Statistics 435/535/711 4 April 2022
P. Shaman

Seasonal ARIMA Model Fits—Examples

Seasonal model notation is introduced in part B of the 28 March notes (page 2).

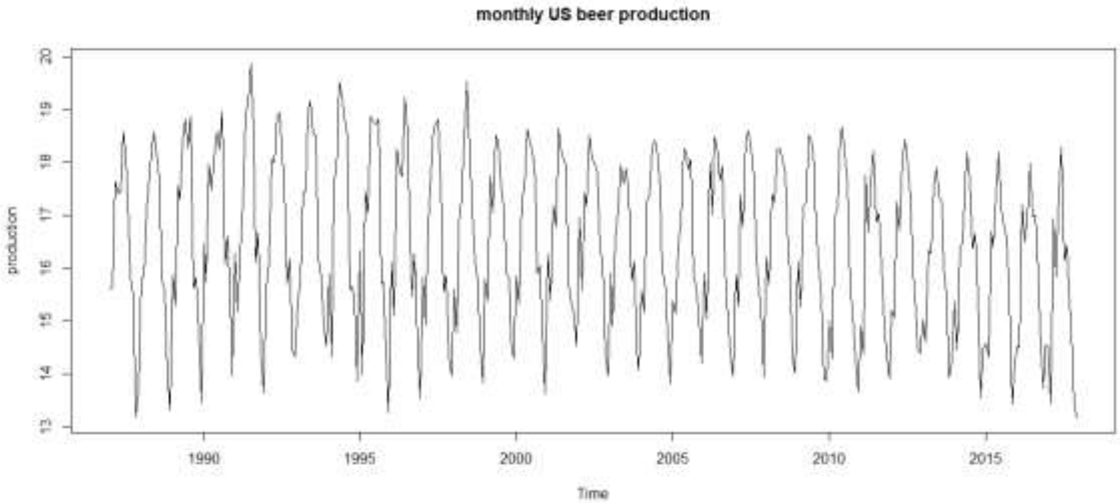
Seasonal ARIMA models can be quite difficult to fit. Here are some comments and suggestions.

- Always begin by looking at a plot of the data. Try to assess from the plot what type of differencing might be required. That is, look for a trend, which calls for ordinary differencing, and bear in mind that a clearly delineated seasonal pattern probably calls for seasonal differencing.
- In examining the plot, look for outliers. Severe outliers can distort ARIMA fitting (recall the January 1975, October 1987, and August 1998 observations in the CRSP equal-weighted monthly index log return data. It may be necessary to adjust for the outlier values in order to achieve a sensible ARIMA fit. If there are more than a few such outlier values, one needs to address volatility and outliers more directly.
- For seasonal ARIMA model fitting one needs to examine the acf and pacf at low lags and also at several multiples of the seasonal period. For example, for monthly data with an annual period, look at the behavior of the acf and the pacf in the neighborhoods of lags 12, 24, and 36. For such seasonal modelling (monthly data with an annual period) I usually plot the acf and pacf out to lag 40. This can be somewhat problematic for short time series, as the estimates at large lags can be unstable (they have high variance). For example, if the length of the time series is 80, I might tabulate the acf and pacf only up to lag 30.
- Sometimes the iterations for a seasonal ARIMA fit don't converge, or many iterations are necessary. If this is the case, you may be attempting to fit an unstable model, for example, one for which the autoregressive operator, or the difference operator, contains a factor that is very close to a factor in the moving average operator. Try to fit a simpler model, to avoid this redundancy of factors. An example: The data are subjected to ordinary differencing, so that $1 - B$ is a factor on the left, and an MA(1) fit on the right results in a factor $1 - 0.99B$.

A. Monthly U.S. beer production, January 1987—December 2017

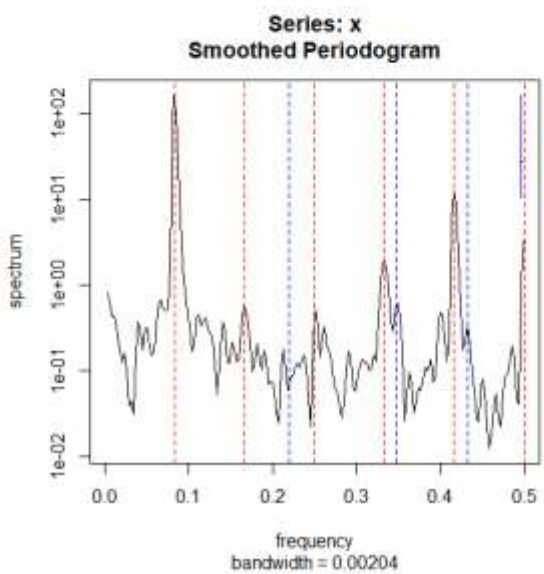
```
> usbeer<-read.csv("F:/Stat71122Spring/beernew.txt",header=T)
> attach(usbeer)
> head(usbeer)
  year month time   beer        c348       s348       c432       s432
1 1987     1    1 15.601 -0.57757270  0.8163393 -0.9101060  0.4143756
2 1987     2    2 15.633 -0.33281954 -0.9429905  0.6565858 -0.7542514
3 1987     3    3 17.656  0.96202767  0.2729519 -0.2850193  0.9585218
4 1987     4    4 17.422 -0.77846230  0.6276914 -0.1377903 -0.9904614
5 1987     5    5 17.436 -0.06279052 -0.9980267  0.5358268  0.8443279
6 1987     6    6 18.584  0.85099448  0.5251746 -0.8375280 -0.5463943
```

First, let's plot the beer production series, as in the 12 January notes. Some downward trending is present.



Next let's inspect the spectral plot of the beer series. We use a narrow bandwidth for the estimation in order to look for peaks at the calendar frequencies.

```
> spectrum(beer,3)
> abline(v=c(1/12,2/12,3/12,4/12,5,12,6,12),lty=2,col="red")
> abline(v=c(0.220,0.348,0.432),lty=2,col="blue")
```



The plot shows very strong seasonal structure, as we expect. There are peaks at frequencies 1/12, 1/6, 1/4, 1/3, 5/12, and 1/2. Calendar structure at frequencies 0.348 and 0.432 is evident. A small rise at frequency 0 indicates a mild trend.

Clearly, the beer time series is measuring a flow variable, as the data give monthly production. We will remove the calendar features first via regression. Then we will fit a seasonal ARIMA model to the regression residuals. However, let's verify that there are significant calendar effects. Note, although we will only remove calendar effects if they are significant, the regression fit we form also includes static seasonal structure. This is done in order to get reasonable hypothesis tests for the calendar variables. Then the seasonality will be addressed in the ARIMA modeling which will follow.

```
> fmonth<-as.factor(month)
> regmodel<-lm(beer~fmonth+c348+s348+c432+s432);summary(regmodel)

Call:
lm(formula = beer ~ fmonth + c348 + s348 + c432 + s432)

Residuals:
     Min       1Q   Median       3Q      Max
-1.67893 -0.32728  0.03547  0.32270  1.68029
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 15.678921   0.094631 165.686  < 2e-16 ***
fmonth2     -0.646738   0.133875  -4.831 2.02e-06 ***
fmonth3      1.564881   0.133825  11.693  < 2e-16 ***
fmonth4      1.438396   0.133796  10.751  < 2e-16 ***
fmonth5      2.588070   0.133863  19.334  < 2e-16 ***
fmonth6      2.835966   0.133818  21.193  < 2e-16 ***
fmonth7      2.354466   0.133814  17.595  < 2e-16 ***
fmonth8      1.928771   0.133842  14.411  < 2e-16 ***
fmonth9      0.252620   0.133843   1.887  0.05992 .
fmonth10    -0.004647   0.133781  -0.035  0.97231
fmonth11    -1.298985   0.133880  -9.703  < 2e-16 ***
fmonth12    -1.732434   0.133808 -12.947  < 2e-16 ***
c348         0.075061   0.038683   1.940  0.05312 .
s348         0.123722   0.038690   3.198  0.00151 **
c432         0.083528   0.038743   2.156  0.03176 *
s432        -0.020949   0.038575  -0.543  0.58742
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5266 on 356 degrees of freedom
Multiple R-squared:  0.8933,    Adjusted R-squared:  0.8889
F-statistic: 198.8 on 15 and 356 DF,  p-value: < 2.2e-16
```

The trigonometric pair with frequency 0.348 is clearly significant. The partial *F* test for the pair with frequency 0.432 follows.

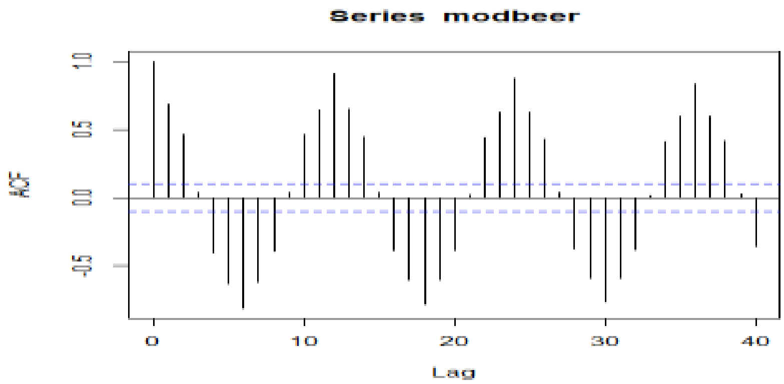
```
> regmodel2<-lm(beer~fmonth+s348+c348)

> anova(regmodel2,regmodel)
Analysis of Variance Table

Model 1: beer ~ fmonth + s348 + c348
Model 2: beer ~ fmonth + c348 + s348 + c432 + s432
  Res.Df     RSS Df Sum of Sq      F  Pr(>F)
1    358 100.103
2    356  98.729  2    1.3741 2.4774 0.08541 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The pair with frequency 0.432 is marginally significant, and we choose to retain it in the model and remove both calendar pairs via regression.

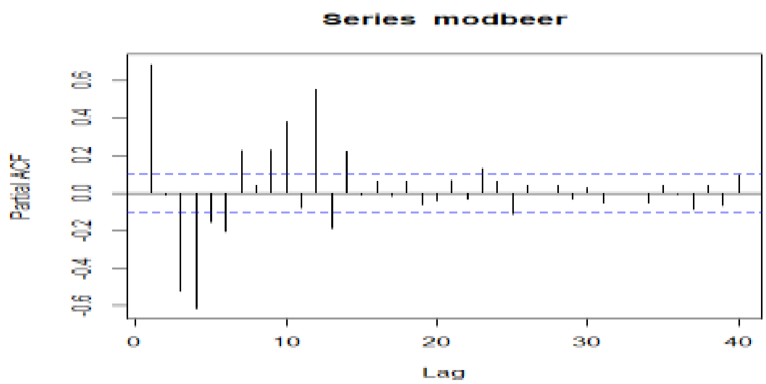
```
> modbeer<-resid(lm(beer~c348+s348+c432+s432))
```

The estimated acf and pacf plots for the modified series (which is stripped of the calendar effects) are shown next. These are given to lag 40, in order to see correlations over a span of three years plus.

```
> acf(modbeer,40)
```



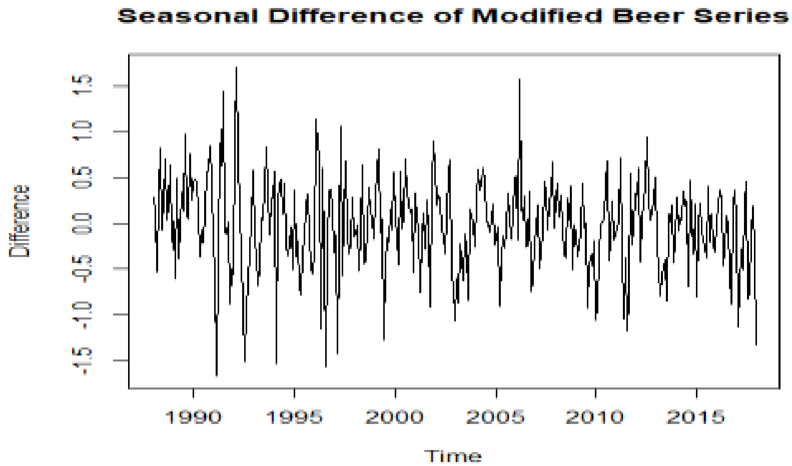

```
> pacf(modbeer,40)
```



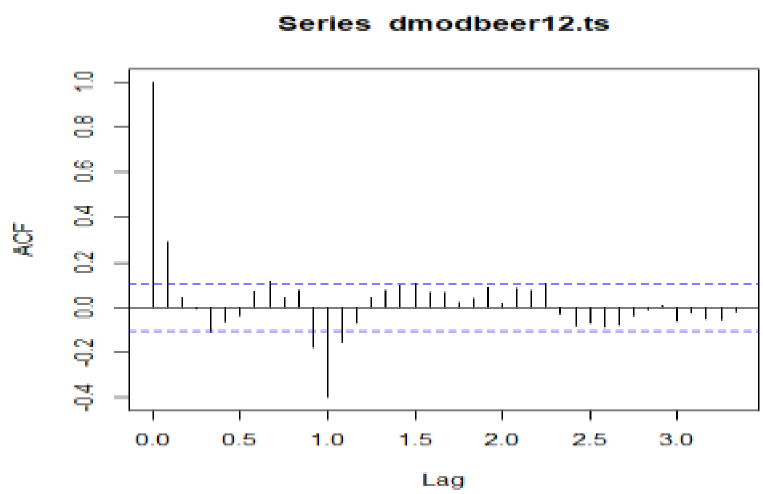
The acf plot shows there is very strong seasonal structure.

Next let's plot the first seasonal difference of the modified data. This produces a nice-looking plot with very weak trend structure. Note that differences cannot be calculated for the first year, 1987. That is, the plot starts at 1988—the length of the differenced series is 360, reduced from 372.

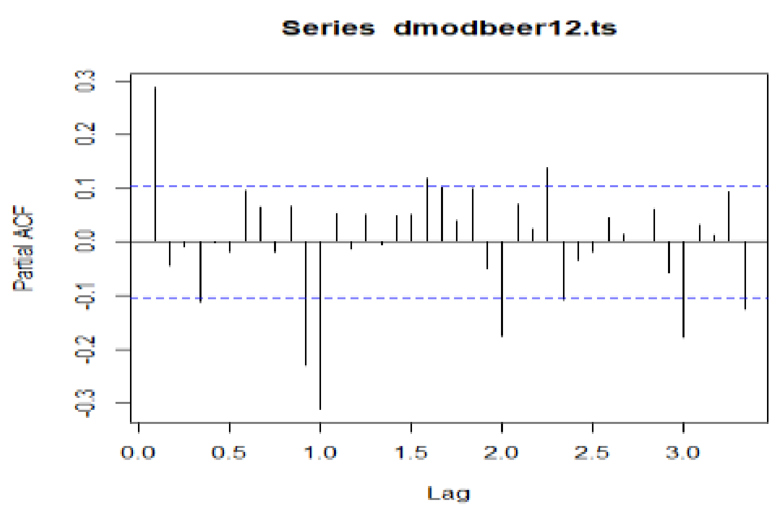
```
> dmodbeer12.ts<-ts(diff(modbeer,12),start=c(1988,1),freq=12)
> plot(dmodbeer12.ts,xlab="Time",ylab="Difference",main="Seasonal
Difference of Modified Beer Series")
```




```
> acf(dmodbeer12.ts)
```




```
> pacf(dmodbeer12.ts)
```



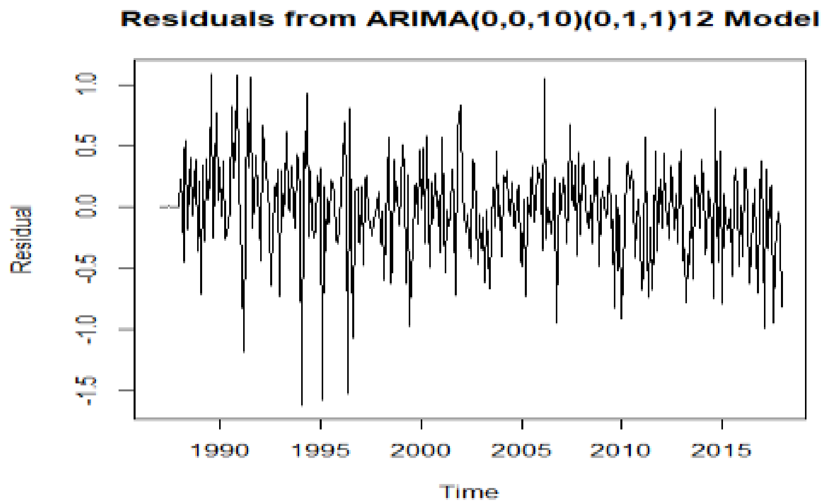
Note that use of seasonal differencing aids in the selection of a model. The seasonal differencing enables one to see structure in the acf and pacf plots that is not visible without the differencing.

Let's use MA structure. After some trial and error, we settle on an ARIMA(0,0,10)(0,1,1)$_{12}$ model.

```
> modelma<-
arima(modbeer,order=c(0,0,10),seasonal=list(order=c(0,1,1),period=12))
> modelma

Call:
arima(x = modbeer, order = c(0, 0, 10), seasonal = list(order = c(0, 1, 
1), 
    period = 12))

Coefficients:
         ma1     ma2     ma3     ma4      ma5     ma6     ma7     ma8     ma9
      0.2923  0.0906  0.1558  0.0243  -0.0113  0.0214  0.0825  0.1079  0.0601
s.e.  0.0522  0.0534  0.0555  0.0560   0.0569  0.0583  0.0609  0.0573  0.0532
        ma10     sma1
      0.1997  -0.6658
s.e.  0.0566   0.0499

sigma^2 estimated as 0.1658:  log likelihood = -191.23,  aic = 406.46

> library("lmtest")

> coeftest(modelma)

z test of coefficients:

      Estimate Std. Error  z value  Pr(>|z|)    
ma1   0.292330   0.052189   5.6014 2.126e-08 ***
ma2   0.090643   0.053423   1.6967  0.089751 .  
ma3   0.155802   0.055453   2.8096  0.004960 ** 
ma4   0.024347   0.056049   0.4344  0.664012    
ma5  -0.011295   0.056923  -0.1984  0.842713    
ma6   0.021436   0.058309   0.3676  0.713147    
ma7   0.082487   0.060908   1.3543  0.175646    
ma8   0.107939   0.057255   1.8852  0.059398 .  
ma9   0.060129   0.053155   1.1312  0.257969    
ma10  0.199723   0.056604   3.5284  0.000418 ***
sma1 -0.665808   0.049883 -13.3473 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Residual analysis follows.

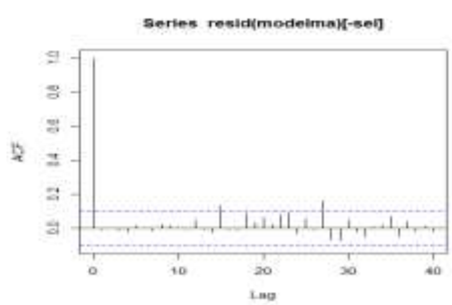
```
> modelmaresid.ts<-ts(resid(modelma),start=c(1987,1),freq=12)
> plot(modelmaresid.ts,xlab="Time",ylab="Residual",main="Residuals from 
ARIMA(0,0,10)(0,1,1)12 Model")
```



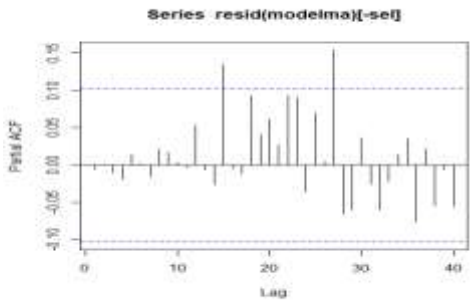

```
> resid(modelma)[1:24]

 [1] -0.0008176336 -0.0007580052  0.0011504442  0.0009323143  0.0010976651
 [6]  0.0020434250  0.0015985514  0.0005348723 -0.0008507176 -0.0008247030
[11] -0.0032400237 -0.0029033201  0.2260553699  0.0265520065 -0.4503976805
[16]  0.4309473313  0.5388605670 -0.1844166676  0.2046468838  0.4019044299
[21] -0.0729689367  0.2135720266  0.3854882756 -0.3650330199
```

The first 12 residuals calculated from the ARIMA model correspond to the year 1987. That is, the R program has done some back forecasting and has thus produced residuals for 1987, but they clearly don't have the right structure (note the shelf at the left end). Recall that the seasonal differencing employed in the estimation did not produce differenced values for 1987. We need to calculate the ARIMA model residual diagnostics without using these first 12 residuals.

```
> sel<-1:12
> acf(resid(modelma)[-sel],40)
```

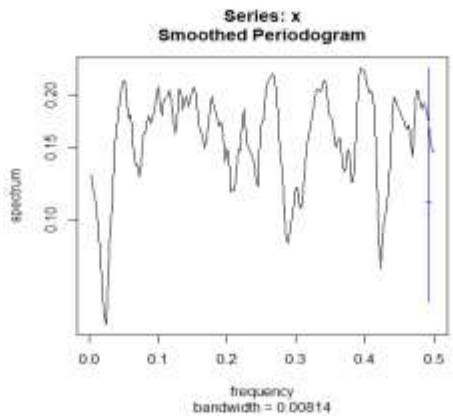


```
> pacf(resid(modelma)[-sel],40)
```



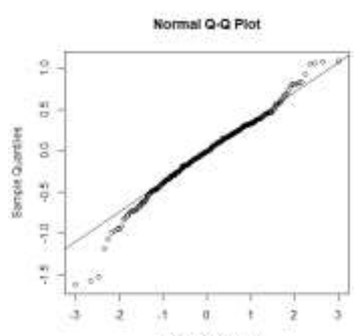
The residual acf and pacf plots both show slightly significant results at lags 15 and 27.

```
> spectrum(resid(modelma)[-sel],span=10)
```




```
> library("hwwntest")
Warning message:
package 'hwwntest' was built under R version 3.5.1

> bartlettB.test(ts(resid(modelma)[-sel]))

        Bartlett B Test for white noise

data:
= 0.32406, p-value = 0.9999
```

Let's also examine the normal quantile plot of the ARIMA residuals.



The lower tail of the residual distribution is long relative to normality. Some low monthly production figures are being overpredicted by the model.
The residual acf and pacf calculations show some modestly significant values. The residual spectral plot is very flat except for the drop near frequency 0. However, this valley is very narrow and Bartlett's version of the Kolmogorov–Smirnov test decisively does not reject the null hypothesis of reduction to white noise. We conclude that the initial adjustment for calendar effects and the subsequent ARIMA model fit have achieved reduction to white noise.

The fitted ARIMA(0,0,10)(0,1,1)$_{12}$ model can be written as

$$(1-B^{12})y_t = (1+0.2923B+0.0906B^2+0.1558B^3+0.0243B^4-0.0113^5+0.0214B^6 \\ +0.0825B^7+0.1079B^8+0.0601B^9+0.1997B^{10})\cdot(1-0.6658B^{12})\,\varepsilon_t\,.$$

In this representation the response is the residual from the initial regression used to remove calendar structure. We have fit a two-stage model.

Let's compare the goodness of fit of this model to that of the additive decomposition regression model employed in the 24 January notes. For the regression fit the residual standard deviation is 0.4455 (see page 3 of the 24 January notes). The residual standard deviation calculation for the ARIMA model fit is:

```
> sd(resid(modelma)[-sel])
[1] 0.4067654
```

Thus, the present two-stage ARIMA fit is closer to the data than is the regression model in the 24 January notes. Both use about the same number of parameters. The regression model cited does not adjust for calendar structure, of course, and it does not achieve reduction to white noise.

Next let's consider how the two competing methods forecast the last 12 observations. We withhold the last 12 observations and use the same model structures employed for the full data sets.

First, we give the regression model forecasts.

```
> usbeer<-data.frame(usbeer,fmonth)
> modela<-
lm(beer~time+I(time^2)+I(time^3)+I(time^4)+fmonth,data=usbeer[1:360,])

> regforecast<-predict(modela,newdata=usbeer[361:372,])
> regforecast

     361      362      363      364      365      366      367      368
14.93749 14.29410 16.45602 16.33971 17.45574 17.68586 17.22378 16.76982
     369      370      371      372
15.05642 14.78223 13.49289 13.03146
```

The calculations to obtain the forecasts with the two-stage ARIMA model follow next. First we calculate predictions from the first stage regression.

```
> regmodel2<-lm(beer~c348+s348+c432+s432,data=usbeer[1:360,])
> regforecast2<-predict(regmodel2,newdata=usbeer[361:372,])

> regforecast2
     361      362      363      364      365      366      367      368 
16.41618 16.48611 16.59317 16.31904 16.58583 16.52820 16.31371 16.66868 
     369      370      371      372 
16.39718 16.44450 16.57589 16.44367
```

Next, we find the forecasts from the second stage, the ARIMA model.

```
> modbeerl12<-head(modbeer,-12)
> modelma2<-
arima(modbeerl12,order=c(0,0,10),seasonal=list(order=c(0,1,1),period=12
))
> modelma2

Call:
arima(x = modbeerl12, order = c(0, 0, 10), seasonal = list(order = c(0, 
1, 1), 
    period = 12))

Coefficients:
         ma1     ma2     ma3     ma4      ma5     ma6     ma7     ma8     ma9
      0.2889  0.0885  0.1421  0.0028  -0.0426  0.0059  0.0721  0.0966  0.0367
s.e.  0.0532  0.0543  0.0562  0.0577   0.0581  0.0584  0.0622  0.0582  0.0524
        ma10     sma1
      0.1722  -0.6689
s.e.  0.0575   0.0515

sigma^2 estimated as 0.1623:  log likelihood = -181.23,  aic = 386.46

> predbeerarima<-predict(modelma2,n.ahead=12)

> predbeerarima
$`pred`
Time Series:
Start = 361 
End = 372 
Frequency = 1 
 [1] -1.61198747 -1.89578333  0.40473882 -0.03243932  0.97845130  1.57103759
 [7]  0.82617558  0.45718672 -0.29071757 -1.23771466 -2.66673489 -2.13506959

$se
Time Series:
Start = 361 
End = 372 
Frequency = 1 
 [1] 0.4029038 0.4193785 0.4208924 0.4247707 0.4247722 0.4251190 0.4251257
 [8] 0.4261164 0.4278885 0.4281442 0.4337298 0.4337298
```

Here are the predictions from the two-stage ARIMA procedure:

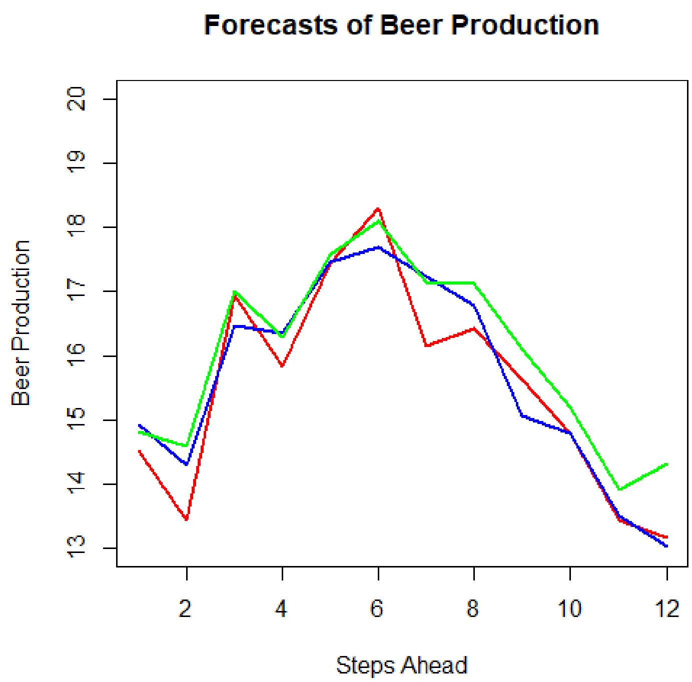
```
> arimaforecast<-regforecast2+predbeerarima$pred
> arimaforecast

     361      362      363      364      365      366      367      368 
14.80420 14.59033 16.99791 16.28660 17.56429 18.09924 17.13988 17.12587 
     369      370      371      372 
16.10646 15.20678 13.90915 14.30860
```

And here are the data values being predicted:

```
> actual<-tail(beer,12)
> actual

 [1] 14.530 13.435 16.935 15.840 17.439 18.299 16.143 16.416 15.628 14.789
[11] 13.432 13.170
```

These numbers are shown in tabular form, and then they are plotted.

```
> cbind(actual,regforecast,arimaforecast)

    actual regforecast arimaforecast
361 14.530    14.93749      14.80420
362 13.435    14.29410      14.59033
363 16.935    16.45602      16.99791
364 15.840    16.33971      16.28660
365 17.439    17.45574      17.56429
366 18.299    17.68586      18.09924
367 16.143    17.22378      17.13988
368 16.416    16.76982      17.12587
369 15.628    15.05642      16.10646
370 14.789    14.78223      15.20678
371 13.432    13.49289      13.90915
372 13.170    13.03146      14.30860

> plot(ts(actual),xlab="Steps Ahead",ylab="Beer 
Production",main="Forecasts of Beer 
Production",ylim=c(13,20),lty=1,lwd=2,col="red")
> lines(ts(regforecast),lty=1,lwd=2,col="blue")
> lines(ts(arimaforecast),lty=1,lwd=2,col="green")
```



The red curve gives the actual data, blue is for the regression forecasts, and green is for the ARIMA forecasts. The regression forecasts appear to be better than the ARIMA forecasts. One method to compare the two sets of forecasts is the root mean square forecast error, and here is this calculation:

```
> sqrt(sum(actual-regforecast)^2/12)
[1] 0.4242103
> sqrt(sum(actual-arimaforecast)^2/12)
[1] 1.756101
```

The regression forecast errors are smaller overall via this measure.

We should note that the U.S. beer data is relatively easy to forecast. It has a gentle trend and a very strong seasonal pattern.

Finally, instead of the two-step approach described above, let's fit an ARIMAX model, giving a one-step procedure. We use the same variables and structure employed with the two-step procedure.

```
> df<-data.frame(c348,s348,c432,s432)
> modelarimax<-
arima(beer,order=c(0,0,10),seasonal=list(order=c(0,1,1),period=12),xreg
=df)
> modelarimax

Call:
arima(x = beer, order = c(0, 0, 10), seasonal = list(order = c(0, 1,
1), period = 12),
    xreg = df)
```

```
Coefficients:
         ma1     ma2     ma3     ma4      ma5     ma6     ma7     ma8
ma9
      0.2942  0.0904  0.1550  0.0257  -0.0123  0.0223  0.0819  0.1089
0.0596
s.e.  0.0522  0.0535  0.0555  0.0561   0.0569  0.0584  0.0610  0.0572
0.0532
        ma10     sma1    c348    s348    c432     s432
      0.1997  -0.6659  0.0734  0.1171  0.0828  -0.0211
s.e.  0.0566   0.0498  0.0206  0.0206  0.0194   0.0194

sigma^2 estimated as 0.1654:  log likelihood = -190.75,  aic = 413.5

> library("lmtest")
> coeftest(modelarimax)

z test of coefficients:

      Estimate Std. Error  z value  Pr(>|z|)
ma1   0.294198   0.052175   5.6387 1.713e-08 ***
ma2   0.090385   0.053461   1.6907 0.0909016 .
ma3   0.155026   0.055462   2.7952 0.0051873 **
ma4   0.025653   0.056065   0.4575 0.6472769
ma5  -0.012301   0.056909  -0.2162 0.8288673
ma6   0.022281   0.058361   0.3818 0.7026287
ma7   0.081934   0.060990   1.3434 0.1791453
ma8   0.108897   0.057248   1.9022 0.0571434 .
ma9   0.059614   0.053156   1.1215 0.2620745
ma10  0.199669   0.056603   3.5275 0.0004195 ***
sma1 -0.665940   0.049779 -13.3778 < 2.2e-16 ***
c348  0.073371   0.020565   3.5678 0.0003599 ***
s348  0.117060   0.020569   5.6910 1.263e-08 ***
c432  0.082820   0.019383   4.2728 1.930e-05 ***
s432 -0.021148   0.019359  -1.0924 0.2746643
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ARIMAX parameter estimates here, and their standard errors, are essentially the same as those obtained with the two-stage procedure (see page 6).

```
> resid(modelarimax)[1:24]
 [1]  0.01563194  0.01569746  0.01759731  0.01739608  0.01753089  0.01851786
 [7]  0.01803397  0.01699126  0.01561126  0.01560949  0.01323052  0.01353543
[13]  0.23491669  0.01433434 -0.43599699  0.41251098  0.55885667 -0.19939129
[19]  0.20579587  0.41437671 -0.09781372  0.24029651  0.36325998 -0.34744408

> bartlettB.test(ts(resid(modelarimax)[-sel]))

        Bartlett B Test for white noise

data:
= 0.32291, p-value = 0.9999
```

The representation of this model is

$$
\begin{aligned}
(1-B^{12})y_t &= \beta_0 + \beta_1 c348_t + \beta_2 s348_t + \beta_3 c432_t + \beta_4 s432_t \\
&\quad + (1 + 0.2942B + 0.0904B^2 + 0.1550B^3 + 0.0257B^4 - 0.0123^5 + 0.0223B^6 \\
&\qquad + 0.0819B^7 + 0.1089B^8 + 0.0596B^9 + 0.1997B^{10}) \cdot (1 - 0.6659B^{12})\,\varepsilon_t\,,
\end{aligned}
$$

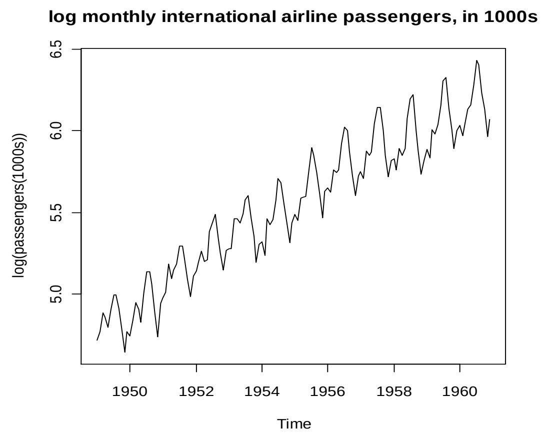
where y_t is monthly beer production.

One may use either the one-step or the two-step procedure. We see that the two have produced essentially the same result. Perhaps the two-step approach is more helpful in the model fitting decision.

B. Monthly international airline passenger data, 1949-1960

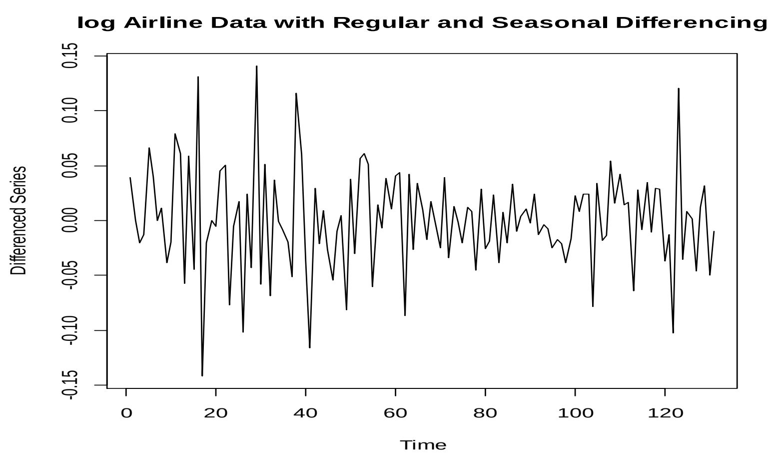
This is a classic example, given in the 1970 book by Box and Jenkins. Usually an ARIMA$(0,1,1)(0,1,1)_{12}$ model is fit to the logarithms of the passenger series. The $(0,1,1)(0,1,1)_{12}$ model is now commonly referred to as the airline model. It is used in many applications. See initial discussion of this time series in the 12 January notes.

Recall that the logarithmic transformation helps to stabilize the variance of the passenger series.

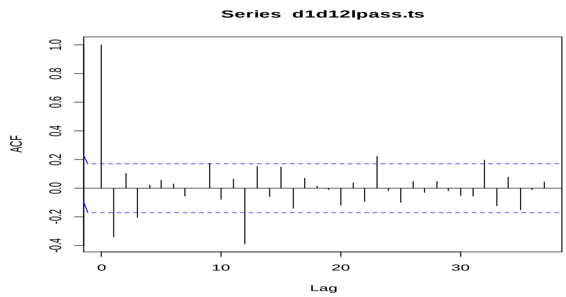


The plot clearly indicates first-order differencing is needed. Thus we will be analyzing log differences, essentially the time series of percentage changes.

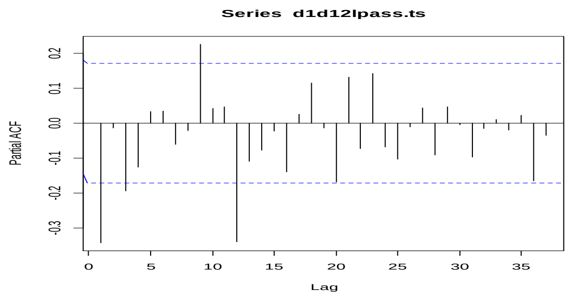
The strong and somewhat regular seasonal pattern also calls for seasonal differencing. Here are the plot of the data and the acf and pacf estimates after both regular differencing and seasonal differencing. There is clear evidence of changing volatility and of dynamic seasonal structure, and this may cause some trouble. The acf and pacf are tabulated out to 37 lags.




```
> lpass<-log(passengers)
> d1d12lpass.ts<-ts(diff(diff(lpass),12))
> acf(d1d12lpass.ts,37)
```




```
> pacf(d1d12lpass.ts,37)
```



The acf and pacf plots do suggest trying an ARIMA(0,1,1)(0,1,1)$_{12}$ fit.

```
> lpass.ts<-ts(lpass)
> airlinemodel<-
arima(lpass.ts,order=c(0,1,1),seasonal=list(order=c(0,1,1),period=12))
> airlinemodel

Call:
arima(x = lpass.ts, order = c(0, 1, 1), seasonal = list(order = c(0, 1,
1),
    period = 12))

Coefficients:
          ma1     sma1
      -0.4018  -0.5569
s.e.   0.0896   0.0731

sigma^2 estimated as 0.001348:  log likelihood = 244.7,  aic = -483.4

> coeftest(airlinemodel)

z test of coefficients:

      Estimate Std. Error z value  Pr(>|z|)
ma1  -0.401828   0.089644 -4.4825 7.377e-06 ***
sma1 -0.556945   0.073100 -7.6190 2.557e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This is the fit published in the 1970 book by Box and Jenkins. It states that

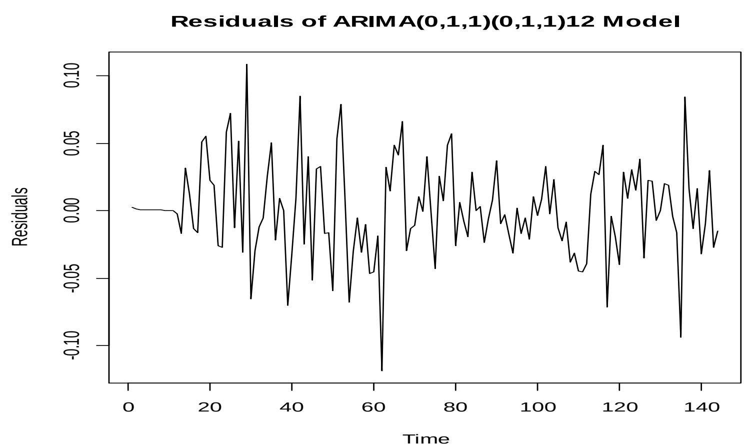
$$(1-B)(1-B^{12})y_t = (1-0.4018B)(1-0.5569B^{12})\varepsilon_t,$$

or

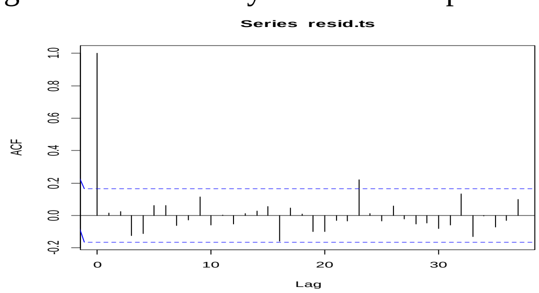
$$y_t = y_{t-1} + y_{t-12} - y_{t-13} + \varepsilon_t - 0.4018\varepsilon_{t-1} - 0.5569\varepsilon_{t-12} + 0.2238\varepsilon_{t-13}.$$

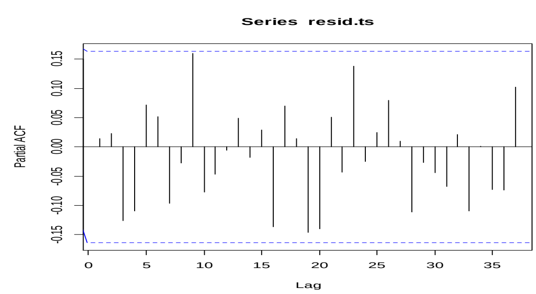
The seasonal differencing (D = 1) included in the model estimates the static seasonal structure, and the remaining part of the seasonal ARIMA command (here Q = 1) estimates dynamic seasonal structure.

```
> resid.ts<-ts(resid(airlinemodel))
> plot(resid.ts,xlab="Time",ylab="Residuals",main="Residuals of
ARIMA(0,1,1)(0,1,1)12 Model",lty=1)
```

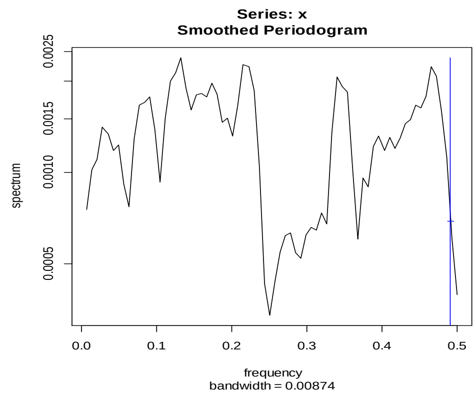


The residual plot still exhibits some change in volatility over time. Moreover, there are apparent outliers at times 62 and 135. I tried modifying the data values at these times, but the fit did not change in a material way. The acf and pacf of the residuals follow.





There is only one significant residual autocorrelation, at lag 23, perhaps a spillover from lag 24. None of the residual partial autocorrelations is significant.




```
> bartlettB.test(resid.ts)

        Bartlett B Test for white noise

data:
= 0.6863, p-value = 0.7339
```

The Bartlett test does not reject the white noise hypothesis. However, the residual spectral plot does show some spectral activity at the calendar frequencies 0.220 and 0.348.

In the next set of notes, we will revisit both the beer and airline series, with additional analysis, to obtain seasonal index estimates from the ARIMA model fits. The analysis will also attempt to address calendar frequency components for the airline series.

Summary and additional remarks

1. ARIMA seasonal models have both nonseasonal and seasonal parts, with the two connected multiplicatively. The models provide a parsimonious way to describe both nonseasonal and seasonal structure at the same time.

2. To fit an ARIMA seasonal model to a time series, one first examines the time series plot to determine if differencing is needed to eliminate trending. If there is a very strong and clear seasonal pattern, seasonal differencing will also be needed. One performs the requisite differencing and then examines the sample autocorrelations and partial correlations of the differenced series (with both ordinary and seasonal differencing performed, if appropriate). Given this view, one selects the AR and MA orders for both the nonseasonal and seasonal parts. Then one fits an ARIMA(p, d, q)(P, D, Q)$_s$ model, where s is the seasonal period.

3. Forecasts with a regression model and with an ARIMA model are compared for the U.S. beer data. The regression model provides slightly better predictions when the last year of data is withheld.

4. The international airline passenger data series has been fit with an $ARIMA(0,1,1)(0,1,1)_{12}$ model, which has come to be called the airline model. This ARIMA structure is quite commonly used to fit time series data. The notes explore the residuals from this model and suggest that some calendar structure remains.

5. The next set of notes will revisit both the U.S. beer and airline series.