Journal of Official Statistics
Vol. 6. No. 1, 1990, pp. 3–73
© Statistics Sweden

STL: A Seasonal-Trend Decomposition Procedure Based on Loess

Robert B. Cleveland,[1] William S. Cleveland,[2] Jean E. McRae,[2] and Irma Terpenning[2]

Abstract: STL is a filtering procedure for decomposing a time series into trend, seasonal, and remainder components. STL has a simple design that consists of a sequence of applications of the loess smoother; the simplicity allows analysis of the properties of the procedure and allows fast computation, even for very long time series and large amounts of trend and seasonal smoothing. Other features of STL are specification of amounts of seasonal and trend smoothing that range, in a nearly continuous way, from a very small amount of smoothing to a very large amount; robust estimates of the trend and seasonal components that are not distorted by aberrant behavior in the data; specification of the period of the seasonal component to any integer multiple of the time sampling interval greater than one; and the ability to decompose time series with missing values.

Key words: Seasonal adjustment; time series; loess.

1. Introduction

STL is a filtering procedure for decomposing a seasonal time series into three components: trend, seasonal, and remainder. Figure 1 shows an example. The data, graphed in the first (top) panel, are daily average measurements of atmospheric carbon dioxide (CO_2) made at the Mauna Loa Observatory in Hawaii (Komhyr and Harris 1977). The second panel graphs a trend component: the low frequency variation in the data together with nonstationary, long-term changes in level. The third panel graphs a seasonal component: variation in the data at or near the seasonal frequency, which in this case is one cycle per year. The remainder component, shown in the fourth panel, is the remaining variation in the data beyond that in the seasonal and trend components. That is, suppose the data, the trend component, the seasonal component, and the remainder component are denoted by Y_v, T_v, S_v, and R_v, respectively, for $v = 1$ to N. Then

$$Y_v = T_v + S_v + R_v.$$

The measurements in Figure 1 were made by the U.S. National Oceanic and Atmospheric Administration, or NOAA, as part of a worldwide government program to monitor CO_2 concentrations. The measurements

[1] University of Michigan, Ann Arbor, MI 48109, U.S.A.
[2] AT&T Bell Laboratories, 600 Mountain Avenue, Murrey Hill, N.J. 07974, U.S.A.

Acknowledgements: We are very grateful to Trevor Hastie for discussions that led to the material in Section 5.

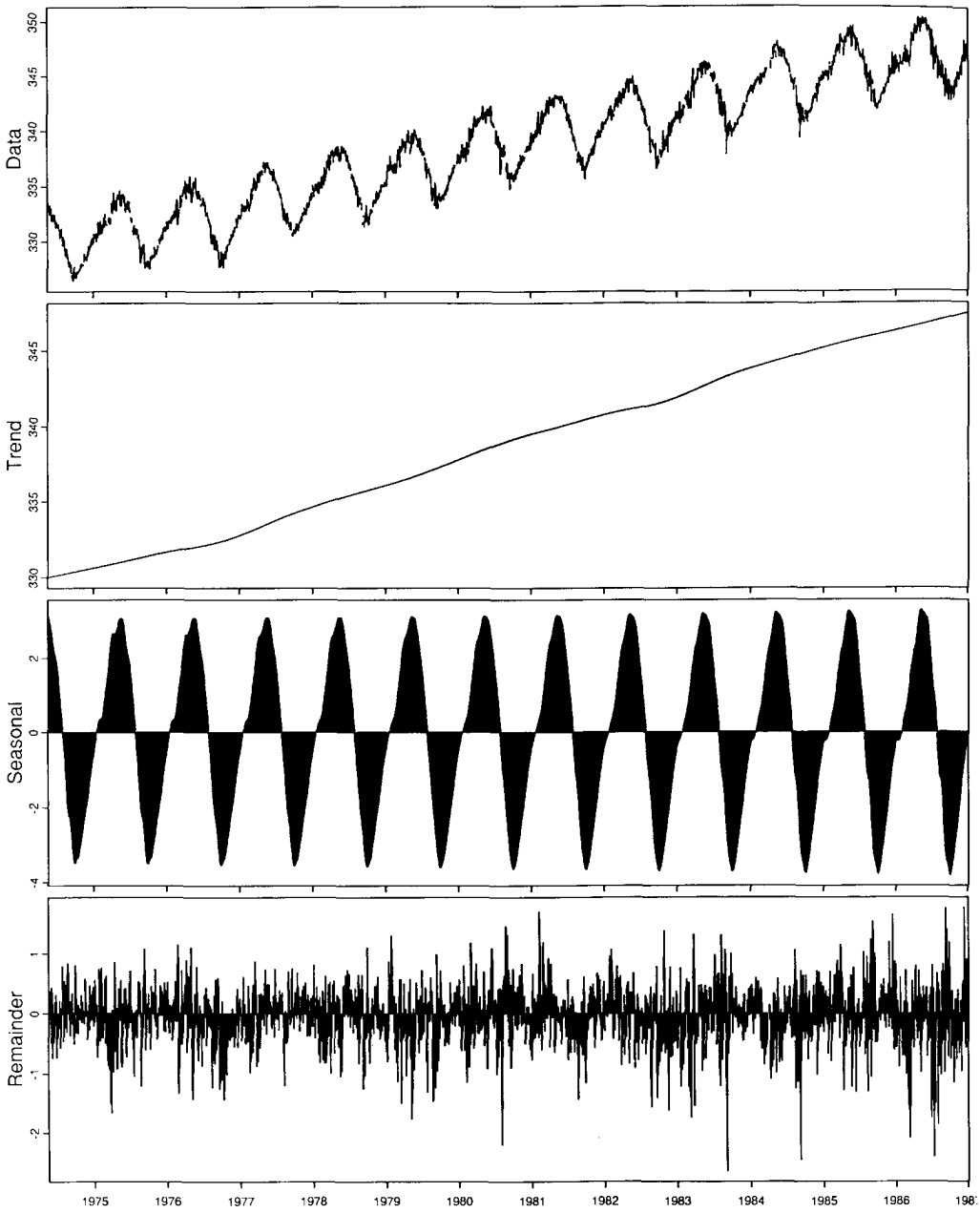

Fig. 1. Decomposition Plot of Daily Carbon Dioxide Data. The units on the vertical scales are ppm.

span the period April 17, 1974 to December 31, 1986. We deleted all occurrences of February 29, as if that day did not exist, to keep the period equal to 365 days; thus the data, with these days deleted, span 4609 days. Data are missing for 416 of these days, so altogether there are 4193 CO_2 measurements.

1.1. Design goals

Our goal for STL was to develop a decomposition procedure and a companion computer implementation that satisfy the following interdependent criteria:

1. Simple design and straightforward use.
2. Flexibility in specifying the amounts of variation in the trend and seasonal components.
3. Specification of the number of observations per cycle of the seasonal component to any integer greater than 1.
4. The ability to decompose series with missing values.
5. Robust trend and seasonal components that are not distorted by transient, aberrant behavior in the data.
6. Easy computer implementation and fast computation, even for long time series.

STL consists of a sequence of smoothing operations each of which, with one exception, employs the same smoother: locally-weighted regression, or loess (Cleveland and Grosse 1990; Cleveland and Devlin 1988; Cleveland, Devlin, and Grosse 1988). In Section 2, loess is described and then the operations that make up STL are given.

STL has several parameters that must be chosen by the data analyst. Section 3 discusses how to choose them. For some parameters, a priori prescribed values can be used. The choices of the others must be based on the properties of the data; diagnostic methods are given that help the data analyst to make these choices.

Computation is a critical issue. To achieve the widest possible applicability, computer routines that implement a seasonal-trend decomposition procedure must run fast, even for long time series such as that in Figure 1, and should have a simple, modular structure. Implementation of STL is discussed in Section 4.

The design of STL and the choices of parameters in practice are based on an understanding of which part of the variation in a time series becomes the seasonal component and which part becomes the trend component. This understanding comes from eigenvalue and frequency response analyses in Section 5.

Section 6 is a general discussion of the following topics: a summary of the choice of the STL parameters in practice; a review of the salient features of STL; two examples (multiplicative decomposition and trading-day components) that illustrate how the design of STL leads to easy modification to achieve other goals; combining STL with a time series model to get confidence intervals for components; a comparison with X-11 (Shiskin, Young, and Musgrave 1967); and information on how to acquire public-domain Fortran routines that implement STL.

2. The Definition of STL

In this section we will describe the loess smoother and the STL operations. Our goal is to give a straightforward account of the details; the justification for various aspects is given in later sections.

2.1. Loess

Suppose x_i and y_i for $i = 1$ to n are measurements of an independent and dependent variable, respectively. The loess regression curve, $\hat{g}(x)$, is a smoothing of y given x that can be computed for any value x along the scale of the independent variable. That is, loess is defined everywhere and not just at the x_i; as we shall see, this is an important feature that in STL will allow us to deal with missing values and detrend the seasonal component in a straightforward way. Actually, loess can be used to smooth y as a

function of any number of independent variables, but for STL, only the case of one independent variable is needed.

$\hat{g}(x)$ is computed in the following way. A positive integer, q, is chosen. For the moment suppose $q \leqslant n$. The q values of the x_i that are closest to x are selected and each is given a *neighborhood weight* based on its distance from x. Let $\lambda_q(x)$ be the distance of the qth farthest x_i from x. Let W be the tricube weight function:

$$W(u) = \begin{cases} (1 - u^3)^3 & \text{for } 0 \leqslant u < 1 \\ 0 & \text{for } u \geqslant 1. \end{cases}$$

The neighborhood weight for any x_i is

$$v_i(x) = W\left(\frac{|x_i - x|}{\lambda_q(x)}\right).$$

Thus the x_i close to x have the largest weights; the weights decrease as the x_i increase in distance from x and become zero at the qth farthest point. The next step is to fit a polynomial of degree d to the data with weight $v_i(x)$ at (x_i, y_i). The value of the locally-fitted polynomial at x is $\hat{g}(x)$. In this paper we will use $d = 1$ or 2; that is, the fitting is *locally-linear* or *locally-quadratic*.

Now suppose that $q > n$. $\lambda_n(x)$ is the distance from x to the farthest x_i. For $q > n$ we define $\lambda_q(x)$ by

$$\lambda_q(x) = \lambda_n(x)\frac{q}{n}.$$

Then we proceed as before in the definition of the neighborhood weights using this value of $\lambda_q(x)$.

To use loess, d and q must, of course, be chosen. The choices, in the context of STL, will be discussed in detail in this section and in Section 3. As q increases, $\hat{g}(x)$ becomes smoother. As q tends to infinity, the $v_i(x)$ tend to one and $\hat{g}(x)$ tends to an ordinary least-squares polynomial fit of degree d. Taking $d = 1$ is reasonable if the underlying pattern in the data has gentle curvature. But if the pattern has substantial curvature, for example, peaks and valleys, then $d = 2$ is a better choice.

Suppose each observation (x_i, y_i) has a weight ρ_i that expresses the reliability of the observation relative to the others. For example, if the y_i had variances $\sigma^2 k_i$ where the k_i were known, then ρ_i might be $1/k_i$. We can incorporate these weights into the loess smoothing in a straightforward way by using $\rho_i v_i(x)$ as the weights in the local least-squares fitting. As we will see, this provides a mechanism by which we can easily build robustness into STL.

2.2. *The overall design: inner and outer loops*

STL consists of two recursive procedures: an inner loop nested inside an outer loop. In each of the passes through the inner loop, the seasonal and trend components are updated once; each complete run of the inner loop consists of $n_{(i)}$ such passes. Each pass of the outer loop consists of the inner loop followed by a computation of robustness weights; these weights are used in the next run of the inner loop to reduce the influence of transient, aberrant behavior on the trend and seasonal components. An initial pass of the outer loop is carried out with all robustness weights equal to 1, and then $n_{(o)}$ passes of the outer loop are carried out. The choices of $n_{(i)}$ and $n_{(o)}$ are discussed in Section 3. For the decomposition in Figure 1, $n_{(i)} = 1$ and $n_{(o)} = 10$.

Suppose the number of observations in each period, or cycle, of the seasonal component is $n_{(p)}$. For example, if the series is monthly with a yearly periodicity, then $n_{(p)} = 12$. We need to be able to refer to the subseries of values at each position of the

seasonal cycle. For example, for a monthly series with $n_{(p)} = 12$, the first subseries is the January values, the second is the February values, and so forth. We will refer to each of these $n_{(p)}$ subseries as *cycle-subseries*.

2.3 *The inner loop*

Each pass of the inner loop consists of a seasonal smoothing that updates the seasonal component, followed by a trend smoothing that updates the trend component. Suppose $S_v^{(k)}$ and $T_v^{(k)}$ for $v = 1$ to N are the seasonal and trend components at the end of the kth pass; these two components are defined at all times $v = 1$ to N, even at times where Y_v is missing. The updates of the $(k + 1)$st step, $S_v^{(k+1)}$ and $T_v^{(k+1)}$, are computed in the following way.

Step 1: *Detrending.* A detrended series, $Y_v - T_v^{(k)}$, is computed. If Y_v is missing at a particular time position, then the detrended series is also missing at that position.

Step 2: *Cycle-subseries Smoothing.* Each cycle-subseries of the detrended series is smoothed by loess with $q = n_{(s)}$ and $d = 1$. Smoothed values are computed at all time positions of the cycle-subseries, including those with missing values, and at the positions just prior to the first time position of the subseries and just after the last. For example, suppose the series is monthly, $n_{(p)} = 12$, and the January cycle-subseries ranges from January 1943 to January 1985 with a missing value at January 1960; then the smoothed values are computed at all positions from January 1942 to January 1986. The collection of smoothed values for all of the cycle-subseries is a temporary seasonal series, $C_v^{(k+1)}$, consisting of $N + 2n_{(p)}$ values that range from $v = -n_{(p)} + 1$ to $N + n_{(p)}$. For the decomposition shown in Figure 1, $n_{(s)} = 35$. The choice of $n_{(s)}$ will be discussed in Sections 3 and 5.

Step 3: *Low-Pass Filtering of Smoothed Cycle-Subseries.* A low-pass filter is applied to $C_v^{(k+1)}$. The filter consists of a moving average of length $n_{(p)}$, followed by another moving average of length $n_{(p)}$, followed by a moving average of length 3, followed by a loess smoothing with $d = 1$ and $q = n_{(l)}$. The choice of $n_{(l)}$ will be discussed in Sections 3 and 5. For the decomposition in Figure 1, $n_{(l)} = 365$. The output, $L_v^{(k+1)}$, is defined at time positions $v = 1$ to N because the three moving averages cannot extend to the ends; $n_{(p)}$ positions are lost at each end. The seasonal smoothing in Step 2 was extended $n_{(p)}$ positions at each end in anticipation of this loss.

Step 4: *Detrending of Smoothed Cycle-Subseries.* The seasonal component from the $(k + 1)$st loop is $S_v^{(k+1)} = C_v^{(k+1)} - L_v^{(k+1)}$ for $v = 1$ to N. $L_v^{(k+1)}$ is subtracted to prevent low-frequency power from entering the seasonal component.

Step 5: *Deseasonalizing.* A deseasonalized series $Y_v - S_v^{(k+1)}$ is computed. If Y_v is missing at a particular time position, then the deseasonalized series is also missing.

Step 6: *Trend Smoothing.* The deseasonalized series is smoothed by loess with $q = n_{(t)}$ and $d = 1$. Smoothed values are computed at all time positions $v = 1$ to N, even those with missing values. The trend component from the $(k + 1)$st loop, $T_v^{(k+1)}$ for $v = 1$ to N, is this set of smoothed values. For the decomposition in Figure 1, $n_{(t)} = 573$. The choice of $n_{(t)}$ will be discussed in Section 3 and 5.

Thus the seasonal-smoothing portion of the inner loop is Steps 2, 3, and 4, and the trend-smoothing portion is Step 6.

To carry out Step 1 on the initial pass through the inner loop we need starting values, $T_v^{(0)}$, for the trend component. Using $T_v^{(0)} \equiv 0$ works quite well. The trend becomes part of the smoothed cycle-

subseries, $C_v^{(1)}$, but is largely removed in Step 4, the detrending.

2.4. *The outer loop*

Suppose we have carried out an initial run of the inner loop to get estimates, T_v and S_v, of the trend and seasonal components. Then the remainder is

$$R_v = Y_v - T_v - S_v.$$

(Note that the remainder, unlike T_v and S_v, is not defined where Y_v has missing values.) We will define a weight for each time point where Y_v is observed. These *robustness weights* reflect how extreme R_v is. An outlier in the data that results in a very large $|R_v|$ will have a small or zero weight. Let

$$h = 6 \text{ median}(|R_v|).$$

Then the robustness weight at time point v is

$$\rho_v = B(|R_v|/h)$$

where B is the bisquare weight function:

$$B(u) = \begin{cases} (1 - u^2)^2 & \text{for } 0 \leqslant u < 1 \\ 0 & \text{for } u > 1. \end{cases}$$

Now the inner loop is repeated, but in the smoothings of Steps 2 and 6, the neighborhood weight for a value at time v is multiplied by the robustness weight, ρ_v. This is just a use of the reliability weights discussed in Section 2.1. These robustness iterations of the outer loop are carried out a total of $n_{(o)}$ times. Each time we enter the inner loop after the initial pass we do not set $T_v^{(0)} \equiv 0$ as we did on the initial pass, but rather use the trend component from Step 6 of the previous inner loop.

2.5. *Post-smoothing of the seasonal*

Consider the daily CO_2 series in Figure 1. Step 2 of STL, the basic one in the formation of the seasonal component, has the following form: the 365 cycle-subseries of the detrended series are smoothed separately and then are put together to form the temporary seasonal component. This means that each cycle-subseries of the seasonal will be smooth across years. For example, the values of the seasonal component for July 4 change smoothly from one year to the next. But the smoothing does not guarantee that the seasonal component from one day to the next will be smooth. Such smoothness is not imposed because there are many time series for which it is inappropriate. The top panel of Figure 2 shows the seasonal component of the daily CO_2 series that results from STL; there is clearly local roughness. But for this series we want a component that is smooth from one day to the next.

A simple solution to a locally-rough seasonal is a post-smoothing in which the seasonal component from STL is smoothed by loess. The smoothed values are the final seasonal component. In carrying out this smoothing we want to be sure to use locally-quadratic fitting because there are many peaks and valleys in the seasonal component. Also, there is no need for robustness iterations in loess since STL produces a well-behaved component apart from the local roughness. Finally, q can typically be small or moderate since the roughness will typically have a small variance. For the daily CO_2 data, q was taken to be 51 for the post-smoothing. The resulting seasonal component is shown in the bottom panel of Figure 2; this is the component shown in Figure 1.

3. Choosing the STL Parameters

STL has 6 parameters:

$n_{(p)}$ = the number of observations

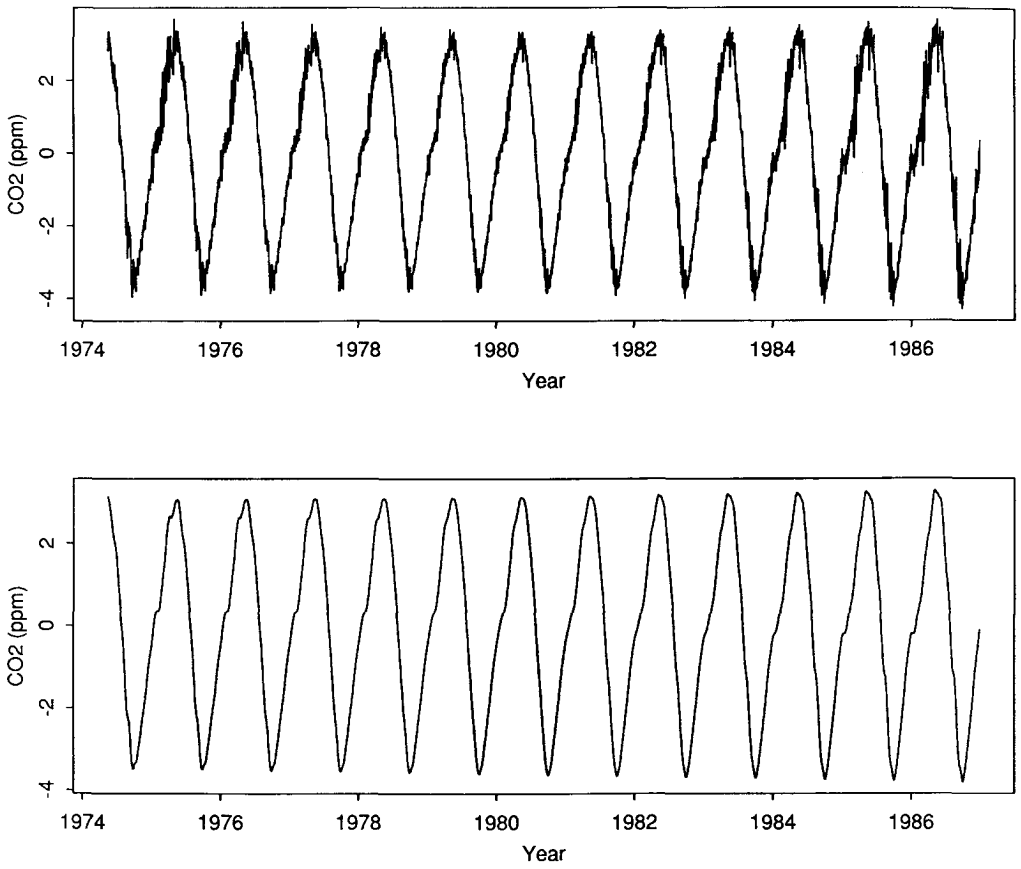

Fig. 2. (Top Panel) Seasonal Component of Daily Carbon Dioxide. (Bottom Panel) Seasonal Component After Post-Smoothing. The units on the vertical scales are ppm.

in each cycle of the seasonal component,

$n_{(i)}$ = the number of passes through the inner loop,

$n_{(o)}$ = the number of robustness iterations of the outer loop,

$n_{(l)}$ = the smoothing parameter for the low-pass filter,

$n_{(t)}$ = the smoothing parameter for the trend component,

and

$n_{(s)}$ = the smoothing parameter for the seasonal component.

Choosing the first five is straightforward. The last parameter, $n_{(s)}$, however, must be carefully tailored to each application; we provide some diagnostic methods to help do this. In this section we discuss the choices.

3.1. Scripps CO_2 and unemployed males

We will use two time series as examples in this section. The first is monthly averages of measurements of atmospheric CO_2 made at the Mauna Loa Observatory in Hawaii by the Scripps Institute of Oceanography (Keeling, Bacastow, and Whorf 1982). The *decomposition plot* in Figure 3 shows the data and three components. The time frame of the data ranges from January 1959 to

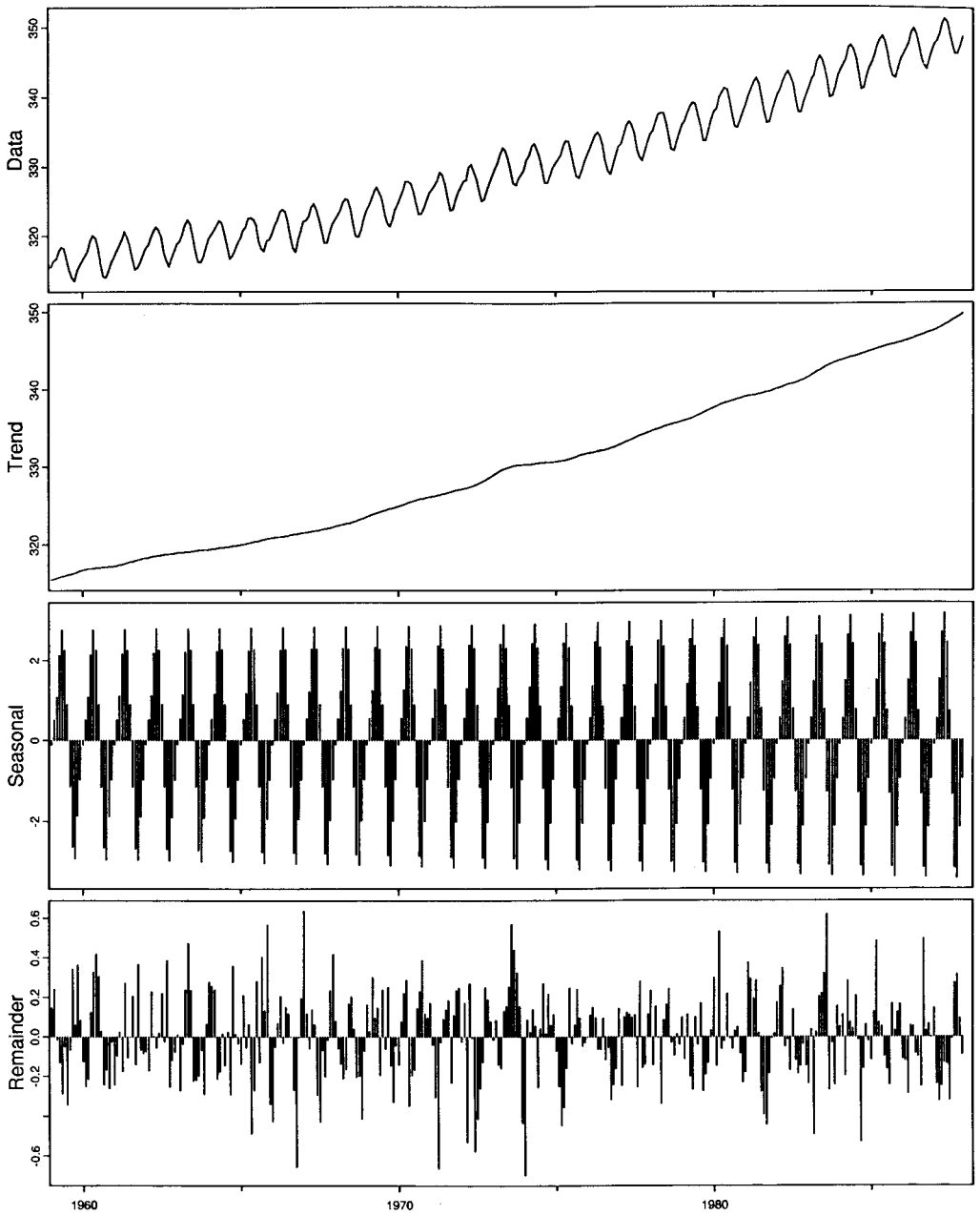

Fig. 3. Decomposition Plot of Monthly Carbon Dioxide Data. The units on the vertical scales are ppm.

December 1987; this was all of the data available from our source, the Carbon Dioxide Information Analysis Center of the Oak Ridge National Laboratory, at the time our analysis was carried out. There is a yearly periodicity so $n_{(p)} = 12$. The other parameters of the decomposition are $n_{(i)} = 2$, $n_{(o)} = 0$, $n_{(l)} = 13$, $n_{(t)} = 19$, and

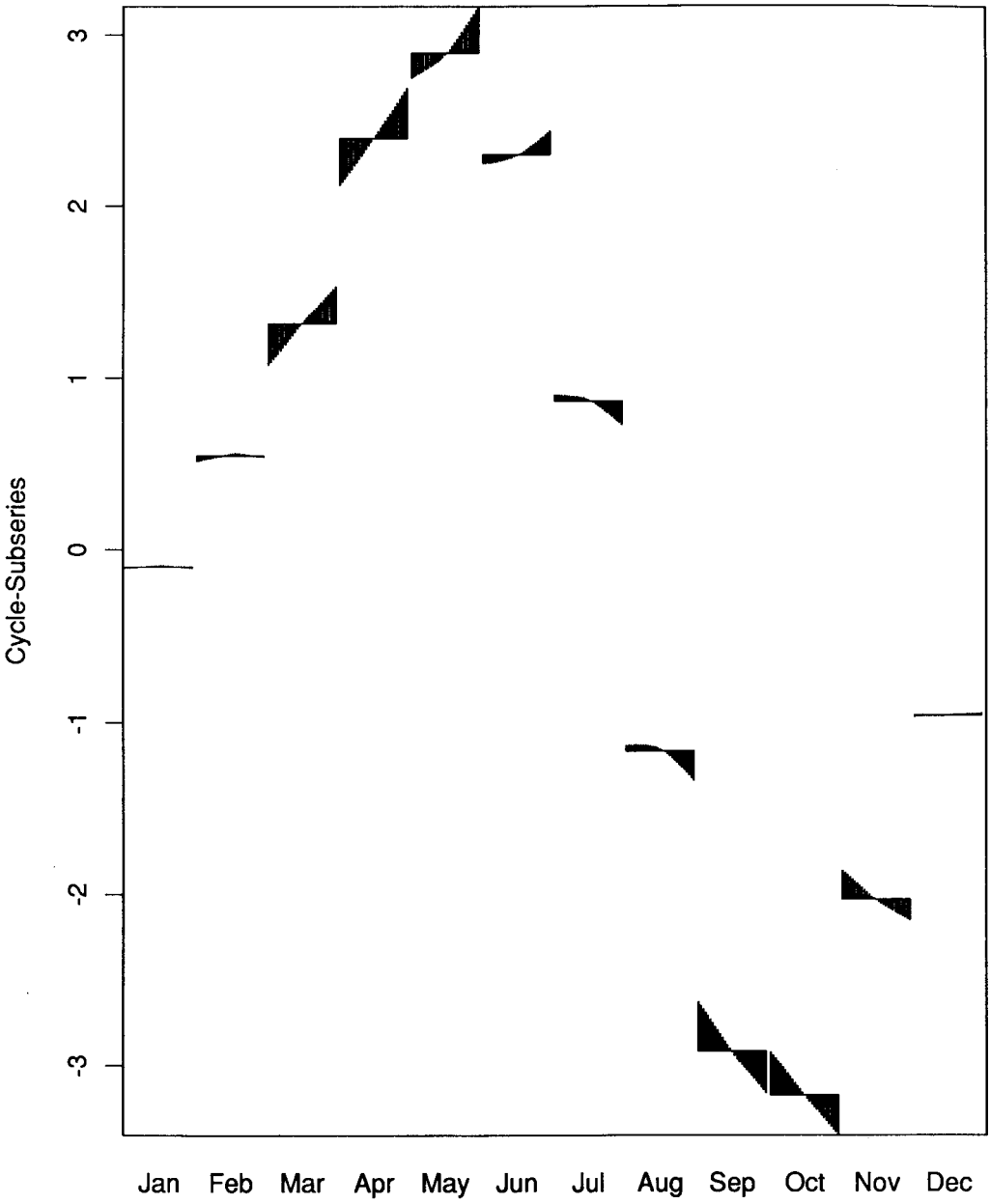

Fig. 4. Cycle-Subseries Plot for Monthly Carbon Dioxide Data. The units on the vertical scale are ppm.

$n_{(s)} = 35$. Figure 4 is a *cycle-subseries plot* of the seasonal component. Each cycle-subseries is graphed separately against time. First the January values are graphed, then the February values are graphed, and so forth. The midmean of the values is portrayed by the horizontal line and the values themselves are portrayed by the ends of the vertical lines emanating from the horizontal line.

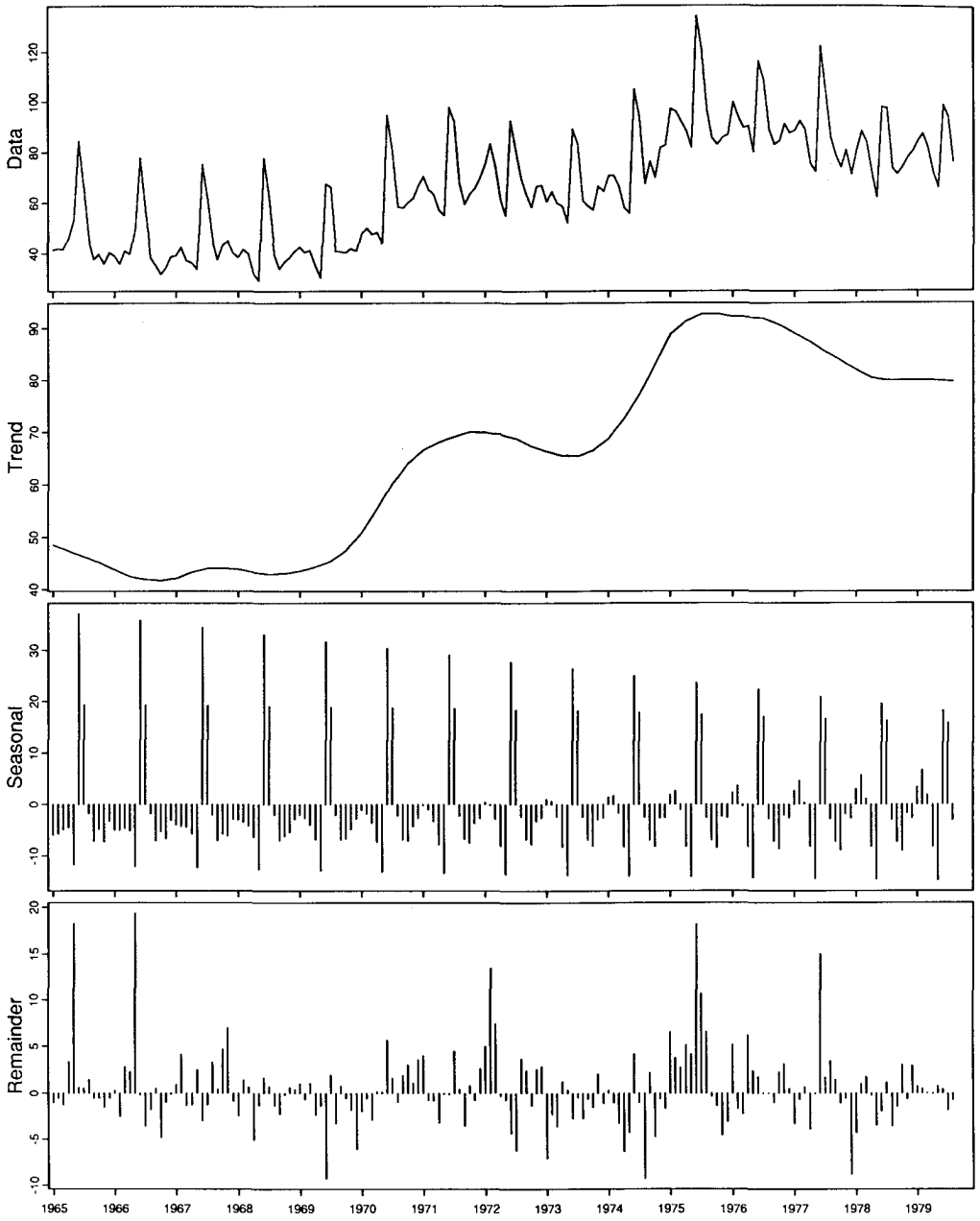

Fig. 5. Decomposition Plot of U.S. Unemployed Males Ages 16–19. The units on the vertical scales are tens of thousands.

The second time series is UM16, the number of unemployed males aged 16 to 19 in the U.S. for each month from January 1965 to August 1979. We used this time frame so that readers can compare our discussion with earlier discussions of these data by Hillmer and Tiao (1982), Hillmer (1985), and Carlin and Dempster (1989). A decom-

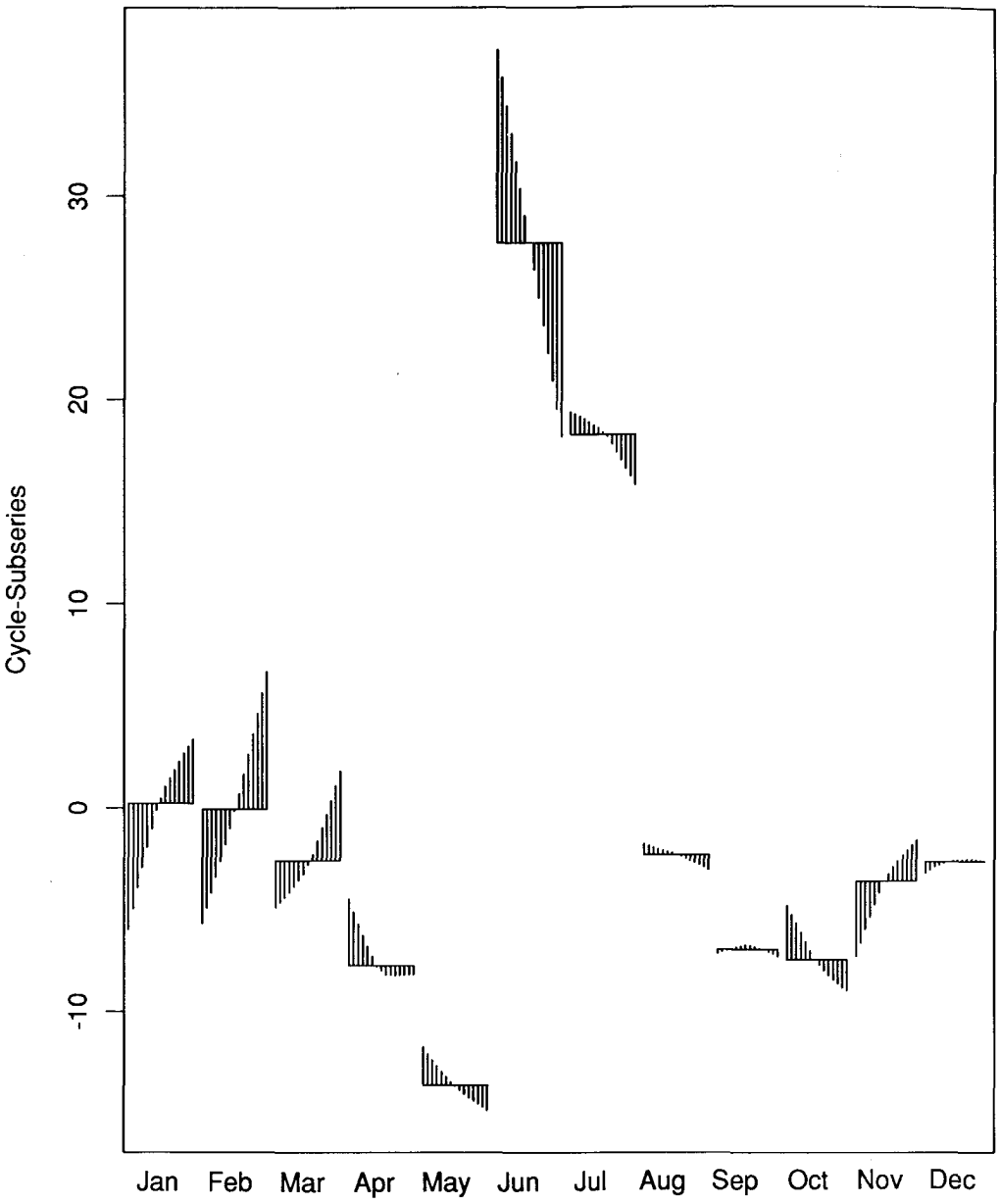

Fig. 6. Cycle-Subseries Plot for U.S. Unemployed Males Ages 16–19. The units on the vertical scale are tens of thousands.

position plot is shown in Figure 5 and a cycle-subseries plot is shown in Figure 6. There is a yearly periodicity so $n_{(p)} = 12$. The other parameters of the decomposition are $n_{(i)} = 1$, $n_{(o)} = 5$, $n_{(l)} = 13$, $n_{(t)} = 21$, and $n_{(s)} = 17$.

3.2. $n_{(p)}$, the number of observations per seasonal cycle

This parameter arises in an obvious way from the application. For example, for the two CO_2 series presented earlier, there is a

yearly periodicity so $n_{(p)} = 365$ for the daily data and $n_{(p)} = 12$ for the monthly data. It is entirely possible that a time series can have two or more periodic components; for example, a series measured daily might have weekly and yearly periodicities. In such a case one can use STL to successively estimate the components by proceeding from the shortest-period component to the longest-period component, estimating each component, subtracting it out, and estimating the next component from the residuals.

3.3. $n_{(i)}$, the number of passes of the inner loop and $n_{(o)}$, the number of robustness iterations

The STL robust estimation is needed when prior knowledge of the data or diagnostic checking indicates that non-Gaussian behavior in the time-series leads to extreme, transient variation. Otherwise we can omit the robustness iterations and set $n_{(o)} = 0$; in this case, there is no outer loop and STL consists of the inner loop. The monthly CO_2 data in Figure 3 show no aberrant behavior, so $n_{(o)}$ was taken to be 0. For the unemployed-males data in Figure 5, however, the first two May values are outliers, so the robust STL was used; we will study this aberrant behavior later.

First, suppose we need no robustness. We want to choose $n_{(i)}$ large enough so that the updating of the trend and seasonal components converges. But for reasons that will be given in Section 4, the convergence is very fast. In many cases, $n_{(i)} = 1$ is sufficient, but we recommend $n_{(i)} = 2$ to provide near certainty of convergence.

Suppose now that we need robustness iterations. We want to choose $n_{(o)}$ large enough so that the robust estimates of the trend and seasonal components converge. In doing this, there are two reasons for always taking $n_{(i)} = 1$. The first is the reason given above – convergence of the inner loop is very rapid. The second has to do with a general principle of unconstrained optimization when there are nested iterations; it does not pay to excessively refine an inner loop to get overall convergence. With $n_{(i)} = 1$, we have found that $n_{(o)} = 5$ is a very safe value and that $n_{(o)} = 10$ provides near certainty of convergence. For the unemployed-males data, $n_{(o)}$ was taken to be 5 and convergence had occurred by the 5th iteration. However, for the daily CO_2 data in Figure 1, convergence was slower and 10 iterations were required.

Of course, one could develop a convergence criterion and stop the iterations when the criterion is satisfied. In our investigations we used the following criterion to judge convergence: Suppose $U_v^{(k)}$ and $U_v^{(k+1)}$ are successive iterates of either a trend or seasonal component, then $U_v^{(k)}$ was judged to have been a converged component if

$$\frac{\max_v |U_v^{(k)} - U_v^{(k+1)}|}{\max_v U_v^{(k)} - \min_v U_v^{(k)}} < 0.01.$$

3.4. $n_{(l)}$, the smoothing parameter of the low-pass filter

For reasons given in Section 5, $n_{(l)}$ always can be taken to be equal to the least odd integer greater than or equal to $n_{(p)}$. This choice of $n_{(l)}$, which contributes to achieving the goal of preventing the trend and seasonal components from competing for the same variation in the data, is used in all decompositions in this paper. For the decomposition in Figure 1, $n_{(l)} = 365$; and in Figures 3 and 5, $n_{(l)} = 13$.

3.5. $n_{(s)}$, the seasonal smoothing parameter

As $n_{(s)}$ increases, each cycle-subseries becomes

smoother. We will always take $n_{(s)}$ to be odd. For reasons given in Section 5, we also want $n_{(s)}$ to be at least 7.

The choice of $n_{(s)}$ determines the variation in the data that makes up the seasonal component; the choice of the appropriate variation depends critically on the characteristics of the series. It should be emphasized that there is an intrinsic ambiguity in the definition of seasonal variation. The data analyst defines the seasonal variation in choosing the seasonal smoothing parameter. We will describe a diagnostic method that can assist the data analyst in choosing a definition; but these methods do not always lead to a unique choice, and in many applications the final decision must be based on knowledge about the mechanism generating the series and the goals of the analysis. The ambiguity is true of all seasonal decomposition procedures, not just STL. A lucid discussion of this point is given by Carlin and Dempster (1989).

Figure 7 illustrates a diagnostic graphical method that can help in the choice of $n_{(s)}$. Each panel of Figure 7 graphs two sets of values for a particular month. Let $\bar{s}_k$ be the mean of the values of the cycle-subseries of the seasonal component for the k-th month. The curve on the panel for the k-th month graphs those seasonal values minus their mean $\bar{s}_k$. The circles graph the values of the k-th cycle-subseries of the seasonal plus the remainder, also with $\bar{s}_k$ subtracted. (The reason for subtracting $\bar{s}_k$ is to center the values on each panel at zero; note that the vertical scales of all panels are the same so that we can graphically compare the variation of values on different panels.) This diagnostic method, which we will call a *seasonal-diagnostic plot*, helps us decide how much of the variation in the data other than trend should go into the seasonal component and how much into the remainder.

The values graphed in Figure 7 are from the decomposition of the monthly CO_2 series in Figure 3 where $n_{(s)} = 35$. Figure 8 is the seasonal-diagnostic display for a decomposition of the same series with $n_{(s)}$ decreased to 11. Each cycle-subseries of the seasonal component is now much less smooth. The additional variation in these seasonal values, compared with the seasonal values for $n_{(s)} = 35$, appears to be noise and not meaningful seasonal variation because the cycle in the CO_2 series is caused mainly by the seasonal cycle of foliage in the Northern Hemisphere, and one would expect a smooth evolution of this cycle over years.

Figure 9 is the seasonal-diagnostic display for the robust decomposition of the unemployed-males series shown in Figure 5 where $n_{(s)} = 17$. The cycle-subseries of the seasonal component appear to follow the patterns in the values of the seasonal plus remainder without introducing undue noise. Note that the values of the seasonal component for May are nearly linear and follow the pattern of the majority of the values of $S_v + R_v$ without being distorted by the initial two outliers. This is a result of the robust estimation. Without the robustness, the seasonal component is distorted. This is illustrated by the seasonal-diagnostic display in Figure 10 for a decomposition that uses no robustness iterations; the parameters are $n_{(i)} = 2$, $n_{(o)} = 0$, $n_{(l)} = 13$, $n_{(t)} = 21$, and $n_{(s)} = 17$. The resulting cycle-subseries of the seasonal component are similar to those for the other decomposition except for the months of May and June. For May, the outliers distort the seasonal values; these values neither account for the outliers nor follow the pattern of the remaining values. (*N.B.* We have taken the position that the aberrant May behavior is a transient noise phenomenon. If those with more detailed knowledge of the mechanism generating the

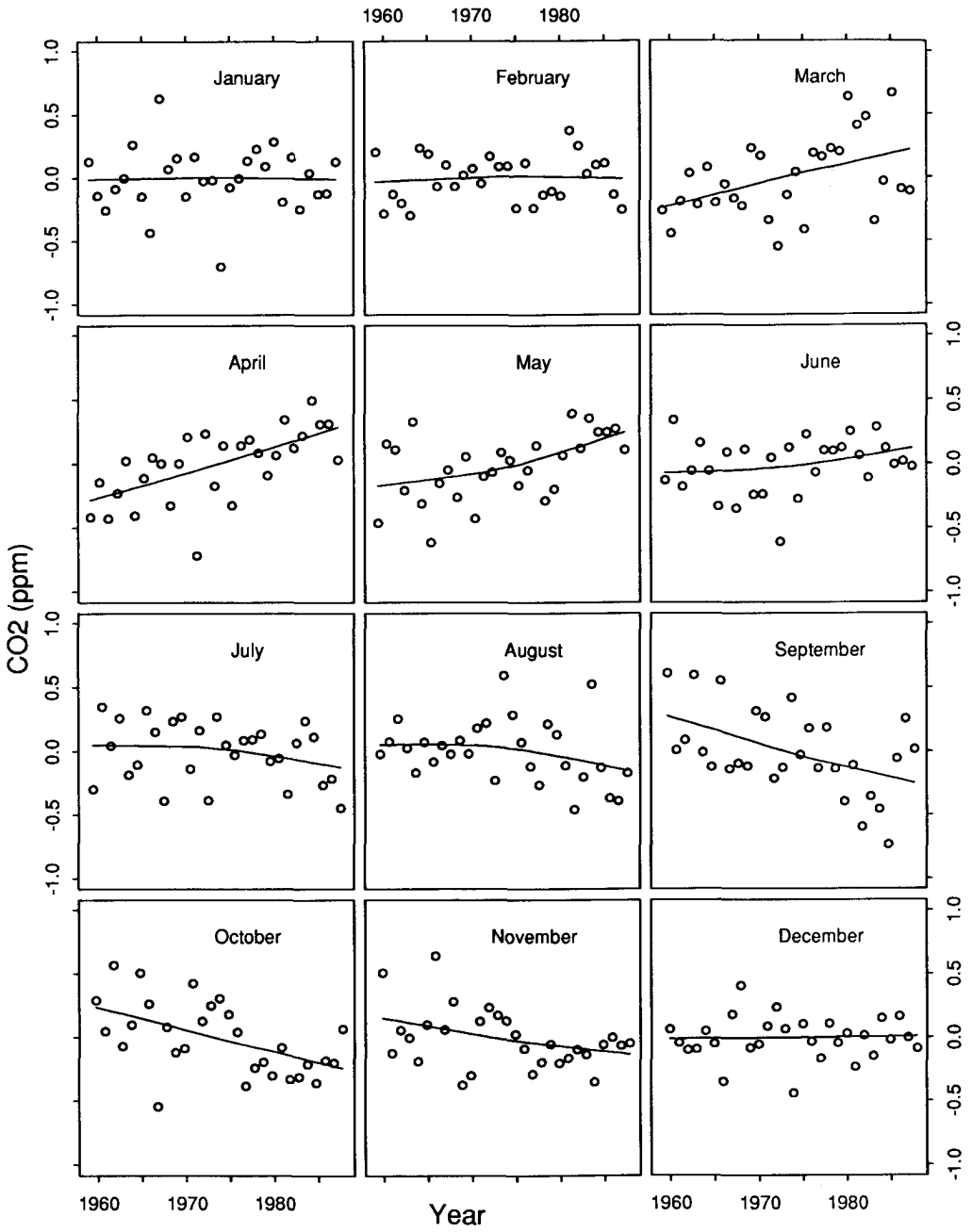

Fig. 7. Seasonal-Diagnostic Plot for Monthly Carbon Dioxide Data with the Seasonal Smoothing Parameter Equal to 35. The units on the vertical scales are ppm.

data had a convincing argument that the behavior is a rapidly evolving seasonal component, then we would reduce $n_{(s)}$ to account for it in the seasonal.)

3.6. The trend smoothing parameter, $n_{(t)}$

As $n_{(t)}$ increases, the trend component, T_v, extracts less variation from X_v and

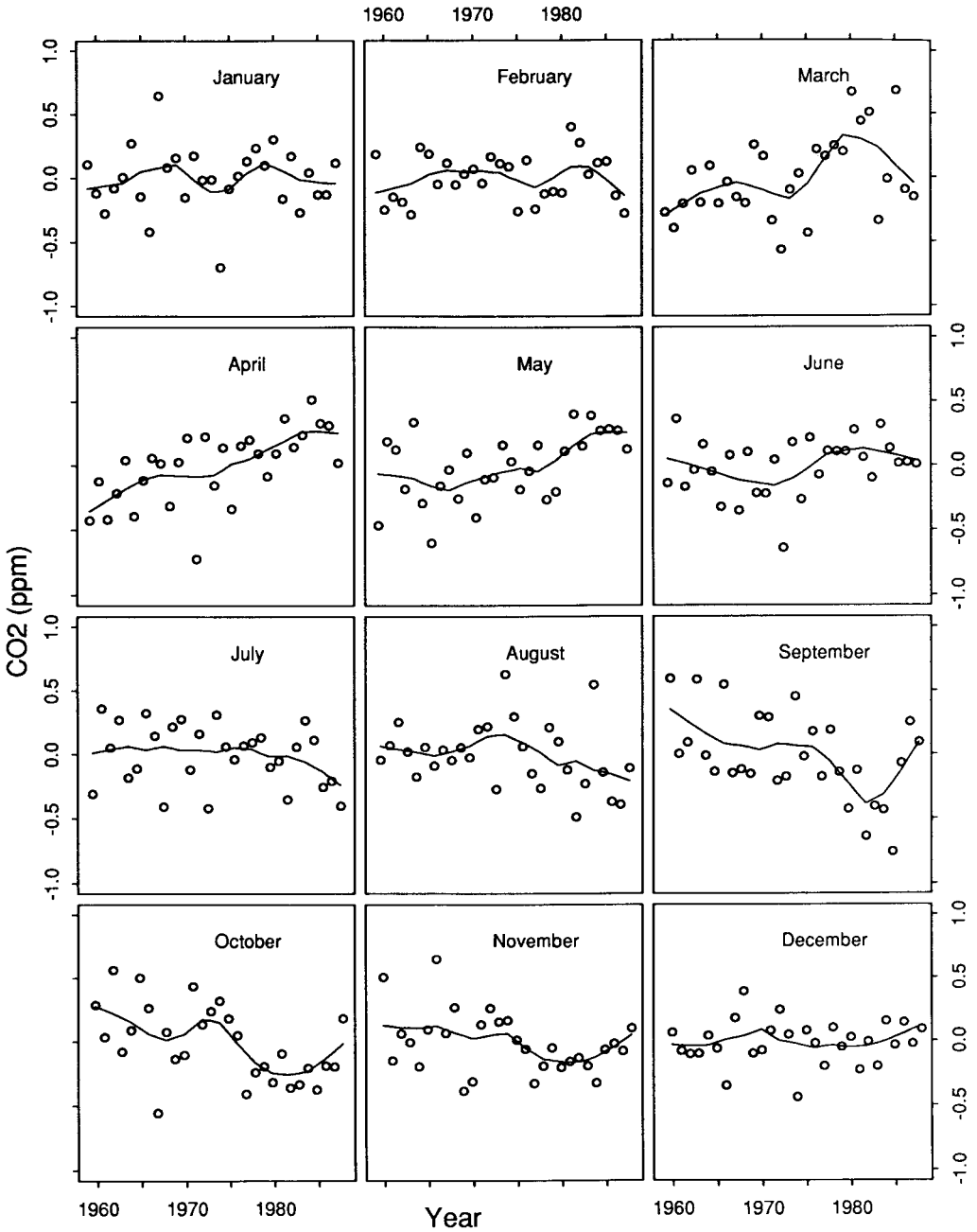

Fig. 8. Seasonal-Diagnostic Plot for Monthly Carbon Dioxide Data with the Seasonal Smoothing Parameter Equal to 11. The units on the vertical scales are ppm.

becomes smoother. We will always take $n_{(t)}$ to be odd.

We recommend the following approach to the trend component. Consider it to be a component whose estimation is needed to form an estimate of the seasonal; in other words, regard the primary goal of STL to be the estimation of the seasonal component. If

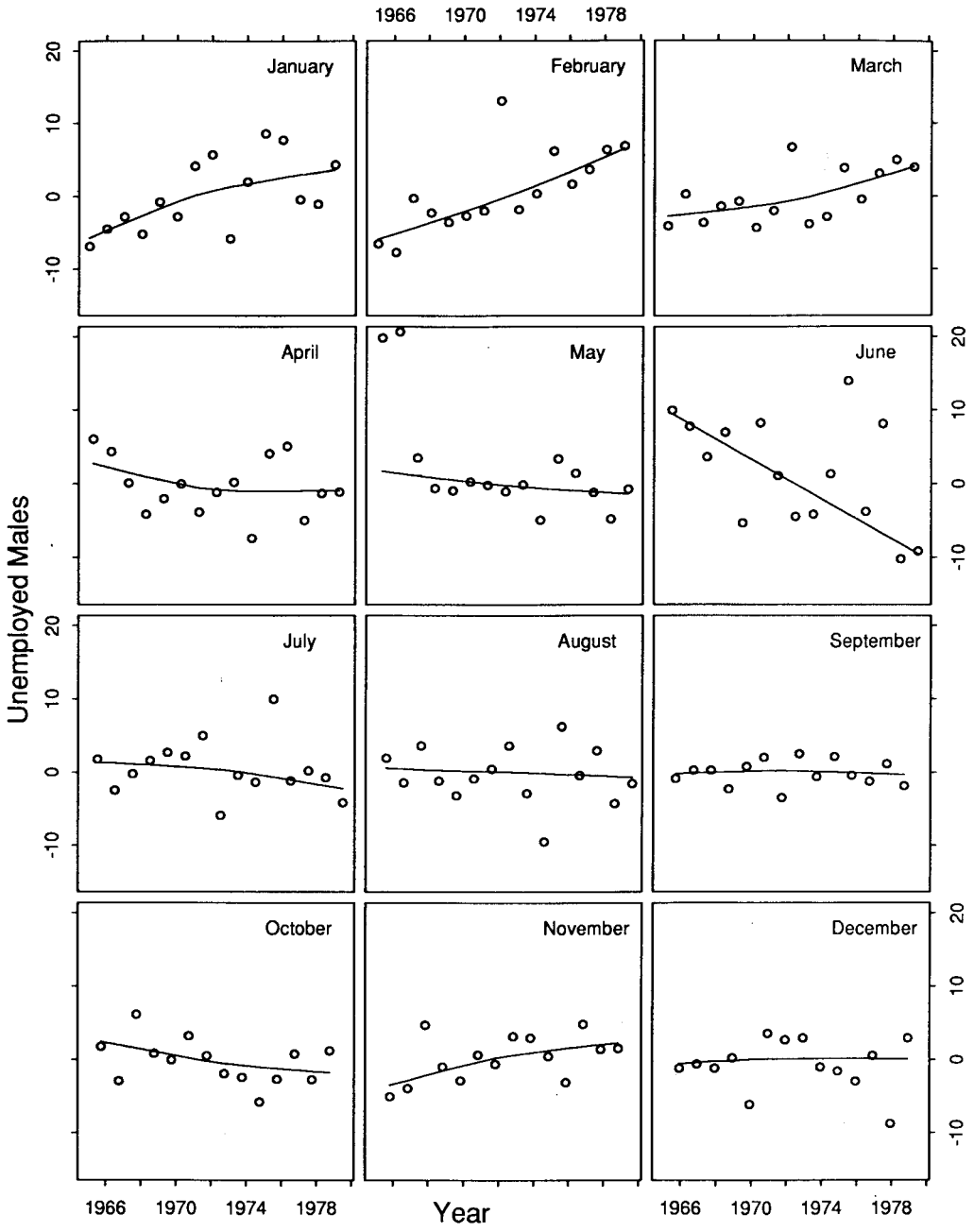

Fig. 9. Seasonal-Diagnostic Plot for the Unemployed-Males Data with the Number of Robustness Iterations Equal to 5. The units on the vertical scales are tens of thousands.

a component is needed that describes certain low-frequency variation in the data, then we can carry out a *post-trend smoothing*. This means that a low pass filter, such as loess, is applied to $T_v + R_v$, the data with the seasonal component removed, to get a component with the desired variation. As we will see, we often are forced to do this since our

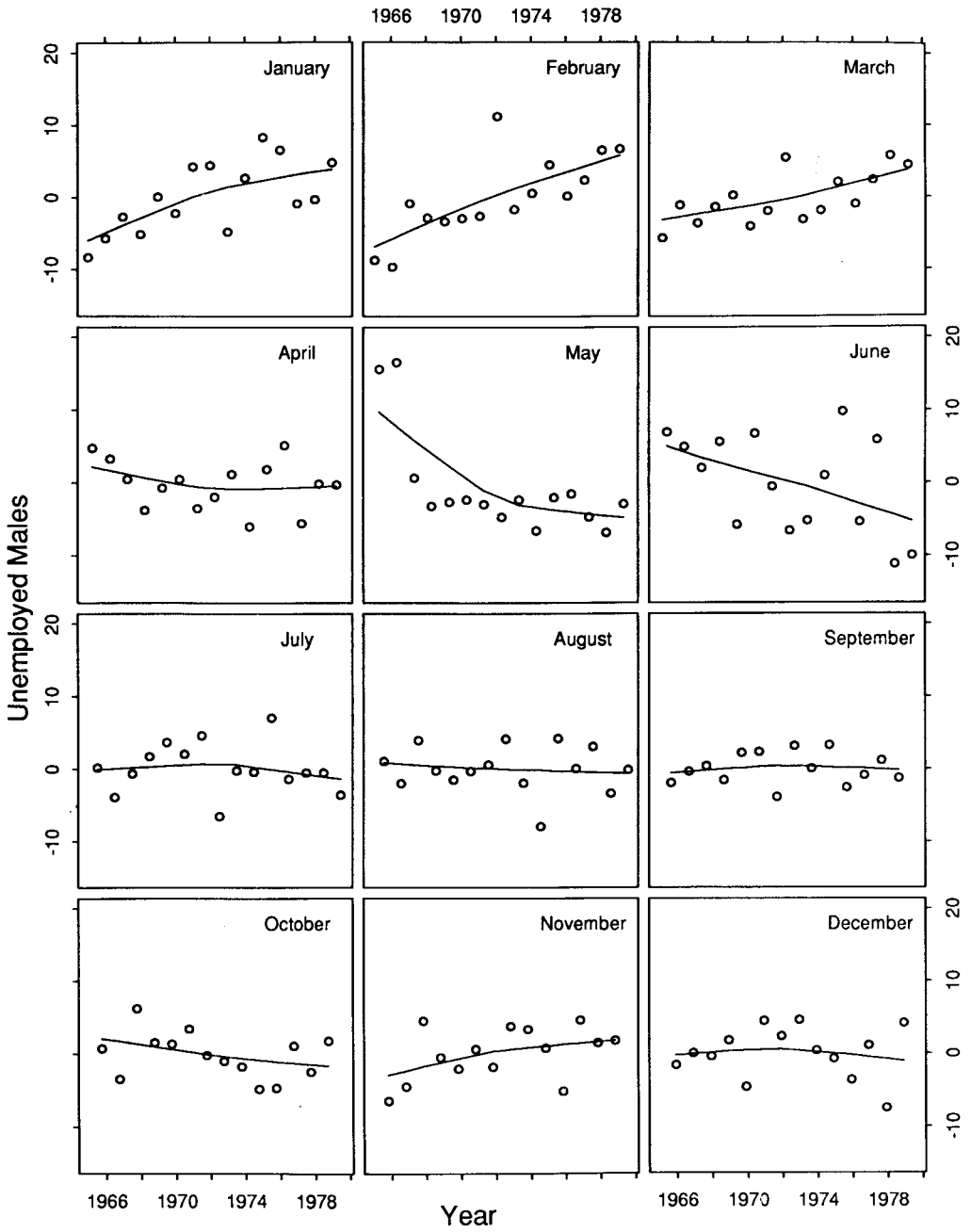

Fig. 10. Seasonal-Diagnostic Plot for the Unemployed-Males Data with No Robustness Iterations. The units on the vertical scales are tens of thousands.

choice of $n_{(t)}$ often is restricted by the needs of the decomposition and cannot necessarily be chosen so that the trend component describes a certain prescribed component of variation in the data.

There are two roles that the trend com-

ponent plays in helping to estimate the seasonal component. One is to remove persistent, long-term variation, such as the persistent increase in the CO_2 concentrations in Figures 1 and 3. If we did not remove such behavior, it would distort the seasonal component. (The presence of such behavior is what prevents us from simply applying an ordinary digital filter to the data that passes in bands centered at the fundamental seasonal frequency and its harmonics.) This role is achieved unless $n_{(t)}$ gets so large that the smoother misses even persistent effects.

The trend component also plays a role in the robustness iterations. The robustness weights, whose purpose is to decrease the influence of aberrant behavior, are based on the magnitudes of the remainder component. Values of the remainder that are large in absolute value are given reduced weight. We do not want to allow major low-frequency effects to go into the remainder because we want to give reduced weight only to extreme, transient behavior and not to peaks and troughs of major, slow oscillations. Thus, for this purpose, we want $n_{(t)}$ to be small.

But we cannot allow $n_{(t)}$ to become too small. Having chosen $n_{(s)}$, and thus the variation that should go into the seasonal component, we do not want $n_{(t)}$ so small that some of this variation winds up in the trend component. In other words, we do not want the trend and seasonal components to compete for variation in the data. In Section 5 we show that to do this we need to choose $n_{(t)}$ so that

$$n_{(t)} \geq \frac{1.5n_{(p)}}{1 - 1.5n_{(s)}^{-1}}.$$

(Since $n_{(s)} \geq 7$, $n_{(t)}$ ranges from about $1.5n_{(p)}$ to $2n_{(p)}$.) Thus, all of the above goals for the trend component are satisfied if we take $n_{(t)}$ to be the smallest odd integer satisfying the above inequality. This value of $n_{(t)}$ was used for the three examples in this paper. For the monthly CO_2 series, $n_{(p)} = 12$ and $n_{(s)}$ was chosen to be 35; the right side of the inequality is 18.8, so $n_{(t)} = 19$. For the daily CO_2 series $n_{(p)} = 365$ and $n_{(s)} = 35$; the right side in this case is 572.0, so $n_{(t)} = 573$. For the unemployed-males series, $n_{(p)} = 12$ and $n_{(s)} = 17$; the right side in this case is 19.7, so $n_{(t)} = 21$.

To assess the trend component that results from the choice of $n_{(t)}$ it is helpful to make a *trend-diagnostic plot* as shown in Figure 11 for the decomposition of the unemployed-males data. The circles in the top panel graph the trend plus remainder, $T_v + R_v$, and the superposed curve graphs T_v. The bottom panel graphs the remainder, R_v; thus the bottom panel is a graph of the residuals from the curve in the top panel. The May outliers appear as two large positive values in the remainder.

4. Computational Methods

There is a general principle about the computation of a loess smoothing that allows fast computation. Because it is smooth, $\hat{g}(x)$ does not need to be computed exactly at all values of x where we need it, but rather can be computed exactly at a sufficiently dense set of points and interpolated everywhere else. The implementation of this general principle of computing loess can vary substantially and depends on the setting in which loess is applied.

We implemented STL in two ways and each employed this principle of fast computation of loess, but the details of the two cases were quite different. One implementation was in Fortran and the other was in the S environment for graphics and data analysis (Becker, Chambers, and Wilks 1988). In the Fortran implementation, there

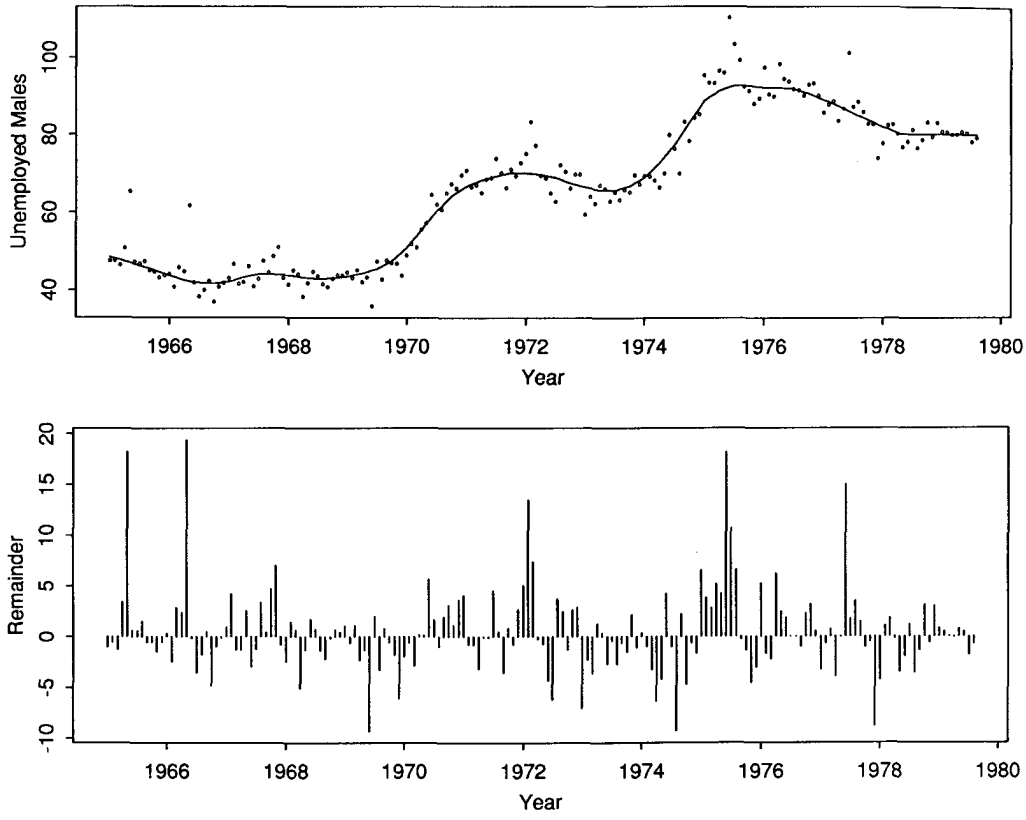

Fig. 11. Trend-Diagnostic Plot for the Unemployed-Males Data. The units on the vertical scales are tens of thousands.

are three computation parameters: $n_{(l)}^{(jump)}$, $n_{(t)}^{(jump)}$, and $n_{(s)}^{(jump)}$. The trend smoothing is carried out at positions 1, $1 + n_{(t)}^{(jump)}$, $1 + 2n_{(t)}^{(jump)}$, and so forth, and at position N. The trend component at other positions is computed by linear interpolation. A similar procedure with the parameter $n_{(s)}^{(jump)}$ is used for each loess smoothing in the cycle-subseries smoothing, and with the parameter $n_{(l)}^{(jump)}$ for the loess smoothing in the low-pass filter. We have found that taking $n_{(t)}^{(jump)}$ to be the smallest integer greater than or equal to $n_{(t)}/10$ or even $n_{(t)}/5$ works quite well. Similar statements hold for $n_{(l)}^{(jump)}$ and $n_{(s)}^{(jump)}$. In the S implementation, each loess smoothing is carried out by a general-purpose loess routine where the choice of positions at which smoothed values are computed exactly is determined by an algorithm that uses $k - d$ trees, and interpolation is carried out by blending functions; the details of this procedure are described by Cleveland, Devlin, and Grosse (1988).

We carried out an analysis of the computation time of the Fortran implementation. The Fortran routines were machine-generated from programs written in Ratfor, a Fortran preprocessor (Kernighan 1975). No missing values are allowed because even greater speed can be achieved. (The S implementation mentioned above allows for missing values.) The following function of the STL parameters provides a reasonable approximation to the run time:

$$n_{(i)}(n_{(o)} + 1)N\left[\frac{h(N, n_{(t)})}{n_{(t)}^{(jump)}} + \frac{h(N/n_{(p)}, n_{(s)})}{n_{(s)}^{(jump)}} + \frac{h(N, n_{(l)})}{n_{(l)}^{(jump)}}\right]$$

where

$$h(k, l) = c_1 + c_2 \min(k, l).$$

The constants c_1 and c_2 depend on the machine on which STL is run. For a VAX 8550 with run time measured in milliseconds, $c_1 = 0.0635$ and $c_2 = 0.0289$. For example, consider the decomposition of the monthly CO_2 data in Section 3 where $N = 348$, $n_{(p)} = 12$, $n_{(i)} = 2$, $n_{(o)} = 0$, $n_{(l)} = 13$, $n_{(t)} = 19$, and $n_{(s)} = 35$. For $n_{(t)}^{(jump)} = 2$, $n_{(s)}^{(jump)} = 4$, and $n_{(l)}^{(jump)} = 2$, the VAX run time from the formula is 0.52 sec. (The actual run time is 0.65 sec.) For the unemployed-males decomposition, we have $N = 176$, $n_{(p)} = 12$, $n_{(i)} = 1$, $n_{(o)} = 5$, $n_{(l)} = 13$, $n_{(t)} = 21$, and $n_{(s)} = 17$. For $n_{(t)}^{(jump)} = 3$, $n_{(s)}^{(jump)} = 2$, and $n_{(l)}^{(jump)} = 2$, the VAX run time from the formula is 0.73 sec. (The actual run time is 0.88 sec.) The formula, as in these two examples, underestimates the actual run time, but its accuracy is quite sufficient for our purpose: showing, roughly, the change in run time as we change the parameters. For example, we can see that if $n_{(t)}^{(jump)}$, $n_{(t)}^{(jump)}$, and $n_{(s)}^{(jump)}$ are all multiplied by a factor d, then the run time is approximately divided by d. The formula also shows that the run time is proportional to $n_{(i)}(n_{(o)} + 1)$, the total number of passes through the inner loop.

5. Eigenvalue Analysis

The guidance given in the previous sections about the choices of the smoothing parameters $n_{(s)}$, $n_{(t)}$, and $n_{(l)}$ depends critically on the analysis in this section, which consists of a study of the eigenvalues and frequency response functions of the linear filtering operations of the inner loop (with no robustness weights). Using the eigenvalues and frequency response functions we will apply results of Buja, Hastie, and Tibshirani (1989) to determine properties of the seasonal and trend components and the dependence of these properties on the parameters.

5.1. *Operator matrices*

Each smoothing step of the STL inner loop consists of an operation in which an input time series, w_v, is linearly filtered to produce an output time series, x_v; that is,

$$x_v = \sum_u d_{vu} w_u.$$

Thus, if w and x are vectors whose uth elements are w_u and x_u, respectively, and D is a matrix whose (v, u)th element is d_{vu}, then

$$x = Dw.$$

We will refer to D as the *operator matrix* of the filter.

Let I be the $N \times N$ identity matrix, let C be the $(N + 2n_{(p)}) \times N$ operator matrix of the cycle-subseries smoothing in Step 2, let L be the $N \times (N + 2n_{(p)})$ operator matrix of the low-pass filter in Step 3, and let P be an $N \times (N + 2n_{(p)})$ matrix whose first $n_{(p)}$ columns consist of zeros, whose next N rows are the identity matrix, I, and whose last $n_{(p)}$ columns consist of zeros. Then the operator matrix for the seasonal smoothing, which consists of Steps 2 to 4, is

$$S = (P - L)C.$$

Let T be the $N \times N$ matrix of the trend smoothing in Step 6. After the first pass of the inner loop, the seasonal component is

$$SY$$

and the trend component is

$$T(I - S)Y.$$

Thus the operator matrix for the trend component from the first pass is

$$T_1 = T(I - S) = T - TS$$

and the operator matrix for the seasonal component from the first pass is

$$S_1 = S.$$

Let T_k and S_k be the operator matrices for the kth pass, where k is any positive integer, and let T_0 be the zero operator matrix, that is, with all elements zero. Then

$$S_k = S(I - T_{k-1})$$

and

$$T_k = T(I - S_k).$$

For example,

$$S_2 = S(I - T_1)$$

$$= S(I - T(I - S))$$

$$= S - ST + STS$$

and

$$T_2 = T(I - S_2)$$

$$= T - TS + TST - TSTS.$$

Suppose for each positive integer m,

$$A_m = \begin{cases} (ST)^{m/2} & \text{for } m \text{ even} \\ (ST)^{(m-1)/2}S & \text{for } m \text{ odd} \end{cases}$$

and

$$B_m = \begin{cases} (TS)^{m/2} & \text{for } m \text{ even} \\ (TS)^{(m-1)/2}T & \text{for } m \text{ odd,} \end{cases}$$

then

$$S_k = \sum_{m=1}^{2k-1} (-1)^{m-1} A_m$$

and

$$T_k = \sum_{m=1}^{2k} (-1)^{m-1} B_m.$$

5.2. *Eigenvalues and frequency response functions*

To simplify the ensuing analysis we will suppose that the time series being decomposed, Y_v, is circular in the sense that Y_v is defined for all integer v and $Y_v = Y_u$ if $v = u \pmod N$. While this is a fiction, it nevertheless provides a good approximation if $n_{(t)}$ and $n_{(l)}$ are small compared with N and if $n_{(s)}$ is small compared with $N/n_{(p)}$; in other words, in this analysis, we ignore end effects. Also, for notational convenience, we shall suppose that N is even and equal to $2M$ where M is a positive integer.

One result of the circularity assumption is that the operation associated with any operator matrix, D, defined in Section 5.1 is a stationary, symmetric, linear filter. That is, if w_v is the input series of the operation and x_v is the output series of the operation, then

$$x_v = \sum_{j=-r}^{r} d_j w_{v-j}$$

where $d_j = d_{-j}$, and D is an $n \times n$ circulant matrix (Grenander and Szego 1958). For example, for the trend smoothing,

$$d_j = \frac{W(j/r)}{\sum_{i=-r}^{r} W(i/r)}$$

where $r = (n_{(t)} - 1)/2$ and W is the tricube weight function.

Let

$$D^*(f) = \sum_{j=-r}^{r} d_j \cos 2\pi jf$$

be the Fourier transform of the filter coefficients d_j at frequency f. Let $f_k = k/N$, for $k = 0$ to M, be the Fourier frequencies.

Since D is a circulant matrix, its eigenvalues are

$$d_k^* = D^*(f_k) \quad \text{for} \quad k = 0 \quad \text{to} \quad M,$$

where the multiplicity of d_k^* for $k > 0$ is 2 and the multiplicity of d_0^* is 1. For $k > 1$, the cosinusoids at frequency f_k, $\sin(2\pi v f_k)$ and $\cos(2\pi v f_k)$ for $v = 1$ to N, are eigenvectors with eigenvalue d_k^*; the cosine at frequency $f_0 = 0$, $\cos 2\pi v f_0 = 1$ for $v = 1$ to N, is an eigenvector with eigenvalue d_0^*.

Following the previous convention, the Fourier transforms of the STL operators T, C, L, and S will be denoted by T^*, C^*, L^*, and S^*. Since $S = (I - L)C$, we have

$$S^*(f) = [1 - L^*(f)]C^*(f).$$

Furthermore, it is easy to see that for $n_{(s)} = n_{(t)}$, T^* and C^* are related by

$$C^*(f) = T^*(n_{(p)}f).$$

L is a low-pass filter so $H = I - L$ is a high-pass filter. Let H^* be the Fourier transform of H.

Figure 12 shows four *frequency response functions* - $T^{*2}(f)$, $C^{*2}(f)$, $H^{*2}(f)$, and $S^{*2}(f)$ - for the parameter values $n_{(p)} = 12$, $n_{(t)} = 23$, $n_{(s)} = 7$, and $n_{(l)} = 13$.

5.3. *Design goal*

Let t_k^* and s_k^* for $k = 1$ to M be the eigenvalues of T and S, respectively. Since $|T^*(f)| \leqslant 1$, we have $|t_k^*| \leqslant 1$. However, we can have $|s_k^*| > 1$ because $|H^*(f)|$ can be greater than one; but if $n_{(s)}$, $n_{(t)}$, and $n_{(l)}$ are chosen to satisfy the criteria we will set out later, then $|S^*(f)| \leqslant 1$. Thus, we shall suppose that $|s_k^*| \leqslant 1$.

We will now apply general results of Buja, Hastie, and Tibshirani (1989) on methods of iterative fitting such as the inner loop of STL. Because $|s_k^*| \leqslant 1$ and $|t_k^*| \leqslant 1$, the updates of the trend and seasonal components in each pass of the inner loop converge to final trend and seasonal components. Let $\tilde{S}$ be the operator matrix of the final seasonal component and let $\tilde{T}$ be the operator matrix of the final trend component. $\tilde{S}$ and $\tilde{T}$ are also circulant matrices and thus have the above cosinusoids as eigenvectors. Let $\tilde{s}_k^*$ be the eigenvalue of $\tilde{S}$ that corresponds to the cosinusoids of frequency f_k, and let $\tilde{t}_k^*$ be defined similarly for $\tilde{T}$. If $|s_k^*| = 1$, then $|\tilde{s}_k^*| = 1$ and $\tilde{t}_k^* = 0$, so all of the variation at frequency f_k in the series being decomposed goes into the final seasonal component and none into the trend. This would be true even if $|t_k^*| = 1$ because the seasonal operator S is applied first in the initial pass; but, in fact, our parameter selection will prevent t_k^* and s_k^* from both being 1. If $|t_k^*| = 1$ and $|s_k^*| < 1$, then $\tilde{t}_k^* = 1$ and $\tilde{s}_k^* = 0$; thus all variation at frequency f_k goes into the final trend component. If $|t_k^*| < 1$ and $|s_k^*| < 1$, then

$$\tilde{s}_k^* = \frac{s_k^*(1 - t_k^*)}{1 - t_k^* s_k^*}$$

and

$$\tilde{t}_k^* = \frac{t_k^*(1 - s_k^*)}{1 - t_k^* s_k^*};$$

thus variation at frequency f_k goes into both final components, and the relative amounts are determined by the relative values of s_k^* and t_k^*.

As we stated in Section 3, we do not want the trend and seasonal components competing for variation in the data. In view of the above results, we avoid competition if we choose $n_{(s)}$, $n_{(t)}$, and $n_{(l)}$ to satisfy the following eigenvalue criterion: $|s_k^*|$ and $|t_k^*|$ should not both be nonnegligible.

Note that if we satisfy the criterion, we have $\tilde{s}_k^* \approx s_k^*$ and $\tilde{t}_k^* \approx t_k$. Thus $\tilde{S} \approx S_1 \approx S$, and $\tilde{T} \approx T_1 \approx T$, which means we get very rapid convergence, essentially after

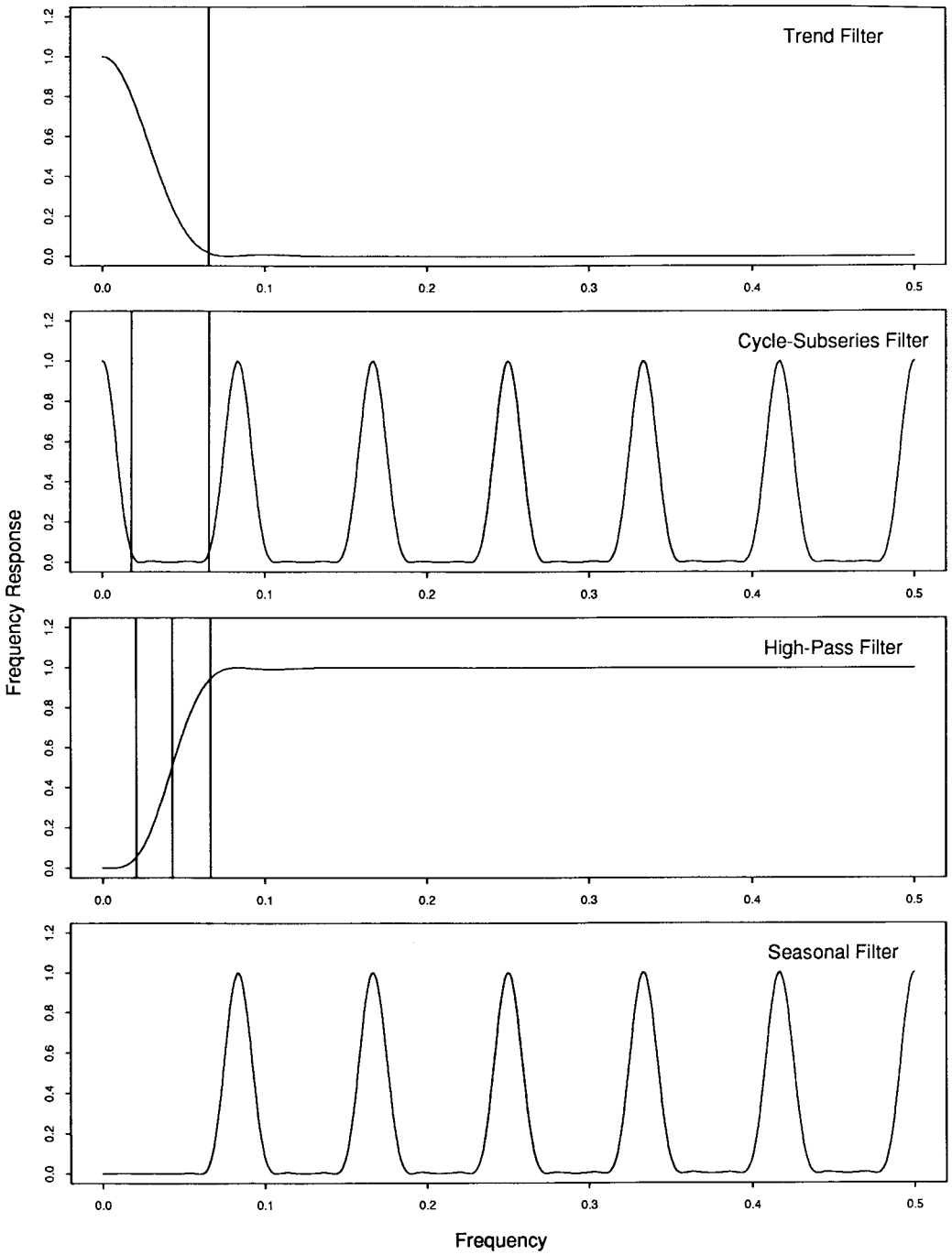

Fig. 12. Four Frequency Response Functions.

the first pass. This explains the rapid convergence we have observed in the inner loop and provides support for our recommendations about the choice of $n_{(i)}$. End effects and persistent, long-term components often prevent convergence on the very first pass in actual applications, but two passes almost always are sufficient.

5.4. *The choices of $n_{(s)}$, $n_{(p)}$, and $n_{(l)}$*

To achieve the eigenvalue criterion set out in Section 5.3 we will consider the four frequency response functions T^{*2}, C^{*2}, H^{*2}, and S^{*2}. We want to choose $n_{(s)}$, $n_{(t)}$, and $n_{(l)}$ so that if $S^{*2}(f)$ is not small, then $T^{*2}(f)$ is small, and if $T^{*2}(f)$ is not small, then $S^{*2}(f)$ is small. Notice that this criterion is satisfied by the parameter choices for the example of Figure 12.

The operator T is, in frequency terms, a low-pass filter; that is, it passes only cosinusoids with low frequencies. We will define the *critical frequency* for T to be the smallest frequency for which $T^{*2}(f) = 0.05$. Cosinusoids above this frequency are reduced to nearly zero. Since $T^{*2}(f)$ depends on $n_{(t)}$, so does the critical frequency, and a good approximation of the frequency is $f_{(t)} = 1.5n_{(t)}^{-1}$. In Figure 12, $f_{(t)}$ is shown by the vertical line on the graph of $T^{*2}(f)$. We will also define critical frequencies for $C^{*2}(f)$. As the second panel of Figure 12 illustrates, $C^{*2}(f)$ has peaks at $f = 0$ and at $f = n_{(p)}^{-1}$ and its harmonics. Starting at any one of these peak frequencies and moving left or right we will define a critical frequency to be the first value of f where $C^{*2}(f) = 0.05$. We need only the smallest two of these critical frequencies. Since $C^*(f) = T^*(n_{(p)}f)$ for $n_{(t)} = n_{(s)}$, and since $f_{(t)} = 1.5n_{(t)}^{-1}$ is a good approximation of the critical frequency for $T^{*2}(f)$, we have that good approximations of the two critical frequencies for $C^{*2}(f)$ are

$$f_{(c)}^{(lower)} = 1.5n_{(p)}^{-1}n_{(s)}^{-1}$$

and

$$f_{(c)}^{(upper)} = n_{(p)}^{-1}(1 - 1.5n_{(s)}^{-1}).$$

In Figure 12, $f_{(c)}^{(lower)}$ and $f_{(c)}^{(upper)}$ are shown by the vertical lines on the graph of $C^{*2}(f)$.

The purpose of the high-pass filter H is to remove the power in $C^{*2}(f)$ below $f_{(c)}^{(lower)}$. If we did not do this, and took $S = C$, then $S^{*2}(f)$ and $T^{*2}(f)$ would compete for variation at low frequencies. On the other hand, we do not want $C^{*2}(f)$ to be affected for f above $f_{(c)}^{(upper)}$. That is, we want H to remove low-frequency power but otherwise not alter the power at the seasonal frequencies. Thus we want to choose $n_{(l)}$ so that the following criterion is met:

$$H^{2*}(f) \approx 0 \quad \text{for} \quad f \leqslant f_{(c)}^{(lower)}$$

and

$$H^{2*}(f) \approx 1 \quad \text{for} \quad f \geqslant f_{(c)}^{(upper)}.$$

We will define three critical frequencies for $H^{*2}(f)$ in the following way: Let $f_{(h)}^{(lower)}$ be the lowest frequency such that $H^{*2}(f) = 0.05$; let $f_{(h)}^{(middle)}$ be defined by $H^{*2}(f) = 0.5$, and let $f_{(h)}^{(upper)}$ be the highest frequency such that $|1 - H^{*2}(f)| = 0.05$. In Figure 12, these three critical frequencies are shown by the vertical lines on the plot of $H^{*2}(f)$. We can satisfy the above criterion if we can choose $n_{(l)}$ so that $f_{(h)}^{(lower)} \geqslant f_{(c)}^{(lower)}$ and $f_{(h)}^{(upper)} \leqslant f_{(c)}^{(upper)}$. In other words, the upper and lower critical frequencies for $H^{*2}(f)$ must be in the interval formed by the critical frequencies of $C^{*2}(f)$.

The smallest value that $n_{(s)}$ can take and still smooth the data is 5. (A value of $n_{(s)}$ equal to 1 or 3 results in no smoothing.) In fact, $n_{(s)} = 5$ is a rather small value that results in very little smoothing and that would be unlikely to arise in practice. Now

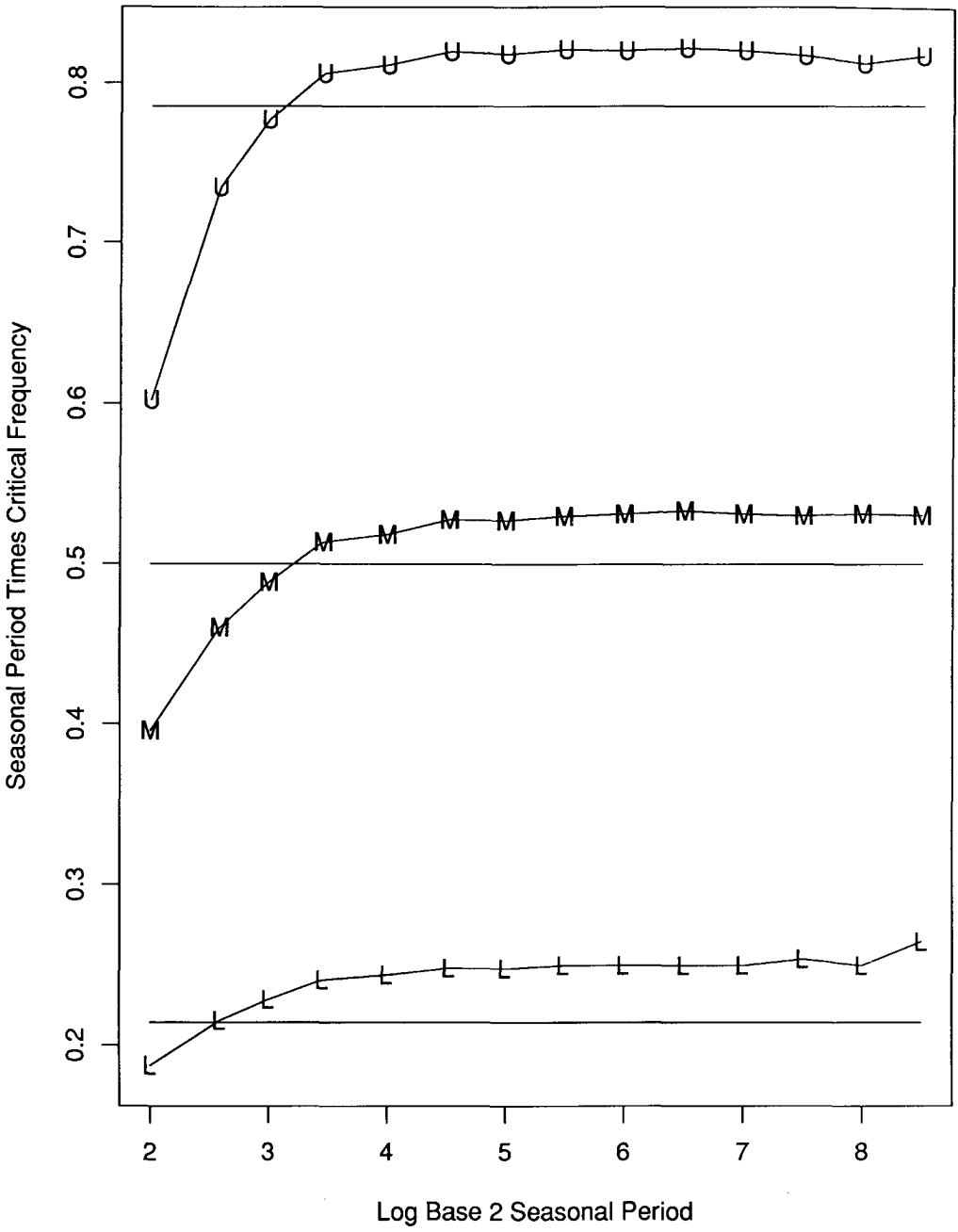

Fig. 13. Critical Frequencies.

the smaller $n_{(s)}$ is, the harder it is for $H^{*2}(f)$ to satisfy the above criterion since $f_{(c)}^{(upper)}$ decreases and $f_{(c)}^{(lower)}$ increases as $n_{(s)}$ decreases; that is, the specification gets tighter.

By exploring $H^{*2}(f)$ we discovered the following two properties: (1) if we choose $n_{(l)}$ so that $f_{(h)}^{(middle)}$ is close to the midpoint, $(2n_{(p)})^{-1}$, of $f_{(c)}^{(lower)}$ and $f_{(c)}^{(upper)}$, then the above

criterion for $H^{*2}(f)$ is met or nearly met provided $n_{(s)} \geqslant 7$; (2) if we take $n_{(l)}$ to be the smallest odd integer greater than or equal to $n_{(p)}$, $f_{(h)}^{(middle)}$ is close to $(2n_{(p)})^{-1}$. This is the choice of $n_{(l)}$ given in Section 3. The two properties are demonstrated in Figure 13. For 14 values of $n_{(p)}$,

$$n_{(p)} = 2^{2+m/2} \quad \text{for} \quad m = 0 \quad \text{to} \quad 13,$$

we computed the three critical frequencies for $H^{*2}(f)$ for the above choice of $n_{(l)}$. The three frequencies multiplied by $n_{(p)}$ are graphed against $\log_2(n_{(p)})$ in Figure 13; $n_{(p)}f_{(h)}^{(upper)}$ is graphed by the symbol "U", $n_{(p)}f_{(h)}^{(middle)}$ is graphed by the symbol "M"; and $n_{(p)}f_{(h)}^{(lower)}$ is graphed by the symbol "L". The upper horizontal line shows the value of $n_{(p)}f_{(c)}^{(upper)}$ for $n_{(s)} = 7$; this value is

$$n_{(p)}n_{(p)}^{-1}(1 - 1.5/7) = 0.79.$$

Some of the values of $n_{(p)}f_{(h)}^{(upper)}$ are slightly above the limit, but not by an amount that seriously jeopardizes the above criterion. The lower horizontal line shows the value of $n_{(p)}f_{(c)}^{(lower)}$ for $n_{(s)} = 7$; this value is

$$n_{(p)}n_{(p)}^{-1}(1.5/7) = 0.21.$$

Some of the values are slightly below the design limit but, again, the deviation is not serious. The middle horizontal line is the value

$$n_{(p)}(0.5n_{(p)}^{-1}) = 0.5$$

the target for $n_{(p)}f_{(h)}^{(middle)}$. Except for the two smallest values of $n_{(p)}$, $n_{(p)}f_{(h)}^{(middle)}$ is quite close to the target.

Having satisfied the above criterion for $H^{*2}(f)$ we now want to choose $n_{(t)}$ so that $T^{2*}(f)$ and $S^{*2}(f)$ do not both have non-trivial values. We can do this by choosing $n_{(t)}$ so that $f_{(t)} \leqslant f_{(c)}^{(upper)}$. This means that

$$1.5n_{(t)}^{-1} \leqslant n_{(p)}^{-1}(1 - 1.5n_{(s)}^{-1})$$

or

$$n_{(t)} \geqslant \frac{1.5n_{(p)}}{1 - 1.5n_{(s)}^{-1}}.$$

This is the recommendation given for $n_{(t)}$ in Section 3.

The criteria for $n_{(s)}$, $n_{(t)}$, and $n_{(l)}$ that have arisen from this analysis are satisfied by the choices for the example in Figure 12. The goal of separating the power of $S^{*2}(f)$ and $T^{*2}(f)$ has been met. Notice that in this example, $n_{(s)} = 7$, the worst-case value in terms of the design of $H^{*2}(f)$.

6. Summary

6.1. *The choice of the STL parameters in practice*

STL can be used in practice with most of the parameters chosen in an automated way. In the following, which describes this usage, $[x]$ denotes the smallest integer greater than or equal to x, and $[x]_{odd}$ denotes the smallest odd integer greater than or equal to x:

1. $n_{(p)}$ arises in an obvious way from the application.
2. $n_{(l)} = [n_{(p)}]_{odd}$.
3. $n_{(s)}$, an odd integer greater than or equal to 7, is chosen on the basis of knowledge of the time series and on the basis of diagnostic methods.
4. $n_{(t)} = [1.5n_{(p)}/(1 - 1.5/n_{(s)})]_{odd}$.
5. The decision about whether to use robust estimation or not is based on knowledge of the series and diagnostic methods. If robustness is not needed, use $n_{(i)} = 2$ and $n_{(o)} = 0$. If robustness is needed, use $n_{(i)} = 1$; $n_{(o)} = 5$ is a safe value and $n_{(o)} = 10$ provides near certainty of convergence. Of course, a convergence criterion such as that in Section 3.3 can be used and the robustness iterations ended when convergence occurs.
6. For the Fortran implementation of

Section 4, use $n_{(l)}^{(jump)} = [n_{(l)}/10]$, $n_{(s)}^{(jump)} = [n_{(s)}/10]$, and $n_{(t)}^{(jump)} = [n_{(t)}/10]$. If computation time is still likely to be too great, replace the values of 10 by 5.

There are four graphical diagnostic methods that help the data analyst assess the adequacy of the choices of points 3, 4, and 5. They are the decomposition plot, the cycle-subseries plot, the seasonal-diagnostic plot, and the trend-diagnostic plot. In some cases, a seasonal post-smoothing or a trend post-smoothing is desirable.

6.2. *Design criteria*

In Section 1 we set out several criteria for the design of STL – simplicity, straightforward use, flexibility in the amounts of trend and seasonal smoothing and in the period of the seasonal component, allowance for missing values, robust estimation, easy computer implementation, and fast computation.

Simplicity was achieved by basing STL on one smoother (loess), by minimizing the number of smoothing operations in the inner loop, and by using a straightforward weighting procedure to achieve robustness. There are at least two benefits to having a decomposition procedure with a simple design. One is easy computer implementation; for example, the Ratfor implementation discussed in Section 4 consists of 338 lines of code. A second benefit is that simplicity of a procedure allows one to analyze its properties, and an understanding of properties leads to more informed usage. For example, we were able to study the eigenvalue properties of STL in Section 5, and this led to guidelines for the choices of the parameters.

Parameter selection for STL is straightforward. One can use prescribed values for $n_{(l)}$ and $n_{(t)}$ so that for each application, the data analyst focuses on two choices: whether to use robust estimation and the value of the seasonal smoothing parameter, $n_{(s)}$. We have provided several graphical methods to help the data analyst make these choices.

The parameters of STL provide substantial flexibility in its use. The seasonal smoothing parameter, $n_{(s)}$, can vary from small to large, allowing the data analyst to accommodate a wide range of seasonal patterns. Similarly $n_{(t)}$ and $n_{(l)}$ can range from small to large and this allows us to set them to the prescribed values. Also, $n_{(p)}$ can be any integer greater than 1, so STL can accommodate any number of observations per seasonal cycle; values of 4, 7, 12, 24, and 365 occur commonly in practice.

Fast computation of STL is possible because there is a general approach to fast computation of loess – compute at selected values and interpolate elsewhere. Accommodating missing values is possible because the loess smoother can be applied to data with unequally-spaced x values, and because it can provide a fitted value for any value of x.

The method used for robustness in STL is a general one called *iterated weighted least-squares* (Andrews 1974). We chose this method for STL because it has desirable properties (Andrews et al. 1972) and because it has been used successfully in practice for robust estimation in other situations such as fitting parametric regression models (Andrews 1974) and the general loess smoother (Bickel et al. 1989), and also in the SABL seasonal-trend decomposition procedure (Cleveland, Devlin, and Terpenning 1982).

6.3. *Modifications to STL: Multiplicative decomposition and components estimated by regression*

Computer programs that implement STL can typically be easily changed to modify

the STL procedure or make additions; the reasons for the ease is the relatively simple design of STL. For example, it would be relatively easy to modify the Ratfor programs described in Section 4 to make STL a multiplicative rather than an additive decomposition procedure (Shiskin, Young, and Musgrave 1967). Also, the estimation of a component, F_v, by regression methods can be easily inserted into these programs. One situation where such estimation is important is aggregated monthly time series that have a trading-day component (Young 1965). To do this there would be a Step 7 in the inner loop in which the trend and seasonal are subtracted from Y_v and the regression estimation carried out using the residuals. Also, the regression component would be subtracted in Step 1, in Step 5, and in the computation of the remainder in the outer loop. As with the trend component, a starting value of $F_v^{(0)} \equiv 0$ could be used. If robustness iterations are carried out, the robustness weights would be used in the least-squares fitting of the regression estimation.

6.4. *Models and confidence intervals*

Suppose the time series being decomposed is Gaussian and that the data analyst has fitted a model to the data, say a standard ARIMA model (Box and Jenkins 1970). Then robustness iterations would not be used and the STL seasonal component would be a linear operator $\tilde{S}$ applied to the time series; thus confidence intervals for the seasonal component could be straightforwardly computed. In fact, equipped with a model, one could predict the series forward and backward and apply STL to a series consisting of the data and the predicted values; this then would correspond to an "optimal" decomposition in a sense defined by Pierce (1979). In effect, what we would be doing is combining STL and a standard ARIMA model. (This has been done for X-11 in well-known work by Dagum (1978).)

One might legitimately ask the following question: If we go to the trouble of building a model for a series, why not develop a component model for the series (e.g., Hillmer and Tiao 1982; Carlin and Dempster 1989) and use the model to derive a decomposition? There are two reasons for using STL and a standard model. First, the component models so far developed usually do not allow for as flexible a specification of the seasonal component as STL. (The model of Carlin and Dempster appears to be one exception.) Second, developing a standard time series model and using STL is typically considerably easier than fitting a component model. (Fitting the Carlin-Dempster model is not an exception.)

6.5. *X-11*

The standard procedure in use today for decomposing a time series is X-11 (Shiskin, Young, and Musgrave 1967). This method, which dates back to the 1950s and 1960s, is quite interesting from a historical perspective because it incorporated a number of innovative statistical ideas that would later become quite fashionable in other areas of statistics. One is iterative estimation of the trend, seasonal, and regression components (as is done in the inner loop of STL) by what is now known as the *backing-fitting algorithm*. This method is used in projection pursuit regression (Friedman and Stuetzle 1981), in alternating conditional expectation (Breiman and Friedman 1985), and in additive fitting (Hastie and Tibshirani 1986). The backfitting in X-11 also utilized *semiparametric modeling*, another topic of substantial current interest in statistics (Engle,

Granger, Rice, and Weiss 1986; Green 1985). Of course, much more is now known about backfitting than at the time of the development of X-11, and this new knowledge has been important for the design of STL; in particular, the eigenvalue analysis in Section 5 provides important information on how to prevent the seasonal and trend components from competing for the same variation in the data.

A second innovative methodology of X-11 was robust estimation, which was later intensively studied in the 1970s (Andrews et al. 1972; Huber 1977; Mallows 1979). Again, STL has benefited from this later work, in particular, in the use of iterated weighted least-squares for robust estimation.

Despite the impressive innovative work of X-11, it seems reasonable to think in terms of replacing it now by more modern methods. The innovative pieces just cited are now superseded by the later work. For example, the new backfitting work has allowed a scientific design of the inner loop of STL. (It would be interesting to study *X*-11 by an eigenvalue analysis such as that in Section 5 to determine which combinations of amounts of seasonal and trend smoothing lead to the trend and seasonal components competing for the same variation in the data.) And iterated weighted least-squares is a more reliable estimation procedure than the data-modification method used in X-11. (The X-11 robust estimate of location uses the sample standard deviation to determine the data modification; this is a poor method since the standard deviation can itself be very adversely affected by outliers.)

But there are even more serious defects in X-11. First, it is a very complicated procedure; this makes it difficult to select options and makes it difficult to decide how to modify them when diagnostic checking reveals problems. Also, the computer routines that implement X-11 are complex and difficult to deal with. A second major defect of X-11 is inflexibility. It cannot handle missing values (and could not easily be modified to do so), it allows seasonal periods of only 4 and 12, and it has only four levels of seasonal smoothing and three levels of trend smoothing.

6.6. *Software*

The Fortran implementation described in Section 4 is public-domain software that may be obtained in two ways, electronically or on a floppy disk. To obtain it electronically, send the message

send stl from a

to research! netlib on UUCP or to netlib@research.att.com on INTERNET. The message is read by a program and the code is automatically returned. Single-precision Fortran subroutines for the general loess smoother can also be obtained by sending the message

send loess from a

to one of the above addresses. To get double-precision routines replace "loess" by "dloess" in the message.

A floppy disk that includes both the loess and STL programs may be obtained (for a service charge) by writing to

Wadsworth, Inc.
7625 Empire Drive
Florence, Kentucky 41042
U.S.A.

and asking for lowess/loess/stl, ISBN 0-534-12756-8.

7. References

Andrews, D.F. (1974). A Robust Method

for Multiple Linear Regression. Technometrics, 16, 523–531.

Andrews, D.F., Bickel, P.J., Hampel, F.R., Huber, P.J., Rogers, W.H., and Tukey, J.W. (1972). Robust Estimates of Location. Princeton, NJ: Princeton University Press.

Becker, R.A., Chambers, J.M., and Wilks, A.R. (1988). The New S Language. Pacific Grove, CA: Wadsworth.

Box, G.E.P. and Jenkins, G.M. (1970). Time Series Analysis: Forecasting and Control. San Francisco: Holden-Day.

Breiman, L. and Friedman, J.H. (1985). Estimating Optimal Transformation for Correlation and Regression. Journal of the American Statistical Association, 80, 580–598.

Buja, A., Hastie, T.J., and Tibshirani, R.J. (1989). Linear Smoothers and Additive Models. The Annals of Statistics, to appear.

Carlin, J.B. and Dempster, A.P. (1989). Sensitivity Analysis of Seasonal Adjustments: Empirical Case Studies. Journal of the American Statistical Association (with discussion), 84, 6–32.

Cleveland, W.S. and Devlin, S.J. (1988). Locally-Weighted Regression: An Approach to Regression Analysis by Local Fitting. Journal of the American Statistical Association, 83, 596–610.

Cleveland, W.S., Devlin, S.J., and Grosse, E. (1988). Regression by Local Fitting: Methods, Properties, and Computational Algorithms. Journal of Econometrics, 37, 87–114.

Cleveland, W.S., Devlin, S.J., and Terpenning, I.J. (1982). The SABL Seasonal and Calendar Adjustment Procedures. Time Series Analysis: Theory and Practice 1, edited by O.D. Anderson. New York: North-Holland, 539–564.

Cleveland, W.S. and Grosse, E. (1990). Fitting Curves and Surfaces to Data. Monterey, CA: Wadsworth Advanced Books and Software.

Dagum, E.B. (1978). Modelling, Forecasting, and Seasonally Adjusting Economic Time Series with the $X - 11$ ARIMA Method. The Statistician, 27, 203–216.

Engle, R.F., Granger, C.W.F., Rice, J., and Weiss, A. (1986). Semi-Parametric Estimates of the Relation Between Weather and Electricity Sales. Journal of the American Statistical Association, 81, 310–320.

Friedman, J.H. and Stuetzle, W. (1981). Projection Pursuit Regression. Journal of the American Statistical Association, 76, 817–823.

Green, P.J. (1985). Linear Models for Field Trials, Smoothing and Cross-Validation. Biometrika, 72, 527–537.

Grenander, U. and Szego, G. (1958). Toeplitz Forms and Their Applications. Los Angeles: University of California Press.

Hastie, T.J. and Tibshirani, R.J. (1986). Generalized Additive Models. Statistical Science, 1, 297–318.

Hillmer, S.C. (1985). Measures of Variability for Model-Based Seasonal Adjustment Procedures. Journal of Business and Economic Statistics, 3, 60–68.

Hillmer, S.C. and Tiao, G.C. (1982). An ARIMA-Model-Based Approach to Seasonal Adjustment, Journal of the American Statistical Association, 77, 63–70.

Huber, P.J. (1977). Robust Statistical Procedures. Philadelphia: Society for Industrial and Applied Mathematics.

Keeling, C.D., Bacastow, R.B., and Whorf, T.P. (1982). Measurements of the Concentration of Carbon Dioxide at Mauna Loa Observatory, Hawaii. Carbon Dioxide Review: 1982, edited by W.C. Clark,

New York: Oxford University Press, 317–385.

Kernighan, B.W. (1975). A Preprocessor for a Rational Fortran. Software Practice and Experience, 5, 395–406.

Komhyr, W.D. and Harris, T.B. (1977). Measurements of Atmospheric CO_2 at the U.S. GMCC Baseline Stations. Air Pollution Measurement Techniques: Special Environmental Report 10, WMO 460, Geneva. World Meteorological Organization, 9–19.

Mallows, C. (1979). Robust Methods – Some Examples of Their Use. The American Statistician, 33, 179–184.

Pierce, D.A. (1979). Signal Extraction Error in Nonstationary Time Series. Annals of Statistics, 7, 1303–1320.

Shiskin, J., Young, A.H., and Musgrave, J.C. (1967). The $X - 11$ Variant of the Census Method II Seasonal Adjustment Program. Bureau of the Census Technical Paper 15, U.S. Department of Commerce, Washington, D.C.

Young, A. (1965). Estimating Trading-Day Variation in Monthly Economic Time Series. Technical Paper No. 12, U.S. Bureau, of the Census, Washington, D.C.

Received April 1989

Comment

Dennis Trewin[1]

First, I would like to congratulate the authors on the paper. Not only does the paper address the common practical statistical problem of seasonal adjustment with an innovative approach, but the presentation of the material is excellent. I enjoyed the paper thoroughly although I wish I could have experimented with the system more before preparing this comment.

In Section 1.1, the authors outline criteria they wanted STL to satisfy. The paper provides the readers with some confidence that the system does indeed satisfy these six criteria. I particularly like the flexibility in specifying the amounts of variation in the trend and seasonal components – this gives the experienced seasonal adjuster who has a good feeling for the data series a very powerful tool to refine his/her adjustments – more so than for the X-11 method.

The method also allows some flexibility in the specification of the number of cycles in the seasonal component. While this will be of limited value to government statistical agencies where most series are monthly or quarterly, it provides a distinct advantage over X-11 to those who wish to adjust a series that is not monthly or quarterly.

By its nature the method is robust to outliers. This, together with the strong graphics capabilities, must also make it a

[1] Australian Bureau of Statistics, P.O. Box 10, Belconnen, A.C.T. 2616, Australia.

Acknowledgement: I would like to thank Andrew Sutcliffe for the assistance he provided in preparing these comments and for the X-11 analysis of the U.S. unemployed male data.