Statistics 435/535/711 12 January 2022
P. Shaman

Introductory Remarks; Regression Analysis of Time Series

Time series, introductory discussion

A time series is a sequence of observations, $y_t, t = 1,\ldots,T,$ of some phenomenon, where the index t represents time, or space, or time and space. In this course we will restrict attention to indices representing time.

The index t may in truth be continuous, but with observations recorded discretely. For example, if y_t is the temperature at time t at a specific location, there is fluctuation continuously over time, but we typically record temperature discretely, at equally-spaced points in time, such as hourly on the hour, or daily at a specific clock time. Sometimes recording is actually made continuously, as with an EKG or an EEG. However, usually such analog signals are digitized, yielding a sequence of observations in discrete time (e.g., we take 200 equally-spaced readings per second for an EEG trace).

In applications in business, economics and accounting, y_t sometimes represents an aggregation over some period of time, such as weekly, monthly, quarterly, or annual sales. A published time series may actually be a further aggregation of such measurements. For example, sales data for some commodity may have been collected monthly, but the published series may only report quarterly sales. The U.S. government and governments of other countries publish many economic time series. An example is quarterly gross domestic product. A variable which measures aggregation over time is called a *flow* variable. Flow variables typically describe a rate of speed.

Other types of time series are instantaneous measurements, such as the result of an exchange rate transaction or the sale of a lot of common stock. These are examples of *stock* variables. Of course, a stock measurement may have resulted from a flow in or a flow out of some quantity, as with the measurement of total assets of an investment fund.

One can argue that an exchange rate, or the price of a stock, varies almost continuously over time, with changes occurring whenever there is a transaction. It is common to record and study such a phenomenon at equally-spaced discrete points in time, such as hourly, daily or monthly. We shall restrict attention in this course to time series with equally-spaced observations.

If we record a time series discretely at a coarser spacing than otherwise might be chosen (e.g., temperature measurements are recorded daily instead of hourly, or the EEG signal is digitized at the rate 20 observations per second instead of 200 observations per second), there is typically some loss of information. In designing a data collection scheme, we need to choose the sampling interval so that questions of interest can be

properly addressed. For example, if we want to study annual seasonal variation of sales, it would not suffice to record only aggregate annual sales.

Often the structure of a time series depends upon the sampling interval chosen. To illustrate, suppose the value of the S&P500 index is recorded at equally-spaced points in time, and returns are then calculated from these data. The nature and the degree of volatility we observe will vary depending upon the sampling interval (e.g., daily returns, or monthly returns, or annual returns). This results from the action of the central limit theorem in probability theory.

Applications where the index set represents space, or space and time, are common in physical science settings.

Decomposition models

We begin our study of time series data with *decomposition models* and use regression methods to fit these models to the data. We distinguish two types of such models. An *additive decomposition model* has the form

$$y_t = T_t + S_t + \varepsilon_t, \tag{1}$$

where y_t is the response, T_t is a trend component (one that is slowly varying), S_t is a seasonal component, and ε_t is a disturbance term, also called the irregular component. Thus, the irregular component accounts for features of the time series not covered by the trend and seasonal parts.

A *multiplicative decomposition model* is given by

$$y_t = T_t S_t \varepsilon_t. \tag{2}$$

If we log (2), we obtain an additive model,

$$\begin{aligned} \log y_t &= \log T_t + \log S_t + \log \varepsilon_t \\ &= T_t' + S_t' + \varepsilon_t'. \end{aligned} \tag{2′}$$

That is, the operation of logging converts the multiplicative form to one that is additive in the logs of the components. This conversion of the multiplicative model to additive form permits use of the usual statistical methodology for analyzing data with a multiplicative decomposition model.

The multiplicative model is useful when the variance of y_t increases as the level increases. To decide which of these formulations to employ, one plots the time series, with the response on the vertical axis and time on the horizontal axis. If there is

increasing volatility as the level of the response rises, the multiplicative model should be used, rather than the additive model. However, if the volatility does not change as the response level increases, both the additive and multiplicative approaches can be used. In such a case, the two offer different interpretations, and it is often useful to consider the interpretation offered by the multiplicative model.

The multiplicative model addresses percentage changes in y_t, whereas the additive model deals directly with level changes. Suppose that y_t increases from one observation to the next by the percentage 100δ, for small δ. That is,

$$y_{t+1} = y_t(1+\delta).$$

Then, using logarithms to the base e and Taylor series, we have

$$\begin{aligned} \log y_{t+1} - \log y_t &= \log[y_t(1+\delta)] - \log y_t \\ &= \log y_t + \log(1+\delta) - \log y_t \\ &= \log(1+\delta) \\ &\approx \delta. \end{aligned} \tag{3}$$

Thus, the level of $\log y_t$ changes (approximately) by the amount δ from time t to time $t + 1$ if δ is small. The closer δ is to 0, the better is the approximation given by (3).

To consider further the role that logarithms play in describing percentage changes, consider the following simple numerical example. In the first column of the table below the numbers increase multiplicatively by a factor of 1.1. That is, the numbers increase by ten per cent at each step. The second column gives the corresponding log to the base 10 and the third column shows the corresponding natural logarithm (log to the base e).

N	$\log_{10} N$	$\log_e N$
1.0	0.0	0.0
1.1	0.04139	0.09531
1.21	0.08279	0.19062
1.331	0.12418	0.28593
1.4641	0.16557	0.38124
1.61051	0.20696	0.47655
1.771561	0.24836	0.57186
1.9487171	0.28975	0.66717

As we can see, while the numbers in the first column increase at each step by a *constant multiplicative factor*, the numbers in the second and third columns increase at each step

by a *constant additive amount*. Moreover, this holds for any logarithmic base. Furthermore, if the log to the base e is used, the percentage change at each step is approximately equal to the change in the level of the log value, for small percentage changes.

Model assumptions

As noted, we begin our study of time series by using regression methods to model the trend and seasonal components. We assume that the trend and seasonal components are nonrandom, and that the disturbance term is random and satisfies, for the additive model, the following three assumptions:

(i) The ε_t's have mean 0 and constant variance.

(ii) The ε_t's are uncorrelated.

(iii) The ε_t's are normally distributed.

If the model is multiplicative, the assumptions are the same, with ε_t replaced by ε_t', defined in (2′).

To begin, we assume that the seasonal structure is *periodic*. A periodic seasonal component repeats a cycle exactly; e.g., an annual periodic seasonal component repeats its pattern every 12 observations when the data are observed monthly (more about this later).

These assumptions, that the trend and seasonal are nonrandom, that the seasonal is periodic, and that the disturbance term satisfies (i)-(iii), are very restrictive. We will relax these assumptions as we proceed and introduce other methods of analysis. In fact, at the outset we will see that, for the time series analyzed, the assumption that the seasonal component is exactly periodic, is only an approximation to the seasonal structure; there is typically additional structure that such an approach does not capture. We term an exactly periodic seasonal term a *static* structure. Later in the course we will see how to model a seasonal pattern which evolves over time. We term such a seasonal pattern a *dynamic* structure. Most time series we will encounter have some amount of dynamic structure. Later, when we consider ARIMA modeling, we will encounter model structure which assumes the trend is random.

Assumptions (i)-(ii) correspond to a so-called white noise model. In the regression approach, we will attempt to estimate the trend and seasonal components, with the aim of reducing y_t to white noise. If we have succeeded in reducing to white noise, then there is no additional correlation structure which can be used to advantage to improve our modelling of the level of the time series. (This is a valid and complete approach if assumption (iii) also holds, because then the correlation structure completely

characterizes the data. However, if (iii) does not hold, the reduction to white noise does not extract all the useful information contained in the data. We will address this in the last part of the course in studying ARCH and GARCH models.) In modelling the level of the time series with methodologies other than regression, we will also adopt as a goal reduction to white noise.

We note that, when assumption (iii) is not valid, sometimes a transformation of the data will bring about the approximate validity of (iii). The log transformation is commonly useful for this purpose.

Our goals in this course are to model time series, so that we may offer explanations, test hypotheses arising from context and theory, forecast future observations, and isolate (and sometimes remove) trend and seasonal components.

As stated above, we start with regression methods to fit trend and seasonal components, and we will examine the residuals to determine if there has been a successful reduction to white noise. We emphasize again that this is a somewhat restrictive approach. It is, however, a good way to begin our discussion of time series.

Examples

The nine examples which follow illustrate various features of time series. For some there is a strong seasonal component, and several have a prominent trend. One series requires a log transformation, and another has some readings which are unusually low. The data sets for most of these time series will be analyzed in detail in subsequent class notes.

The discussions which follow both display the time series and illustrate R commands for time series.

1. Monthly data for U.S. beer production, in millions of barrels, each of 31 gallons, January 1987 through December 2017. This series contains a trend component, a pronounced seasonal component that is close to being periodic, and calendar effects. There is evidence that model (1) with nonrandom trend and seasonal parts, and calendar components, does not permit reduction to white noise, however. Despite this, we will argue that estimation of the static part of the seasonal structure via the model (1) is reasonably accurate.

The data are stored in the file beernew.txt in comma-separated format. There are eight columns and headers. Let's begin by reading in the dataset and using the `attach` command. To paste from the R window, use Courier New font.

```
> 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
```

Note that the `head` command prints the first six lines of the R object. The `tail` command can be used to print the last six lines.

The last four columns are two trigonometric pairs used to capture calendar effects. Calendar effects will be discussed later.

```
> class(usbeer)
[1] "data.frame"
```

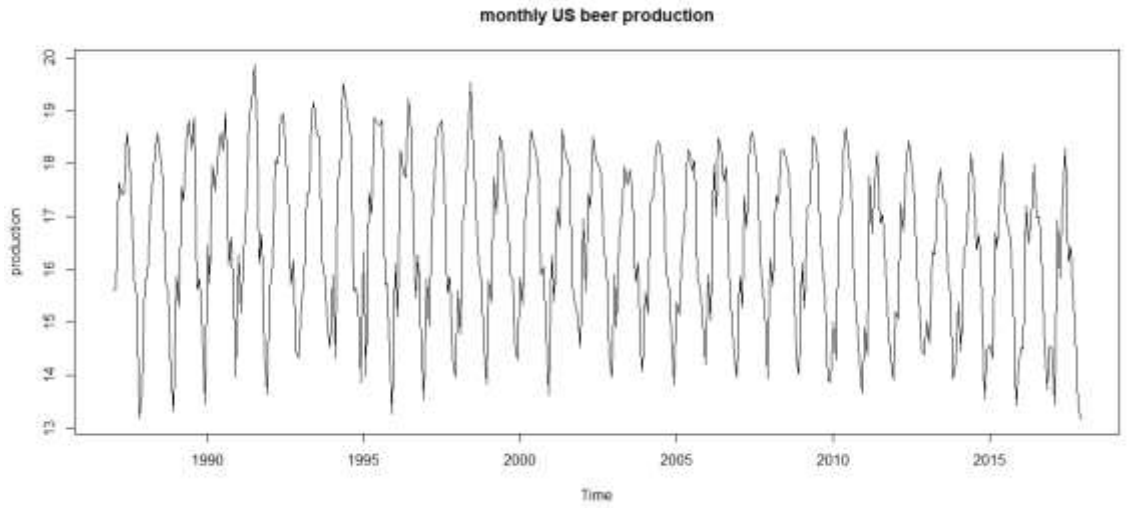
R operates upon *objects*, such as scalars, vectors, matrices, arrays, lists, data frames, and time series. The class of the object `usbeer` is data frame. The following command converts the variable `beer` to a time series object.

```
> usbeer.ts<-ts(beer,start=c(1987,1),freq=12)
```

The frequency specification indicates that 12 monthly observations span a cycle (for the seasonal component).

Let's plot the production series.

```
> plot(usbeer.ts,ylab="production",main="monthly US beer production")
```



Production increased slowly until about 1998, and it has declined slowly in subsequent years. The dominant feature of the plot is a very strong seasonal component with peak production occurring midyear.

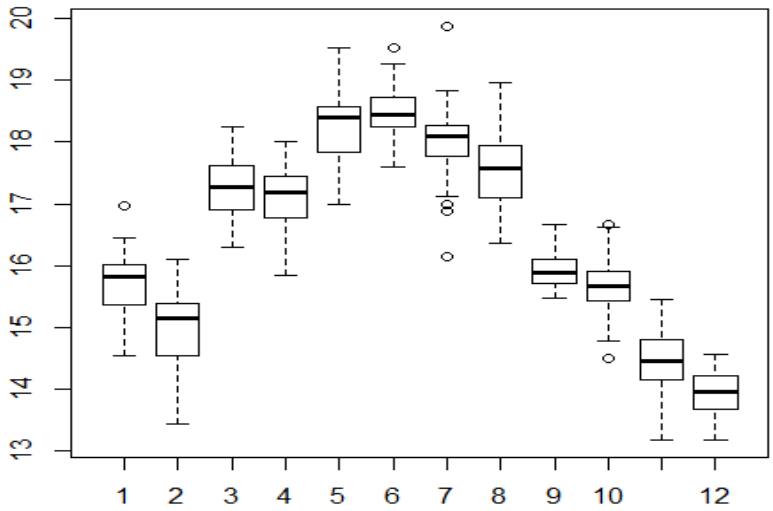
The `cycle` function labels the data according to month.

```
> cycle(usbeer.ts)
```

```
     Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
1987   1   2   3   4   5   6   7   8   9  10  11  12
1988   1   2   3   4   5   6   7   8   9  10  11  12
1989   1   2   3   4   5   6   7   8   9  10  11  12
1990   1   2   3   4   5   6   7   8   9  10  11  12
1991   1   2   3   4   5   6   7   8   9  10  11  12
1992   1   2   3   4   5   6   7   8   9  10  11  12
1993   1   2   3   4   5   6   7   8   9  10  11  12
1994   1   2   3   4   5   6   7   8   9  10  11  12
1995   1   2   3   4   5   6   7   8   9  10  11  12
1996   1   2   3   4   5   6   7   8   9  10  11  12
1997   1   2   3   4   5   6   7   8   9  10  11  12
1998   1   2   3   4   5   6   7   8   9  10  11  12
1999   1   2   3   4   5   6   7   8   9  10  11  12
2000   1   2   3   4   5   6   7   8   9  10  11  12
2001   1   2   3   4   5   6   7   8   9  10  11  12
2002   1   2   3   4   5   6   7   8   9  10  11  12
2003   1   2   3   4   5   6   7   8   9  10  11  12
2004   1   2   3   4   5   6   7   8   9  10  11  12
2005   1   2   3   4   5   6   7   8   9  10  11  12
2006   1   2   3   4   5   6   7   8   9  10  11  12
2007   1   2   3   4   5   6   7   8   9  10  11  12
2008   1   2   3   4   5   6   7   8   9  10  11  12
2009   1   2   3   4   5   6   7   8   9  10  11  12
2010   1   2   3   4   5   6   7   8   9  10  11  12
2011   1   2   3   4   5   6   7   8   9  10  11  12
2012   1   2   3   4   5   6   7   8   9  10  11  12
2013   1   2   3   4   5   6   7   8   9  10  11  12
2014   1   2   3   4   5   6   7   8   9  10  11  12
2015   1   2   3   4   5   6   7   8   9  10  11  12
2016   1   2   3   4   5   6   7   8   9  10  11  12
2017   1   2   3   4   5   6   7   8   9  10  11  12
```

The next plot shows side-by-side boxplots for the months.

```
> boxplot(usbeer.ts~cycle(usbeer.ts))
```



An alternative R command which produces the same plot follows. First, though, we need to change the variable `month` from integer class to factor class.

```
> fmonth<-as.factor(month)
> boxplot(usbeer.ts~fmonth)
```

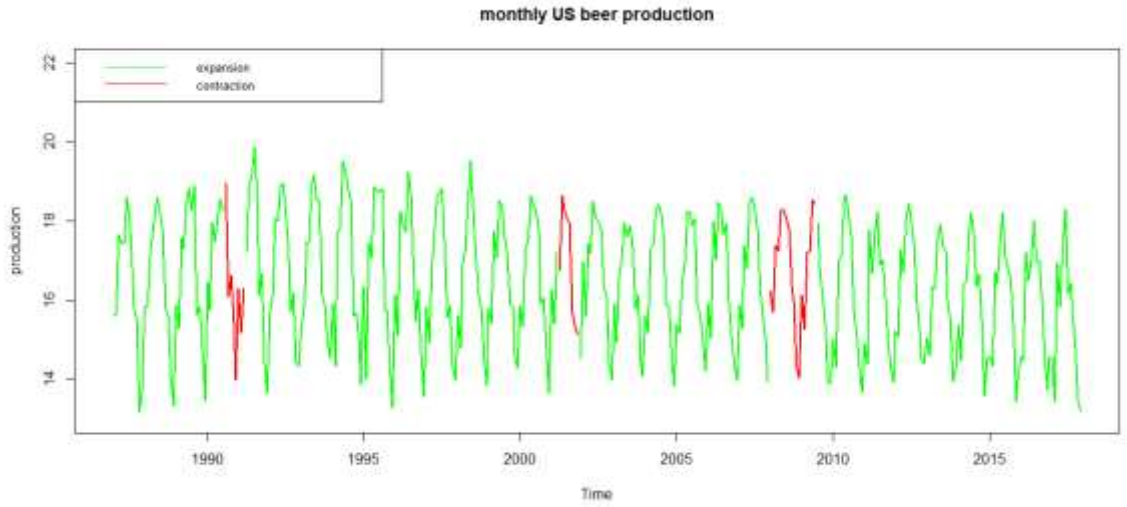
Let's replot the production series to mark U.S. economic contractions. Contraction periods are determined by the Business Cycle Dating Committee of the National Bureau of Economic Research. A later set of notes will provide information and details about the Bureau and the Committee, and how the Committee makes its decisions about occurrences of expansions and contractions. For now we note that there were three time spans judged to be contractions by the Committee during the years 1987 to 2017. They are

1990(8) to 1991(3), 2001(4) to 2001(11), and 2008(1) to 2009(6).

The first two were brief, each lasting eight months. The third one is the recession caused by the financial crisis which began to signal trouble in the summer of 2007.

A simple way to mark these on the time series plot is to create two new series, production for expansion periods and production for contraction periods.

```
beerexpansion<-
c(beer[1:43],rep(NA,8),beer[52:171],rep(NA,8),beer[180:252],rep(NA,18),
beer[271:372])
> beercontraction<-
c(rep(NA,43),beer[44:51],rep(NA,120),beer[172:179],rep(NA,73),beer[253:
270],rep(NA,102))
>
plot(ts(beerexpansion,start=c(1987,1),freq=12),ylim=c(13,22),ylab="prod
uction",main="monthly US beer production",col="green",lwd=2)
> lines(ts(beercontraction,start=c(1987,1),freq=12),col="red",lwd=2)
>
legend("topleft",legend=c("expansion","contraction"),col=c("green","red
"),lty=1,cex=0.8)
```



It is evident that periods of economic contraction have had no impact upon beer production.

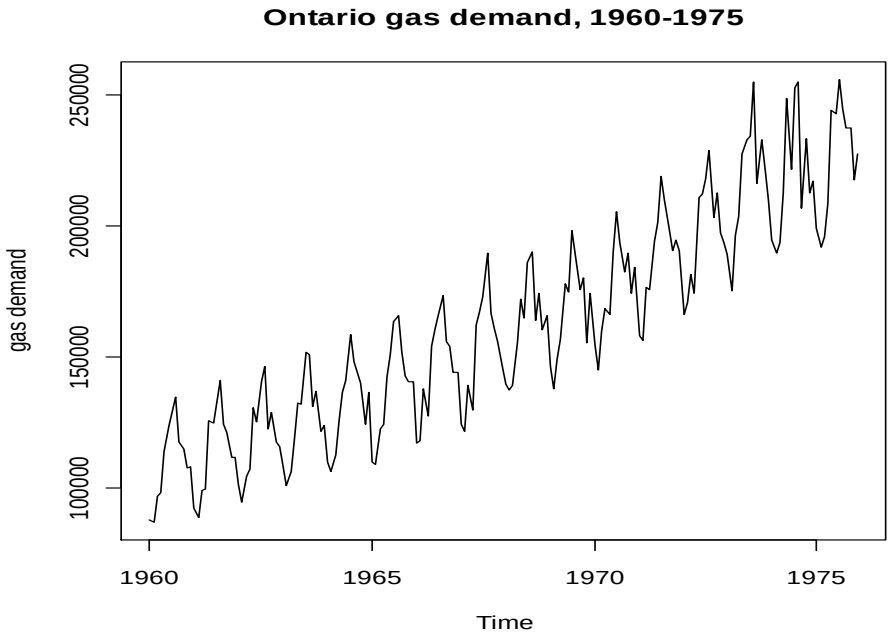
2. Monthly data for gasoline demand in Ontario, in millions of imperial gallons, January 1960 through December 1975. There are a trend component, a prominent seasonal component that is close to periodic, several outliers, and a calendar effect. A fit of the multiplicative model (2) [of course, the additive representation (2′) is used] with nonrandom trend and seasonal parts, the outlier dummies, and the calendar component does not permit reduction to white noise. Estimation of the static portion of the seasonal structure via this modeling is effective, though, we will argue.

```
> ontgas<-read.csv("F:/Stat71122Spring/Ontariogasdemand.txt",header=T)
> attach(ontgas)
> head(ontgas)
  gasdemand loggasdemand year month obs61 obs125 obs177        c348       s348
1     87695     11.38162 1960     1     0      0      0 -0.57757270  0.8163393
2     86890     11.37240 1960     2     0      0      0 -0.33281954 -0.9429905
3     96442     11.47670 1960     3     0      0      0  0.96202767  0.2729519
4     98133     11.49408 1960     4     0      0      0 -0.77846230  0.6276914
5    113615     11.64057 1960     5     0      0      0 -0.06279052 -0.9980267
6    123924     11.72742 1960     6     0      0      0  0.85099448  0.5251746
```

There are three columns for outliers. The last two columns are for a calendar effect.

Let's plot the demand data.

```
> ontariogas.ts<-ts(ontgas[,1],start=c(1960,1),freq=12)
> plot(ontariogas.ts,ylab="gas demand",main="Ontario gas demand, 1960-
1975")
```



There are an upward trend and a strong seasonal component. And there is evidence that a multiplicative model fit may be required.

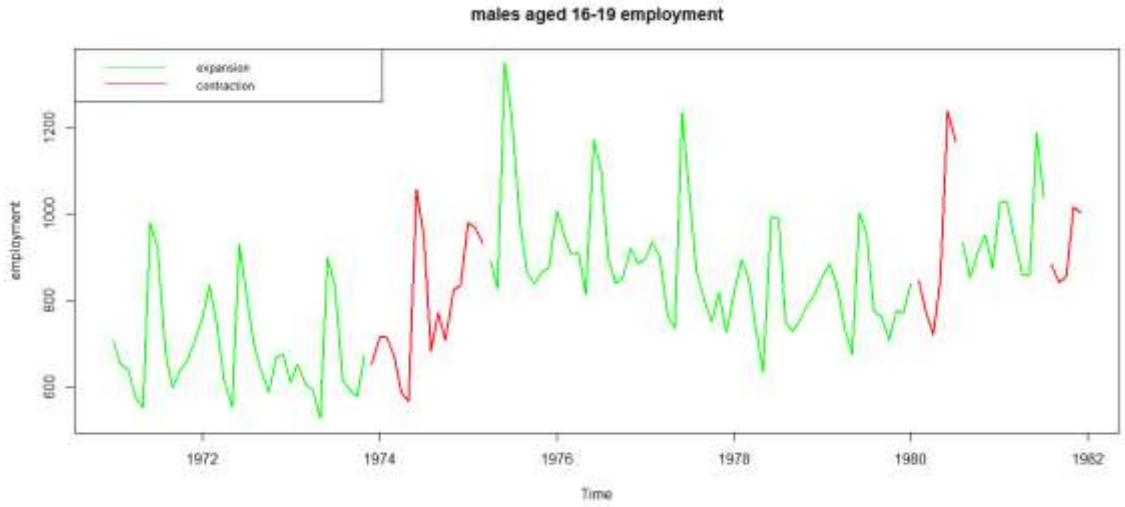
3. Monthly U.S. employment figures for males aged 16-19, in thousands, January 1971 through December 1981. The seasonal component appears to be somewhat regular from year to year, with a secondary and a primary peak each year. The trend is difficult to track with a polynomial fit. A differencing of the time series is of some value—the differencing operation removes the trend and allows one to focus on seasonal estimation. However, differencing does change structure by enhancing fast oscillations.

```
> memp<-read.csv("F:/Stat71122Spring/Memp1619.txt")
> attach(memp)
> head(memp)
  year month employment time
1 1971     1        707    1
2 1971     2        655    2
3 1971     3        638    3
4 1971     4        574    4
5 1971     5        552    5
6 1971     6        980    6
```

There are three periods of economic contraction during 1971 to 1981,

1973(12) to 1975(3), 1980(2) to 1980(7), 1981(8) to 1981(12).

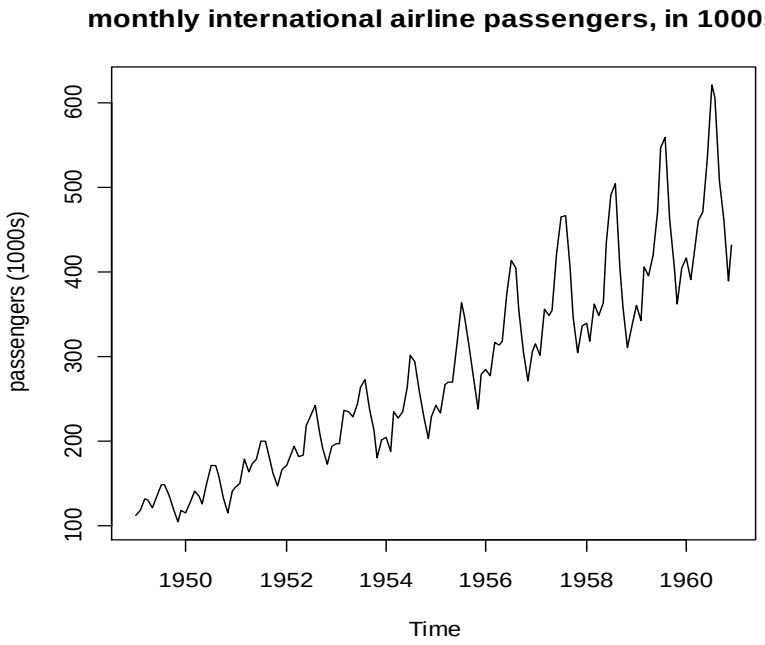
The third contraction period continued until November 1982, in fact.



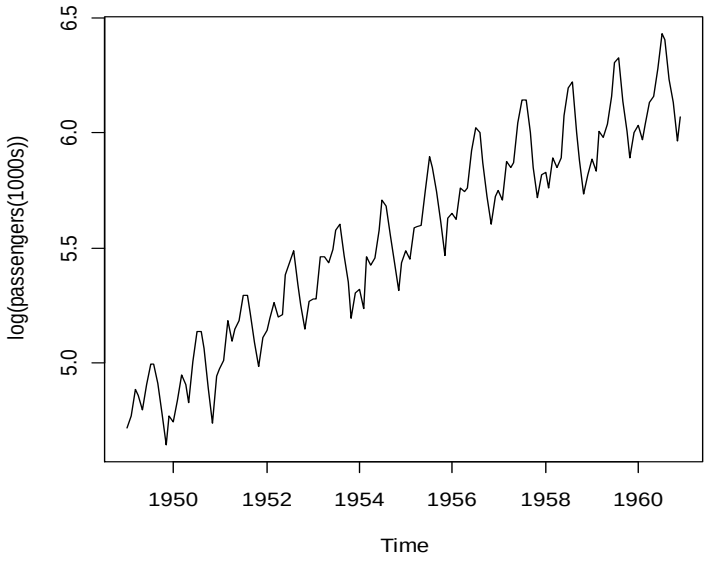
The plot implies that the periods of economic contraction did not have an impact upon monthly employment of young males.

4. Monthly international airline passenger data, January 1949 through December 1960, in thousands. There are a steady upward trend and a strong seasonal component. The latter becomes more variable as the level increases. One should use the multiplicative decomposition model, that is, one should log the series. With the log scale one can see that the structure of the seasonal component changes somewhat over time, perhaps about 1954, and perhaps again in 1958. This is more evident if one examines the residuals from a regression fit. Thus, the seasonal structure is dynamic.

```
> pass<-read.csv("F:/Stat71122Spring/Intair.txt",header=T)
> attach(pass)
> head(pass)
  passengers
1        112
2        118
3        132
4        129
5        121
6        135
> pass.ts<-ts(passengers,start=c(1949,1),freq=12)
> plot(pass.ts,ylab="passengers (1000s)",main="monthly international
airline passengers, in 1000s")
```




```
> lpass<-log(passengers)
> class(lpass)
[1] "numeric"
> lpass.ts<-ts(lpass,start=c(1949,1),freq=12)
> class(lpass.ts)
[1] "ts"
> plot(lpass.ts,ylab="log(passengers(1000s))",main="log monthly
international airline passengers, in 1000s")
```



Next, let's calculate monthly averages for the passenger series. To do so, here we employ a loop. The `window` command extracts the data values for a fixed month in this case.

```
> monthavg<-matrix(rep(0,12),ncol=1)
> for(i in 1:12){
+ monthavg[i]<-mean(window(pass.ts,start=c(1949,i),freq=TRUE))
+ }
> monthavg
          [,1]
 [1,] 241.7500
 [2,] 235.0000
 [3,] 270.1667
 [4,] 267.0833
 [5,] 271.8333
 [6,] 311.6667
 [7,] 351.3333
 [8,] 351.0833
 [9,] 302.4167
[10,] 266.5833
[11,] 232.8333
[12,] 261.8333
```

Let's change the number of digits printed.

```
> options(digits=5)
> monthavg
        [,1]
 [1,] 241.75
 [2,] 235.00
 [3,] 270.17
 [4,] 267.08
 [5,] 271.83
 [6,] 311.67
 [7,] 351.33
 [8,] 351.08
 [9,] 302.42
[10,] 266.58
[11,] 232.83
[12,] 261.83
```

Let's calculate the monthly averages without using a loop. First we define the month variable.

```
> month<-rep(seq(1:12),times=12)
```

The function `tapply` has three arguments. The first specifies the data, the second the grouping of the data, and the third the calculation applied to each group.

```
> matrix(tapply(passengers,month,mean),ncol=1)
        [,1]
 [1,] 241.75
 [2,] 235.00
 [3,] 270.17
 [4,] 267.08
 [5,] 271.83
 [6,] 311.67
 [7,] 351.33
 [8,] 351.08
 [9,] 302.42
[10,] 266.58
[11,] 232.83
[12,] 261.83
```

5. Annual lynx trappings in the Mackenzie River district in northwest Canada, 1821 to 1934. The period appears to be slightly less than ten years. The seasonal structure is evidently dynamic, rather than static. Note that the rise time for each cycle is longer than the fall time. The data have been much discussed in the literature, with several different modelling strategies attempted. One paper in the literature fits the model

$$y_t = \mu + A\cos\lambda t + B\sin\lambda t + \varepsilon_t,$$

with ε_t described as an autoregressive model of order two.

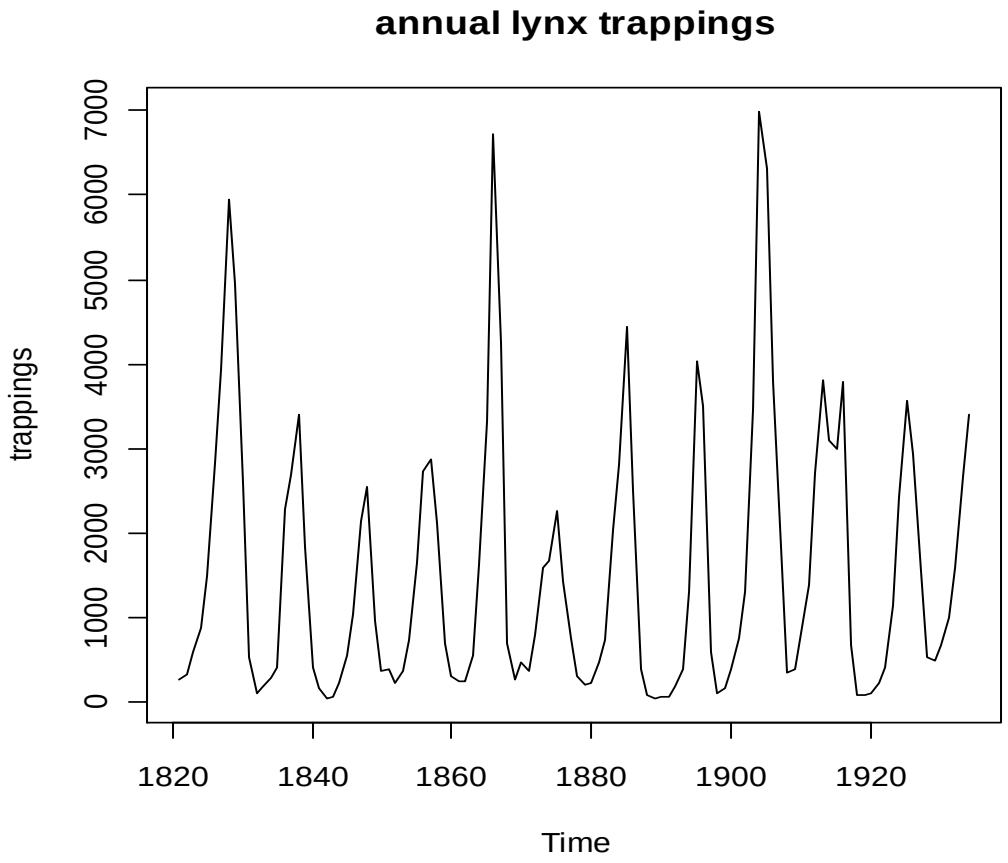
The dataset has five columns which are comma-separated, and headers are present.

```
> trappings<-read.csv("F:/Stat71122Spring/Lynx.txt",header=T)
> attach(trappings)
> head(trappings)
  lynx lnmink lnmuskrat mink muskrat
1  269     NA         NA   NA      NA
2  321     NA         NA   NA      NA
3  585     NA         NA   NA      NA
4  871     NA         NA   NA      NA
5 1475     NA         NA   NA      NA
6 2821     NA         NA   NA      NA

> trappings[28:33,]
   lynx  lnmink lnmuskrat     mink  muskrat
28 2536      NA         NA       NA       NA
29  957      NA         NA       NA       NA
30  361 10.2962   12.0752 29619.85 175465.9
31  377  9.9594   12.1791 21150.10 194677.6
32  225 10.1210   12.5863 24859.62 292523.4
33  360 10.1327   13.1102 25152.18 493955.1
```

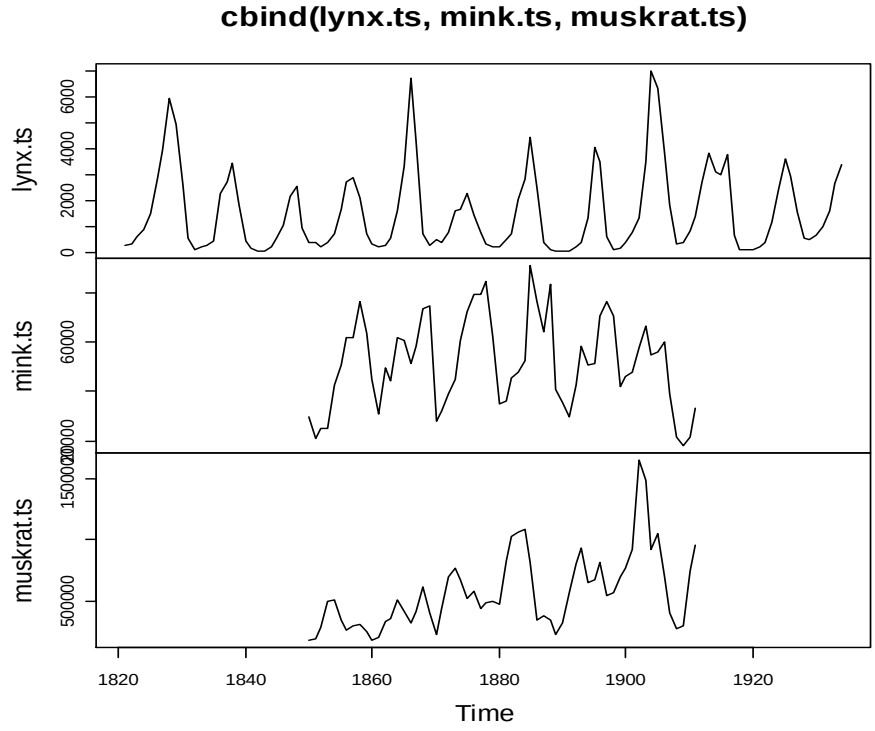
A plot of the lynx series follows.

```
> lynx.ts<-ts(trappings[,1],start=1821)
> plot(lynx.ts,ylab="trappings",main="annual lynx trappings")
```



The mink and muskrat series span the period 1850–1911. Below we use the command `cbind` to plot all three series for this period in one figure. This command combines objects as columns.

```
> lynx.ts<-ts(trappings[,1],start=1821)
> mink.ts<-ts(trappings[,4],start=1821)
> muskrat.ts<-ts(trappings[,5],start=1821)
> plot(cbind(lynx.ts,mink.ts,muskrat.ts))
```



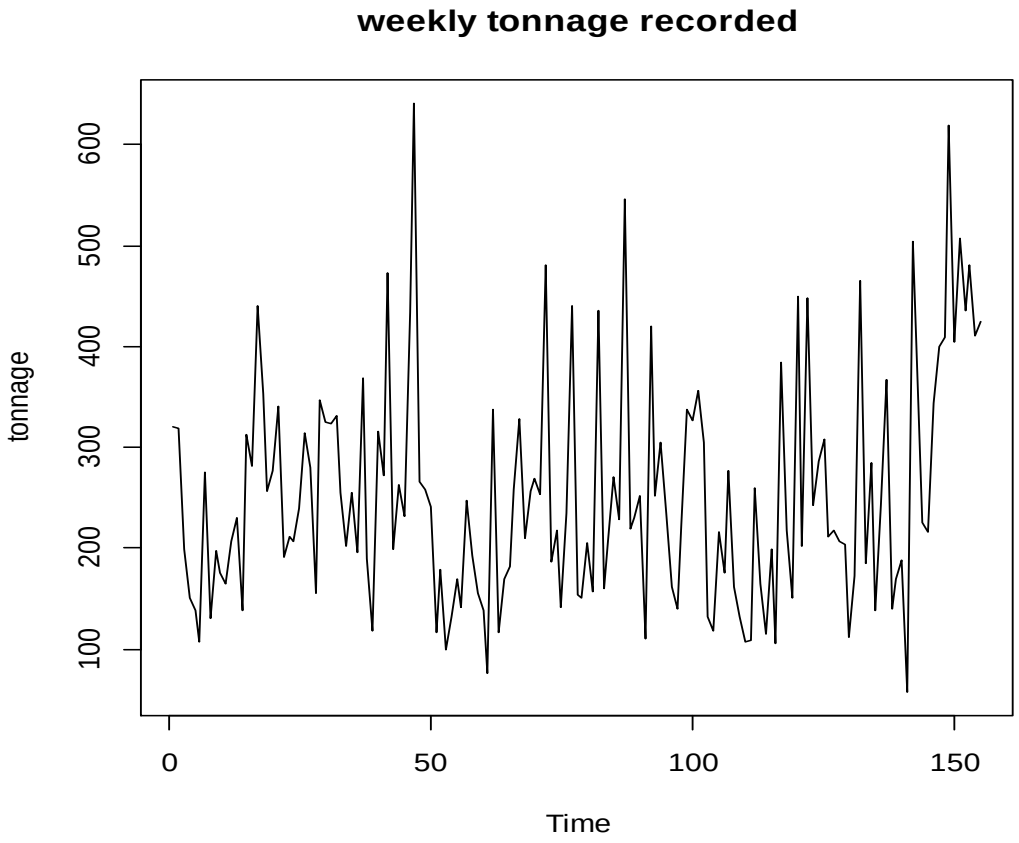
There is some similarity between the lynx and mink series.

6. Weekly garbage deposits, in tons, recorded at the Delaware Solid Waste Authority's Delaware Reclamation Project, 155 consecutive weeks, beginning 30 December 1984. This was a criminal case. There were charges of falsifying weights lodged against a refuse collector and two employees of the Solid Waste Authority. In this case the goal of model construction was estimation of the lost tonnage (and hence lost revenue) attributable to the criminal activity.

```
> garbage<-read.csv("F:/Stat71122Spring/garbage.txt")
> attach(garbage)
> head(garbage)
    date fiscalyr   fee tonnage marlene george mg
1 123084       85 28.26  321.14       0      1  0
2  10685       85 28.26  319.75       0      0  0
3  11385       85 28.26  199.35       1      0  0
4  12085       85 28.26  151.59       1      0  0
5  12785       85 28.26  137.98       0      1  0
6  20385       85 28.26  106.76       0      1  0
```

The variable *tonnage* gives the weight in tons of garbage deposited and for which the deposit fee was paid during a weekly period. The date listed is a Sunday. It marks the beginning of a weekly period. Marlene and George were weighmasters charged with criminal activity, and their variables indicate the weeks during which they worked. The *mg* column identifies the weeks during which both Marlene and George worked.

```
> tonnage.ts<-ts(garbage[,4])
> plot(tonnage.ts,ylab="tonnage",main="weekly tonnage recorded")
```



There are some trending and an evident seasonal pattern. In addition, the weeks when Marlene and George worked have low tonnage figures because of the criminal activity.

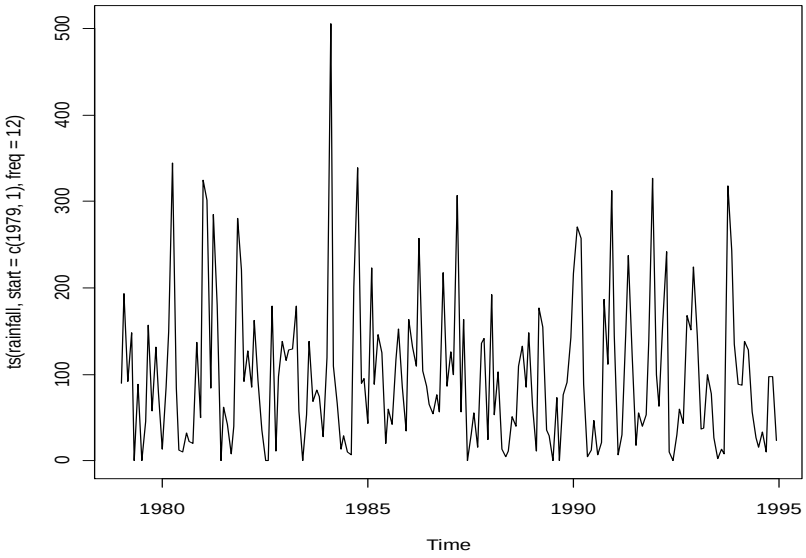
7. Monthly rainfall records, in millimeters, on a farm in Argentina, 1979–1994. A nonrandom seasonal fit does account for some portion of the variation of this series. However, there is still substantial residual variation.

```
> rain<-read.csv("F:/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))
```

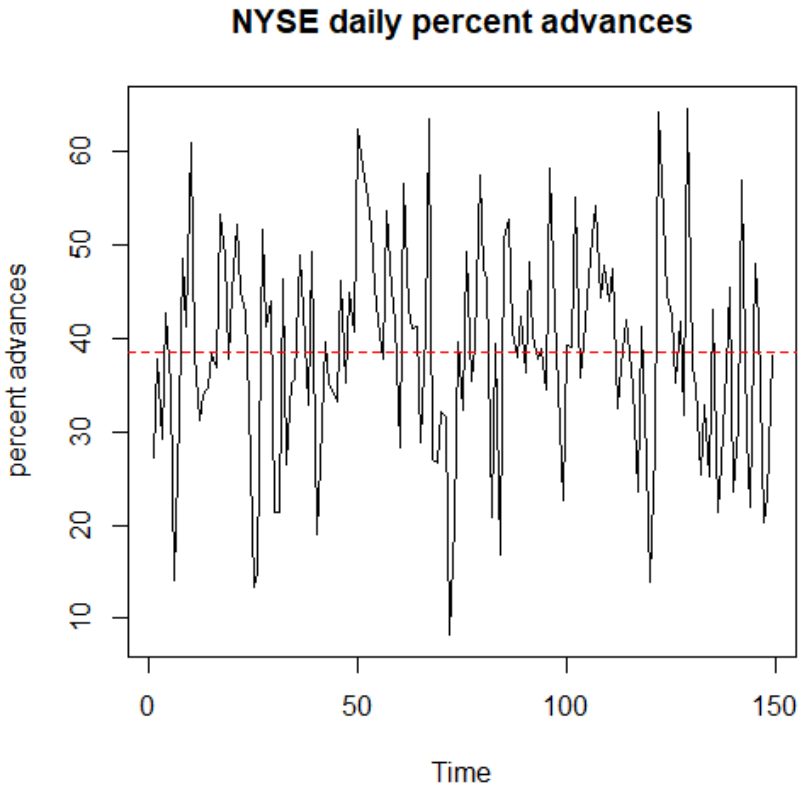


There is no visible trend, as one would expect. There is evidence of changing volatility.

8. Percentage of listings on the New York Stock Exchange advancing, for 149 consecutive trading days. Later we will fit an ARMA model to this data set and will observe that the signal is weak.

```
> nyseadv<-read.csv("F:/Stat71122Spring/nyseadv.txt")
> attach(nyseadv)
> head(nyseadv)
  pctadvnc
1    27.22
2    37.85
3    29.20
4    42.77
5    34.81
6    14.14

> pctadvnc.ts<-ts(pctadvnc)
> plot(pctadvnc.ts,ylab="percent advances",main="NYSE daily percent
advances")
> abline(h=mean(pctadvnc),lty=2,col="red")
```



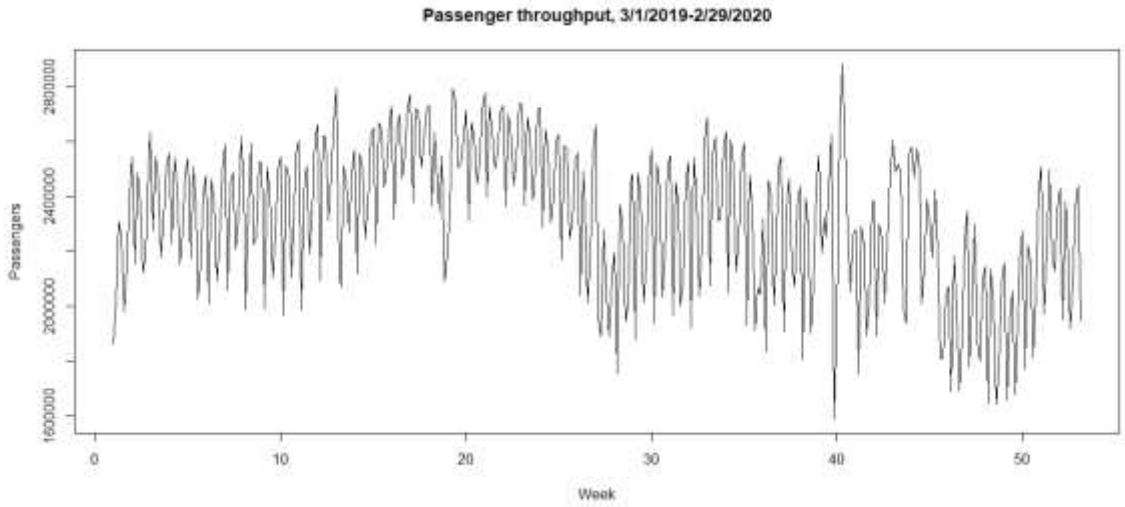
The plot indicates that the time span of these observations was one in which the market was declining.

9. Daily passenger throughout at U.S. airports, 1 March 2019 to 31 December 2021. The data were obtained from the Department of Homeland Security website. The weekdays are coded 1 to 7, with 1 designating Sunday.

```
> tsa<-read.csv("F:/Stat71122Spring/TSAnew2.txt")
> attach(tsa)
> head(tsa)
      Date Year Weekday Travelers logTravelers Time
1 3/1/2019 2019       6   1861286     14.43678    1
2 3/2/2019 2019       7   2015079     14.51617    2
3 3/3/2019 2019       1   2307393     14.65163    3
4 3/4/2019 2019       2   2257920     14.62995    4
5 3/5/2019 2019       3   1979558     14.49838    5
6 3/6/2019 2019       4   2143619     14.57801    6
```

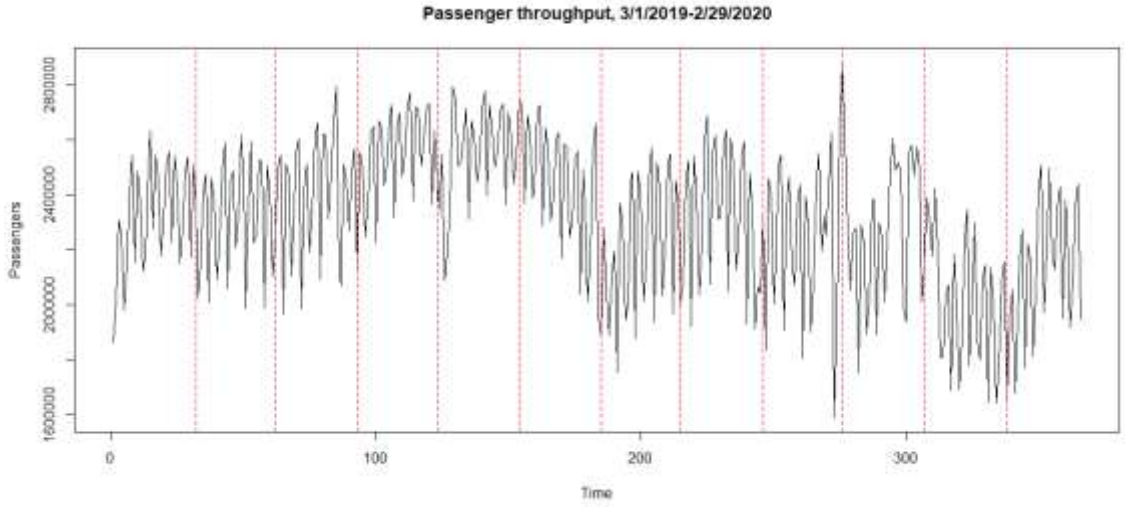
Let's plot the 1 March 2019 to 29 February 2020 series.

```
> tsa19.ts<-ts(Travelers[1:366],start=c(1),freq=7)
> plot(tsa19.ts,ylab="Passengers",xlab="Week",main="Passenger
throughput, 3/1/2019-2/29/2020")
```



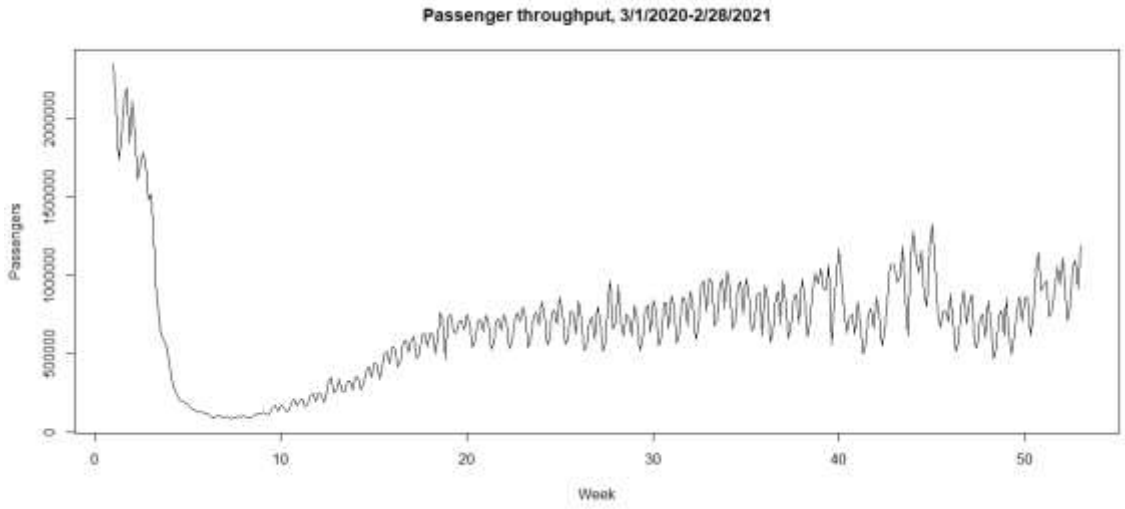
What is responsible for the low passenger throughput numbers in week 18 and the high variability in week 40? To help answer let's redraw this plot to show the first days of the months April, May,…, February.

```
> tsa192.ts<-ts(Travelers[1:366])
> plot(tsa192.ts,ylab="Passengers",xlab="Time",main="Passenger
throughput, 3/1/2019-2/29/2020")
> abline(v=c(32,62,93,123,154,185,215,246,276,307,338),lty=2,col="red")
```



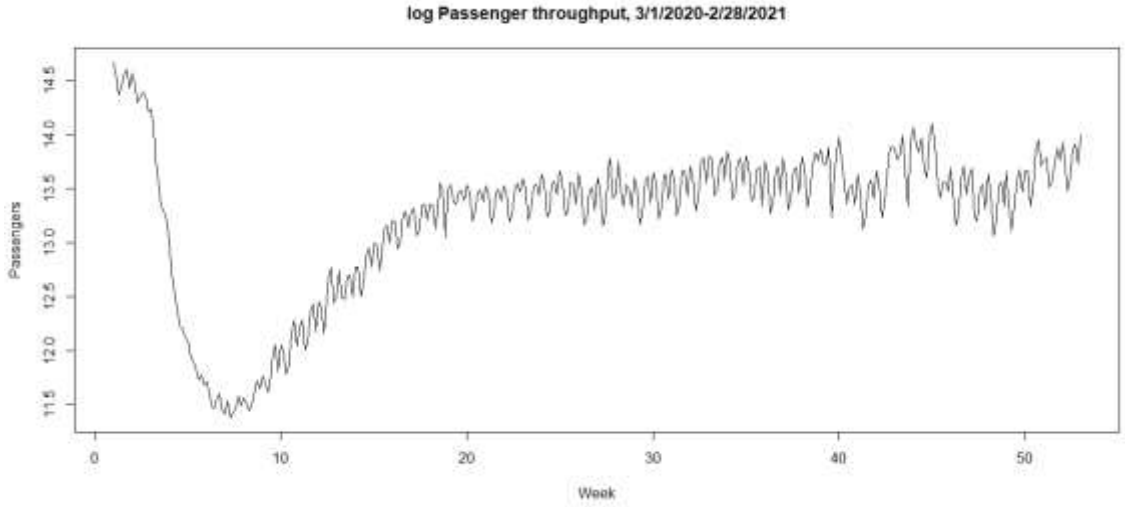
Next, let's plot the daily series for 1 March 2020 to 28 February 2021.

```
> tsa20.ts<-ts(Travelers[367:731],start=c(1),freq=7)
> plot(tsa20.ts,ylab="Passengers",xlab="Week",main="Passenger
throughput, 3/1/2020-2/28/2021")
```



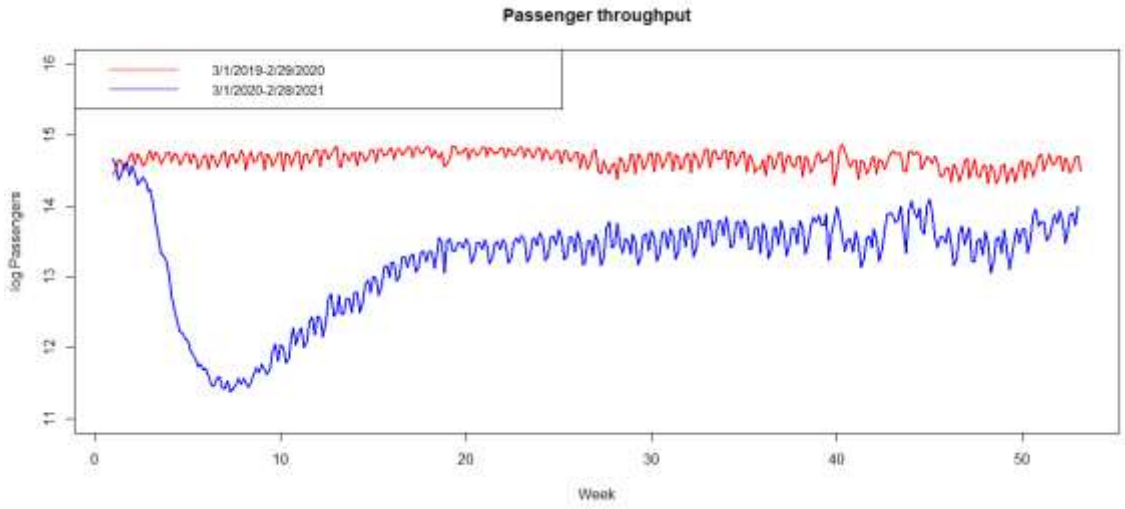
A plot of the logged daily passenger counts is more appropriate.

```
> ltsa20.ts<-ts(log(Travelers)[367:731],start=c(1),freq=7)
> plot(ltsa20.ts,ylab="Passengers",xlab="Week",main="log Passenger
throughput, 3/1/2020-2/28/2021")
```



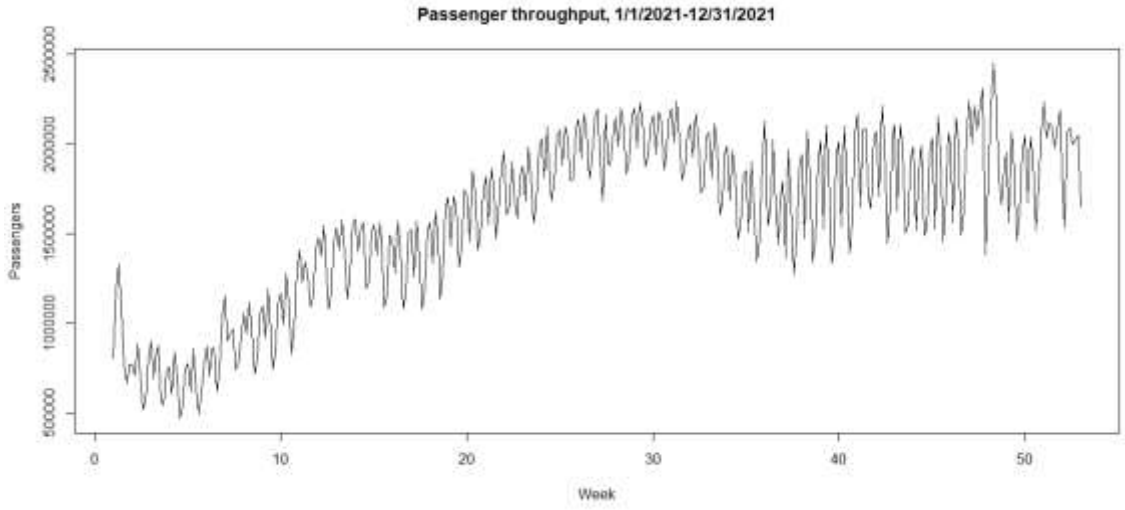
Let's picture the 2019 and 2020 series on one plot. We log the 2019 series in the plot.

```
> ltsa19.ts<-ts(log(Travelers)[1:366],start=c(1),freq=7)
> plot(ltsa19.ts,ylab="log Passengers",xlab="Week",main="Passenger
throughput",ylim=c(11,16),lty=1,lwd=2,col="red")
> lines(ltsa20.ts,lty=1,lwd=2,col="blue")
>
legend("topleft",legend=c("3/1/2019-2/29/2020","3/1/2020-2/28/2021"),co
l=c("red","blue"),lty=1,cex=0.9)
```



Both series display a prominent seasonal component, describing variation among the weekdays. What pattern has developed since 1 January 2021?

```
> tsa21.ts<-ts(Travelers[673:1037],start=c(1),freq=7)
> plot(tsa21.ts,ylab="Passengers",xlab="Week",main="Passenger
throughput, 1/1/2021-12/31/2021")
```



Below we briefly explore variation among days of the week. We calculate for the 1 March 2019–29 February 2020 data weekday averages of the passenger throughput and deviations of the weekday averages from the (approximate) daily grand mean.

```
> weekdayav19<-tapply(Travelers[1:366],Weekday[1:366],mean)
> dmweekdayav19<-weekdayav19-mean(weekdayav19)
> cbind(weekdayav19,dmweekdayav19)
```

```
  weekdayav19 dmweekdayav19
1     2457866      135980.1
2     2428052      106166.6
3     2128908     -192977.3
4     2217986     -103899.9
5     2446768      124881.9
6     2501225      179339.4
7     2072395     -249490.9
```

The same calculation for the period 1 January 2021 to 31 December 2021 follows.

```
> weekdayav21<-tapply(Travelers[673:1037],Weekday[673:1037],mean)
> dmweekdayav21<-weekdayav21-mean(weekdayav21)
> cbind(weekdayav21,dmweekdayav21)

  weekdayav21 dmweekdayav21
1     1767802     179305.47
2     1664594      76098.07
3     1355662    -232834.39
4     1430292    -158203.87
5     1690873     102376.67
6     1728013     139517.12
7     1482237    -106259.07
```

The results suggest some change in the weekday pattern in 2021, relative to 2019−2020.

Summary and additional remarks

1. A *time series* is a sequence of observations, $y_t, t = 1,\ldots,T,$ of some phenomenon, where the index t represents time. When analyzing time series data, we need to preserve the order in time of the data values. Examples of time series are monthly U.S. production of automobiles, daily high temperature readings at some location, the quarterly growth of U.S. GDP, and the exchange rate between the U.S. dollar and the euro recorded at the end of each month.

2. A time series variable which measures accumulation over some period of time, such as a month or a quarter, is called a *flow* variable. Examples of flow variables are monthly sales and quarterly production. For such variables there are calendar considerations, such as the differing lengths and differing trading day counts for the months and the quarters. On the other hand, time series variables which represent an instantaneous type of measurement are called *stock* variables. An example is the exchange rate between two currencies at the time of a transaction.

3. A *decomposition model* is a time series structure which includes trend, seasonal, and irregular components. The additive form of the model is $y_t = T_t + S_t + \varepsilon_t$. The trend component T_t is slowly varying, the seasonal component S_t is taken to be periodic or approximately periodic, and ε_t is the disturbance term, also called the irregular component. For estimation of this model with regression analysis, we take the trend and

seasonal components to be nonrandom, the seasonal component to be periodic (that is, to be a static seasonal), and the disturbance variable ε_t to be random with (i) mean 0 and constant variance, and to be (ii) uncorrelated. These two assumptions define a *white noise* structure.

The aim of estimation for the decomposition model is to describe the trend and seasonal components and extract them so that the remainder satisfies the definition of a white noise sequence. We speak of "reducing the data to white noise." If this is accomplished, we conclude that we have completely described the structure of the level of the time series. A third assumption about the disturbance variables, that they are (iii) normally distributed, is made to allow hypothesis testing and confidence interval construction for parameters describing the decomposition model.

4. The multiplicative decomposition model has the form $y_t = T_t S_t \varepsilon_t$. This form is needed for many economic time series. We log this form to convert the multiplicative structure to additive structure,

$$\log y_t = \log T_t + \log S_t + \log \varepsilon_t = T_t^{'} + S_t^{'} + \varepsilon_t^{'}.$$

The logging operation permits estimation with standard statistical methodology to proceed conveniently. In this additive form, we estimate the log of the trend and the log of the seasonal, and thus we need to exponentiate these estimates if we want to return to the original scale of measurement.

The multiplicative formulation is typically used when the variance of y_t increases as the level increases. While additive decomposition models describe level changes over time of the series, the multiplicative structure addresses percentage changes over time of the series. When the response variance increases as the level of the time series increases, a multiplicative model should be used. When the time series exhibits constant variance across time, either the additive or the multiplicative form can be used to analyze the data. The two approaches offer alternative interpretations, and it is often useful to employ both of them.

Polynomials in time for trend estimation

A polynomial in time can be used to estimate the trend of an additive decomposition model with regression. If $y_t, t = 1,\ldots,T,$ denotes the observed time series, a polynomial in time of degree k given by

$$\beta_0 + \beta_1 t + \cdots + \beta_k t^k$$

will serve to perform the estimation. One needs to decide the degree of polynomial to employ. The following steps can be used:

1. Plot the time series response (or its logarithm if appropriate), as in the examples above.

2. Count the number of turns observed in the plotted series.

3. Choose the polynomial degree to be one greater than the observed number of turns, and fit the regression model with such a polynomial trend included.

4. Test the significance of the highest order coefficient in the polynomial fit. If this coefficient is significant, stop and use this polynomial fit as the estimate of trend. Don't refit with insignificant lower order polynomial terms removed (in fact, don't even bother to test them). That is, if you find the highest order coefficient to be significant, retain all the lower order polynomial terms regardless of their significance.

5. If the highest order coefficient in the polynomial fit is not significant, refit with the polynomial degree lowered by 1. Repeat step 4.

Some comments:

1. It is of course okay, and sometimes useful, to experiment with several different polynomial degrees in fitting the trend component. It is usually unwise, however, to fit polynomials of very high degree. These may fit the data well, but they will cause problems if one is trying to forecast future data points with a regression model. Except for some specific purposes we will encounter, I am reluctant to fit a polynomial of degree higher than five or six to track the trend.

2. Some time series contain many turns in the trend. Or there may be rather severe trend fluctuations, as, for example, in a series affected by an economic recession. When this is the case, polynomial fitting will typically not be able to track the movements of the trend. If this occurs, one can difference the data and analyze the differences $y_t - y_{t-1}, t = 2,\ldots,T.$ We will often employ this approach. It is common to find that the

differencing operation eliminates the trend, and then one can estimate the seasonal structure of the original series by analyzing the differenced data.

Seasonal indices

Consider the additive decomposition model $y_t = T_t + S_t + \varepsilon_t$, or the logged version of the multiplicative form, $\log y_t = \log T_t + \log S_t + \log \varepsilon_t$.

For this discussion and for estimation via regression methods, we assume that the seasonal structure is static. That is, there is a perfectly repeating seasonal pattern. Each repetition defines a cycle.

The main case we will encounter is monthly data with an annual cycle. For this the *seasonal indices* are the values $S_1, \ldots, S_{12}$, each corresponding to a month of the year. Because the pattern perfectly repeats with the static structure, $S_t = S_{t+12}$ for all t. In the additive decomposition model, the indices add to 0,

$$S_1 + \cdots + S_{12} = 0.$$

The value of S_t is the amount of the deviation of the expected response from the level of the trend for month t.

If the decomposition model is multiplicative, the logs of the indices add to 0,

$$\log S_1 + \cdots + \log S_{12} = 0.$$

Thus, then the product of the indices is equal to 1,

$$S_1 \cdots S_{12} = 1.$$

For the multiplicative model, the value of S_t assesses the percentage of the level of the trend for month t for the expected response.

If the data are quarterly with an annual cycle, there are four seasonal indices, $S_1, S_2, S_3,$ and S_4. If the data are daily and there is a weekly cycle, there are seven seasonal indices, $S_1, \ldots, S_7$.

Dummy variables for estimation of periodic seasonal structure

Month dummies. Suppose we have monthly data with an annual cycle, and we want to estimate periodic seasonal structure. With R we can use a factor variable to represent the

months. When this variable is included in a regression *which contains an intercept*, R creates 11 dummies (the number of dummies created is one less than the number of categories of the factor variable if the regression contains an intercept). The category which is first in the ordering of the labels of the factor variable (we assume this month is January) does not receive a dummy, and the 11 dummies are as follows:

$x_{2,t}$ = 1 if t corresponds to February

= 0 otherwise,

$x_{3,t}$ = 1 if t corresponds to March

= 0 otherwise,

.
.
.

$x_{12,t}$ = 1 if t corresponds to December

= 0 otherwise.

This choice of dummy variables to describe a static seasonal structure is convenient when writing R code to specify the model we want to fit, because we don't have to construct the dummies—R does it for us. However, the regression coefficient estimates from the model estimation do not directly specify the seasonal index estimates S_t—interpretation of the coefficient estimates is a bit tricky, and we have to write several lines of R code to obtain estimates of the S_t values.

We describe now an alternative set of dummy variables to track a static seasonal structure. Its advantage is that the estimated regression coefficients directly give the seasonal index estimates and interpretation of the coefficient estimates is direct and straightforward. (We do have to calculate S_{12} as the negative sum of the other 11 estimates, however.) The disadvantage of this alternative approach, though, is that we have to write R code to specify the dummies. The dummies with this approach follow. We assume that the regression contains an intercept, and then the category which is last in the ordering of the labels of the factor variable does not receive a dummy.

$x_{1,t}$ = 1 if t corresponds to January

= −1 if t corresponds to December

= 0 otherwise,

$x_{2,t}$ = 1 if t corresponds to February

= −1 if t corresponds to December

= 0 otherwise,

.
.
.

$x_{11,t}$ = 1 if t corresponds to November

= −1 if t corresponds to December

= 0 otherwise.

If we choose to fit the regression without an intercept, each of the two formulations above has be modified.

The two approaches described above give the same seasonal index estimates.

The dummies required for quarterly data with an annual cycle and daily data with a weekly cycle are specified similarly.

Cosine and sine dummies. Still another set of dummy variables to describe a periodic seasonal structure consists of cosine and sine functions. The use of such dummies gives a useful interpretation of the estimated seasonal structure, and it offers more flexibility than the two procedures discussed above. Moreover, the seasonal index estimates produced with a full set of 11 trigonometric functions are identical to those obtained from the two dummy variable formulations described above.

First, let's discuss some features of trigonometric functions. The cosine function

(4) $$f(t) = R \cos(\lambda t + \alpha)$$

has *amplitude* R, *frequency* λ (in radians per unit time; with $\lambda = 2\pi f$, f is the frequency in cycles per unit time), and *phase angle* α. The *period* is $2\pi/\lambda$, because

$$f(t + 2\pi/\lambda) = R\cos(\lambda(t + 2\pi/\lambda) + \alpha) = R\cos(\lambda t + \alpha + 2\pi) = f(t).$$

Clearly it suffices to consider $0 \le \alpha < 2\pi$. The period indicates the length of an individual cycle.

Next we develop an alternative representation of (4). Recall the double angle formula

$$\cos(x + y) = \cos x \cos y - \sin x \sin y.$$

Thus (4) is

(5) $$f(t) = R\cos\alpha\cos\lambda t + (-R\sin\alpha)\sin\lambda t$$

$$= A\cos\lambda t + B\sin\lambda t,$$

with $A = R\cos\alpha$ and $B = -R\sin\alpha$. The amplitude R is equal to $\sqrt{A^2 + B^2}$, and the phase angle α is obtained by solving $\tan\alpha = -B/A$. The phase angle positions the cosine curve (4) to align its peak on the time axis.

Trigonometric functions can be used to construct dummy independent variables in a regression to model seasonality if the seasonal component is taken to be static. When the trigonometric dummy variables are employed, the regression has to be fit with an intercept. Here are some examples, corresponding to different intervals of observation for the time series.

1. If the data are observed monthly and we have an annual seasonal component, we use

$$\cos\frac{2\pi j}{12}t, \quad \sin\frac{2\pi j}{12}t, \quad j = 1,...,5; \quad \cos\frac{2\pi 6}{12}t = (-1)^t$$

as dummy independent variables. The frequencies are $j/12$, $j = 1,...,6$, with corresponding periods 12, 6, 4, 3, 2.4, 2 months. The *fundamental* frequency is 1/12, and the *overtone* frequencies are $j/12$, $j = 2,\ldots,6$. We also say that the pair with frequency 1/12 forms the *first harmonic*, the pair with frequency 2/12 forms the *second harmonic*, and so on.

2. If the data are observed quarterly and the seasonal component is annual, we use

$$\cos\frac{2\pi}{4}t, \quad \sin\frac{2\pi}{4}t; \quad \cos\frac{2\pi 2}{4}t = (-1)^t.$$

The frequencies are 1 /4 and 2 /4, with periods 4 and 2 quarters. There are a fundamental component and one overtone.

3. If the data are observed daily, to model a weekly periodic component we use

$$\cos\frac{2\pi j}{7}t, \quad \sin\frac{2\pi j}{7}t, \quad j = 1,2,3.$$

The frequencies are $j / 7$, $j = 1, 2, 3$, with periods 7, 3.5, 2.33 days. There are a fundamental component and two overtones.

4. If the data are observed weekly and there is an annual seasonal component, we use

$$\cos\frac{2\pi 7 j}{365}t, \quad \sin\frac{2\pi 7 j}{365}t, \quad j = 1,\ldots, J,$$

where J is typically chosen to be less than 25. We often choose J to be substantially less than 25, because a relatively small number of parameters is usually sufficient to describe the seasonal variation.

To be more precise with weekly data, we can use frequencies which are multiples of $2\pi 7 / 365.25,$ to take leap years into account.

For all of the procedures discussed one counts dummy variables (or parameters) to match a full seasonal description. An annual seasonal component with monthly data requires 11 dummy variables when the regression includes an intercept. An annual seasonal component with quarterly data requires three dummy variables with an intercept. A weekly component with daily data requires six dummy variables with an intercept.

The use of trigonometric functions allows us to conveniently use fewer dummy variables than correspond to a full seasonal description, if appropriate. Some seasonal patterns do not require the full set of dummy variables. In such cases one can build a more parsimonious model (that is, one with fewer parameters). We will encounter examples which illustrate this.

With monthly data and an annual cycle, we emphasize that use of the monthly dummy variables with the two procedures given in the previous section will produce *exactly the same regression fit* as that obtained from the full set of 11 trigonometric dummies shown above. The two regression fits offer different interpretations.

Finally, we should note that seasonal structures which are not periodic (not static), or not close to periodic, may be poorly tracked by these dummy variable formulations.

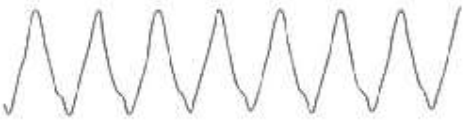
The treatment of monthly flow time series often ignores the fact that the months are not of equal length and that they contain different numbers of trading days, different numbers of Mondays, of Tuesdays, …, etc. As mentioned above, when appropriate, we will use calendar variables to address this matter. Details will appear in subsequent notes.

Musical signals

Musical sounds arise from variations in air pressure. To picture the sound, we graph air pressure on the vertical scale and time on the horizontal scale. The greater the vertical variation is, the louder the sound is. If we picture a note played on a musical instrument, we see a repeating pattern. Two examples are shown on the next page. Note the perfect regularity. Each pattern lasts a fraction of a second, and the rate over time at which the pattern recurs is the frequency of the note. For example, middle C on the piano repeats about 262 times per second, and A above middle C at about 440 times per second. In music this frequency is called pitch. We say that A above middle C has a frequency of 440 cycles per second, or 440 Hertz, written 440 Hz. We term the magnitude of vertical variation the amplitude.

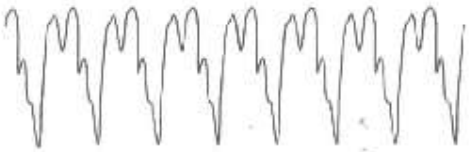
The two examples on the next page are traces of two notes with the same pitch (frequency) and the same volume (amplitude). However, the repeating patterns look very different. The two notes differ in tonal character (timbre is the musical term). The trace at the top of the page is for the note played on a flute, and the other trace is for the note played on a guitar. Note that the traces have the types of patterns we observe for periodic seasonal components of time series.

We can construct the graph of any periodic seasonal pattern by using a weighted summation of cosine curves. The same is true for the graph of a musical note of a given pitch and volume, and any timbre. If the note has frequency f, we use a weighted summation of cosine curves with frequencies f, $2f$, $3f$, … . These are the fundamental component and its sequence of overtones. By varying the weights we vary the timbre. That is, each musical instrument employs a specific sequence of weights to achieve its characteristic tonal character.



Flute

Fig. 10



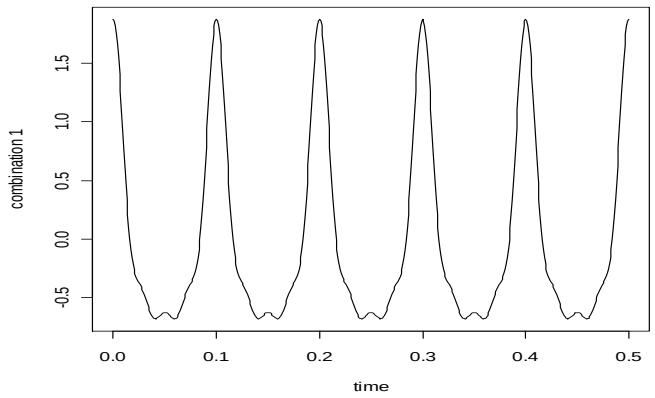
Guitar

But that's not the only way to change a tone. Two notes of the same pitch and the same volume can still sound quite different if they come from different instruments. Figure 10 displays graphs corresponding to a flute and a guitar. Aside from any considerations of pitch or volume, the two graphs clearly set themselves apart by their shapes. Again, when you repeat a pressure pattern at a certain frequency, you get a note at a certain pitch. The nature of the repeated pattern itself determines the note's tonal character (in musical terminology *timbre*), distinguishing a flute from a guitar from a French horn.

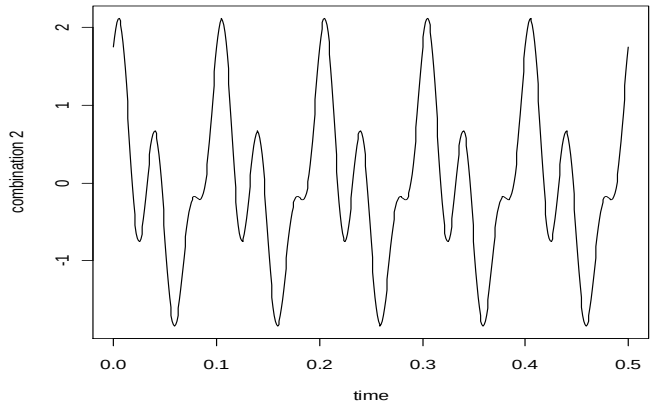
At first glance the infinite variety of possible shapes might seem to make timbre intractable to analyze. Fortunately, some structure comes to the rescue. Every tone can be obtained by combining certain simple building blocks, called *pure tones*. What do I mean by "combine"? Musically, just to play the tones simultaneously. In terms of the air-pressure graphs, for each moment of time you

Let's look at three examples of creating a curve from a fundamental cosine–sine pair and its overtones. The first one below is constructed from a fundamental and its first three overtones, the second one from a fundamental and its first two overtones, and the third from a fundamental and its first three overtones. We will see more examples as the course progresses, as we analyze time series data.

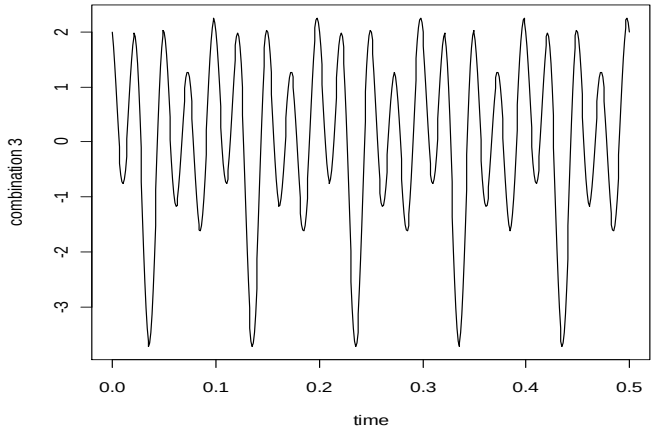
```
> cs<-matrix(rep(0,501*8),nrow=501,ncol=8)
> t<-seq(0,0.5,by=0.001)
> arg<-matrix(c(20*pi*t,40*pi*t,60*pi*t,80*pi*t),nrow=501,ncol=4)
> for(j in 1:4){
+ j2<-2*j;j1<-j2-1
+ cs[,j1]<-cos(arg[,j]);cs[,j2]<-sin(arg[,j])
+ }
> comb1<-cs[,1]+0.5*cs[,3]+0.25*cs[,5]+0.125*cs[,7]
> plot(t,comb1,type="l",xlab="time",ylab="combination 1")
```



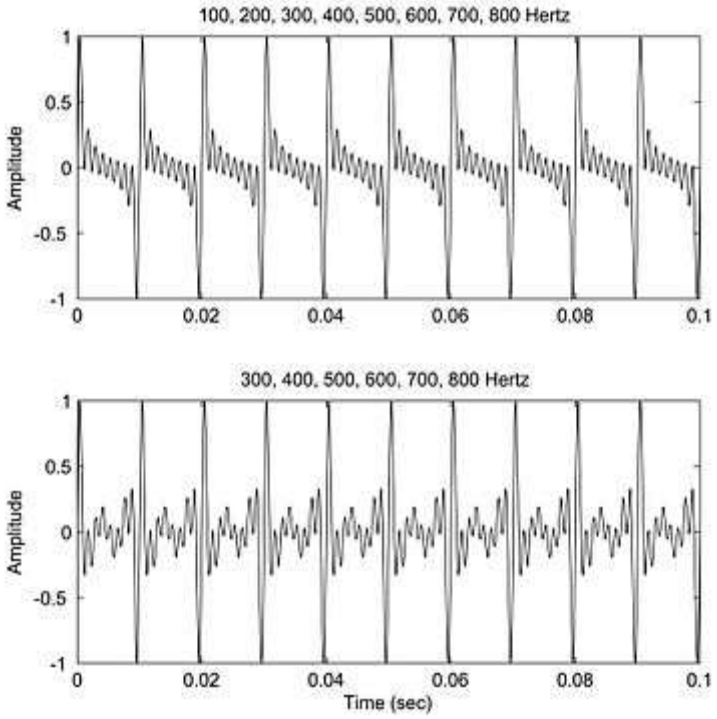

```
> comb2<-cs[,1]+0.5*cs[,2]+0.5*cs[,3]-
0.25*cs[,4]+0.25*cs[,5]+0.75*cs[,6]
> plot(t,comb2,type="l",xlab="time",ylab="combination 2")
```



```
> comb3<-0.5*cs[,1]-0.25*cs[,2]+0.5*cs[,3]+0.75*cs[,4]-0.5*cs[,5]-
0.25*cs[,6]+1.5*cs[,7]-cs[,8]
> plot(t,comb3,type="l",xlab="time",ylab="combination 3")
```



An aside: There is an interesting auditory illusion called the missing fundamental. The following is from Wikipedia, https://en.wikipedia.org/wiki/Missing_fundamental



The bottom waveform is missing the fundamental frequency, 100 hertz, and the second harmonic, 200 hertz. The periodicity is nevertheless clear when compared to the full-spectrum waveform on top.

A harmonic sound is said to have a **missing fundamental**, **suppressed fundamental**, or **phantom fundamental** when its overtones suggest a fundamental frequency but the sound lacks a component at the fundamental frequency itself. The brain perceives the pitch of a tone not only by its fundamental frequency, but also by the periodicity implied by the relationship between the higher harmonics; we may perceive the same pitch (perhaps with a different timbre) even if the fundamental frequency is missing from a tone.

For example, when a note (that is not a pure tone) has a pitch of 100 Hz, it will consist of frequency components that are integer multiples of that value (e.g. 100, 200, 300, 400, 500.... Hz). However, smaller loudspeakers may not produce low frequencies, and so in our example, the 100 Hz component may be missing. Nevertheless, a pitch corresponding to the fundamental may still be heard

Summary and additional remarks

1. To estimate the trend of a decomposition model with regression, a polynomial in time may be used. The degree of the polynomial can be chosen to be one more than the number of turns observed in the time series plot of the response. Test the highest order polynomial term with the t test. If the test shows significance, retain the estimated polynomial. If the result is not significant, refit with the degree lowered. Retain all the lower order terms in the polynomial—don't remove components which are not significant.

2. Polynomial estimation of the trend often fails to capture trend structure adequately. It is not wise to fit polynomials of very high degree. High-degree polynomials may track the trend well, but they usually fail to forecast future observations reasonably.

3. There are alternatives to polynomial estimation of trend. One such option is the `decompose` function in R, which we will encounter in the next set of notes.

4. It is common in time series studies to analyze the differenced data. The differencing operation usually removes the trend. This allows one to focus attention on estimation of the seasonal structure. The differencing operation does alter the seasonal and irregular components of a decomposition model, but estimates of the original seasonal structure can be recovered.

5. For estimation of static seasonal structure in a decomposition model with monthly observations and an annual cycle, monthly dummy variables may be used. If a categorical variable labelling the months is converted to a factor variable, R will calculate the dummies for a regression analysis. This procedure may also be used with quarterly data and an annual cycle, and with daily data and a weekly cycle. Further, one may construct an alternative set of monthly dummy variables to provide a direct estimation of seasonal indices.

6. Trigonometric terms provide an alternative to the use of monthly (or quarterly, or daily) dummies in a regression setting. The cosine curve $R \cos(\lambda t + \alpha)$ has *amplitude* R, *frequency* λ (in radians per unit time; with $\lambda = 2\pi f$, f is the frequency in cycles per unit time), and *phase angle* α. The *period* is $2\pi/\lambda$, giving the length of an individual cycle. The cosine curve satisfies

$$R \cos(\lambda t + \alpha) = A \cos \lambda t + B \sin \lambda t.$$

That is, it can be represented as a linear combination of cosine and sine curves with the same frequency and with phase angle equal to 0.

7. To track a periodic seasonal structure in an additive decomposition model, it is convenient to use cosines and sines as explanatory variables in a regression model. For a monthly time series with a periodic annual seasonal pattern one employs

$$\cos\frac{2\pi j}{12}t, \quad \sin\frac{2\pi j}{12}t, \quad j = 1,...,5; \quad \cos\frac{2\pi 6}{12}t = (-1)^t$$

as dummy independent variables. The frequencies are $j/12$, $j = 1,...,6$, and the corresponding periods are 12, 6, 4, 3, 2.4, and 2 months. There are 11 dummies for the 12 values of the categorical variable *month* if the regression includes an intercept.

With quarterly data and a periodic annual seasonal pattern, one uses the variables $\cos\frac{2\pi}{4}t, \quad \sin\frac{2\pi}{4}t; \quad \cos\frac{2\pi 2}{4}t = (-1)^t$. These dummies have frequencies $j/4$, $j = 1, 2$, with periods 4 and 2 quarters. Thus, there are 3 dummies for the 4 values of the categorical variable *quarter* if the regression includes an intercept. And with daily data and a weekly seasonal pattern, the variables are $\cos\frac{2\pi j}{7}t, \quad \sin\frac{2\pi j}{7}t, \quad j = 1, 2, 3,$ with frequencies $j/7$, $j = 1, 2, 3$, and periods 7, 3.5, and 2.33 days.