A **time series** is a sequence of observations collected over time (like temperatures at a certain location, or sales of airline tickets). In the past, we have assumed that all the observations are independent, but we can no longer do this: if the temperature is higher than average today, it may be higher than average tomorrow. In fact, the study of time series involves disentangling the correlation structure that underlies our sequence of numbers.

Let's start by looking at two kinds of correlation structure that make up (in combination) many time series. We need some symbols: Z_1, Z_2 and so on denote *independent* values (with mean 0), which we cannot observe, and let X_1, X_2 and so on denote the actual time series we do observe. Let t be any particular time point, and let r_k be the correlation between each value of X and the value of X k steps before. In other words, if you observe 1, 2, 6, 3, 4, and you make a table like this, with the original series one step back:

1	Missing
2	1
6	2
3	6
4	3
missing	4

0

and calculate the correlation between the two columns, ignoring any missing values, you are figuring out r_1.

The first kind of series is a very dull one: $X_t=Z_t$, so that each value we observe is independent of all the others. This is called a **purely random process**. The theoretical correlation between any one X and any other X is 0, so that r_1, r_2 and so on are all 0. If we observed such a process, the correlations wouldn't be exactly zero, but they would be small.

The second kind of series is called an **autoregressive process**, where $X_t=a_1 X_{t-1}+a_2X_{t-2}+...+a_kX_{t-k}+ Z_t$. In other words, the current value of the series depends on one or more of the previous values, plus a random error. This is just like regression of the series on itself, hence the name. It's symbolized by AR(k).

Let's look at AR(1), which is $X_t=a_1X_{t-1}+Z_t$. We'll suppose that $X_1=Z_1$, and build things up from there. Suppose that $-1<a_1<1$. $X_2=a_1X_1+Z_2=a_1Z_1+Z_2$, so X_2 contains a small part of Z_1 along with its new random piece Z_2. $X_3=a_1X_2+Z_3=a_1(a_1Z_1+Z_2)+Z_3=a_1^2Z_1+a_1Z_2+Z_3$, so that X_3 contains an even smaller part of Z_1, a small part of Z_2 along with its own random piece Z_3. And so on. The idea is that each X_t is correlated with *all* the other X's, but the correlation decreases as you go back in time. That is, theoretically, if you ignore the fact that some of the r's could be negative, $r_1>r_2>r_3>...>0$. The same is true for all autoregressive processes.

The third kind of series is called a **moving average**, where $X_t=Z_t+b_1Z_{t-1}+b_2Z_{t-2}+...+b_kZ_{t-k}$. Each X depends only on the unobservable independent random pieces. It is symbolized by MA(k).

Let's look at MA(1), where $X_t=Z_t+b_1Z_{t-1}$, and trace it through from $X_1=Z_1$: $X_2=Z_2+b_1Z_1$, $X_3=Z_3+b_1Z_2$ and so on. But see that X_1 and X_3 have no Z's in common, so there is no correlation at all between them.

That is, theoretically, r_2, r_3, and so on are all 0. In general, for an MA(k) process, r_{k+1} and beyond are theoretically 0 (and so for real data from an MA(k) process they should be small).

You see that r_1, r_2 and so on have a crucial role to play in understanding time series. They are called **autocorrelations**, and the complete set of them is called the **autocorrelation function** (acf). One of the important parts of time series analysis is to plot the acf (plot r_t against t), and this gives one clue as to what kind of time series you have: if the r's become zero beyond a certain point, you have an MA series, and if they just decay to zero you have an AR series.

Judging whether the acf is decaying is hard, and a second useful tool is the **partial autocorrelation function (pacf)**. This, for an AR(k) series, is zero beyond the k-th partial autocorrelation. Thus, if you look at both plots, you'll get a sense of whether you have an AR or an MA process, and the "order" (value of k) for either.

Here's an example. The data are in ma1.dat. I generated them at random according to an MA(1) process, so we would hope that the acf and pacf suggest this as a possible model.

First a plot of the series, which SAS makes rather difficult (and incomprehensible).

Code:

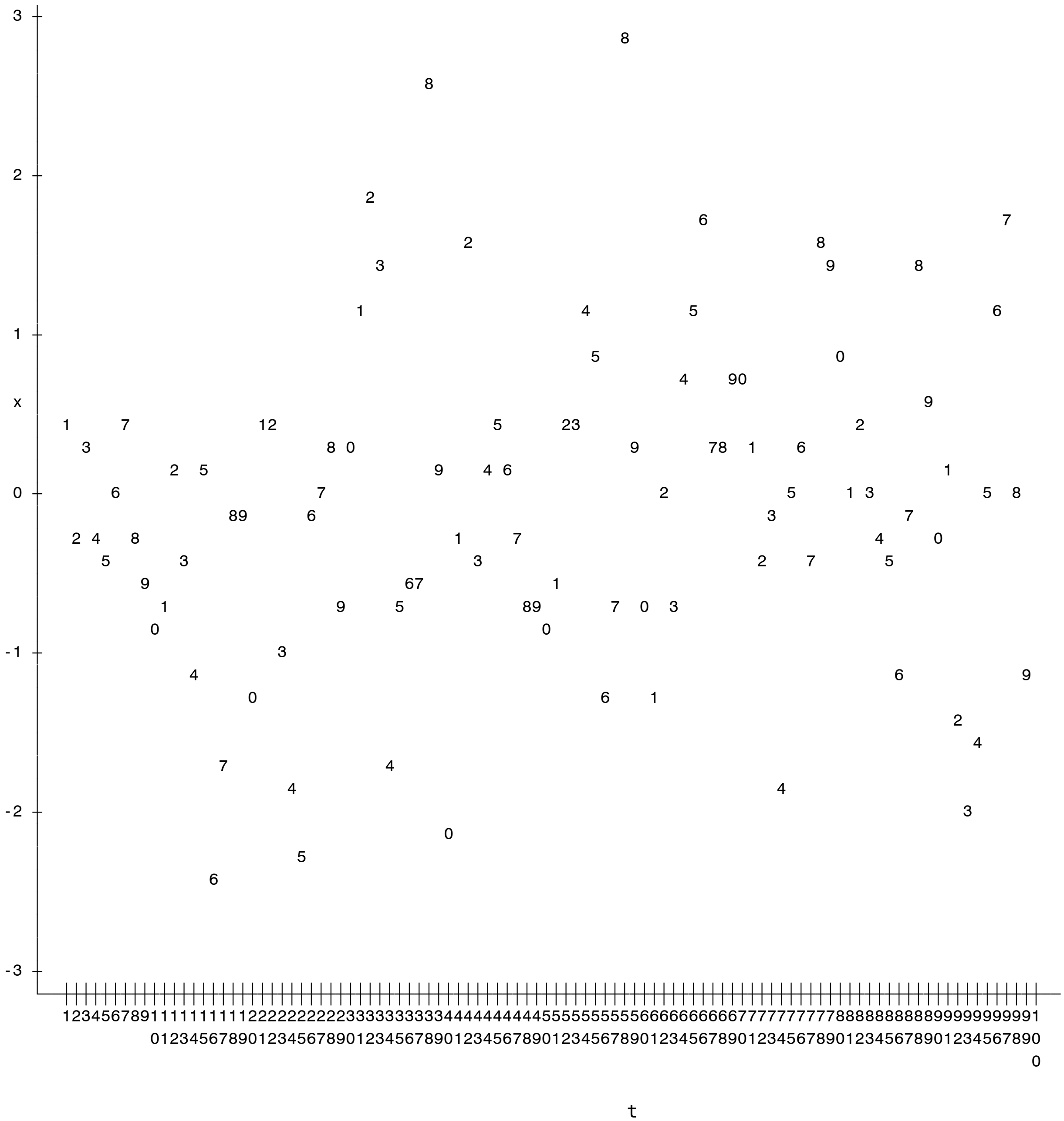
```
options ls=120;

data ts;
  infile "ma1.dat";
  input x;
  t=_n_;
  r=mod(t,10);

proc plot;
  plot x*t=r / haxis=1 to 100 by 1;
```

There are 100 observations in our series (well, actually 99), so the first thing is to make sure SAS plots them all. Then read in the data, with a twist: the special name "n" with underscores is the observation number, and r is the remainder when t is divided by 10, so our data set actually has three variables, t (time) r, and x (the actual series). Finally, the actual plot. We don't want SAS to mess around with the horizontal scale, so we specify the scale we want: "1 to 100" because we have (about) 100 observations, and we use the remainder as the plotting symbol, so that the plot labels the observations 1,2,3,4 etc in order, and it's easier to tell which one comes next. Here's the plot, shrunk a bit to get it on the page:

Plot of x*t. Symbol is value of r.



The first value is slightly positive, the second is negative, the third is positive, the fourth and fifth are negative, and so on. There is a small tendency for an observation to be followed by another of the same sign.

Now some actual analysis. PROC ARIMA is the tool, and it has an IDENTIFY option to help us identify what kind of series this is. We don't need that underscore-n stuff any more, so the data step is a lot simpler.

Code:

```
data ts;
  infile "ma1.dat";
  input x;

proc arima;
  identify var=x esacf scan;
```

On the “identify” line are some other options that will help us identify the series (as if we didn't know). The output looks like this:

```
                              The ARIMA Procedure

                              Name of Variable = x

                       Mean of Working Series    -0.08366
                       Standard Deviation        1.012551
                       Number of Observations          99


                                Autocorrelations

Lag    Covariance    Correlation    -1 9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9 1

  0      1.025260        1.00000    |                    |********************|
  1      0.307514        0.29994    |                .   |******              |
  2     -0.137509        -.13412    |                .***|   .                |
  3     -0.088090        -.08592    |                . **|   .                |
  4     -0.086153        -.08403    |                . **|   .                |
  5     -0.067767        -.06610    |                .  *|   .                |
  6      0.092516        0.09024    |               .    |**  .               |
  7      0.043621        0.04255    |               .    |*   .               |
  8     -0.035104        -.03424    |               .   *|    .               |
  9      0.084939        0.08285    |               .    |**  .               |
 10      0.095591        0.09324    |               .    |**  .               |
 11      0.126371        0.12326    |               .    |**  .               |
 12      0.114410        0.11159    |               .    |**  .               |
 13      0.034411        0.03356    |               .    |*   .               |
 14     -0.035892        -.03501    |               .   *|    .               |
 15     -0.087562        -.08541    |               .  **|    .               |
 16     0.0013234        0.00129    |               .    |    .               |
 17     -0.028465        -.02776    |               .   *|    .               |
 18     -0.129693        -.12650    |               . ***|    .               |
 19     -0.099976        -.09751    |               .  **|    .               |
 20      0.166057        0.16197    |               .    |*** .               |
 21     0.0087638        0.00855    |               .    |    .               |
 22     -0.116710        -.11383    |               .  **|    .               |
 23      0.019285        0.01881    |               .    |    .               |
 24      0.052205        0.05092    |               .    |*   .               |

                         "." marks two standard errors
```

The “lag-0” autocorrelation is the correlation of the series with itself, so this is always 1. Interest is in whether any of the other autocorrelations are significantly different from zero: the lag 1 autocorrelation is, but none of the others beyond it are; they appear to be random rubbish. So this would suggest (correctly) an MA(1) process.

Next, the inverse autocorrelations. These are used the same way as the pacf, coming up after that.

```
                       Inverse Autocorrelations

 Lag    Correlation    -1 9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9 1

   1       -0.41502    |            ********|   .                |
   2        0.23876    |                .   |*****               |
   3       -0.03695    |                .  *|   .                |
   4        0.05706    |                .   |*  .                |
   5        0.04956    |                .   |*  .                |
   6       -0.10138    |                . **|   .                |
   7       -0.03775    |                .  *|   .                |
   8        0.04701    |                .   |*  .                |
   9       -0.13758    |                .***|   .                |
  10        0.00502    |                .   |   .                |
  11       -0.05593    |                .  *|   .                |
  12       -0.07714    |                . **|   .                |
  13        0.01631    |                .   |   .                |
  14       -0.03038    |                .  *|   .                |
  15        0.10212    |                .   |** .                |
  16       -0.10156    |                . **|   .                |
  17        0.10709    |                .   |** .                |
  18       -0.04881    |                .  *|   .                |
  19        0.20234    |                .   |****                |
  20       -0.19094    |                ****|   .                |
  21        0.09889    |                .   |** .                |
  22        0.03561    |                .   |*  .                |
  23        0.00099    |                .   |   .                |
  24       -0.02677    |                .  *|   .                |
```

The first 2 look significant, which means an AR(2) process might also fit these data. The pacf:

```
                       Partial Autocorrelations
 Lag    Correlation    -1 9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9 1
   1        0.29994    |                .   |******              |
   2       -0.24624    |               *****|   .                |
   3        0.04453    |                .   |*  .                |
   4       -0.11673    |                . **|   .                |
   5       -0.01372    |                .   |   .                |
   6        0.10442    |                .   |** .                |
   7       -0.06044    |                .  *|   .                |
   8        0.00245    |                .   |   .                |
   9        0.11227    |                .   |** .                |
  10        0.02292    |                .   |   .                |
  11        0.15815    |                .   |***.                |
  12        0.03723    |                .   |*  .                |
  13        0.04379    |                .   |*  .                |
  14        0.01271    |                .   |   .                |
  15       -0.07647    |                . **|   .                |
  16        0.08191    |                .   |** .                |
  17       -0.11055    |                . **|   .                |
  18       -0.12760    |                .***|   .                |
  19       -0.05367    |                .  *|   .                |
  20        0.16091    |                .   |***.                |
  21       -0.20208    |                ****|   .                |
  22       -0.05618    |                .  *|   .                |
  23        0.03064    |                .   |*  .                |
  24        0.03932    |                .   |*  .                |
```

Again, the values for lags 1 and 2 are significant, so AR(2) is possible. Next:

Autocorrelation Check for White Noise

To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	13.89	6	0.0309	0.300	-0.134	-0.086	-0.084	-0.066	0.090
12	19.11	12	0.0860	0.043	-0.034	0.083	0.093	0.123	0.112
18	22.32	18	0.2181	0.034	-0.035	-0.085	0.001	-0.028	-0.126
24	28.91	24	0.2234	-0.098	0.162	0.009	-0.114	0.019	0.051

This is a test of the null hypothesis that the autocorrelations are no better than random (if we were looking at a random process). The first 6 autocorrelations are bigger than random (the P-value is small), which is good because the first one was nonzero by design. Also, none of the remaining autocorrelations are better than random, also good because none of the autocorrelations beyond the first are “really” bigger than zero.

Next are some procedures that also help identify what kind of process we have. There's some output which is hard to read, but with a summary table below:

ARMA(p+d,q) Tentative
Order Selection Tests

SCAN p+d	SCAN q	ESACF p+d	ESACF q
0	1	0	1
2	0	2	2
		3	2
		4	2
		5	2

(5% Significance Level)

At the moment, d=0, p is the order of the suggested AR series, and q is the order of the suggested MA series. Both methods rate as their first choice the correct MA(1) series; SCAN's second choice is an AR(2) series that was also suggested by the autocorrelations.

The rest of the output:

Squared Canonical Correlation Estimates

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	0.0913	0.0183	0.0077	0.0075	0.0047	0.0089
AR 1	0.0639	0.0118	0.0009	0.0004	0.0091	0.0048
AR 2	0.0018	0.0107	0.0003	0.0013	0.0126	0.0004
AR 3	0.0147	0.0109	0.0125	0.0141	0.0047	0.0031
AR 4	0.0002	0.0131	0.0008	0.0007	0.0020	0.0041
AR 5	0.0136	0.0050	0.0003	0.0019	<.0001	0.0001

SCAN Chi-Square[1] Probability Values

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	0.0021	0.2168	0.4323	0.4445	0.5517	0.4151
AR 1	0.0110	0.3567	0.7982	0.8662	0.4204	0.5848
AR 2	0.6781	0.3548	0.8878	0.7563	0.3807	0.8896
AR 3	0.2328	0.3982	0.3212	0.3640	0.6319	0.7104
AR 4	0.9028	0.3578	0.8338	0.8453	0.7508	0.6326
AR 5	0.2569	0.5786	0.9016	0.7271	0.9830	0.9342

What you are looking for is a triangle of P-values, all of which are nonsignificant. The yellow triangle is an example; the top of the triangle is a suggested process (MA(1)). There is another triangle starting at AR 2, MA 0, so AR(2) is another possibility.

The other method is called ESACF. ESACF's other choices are so-called ARMA processes with both an AR part and an MA part. ARMA(2,2) looks like this: $X_t=a_1X_{t-1}+a_2X_{t-2}+Z_t+b_1Z_{t-1}+b_2Z_{t-2}$. Other ARMA processes use the same idea: an autoregressive part with X's in, and a moving-average part with Z's in. Here's the ESACF part of the output:

Extended Sample Autocorrelation Function

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	0.2999	-0.1341	-0.0859	-0.0840	-0.0661	0.0902
AR 1	0.5076	-0.3225	0.0311	-0.0224	-0.0575	0.1351
AR 2	0.1658	-0.3672	-0.1377	0.0023	-0.0953	0.0820
AR 3	0.3103	-0.3856	0.0476	-0.0519	-0.1197	0.0625
AR 4	-0.1013	0.4918	-0.0411	0.1573	0.1364	0.0652
AR 5	0.1001	0.4338	0.1456	0.1423	-0.0257	-0.0344

ESACF Probability Values

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	0.0028	0.2192	0.4382	0.4510	0.5556	0.4226
AR 1	<.0001	0.0017	0.7866	0.8439	0.6584	0.2261
AR 2	0.1026	0.0004	0.2623	0.9842	0.4961	0.4477
AR 3	0.0024	0.0002	0.6842	0.6291	0.4138	0.5856
AR 4	0.3235	<.0001	0.7561	0.2555	0.2404	0.6134
AR 5	0.3316	0.0006	0.3546	0.2950	0.8428	0.7925

The yellow triangle shows that AR 0, MA 1 (ie. MA(1)) is a possibility (the triangle can go either down or to the right). The blue triangle shows that AR 3, MA 2 would also work (ie. ARMA(3,2)). You can see that ARMA(2,2) would work as well.

If you don`t like this triangle business, you can always consult the summary tables. As ever, though, looking at acf, pacf, SCAN and ESACF may not give you the exact same recommendations, but they all have advice to offer.

The process we saw above was **stationary**, in that its mean didn`t change (there was no overall trend up or down: the mean stayed around 0), its variance (spread around 0) didn't change over time, and the correlations didn't change over time (which means that it makes sense to talk about $r_1$, because it doesn`t matter whether you`re comparing time 2 with time 1 or time 32 with time 31: all that matters is that your comparisons are 1 time unit apart).

An MA process is always stationary. An AR process may not be; the conditions (in terms of the a_k) are messy, but the same conditions that make an AR process stationary also make an MA process satisfy a condition called invertibility. An ARMA process is stationary if its AR part is.

SAS will test whether a series is stationary or not, using the Dickey-Fuller test. The hypotheses for this test seem the wrong way around: the *null* is that the series is not stationary, and the *alternative* is that it is. A small P-value means that stationarity is OK. To do the stationarity tests, use an IDENTIFY line like this:

```
identify var=x esacf scan stationarity=(adf=(0,1,2));
```

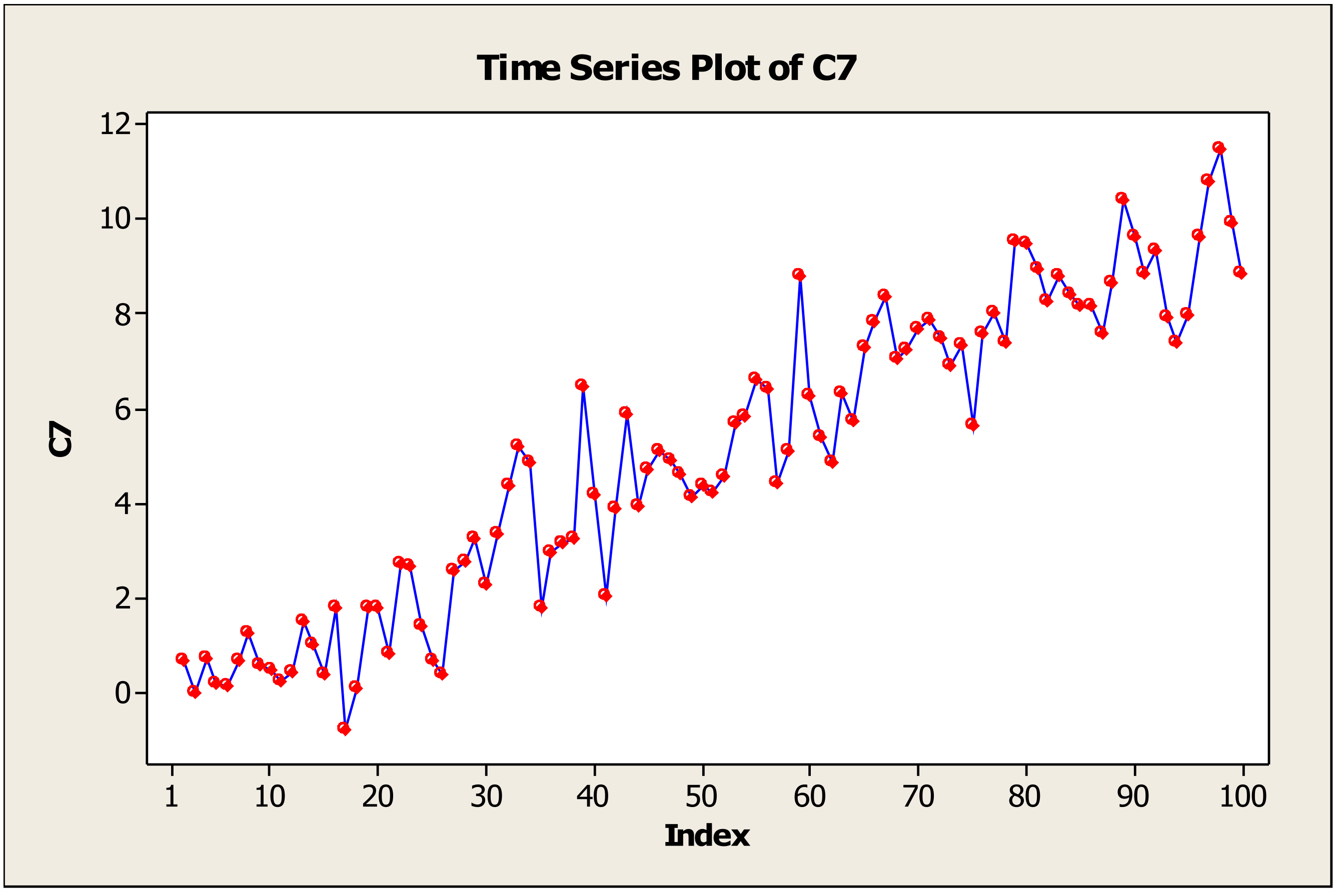
In our MA example, this gives output:

Augmented Dickey-Fuller Unit Root Tests

Type	Lags	Rho	Pr < Rho	Tau	Pr < Tau	F	Pr > F
Zero Mean	0	-67.7902	<.0001	-7.12	<.0001		
	1	-111.110	0.0001	-7.32	<.0001		
	2	-94.7700	<.0001	-5.52	<.0001		
Single Mean	0	-68.2540	0.0009	-7.13	<.0001	25.41	0.0010
	1	-112.784	0.0001	-7.35	<.0001	27.00	0.0010
Single Mean	2	-98.3958	0.0009	-5.57	<.0001	15.51	0.0010
Trend	0	-70.1101	0.0003	-7.22	<.0001	26.10	0.0010
	1	-120.468	0.0001	-7.52	<.0001	28.30	0.0010
	2	-113.146	0.0001	-5.81	<.0001	16.88	0.0010

Since all the P-values are small, there are no problems with stationarity.

One place where stationarity fails is if you have a **trend**: the mean goes up or down over time. Here is a series with a trend:



This one should fail stationarity, because the mean isn't constant. Run the Dickey-Fuller tests on this one (data in trend.dat):

```
data ts;
  infile "trend.dat";
  input x;

proc arima;
  identify var=x stationarity=(adf=(0,1,2));
```

with output:

```
                          Augmented Dickey-Fuller Unit Root Tests

 Type           Lags         Rho    Pr < Rho        Tau    Pr < Tau          F    Pr > F

 Zero Mean         0     -0.9158      0.4857      -0.45      0.5157
                   1     -0.2662      0.6205      -0.15      0.6285
                   2      0.7163      0.8555       0.62      0.8489
 Single Mean       0     -7.2605      0.2489      -1.95      0.3066       2.15    0.5286
                   1     -5.7542      0.3582      -1.75      0.4036       1.94    0.5819
                   2     -2.3085      0.7376      -1.03      0.7392       1.47    0.7002
 Trend             0    -70.1100      0.0003      -7.22      <.0001      26.10    0.0010
                   1    -120.469      0.0001      -7.52      <.0001      28.30    0.0010
                   2    -113.148      0.0001      -5.81      <.0001      16.88    0.0010
```

Most of these indicate that stationarity doesn't work. The tests labelled “trend”, however, show that stationarity is OK *once you allow for the trend.*

Another indication that stationarity doesn't work for these data comes from the acf:

```
                                         Autocorrelations

  Lag    Covariance    Correlation   -1 9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9 1      Std Error

    0     10.044854        1.00000    |                    |********************|             0
    1      9.161191        0.91203    |                .   |******************  |      0.100504
    2      8.453074        0.84153    |             .      |*****************   |      0.164027
    3      8.188363        0.81518    |            .       |****************    |      0.203006
    4      7.873302        0.78381    |           .        |****************    |      0.233744
    5      7.619229        0.75852    |          .         |***************     |      0.258936
    6      7.607151        0.75732    |         .          |***************     |      0.280483
    7      7.437931        0.74047    |        .           |***************     |      0.300429
    8      7.169507        0.71375    |       .            |**************      |      0.318330
    9      7.013198        0.69819    |       .            |**************      |      0.334104
   10      6.751591        0.67214    |      .             |*************.      |      0.348531
   11      6.472376        0.64435    |      .             |*************.      |      0.361387
   12      6.153731        0.61263    |     .              |************  .     |      0.372811
   13      5.819358        0.57934    |     .              |************  .     |      0.382845
   14      5.510669        0.54861    |    .               |***********    .    |      0.391600
   15      5.253492        0.52300    |    .               |**********     .    |      0.399288
   16      4.996807        0.49745    |    .               |**********     .    |      0.406148
   17      4.645043        0.46243    |    .               |*********      .    |      0.412257
   18      4.276685        0.42576    |   .                |*********       .   |      0.417463
   19      4.058695        0.40406    |   .                |********        .   |      0.421827
   20      3.983756        0.39660    |   .                |********        .   |      0.425718
   21      3.543312        0.35275    |   .                |*******         .   |      0.429434
   22      3.122685        0.31087    |   .                |******          .   |      0.432351
   23      2.997235        0.29839    |   .                |******          .   |      0.434603
   24      2.691329        0.26793    |   .                |*****           .   |      0.436667
```

Normally the acf shows a dying-out (if the series is AR or ARMA) or it cuts off completely (if the series is MA). But this time the correlations don't decrease fast: even the autocorrelation at lag 20 is over 0.30.

What to do about nonstationarity? The usual procedure is to replace the actual series by the series of differences between one observation and the next. Why does this help? Imagine you have the series 1,2,3,4,5,6: the differences are 1,1,1,1,1, constant, so that you have removed the trend. (Of course, you wouldn't have a perfectly linear trend, so you wouldn't have perfectly constant results, but you would have removed the trend.) Here is another kind of trend: 2,6,9,11,12,12,11, 9, 6, 2, -3: taking differences gives you 4,3,2,1,0,-1,-2,-3,-4,-5, which is a linear trend, and taking differences again gives you a constant -1. So you might have to take differences more than once, but as long as you start with a "polynomial" trend, taking differences will eventually get rid of the trend. The one case where this won't work is exponential growth (or exponential decay): the series 1,2,4,8,16,32,... on differencing gives you back the same thing. If you have exponential growth or decay you do better to take logs and work with the logged series.

So let's work with first differences of our series with trend. In the "identify" statement, X(1) means "first differences". Curiously, X(2) doesn't mean "second differences": it means "subtract off the value two time periods ago"; X(1,1) means "take differences, then take differences again", which is what second differences are. But we don't have to worry about that for our trend, which is "obviously" linear, so first differences should fix things:

```
proc arima;
  identify var=x(1) esacf scan stationarity=(adf=(0,1,2));
```

Augmented Dickey-Fuller Unit Root Tests

Type	Lags	Rho	Pr < Rho	Tau	Pr < Tau	F	Pr > F
Zero Mean	0	-113.933	0.0001	-11.66	<.0001		
	1	-253.796	0.0001	-10.97	<.0001		
	2	-569.931	0.0001	-8.18	<.0001		
Single Mean	0	-114.630	0.0001	-11.68	<.0001	68.22	0.0010
	1	-264.699	0.0001	-11.10	<.0001	61.62	0.0010
	2	-812.518	0.0001	-8.43	<.0001	35.54	0.0010
Trend	0	-114.614	0.0001	-11.62	<.0001	67.54	0.0010
	1	-264.761	0.0001	-11.04	<.0001	60.94	0.0010
	2	-811.065	0.0001	-8.37	<.0001	35.14	0.0010

Autocorrelations

```
 Lag    Covariance    Correlation    -1 9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9 1      Std Error

   0      1.435016        1.00000    |                    |********************|             0
   1     -0.258222        -.17994    |                ****|   .                |      0.101015
   2     -0.484497        -.33762    |             *******|   .                |      0.104235
   3      0.045101        0.03143    |               .    |*   .               |      0.114853
   4     -0.027354        -.01906    |               .    |    .               |      0.114941
   5     -0.163172        -.11371    |               .  **|    .               |      0.114973
   6      0.202829        0.14134    |               .    |*** .               |      0.116115
   7      0.041387        0.02884    |               .    |*   .               |      0.117857
   8     -0.182662        -.12729    |               . ***|    .               |      0.117929
   9      0.107886        0.07518    |               .    |**  .               |      0.119323
  10     -0.011167        -.00778    |               .    |    .               |      0.119806
  11      0.047958        0.03342    |               .    |*   .               |      0.119811
  12      0.053929        0.03758    |               .    |*   .               |      0.119906
  13     -0.017045        -.01188    |               .    |    .               |      0.120026
  14     -0.018572        -.01294    |               .    |    .               |      0.120038
  15     -0.125546        -.08749    |               .  **|    .               |      0.120052
  16      0.119275        0.08312    |               .    |**  .               |      0.120701
  17      0.069112        0.04816    |               .    |*   .               |      0.121284
  18     -0.136781        -.09532    |               .  **|    .               |      0.121479
  19     -0.223790        -.15595    |               . ***|    .               |      0.122239
  20      0.425141        0.29626    |               .    |******              |      0.124253
  21     -0.029889        -.02083    |               .    |    .               |      0.131263
  22     -0.279004        -.19443    |               .****|    .               |      0.131297
  23      0.117075        0.08158    |               .    |**  .               |      0.134203
  24      0.033840        0.02358    |               .    |    .               |      0.134708

                          "." marks two standard errors
```

They do more or less appear to be decaying away, although the lag-20 autocorrelation appears tobe significantly different from zero, which we hope is just coincidence.

I got this series by taking my MA(1) series and adding a linear trend, so now we see whether the ESACF and SCAN diagnostics pick this out (it's harder now):

```
 ARMA(p+d,q) Tentative
 Order Selection Tests

---SCAN--    --ESACF--
p+d     q    p+d     q

  0     2      0     2
  2     1      1     2
  5     0      2     2
               4     3
               5     3

(5% Significance Level)
```

d is the number of times *more* we should take differences (not counting the one time we already did), so the first recommendation is an MA(2) process for the differences.

When you have also done differencing, this is called an ARIMA model, and the notation is ARIMA(p,d,q): here ARIMA(0,1,2). So PROC ARIMA didn't quite get this one right: we should have had ARIMA(0,1,1).

One other issue we should look at is **seasonality**: some series will tend to have “highs” at the same time of year (for example, sales, or temperatures). If you have monthly measurements, you would expect any value to be similar to the one 12 months back.

An example of seasonality: here are the monthly average air temperatures for Recife (Brazil) for 1953 to 1962:

```
26.8 27.2 27.1 26.3 25.4 23.9 23.8 23.6 25.3 25.8 26.4 26.9
27.1 27.5 27.4 26.4 24.8 24.3 23.4 23.4 24.6 25.4 25.8 26.7
26.9 26.3 25.7 25.7 24.8 24.3 23.4 23.5 24.8 25.6 26.2 26.5
26.8 26.9 26.7 26.1 26.2 24.7 23.9 23.7 24.7 25.8 26.1 26.5
26.3 27.1 26.2 25.7 25.5 24.9 24.2 24.6 25.5 25.9 24.4 26.9
27.1 27.1 27.4 26.8 25.4 24.8 23.6 23.9 25.0 25.9 26.3 26.6
26.8 27.1 27.4 26.4 25.5 24.7 24.3 24.4 24.8 26.2 26.3 27.0
27.1 27.5 26.2 28.2 27.1 25.4 25.6 24.5 24.7 26.0 26.5 26.8
26.3 26.7 26.6 25.8 25.2 25.1 23.3 23.8 25.2 25.5 26.4 26.7
27.0 27.4 27.0 26.3 25.9 24.6 24.1 24.3 25.2 26.3 26.4 26.7
```

We do the usual business: autocorrelations, suggested models, tests for stationarity. The autocorrelations:

```
                                     Autocorrelations

 Lag    Covariance    Correlation    -1 9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9 1      Std Error

   0      1.357844        1.00000    |                    |********************|             0
   1      1.026322        0.75585    |                .   |***************     |      0.091287
   2      0.543664        0.40039    |               .    |********            |      0.133623
   3     -0.011639        -.00857    |              .     |     .              |      0.143272
   4     -0.543870        -.40054    |            ********|     .              |      0.143276
   5     -0.868829        -.63986    |       *************|     .              |      0.152322
   6     -0.929404        -.68447    |      **************|      .             |      0.173279
   7     -0.796936        -.58691    |        ************|       .            |      0.194510
   8     -0.467625        -.34439    |            .*******|       .            |      0.208747
   9     -0.019991        -.01472    |           .        |        .           |      0.213429
  10      0.472352        0.34787    |           .        |******* .           |      0.213437
  11      0.851590        0.62716    |           .        |*************       |      0.218111
  12      0.980109        0.72181    |           .        |**************      |      0.232654
  13      0.836181        0.61582    |          .         |************        |      0.250622
  14      0.436367        0.32137    |         .          |******    .         |      0.262930
  15     -0.069259        -.05101    |         .         *|          .         |      0.266183
  16     -0.476458        -.35089    |         .   *******|          .         |      0.266264
  17     -0.757262        -.55769    |         ***********|          .         |      0.270090
  18     -0.839461        -.61823    |        ************|          .         |      0.279522
  19     -0.680879        -.50144    |        . **********|           .        |      0.290693
  20     -0.393349        -.28969    |        .     ******|           .        |      0.297814
  21    -0.0095898        -.00706    |        .           |           .        |      0.300153
  22      0.423419        0.31183    |        .           |******     .        |      0.300155
  23      0.728418        0.53645    |        .           |***********.        |      0.302842
  24      0.854990        0.62967    |        .           |*************       |      0.310660

                               "." marks two standard errors
```

These certainly don't die away: a value will be highly (positively) correlated with the one 12, 24 etc months ago, and highly negatively correlated with the one 6 months (18 months) ago because one measurement is winter and the other is summer. So stationarity ought to fail: we know this, so there is no need to test it.

The following replaces each observation by the difference from the same time the previous year, and re-does the stationarity tests:

```
proc arima;
  identify var=x(12) stationarity=(adf=(0,1,2));
```

```
                       Augmented Dickey-Fuller Unit Root Tests

 Type           Lags         Rho    Pr < Rho        Tau    Pr < Tau          F    Pr > F

 Zero Mean         0    -71.8517      <.0001      -7.33      <.0001
                   1    -58.1755      <.0001      -5.35      <.0001
                   2    -34.3407      <.0001      -3.79      0.0002
 Single Mean       0    -71.9233      0.0010      -7.30      <.0001      26.62    0.0010
                   1    -58.2650      0.0010      -5.33      <.0001      14.20    0.0010
                   2    -34.3876      0.0010      -3.78      0.0042       7.13    0.0010
 Trend             0    -72.1321      0.0004      -7.28      <.0001      26.50    0.0010
                   1    -58.6631      0.0004      -5.33      0.0001      14.19    0.0010
                   2    -34.7596      0.0015      -3.79      0.0211       7.19    0.0267
```

and everything is OK. But we should also check for “seasonal stationarity”, which is done like this:

```
proc arima;
  identify var=x stationarity=(adf=(0,1,2) dlag=12);
```

(the “dlag” thing says that we are looking at a 12-month seasonality in our monthly data):

```
                  Seasonal Augmented Dickey-Fuller Unit Root Tests

        Type            Lags          Rho     Pr < Rho        Tau     Pr < Tau

        Zero Mean          0       0.0581       0.5501       0.20       0.6211
                           1      -0.0035       0.5454      -0.01       0.5416
                           2      -0.0402       0.5426      -0.09       0.5106
        Single Mean        0     -22.5056       0.0054      -3.68       0.0007
                           1     -32.2512       0.0013      -4.47       <.0001
                           2     -32.1476       0.0013      -4.47       <.0001
```

Stationarity fails for “zero mean”, but the mean temperature (for the original data) isn't zero anyway. The “single mean” tests say that after you allow for seasonality, estimating one mean is enough to make the series stationary (the values are all somewhere near 25).

The ESACF and SCAN methods don't appear to work for seasonal series, so a different strategy appears to be necessary to identify a seasonal model. We'll look at this again when we come back to this series.

The next stage is “estimation”. We have p+q parameters to estimate in general for a non-seasonal series. This will help us in the third stage, forecasting the future.

Going back to our original process (the one I generated and knew was MA(1)), the following code will estimate the one parameter (plus the overall mean). The “identify” line is necessary again, even though we've already done identification, because this is how we specify what variable to analyze and what differencing to use:

```
proc arima;
  identify var=x;
  estimate p=0 q=1 plot;
```

and the important part of the output is:

```
                      Conditional Least Squares Estimation

                                    Standard                 Approx
     Parameter       Estimate          Error    t Value    Pr > |t|      Lag

     MU              -0.08166        0.13699      -0.60      0.5525        0
     MA1,1           -0.44255        0.09121      -4.85      <.0001        1
```

The overall mean of the data as generated was 0, and the MA parameter was -0.5, so these estimates are each within one standard error of the truth. The P-value (<0.0001 for the MA term) answers the question “is this parameter estimate significantly different from zero”, which it should be for the MA term because we decided this was necessary.

```
                       Autocorrelation Check of Residuals

  To        Chi-             Pr >
  Lag      Square     DF    ChiSq    ---------------Autocorrelations--------------

    6        3.86      5    0.5698   -0.037   -0.103   -0.038   -0.037   -0.093    0.114
   12        7.18     11    0.7840    0.035   -0.079    0.100    0.024    0.082    0.070
   18        9.19     17    0.9341    0.005    0.006   -0.102    0.048   -0.024   -0.058
   24       21.35     23    0.5599   -0.157    0.225   -0.034   -0.125    0.057    0.023
```

The residuals should behave like a random process, because any dependence should be included in our model. That appears to be the case here.

```
                       Autocorrelation Plot of Residuals

Lag    Covariance    Correlation    -1 9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9 1

  0      0.900656        1.00000    |                    |********************|
  1     -0.032925        -.03656    |                .  *|   .                |
  2     -0.092865        -.10311    |                . **|   .                |
  3     -0.034083        -.03784    |                .  *|   .                |
  4     -0.033669        -.03738    |                .  *|   .                |
  5     -0.083851        -.09310    |                . **|   .                |
  6      0.103116        0.11449    |                .   |** .                |
  7      0.031613        0.03510    |                .   |*  .                |
  8     -0.070736        -.07854    |                . **|   .                |
  9      0.089956        0.09988    |                .   |** .                |
 10      0.021389        0.02375    |                .   |   .                |
 11      0.074134        0.08231    |                .   |** .                |
 12      0.063463        0.07046    |                .   |*  .                |
 13     0.0041397        0.00460    |                .   |   .                |
 14     0.0051498        0.00572    |                .   |   .                |
 15     -0.091578        -.10168    |                . **|   .                |
 16      0.043604        0.04841    |                .   |*  .                |
 17     -0.021378        -.02374    |                .   |   .                |
 18     -0.052501        -.05829    |                .  *|   .                |
 19     -0.141297        -.15688    |                .***|   .                |
 20      0.202301        0.22461    |                .   |****                |
 21     -0.030830        -.03423    |               .   *|    .               |
 22     -0.112891        -.12534    |               . ***|    .               |
 23      0.051626        0.05732    |               .    |*   .               |
 24      0.020786        0.02308    |               .    |    .               |

                          "." marks two standard errors
```

Likewise, none of the residual autocorrelations above lag 0 are at all large. This is good.

When you have a differenced series, like my "trend" one, you specify the differencing on the IDENTIFY line. Before, the identification suggested an MA(2) for the first differences of the data:

```
proc arima;
  identify var=x(1);
  estimate p=0 q=2;
```

Conditional Least Squares Estimation

Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	0.09791	0.0066892	14.64	<.0001	0
MA1,1	0.50641	0.09197	5.51	<.0001	1
MA1,2	0.44524	0.09239	4.82	<.0001	2

and make sure you are looking at the autocorrelations *of the residuals:*

Autocorrelation Check of Residuals

To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	3.02	4	0.5548	-0.026	-0.072	-0.015	-0.011	-0.071	0.132
12	7.25	10	0.7020	0.057	-0.058	0.119	0.038	0.094	0.084
18	8.85	16	0.9197	0.021	0.016	-0.082	0.054	-0.024	-0.050
24	20.05	22	0.5800	-0.152	0.218	-0.035	-0.117	0.056	0.018

which is all good.

For our climate data, there was a 12-month seasonal component, but there might be other things as well. We couldn't use SCAN or ESACF, so what we do is to fit a model that (we hope) includes everything that might be significant, and take out the things that turn out not to be. Let's start with

```
proc arima;
  identify var=x(12);
  estimate p=(1 2)(12 24) q=(1 2)(12 24) printall plot;
```

which has two bits: A possible ARIMA (2,1,2) process for the seasonal part (the 12-month differences, plus the (12 24) bits on p and q, and a possible ARMA(2,2) process for the non-seasonal part (the (1 2) part on p and q). “printall” gets some useful printout, and “plot” gets acfs and pacfs for the differenced series and the residuals. The idea is to fit a lot of extra stuff and take out what is not significant. However:

Conditional Least Squares Estimation

Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	0.03774	0.02936	1.29	0.2016	0
MA1,1	0.56624	2.90761	0.19	0.8460	1
MA1,2	-0.14576	1.02332	-0.14	0.8870	2
MA2,1	1.14492	0.91675	1.25	0.2146	12
MA2,2	-0.35803	0.70887	-0.51	0.6146	24
AR1,1	0.87539	2.91575	0.30	0.7646	1
AR1,2	-0.18155	1.94748	-0.09	0.9259	2
AR2,1	0.23164	0.91661	0.25	0.8010	12
AR2,2	-0.11827	0.21230	-0.56	0.5787	24

and nothing is significant! A look at the correlations of parameter estimates reveals what the problem is:

Correlations of Parameter Estimates

Parameter	MU	MA1,1	MA1,2	MA2,1	MA2,2
MU	1.000	-0.006	0.019	0.152	-0.181
MA1,1	-0.006	1.000	-0.978	0.152	-0.146
MA1,2	0.019	-0.978	1.000	-0.143	0.135
MA2,1	0.152	0.152	-0.143	1.000	-0.994
MA2,2	-0.181	-0.146	0.135	-0.994	1.000
AR1,1	-0.006	0.999	-0.976	0.154	-0.147
AR1,2	0.013	-0.996	0.989	-0.149	0.141
AR2,1	0.153	0.165	-0.156	0.994	-0.988
AR2,2	-0.012	0.134	-0.127	0.738	-0.683

Correlations of Parameter Estimates

Parameter	AR1,1	AR1,2	AR2,1	AR2,2
MU	-0.006	0.013	0.153	-0.012
MA1,1	0.999	-0.996	0.165	0.134
MA1,2	-0.976	0.989	-0.156	-0.127
MA2,1	0.154	-0.149	0.994	0.738
MA2,2	-0.147	0.141	-0.988	-0.683
AR1,1	1.000	-0.997	0.166	0.137
AR1,2	-0.997	1.000	-0.162	-0.135
AR2,1	0.166	-0.162	1.000	0.720
AR2,2	0.137	-0.135	0.720	1.000

There are some correlations here very close to 1 or -1. This is the same issue as in regression when you have correlated x's: it makes it hard to figure out which ones actually help with the prediction, and your only hope is to take some of them out and try again. In this case, let's assume that the seasonal process is at most ARIMA(1,1,1), so we take out any references to 24 months:

```
proc arima;
  identify var=x(12);
  estimate p=(1 2)(12) q=(1 2)(12) printall plot;
```

giving

Conditional Least Squares Estimation

Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	0.02989	0.02560	1.17	0.2456	0
MA1,1	0.53603	2.16762	0.25	0.8052	1
MA1,2	-0.16631	0.74392	-0.22	0.8235	2
MA2,1	0.79193	0.08501	9.32	<.0001	12
AR1,1	0.83262	2.17491	0.38	0.7026	1
AR1,2	-0.18704	1.40101	-0.13	0.8941	2
AR2,1	-0.10852	0.12793	-0.85	0.3983	12

and now at least the 12-month MA term is significant. As in regression, pull off the non-significant things one at a time (because taking off one might make other things significant). Here the lag-2 AR term is the first thing to go:

```
proc arima;
  identify var=x(12);
  estimate p=(1)(12) q=(1 2)(12) printall plot;
```

Conditional Least Squares Estimation

Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	0.03016	0.02671	1.13	0.2615	0
MA1,1	0.32095	0.31906	1.01	0.3168	1
MA1,2	-0.02263	0.15394	-0.15	0.8834	2
MA2,1	0.78687	0.08562	9.19	<.0001	12
AR1,1	0.62157	0.30492	2.04	0.0441	1
AR2,1	-0.11194	0.12724	-0.88	0.3811	12

and then the lag-2 MA term:

```
proc arima;
  identify var=x(12);
  estimate p=(1)(12) q=(1)(12) printall plot;
```

Conditional Least Squares Estimation

Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	0.03025	0.02697	1.12	0.2646	0
MA1,1	0.34730	0.23583	1.47	0.1439	1
MA2,1	0.78433	0.08409	9.33	<.0001	12
AR1,1	0.65206	0.19058	3.42	0.0009	1
AR2,1	-0.11309	0.12616	-0.90	0.3721	12

then the seasonal AR term (we still have the seasonal MA term):

```
proc arima;
  identify var=x(12);
  estimate p=(12) q=(1)(12) printall plot;
```

(the brackets on p=(12) stay because you *just* want the lag-12 month AR term, not all the ones up to 12):

Conditional Least Squares Estimation

Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	0.03009	0.01971	1.53	0.1299	0
MA1,1	-0.29290	0.09415	-3.11	0.0024	1
MA2,1	0.77266	0.08544	9.04	<.0001	12
AR1,1	-0.11484	0.12643	-0.91	0.3658	12

Note the effect of removing this term: we no longer need any of the AR terms, but the lag-1 MA term has become significant. This is just like in multiple regression. So finally we remove the last AR term:

```
proc arima;
  identify var=x(12);
  estimate q=(1)(12) printall plot;
```

```
                         Conditional Least Squares Estimation

                                            Standard                 Approx
            Parameter       Estimate           Error     t Value     Pr > |t|      Lag

            MU               0.02894         0.01953        1.48       0.1413        0
            MA1,1           -0.28970         0.09369       -3.09       0.0025        1
            MA2,1            0.81650         0.06313       12.93       <.0001       12
```

and we end up with a nice simple model: for the annual-differenced series, there is an MA term for one month ago and another MA term for 12 months ago.

As in regression, our last job is to check the residuals. In time series, that means looking at the acf and pacf (and inverse acf) of the residuals, to make sure there is no pattern left over. For instance, the acf:

```
                          Autocorrelation Plot of Residuals

 Lag    Covariance    Correlation    -1 9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9 1      Std Error

   0      0.278517        1.00000    |                    |********************|              0
   1      0.011089        0.03981    |                .   |*  .                |       0.096225
   2      0.050019        0.17959    |                .   |****                |       0.096377
   3      0.042130        0.15127    |                .   |***.                |       0.099428
   4     0.0010591        0.00380    |                .   |   .                |       0.101536
   5     -0.010496        -.03768    |                .  *|   .                |       0.101538
   6      0.037598        0.13499    |                .   |***.                |       0.101667
   7    0.00054246        0.00195    |                .   |   .                |       0.103313
   8     0.0070193        0.02520    |                .   |*  .                |       0.103314
   9     -0.019600        -.07037    |                .  *|   .                |       0.103371
  10      0.010346        0.03715    |                .   |*  .                |       0.103813
  11     0.0032771        0.01177    |                .   |   .                |       0.103936
  12     -0.019264        -.06917    |                .  *|   .                |       0.103949
  13    0.00035014        0.00126    |                .   |   .                |       0.104374
  14      0.016488        0.05920    |                .   |*  .                |       0.104374
  15     -0.046932        -.16851    |                .***|   .                |       0.104685
  16      0.012889        0.04628    |                .   |*  .                |       0.107167
  17     -0.015740        -.05651    |                .  *|   .                |       0.107351
  18     -0.033990        -.12204    |                . **|   .                |       0.107627
  19     0.0094179        0.03381    |                .   |*  .                |       0.108900
  20    -0.0002889        -.00104    |                .   |   .                |       0.108997
  21     -0.021469        -.07708    |                . **|   .                |       0.108998
  22      0.035263        0.12661    |                .   |***.                |       0.109501
  23    -0.0050341        -.01807    |                .   |   .                |       0.110848
  24    -0.0009334        -.00335    |                .   |   .                |       0.110876

                               "." marks two standard errors
```

which doesn't appear to show any serious problems. Also, the autocorrelation check of the residuals gives this:

Autocorrelation Check of Residuals

To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	8.67	4	0.0700	0.040	0.180	0.151	0.004	-0.038	0.135
12	10.11	10	0.4305	0.002	0.025	-0.070	0.037	0.012	-0.069
18	16.84	16	0.3958	0.001	0.059	-0.169	0.046	-0.057	-0.122
24	20.07	22	0.5787	0.034	-0.001	-0.077	0.127	-0.018	-0.003

with nothing significant, though it would be nice if those autocorrelations at lags 2 and 3 were smaller.

The last part of time series analysis is forecasting the future: that is, trying to use our knowledge gained about the correlational structure of the series to say what it is going to do. Forecasting is extrapolation, and is based on the assumption that the series will continue to behave as it has done, so there is some danger attached. Nonetheless, there are some things that can be done.

The standard procedure for forecasting is to leave the “identify” line as is (because that specifies the differencing), and leave the “estimate” line as is (because that specifies the AR and MA parts of the time series model), and then to add a “forecast” line like this (for the known MA(1) series):

```
data ts;
  infile "ma1.dat";
  input x;

proc arima;
  identify var=x;
  estimate p=0 q=1 plot;
  forecast;
```

```
                        Forecasts for variable x

        Obs       Forecast    Std Error       95% Confidence Limits

        100        -0.4633       0.9490        -2.3234          1.3968
        101        -0.0817       1.0378        -2.1157          1.9524
        102        -0.0817       1.0378        -2.1157          1.9524
        103        -0.0817       1.0378        -2.1157          1.9524
        104        -0.0817       1.0378        -2.1157          1.9524
        105        -0.0817       1.0378        -2.1157          1.9524
```

SAS produces forecasts for (by default) 24 time periods into the future. Note that here, the forecasts are all the same after the second one, because for an MA(1) series, an observation 2 time periods into the future is uncorrelated with anything you already have, so your best forecast is the overall mean (which is -0.0817). This is not very interesting. Compare my series with trend, to which we fitted an MA(2) process:

```
                        Forecasts for variable x

        Obs       Forecast    Std Error       95% Confidence Limits

        100         9.6371       0.9850         7.7065         11.5676
        101        10.1281       1.0984         7.9752         12.2810
        102        10.2260       1.0995         8.0711         12.3810
        103        10.3239       1.1005         8.1670         12.4809
        104        10.4219       1.1015         8.2629         12.5808
        105        10.5198       1.1026         8.3588         12.6808
        106        10.6177       1.1036         8.4547         12.7807
        107        10.7156       1.1046         8.5506         12.8806
        108        10.8135       1.1056         8.6465         12.9805
        109        10.9114       1.1067         8.7424         13.0805
        110        11.0093       1.1077         8.8383         13.1804
        111        11.1073       1.1087         8.9342         13.2803
        112        11.2052       1.1097         9.0301         13.3802
```

and so on. The forecasts are changing, because of the upward trend. As you go more than 2 steps into the future, the correlation disappears, and the accuracy of the prediction depends on how accurately you're estimating the trend. One other point about these forecasts: the trend series was analyzed by taking first differences, but the forecasts are for the *original series:* the series has been “undifferenced” to get the forecasts.

Here are the forecasts for our temperature data:

Forecasts for variable x

Obs	Forecast	Std Error	95% Confidence Limits	
121	26.8830	0.5277	25.8486	27.9173
122	27.2652	0.5494	26.1883	28.3421
123	26.9550	0.5494	25.8781	28.0319
124	26.5714	0.5494	25.4945	27.6483
125	25.8184	0.5494	24.7416	26.8953
126	24.8243	0.5494	23.7474	25.9012
127	24.1707	0.5494	23.0938	25.2476
128	24.1702	0.5494	23.0933	25.2471
129	25.1889	0.5494	24.1120	26.2658
130	26.0396	0.5494	24.9627	27.1165
131	26.3509	0.5494	25.2740	27.4278
132	26.8968	0.5494	25.8199	27.9737
133	26.9600	0.5579	25.8665	28.0535
134	27.2942	0.5586	26.1993	28.3890
135	26.9840	0.5586	25.8891	28.0789
136	26.6003	0.5586	25.5054	27.6952
137	25.8474	0.5586	24.7525	26.9423
138	24.8533	0.5586	23.7584	25.9482
139	24.1997	0.5586	23.1048	25.2945
140	24.1991	0.5586	23.1042	25.2940
141	25.2179	0.5586	24.1230	26.3127
142	26.0685	0.5586	24.9737	27.1634
143	26.3798	0.5586	25.2850	27.4747
144	26.9257	0.5586	25.8309	28.0206

These make sense: the predicted temperatures have a maximum in February and a minimum in July or August, in line with the original data. (The predictions get slightly less accurate as you go into the future.)

There are a couple of useful options on PREDICT, both related to retrospective “forecasting” of data you already have. A line like PREDICT BACK=3; starts the prediction process from the 3rd-last observation, so that for the last 3 observations you can compare the prediction with the actual value. More useful, at least in my opinion, is the PRINTALL option: you get a prediction for each observation, which you can then compare with the truth. These are “one-step ahead” predictions: a prediction for time 39 uses all the values up to time 38. Here are (some of) the predictions for the temperature data:

Forecasts for variable x

Obs	Forecast	Std Error	95% Confidence Limits		Actual	Residual
13	26.8289	0.5277	25.7946	27.8633	27.1000	0.2711
14	27.3075	0.5277	26.2731	28.3418	27.5000	0.1925
15	27.1847	0.5277	26.1504	28.2191	27.4000	0.2153
16	26.3913	0.5277	25.3569	27.4257	26.4000	0.0087
17	25.4315	0.5277	24.3971	26.4658	24.8000	-0.6315
18	23.7460	0.5277	22.7116	24.7804	24.3000	0.5540
19	23.9894	0.5277	22.9551	25.0238	23.4000	-0.5894
20	23.4582	0.5277	22.4238	24.4925	23.4000	-0.0582
21	25.3121	0.5277	24.2777	26.3464	24.6000	-0.7121
22	25.6226	0.5277	24.5883	26.6570	25.4000	-0.2226
23	26.3644	0.5277	25.3301	27.3988	25.8000	-0.5644
24	26.7654	0.5277	25.7311	27.7998	26.7000	-0.0654
25	26.8887	0.5277	25.8543	27.9230	26.9000	0.0113
26	27.3109	0.5277	26.2765	28.3453	26.3000	-1.0109
27	26.9148	0.5277	25.8804	27.9491	25.7000	-1.2148
28	26.0190	0.5277	24.9846	27.0534	25.7000	-0.3190
29	25.2501	0.5277	24.2157	26.2844	24.8000	-0.4501
30	23.8956	0.5277	22.8612	24.9300	24.3000	0.4044
31	23.8963	0.5277	22.8620	24.9307	23.4000	-0.4963
32	23.4721	0.5277	22.4377	24.5064	23.5000	0.0279
33	25.2322	0.5277	24.1978	26.2666	24.8000	-0.4322
34	25.6540	0.5277	24.6196	26.6883	25.6000	-0.0540
35	26.3268	0.5277	25.2925	27.3612	26.2000	-0.1268
36	26.8791	0.5277	25.8448	27.9135	26.5000	-0.3791
37	26.8253	0.5277	25.7910	27.8597	26.8000	-0.0253
38	27.1443	0.5277	26.1100	28.1787	26.9000	-0.2443
39	26.8891	0.5277	25.8548	27.9235	26.7000	-0.1891
40	26.2220	0.5277	25.1876	27.2563	26.1000	-0.1220
41	25.2365	0.5277	24.2022	26.2709	26.2000	0.9635
42	24.3843	0.5277	23.3499	25.4187	24.7000	0.3157
43	23.8300	0.5277	22.7956	24.8644	23.9000	0.0700
44	23.6438	0.5277	22.6095	24.6782	23.7000	0.0562
45	25.1915	0.5277	24.1571	26.2259	24.7000	-0.4915
46	25.6328	0.5277	24.5985	26.6672	25.8000	0.1672
47	26.3937	0.5277	25.3593	27.4281	26.1000	-0.2937
48	26.7834	0.5277	25.7490	27.8178	26.5000	-0.2834
49	26.8572	0.5277	25.8228	27.8916	26.3000	-0.5572
50	26.9730	0.5277	25.9386	28.0074	27.1000	0.1270
51	26.9779	0.5277	25.9436	28.0123	26.2000	-0.7779
52	26.0479	0.5277	25.0135	27.0822	25.7000	-0.3479
53	25.3703	0.5277	24.3360	26.4047	25.5000	0.1297
54	24.2808	0.5277	23.2465	25.3152	24.9000	0.6192
55	23.9765	0.5277	22.9421	25.0108	24.2000	0.2235
56	23.7313	0.5277	22.6969	24.7656	24.6000	0.8687
57	25.3686	0.5277	24.3343	26.4030	25.5000	0.1314
58	25.8468	0.5277	24.8124	26.8811	25.9000	0.0532
59	26.3446	0.5277	25.3103	27.3790	24.4000	-1.9446
60	26.2665	0.5277	25.2321	27.3008	26.9000	0.6335
61	27.0345	0.5277	26.0001	28.0688	27.1000	0.0655

The residuals as ever are the difference between the observed and predicted values. For example, at time 59, the observed mean temperature was 24.4 but the predicted one was 26.3: nearly two degrees too high.