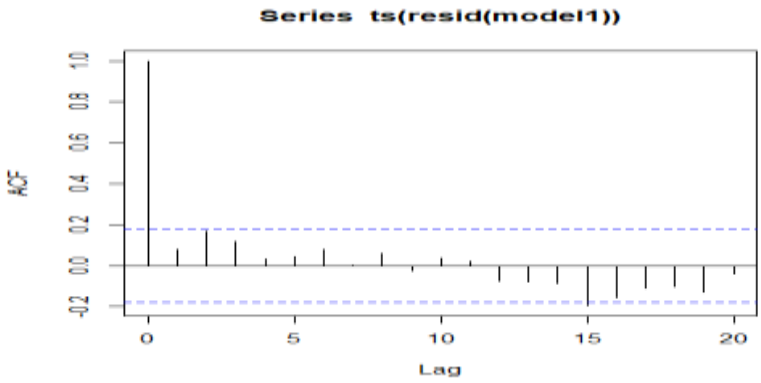


The dominant feature of the seasonal pattern, of course, is the drop in August, with some spillover to July and September. Vacation time is indeed sacrosanct. There are modest peaks in March and June.

Residual diagnostics follow.







Some small trend features remain in the residuals. There is a barely significant lag 15 residual autocorrelation, but, overall, the autocorrelation plot is not troubling. An ARIMA model, which we will consider later, permits a dynamic treatment of the seasonal component.

Let's end by refitting with the cosine and sine basis for seasonal index estimation.

```
> cosm<-matrix(nrow=length(Time),ncol=6)
> sinm<-matrix(nrow=length(Time),ncol=5)
> for(i in 1:5){
+ cosm[,i]<-cos(2*pi*i*Time/12)
+ sinm[,i]<-sin(2*pi*i*Time/12)
+ }
> cosm[,6]<-cos(pi*Time)

> model2<-
lm(logSales~Time+Time2+Time3+Obs80+cosm[,1]+sinm[,1]+cosm[,2]+sinm[,2]+
cosm[,3]+sinm[,3]+cosm[,4]+sinm[,4]+cosm[,5]+sinm[,5]+cosm[,6]);summary
(model2)

Call:
lm(formula = logSales ~ Time + Time2 + Time3 + Obs80 + cosm[,
    1] + sinm[, 1] + cosm[, 2] + sinm[, 2] + cosm[, 3] + sinm[,
    3] + cosm[, 4] + sinm[, 4] + cosm[, 5] + sinm[, 5] + cosm[,
    6])

Residuals:
      Min        1Q    Median        3Q       Max
-0.098858 -0.034024  0.000393  0.029333  0.121768

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.232e+00  1.854e-02 336.125  < 2e-16 ***
Time         7.784e-03  1.325e-03   5.877 5.09e-08 ***
Time2       -6.055e-05  2.544e-05  -2.380   0.0191 *
Time3        2.680e-07  1.385e-07   1.935   0.0556 .
Obs80        2.664e-01  5.179e-02   5.143 1.28e-06 ***
cosm[, 1]    9.165e-02  6.298e-03  14.551  < 2e-16 ***
sinm[, 1]    2.140e-01  6.392e-03  33.483  < 2e-16 ***
cosm[, 2]    7.097e-02  6.298e-03  11.269  < 2e-16 ***
sinm[, 2]   -1.591e-01  6.335e-03 -25.116  < 2e-16 ***
cosm[, 3]   -1.915e-01  6.346e-03 -30.174  < 2e-16 ***
sinm[, 3]   -2.409e-03  6.285e-03  -0.383   0.7023
cosm[, 4]    1.101e-01  6.298e-03  17.485  < 2e-16 ***
sinm[, 4]    1.381e-01  6.327e-03  21.826  < 2e-16 ***
cosm[, 5]    6.063e-02  6.298e-03   9.627 4.60e-16 ***
sinm[, 5]   -1.077e-01  6.324e-03 -17.024  < 2e-16 ***
cosm[, 6]   -6.165e-02  4.464e-03 -13.811  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04865 on 104 degrees of freedom
Multiple R-squared:  0.9802,    Adjusted R-squared:  0.9773
F-statistic: 342.8 on 15 and 104 DF,  p-value: < 2.2e-16
```

All of the trigonometric components are statistically significant. Next, let's calculate amplitudes, phases, peaks, and valleys for the seasonal structure.

```
> ampltd<-c(rep(0,times=6))
> b2<-coef(model2)[6:16]
> for(i in 1:5){
+ i1<-2*i-1
+ i2<-i1+1
+ ampltd[i]<-sqrt(b2[i1]^2+b2[i2]^2)
+ }
> ampltd[6]<-abs(b2[11])

> ampltd
[1] 0.23282258 0.17421897 0.19149143 0.17663168 0.12356312 0.06165372

> phase<-c(rep(0,times=6))
> for(i in 1:5){
+ i1<-2*i-1
+ i2<-i1+1
+ phase[i]<-atan(-b2[i2]/b2[i1])
+ if(b2[i1]<0)phase[i]<-phase[i]+pi
+ if((b2[i1]>0)&(b2[i2]>0))phase[i]<-phase[i]+2*pi
+ }
> if(b2[11]<0)phase[6]<-pi

> phase
[1] 5.116977 1.151210 3.129012 5.385598 1.057902 3.141593

> phase*180/pi
[1] 293.18120  65.95947 179.27917 308.57205  60.61333 180.00000

> peak<-c(rep(0,times=6))
> for(i in 1:5){
+ peak[i]<-(12/i)-6*phase[i]/(pi*i)
+ }
> if(phase[6]>0)peak[6]<-1

> peak
[1] 2.2272933 4.9006754 2.0080092 0.4285663 1.9959111 1.0000000
```

The following table shows peak calculations.

Amplitude and Phase Estimates

	Amplitude	Phase		Peak
		Degrees	Radians	*t* (period)
Fundamental	0.23	293.18	5.117	2.23 (12)
Second harmonic	0.17	65.96	1.151	4.90, 10.90 (6)
Third harmonic	0.19	179.28	3.129	2.01, 6.01, 10.01 (4)
Fourth harmonic	0.18	308.57	5.386	0.43, 3.43, 6.43, 9.43 (3)
Fifth harmonic	0.12	60.61	1.058	2.00, 4.40, 6.80, 9.20, 11.60 (2.4)
Sixth harmonic	0.06	180.00	3.142	1.00, 3.00, 5.00, 7.00, 9.00, 11.00 (2)

The strongest components are the fundamental harmonics two through four. The dominant feature of the seasonal pattern is the strong drop in August, and the next table gives valley calculations.

Amplitude and Phase Estimates

	Amplitude	Phase		Valley
		Degrees	Radians	*t* (period)
Fundamental	0.23	293.18	5.117	8.23 (12)
Second harmonic	0.17	65.96	1.151	1.90, 7.90 (6)
Third harmonic	0.19	179.28	3.129	0.01, 4.01, 8.01 (4)
Fourth harmonic	0.18	308.57	5.386	1.93, 4.93, 7.93, 10.93 (3)
Fifth harmonic	0.12	60.61	1.058	0.80, 3.20, 5.60, 8.00, 10.40 (2.4)
Sixth harmonic	0.06	180.00	3.142	0.00, 2.00, 4.00, 6.00, 8.00, 10.00 (2)

The table shows a valley near the end of August and the start of September.