Chapter 20
EWMACHART Statement

Chapter Table of Contents

OVERVIEW 609

GETTING STARTED 610
Creating EWMA Charts from Raw Data 610
Creating EWMA Charts from Subgroup Summary Data 612
Saving Summary Statistics 615
Saving Control Limit Parameters 616
Reading Preestablished Control Limit Parameters 618

SYNTAX 620
Summary of Options 621
Dictionary of Special Options 630

DETAILS 633
Constructing EWMA Charts 633
Output Data Sets 639
ODS Tables 642
Input Data Sets 642
Methods for Estimating the Standard Deviation 645
Axis Labels 647
Missing Values 648

EXAMPLES 649
Example 20.1 Specifying Standard Values for the Process Mean and Process Standard Deviation 649
Example 20.2 Displaying Limits Based on Asymptotic Values 650
Example 20.3 Working with Unequal Subgroup Sample Sizes 651
Example 20.4 Displaying Individual Measurements on an EWMA Chart . . . 656
Example 20.5 Computing Average Run Lengths 658

Chapter 20
EWMACHART Statement

Overview

The EWMACHART statement creates an exponentially weighted moving average (EWMA) control chart, which is used to determine whether a process is in a state of statistical control and to detect shifts in the process average.

You can use options in the EWMACHART statement to

- specify the weight assigned to the most recent subgroup mean in the computation of the EWMAs
- compute control limits from the data based on a multiple of the standard error of the plotted EWMAs or as probability limits
- tabulate the EWMAs, subgroup sample sizes, subgroup means, subgroup standard deviations, control limits, and other information
- save control limit parameters in an output data set
- save the EWMAs, subgroup sample sizes, subgroup means, and subgroup standard deviations in an output data set
- read control limit parameters from an input data set
- specify one of several methods for estimating the process standard deviation
- specify a known (standard) process mean and standard deviation for computing control limits
- display a secondary chart that plots a time trend removed from the data
- add block legends and symbol markers to reveal stratification in process data
- superimpose stars at points to represent related multivariate factors
- clip extreme points to make the chart more readable
- display vertical and horizontal reference lines
- control axis values and labels
- control layout and appearance of the chart

Getting Started

This section introduces the EWMACHART statement with simple examples that illustrate the most commonly used options. Complete syntax for the EWMACHART statement is presented in the "Syntax" section on page 620, and advanced examples are given in the "Examples" section on page 649.

Creating EWMA Charts from Raw Data

See MACEW1 in the SAS/QC Sample Library

In the manufacture of a metal clip, the gap between the ends of the clip is a critical dimension. To monitor the process for a change in the average gap, subgroup samples of five clips are selected daily. The data are analyzed with an EWMA chart. The gaps recorded during the first twenty days are saved in a SAS data set named CLIPS1.

```
data clips1;
   input day @ ;
   do i=1 to 5;
      input gap @ ;
      output;
      end;
   drop i;
   datalines;
 1  14.76  14.82  14.88  14.83  15.23
 2  14.95  14.91  15.09  14.99  15.13
 3  14.50  15.05  15.09  14.72  14.97
 4  14.91  14.87  15.46  15.01  14.99
 5  14.73  15.36  14.87  14.91  15.25
 6  15.09  15.19  15.07  15.30  14.98
 7  15.34  15.39  14.82  15.32  15.23
 8  14.80  14.94  15.15  14.69  14.93
 9  14.67  15.08  14.88  15.14  14.78
10  15.27  14.61  15.00  14.84  14.94
11  15.34  14.84  15.32  14.81  15.17
12  14.84  15.00  15.13  14.68  14.91
13  15.40  15.03  15.05  15.03  15.18
14  14.50  14.77  15.22  14.70  14.80
15  14.81  15.01  14.65  15.13  15.12
16  14.82  15.01  14.82  14.83  15.00
17  14.89  14.90  14.60  14.40  14.88
18  14.90  15.29  15.14  15.20  14.70
19  14.77  14.60  14.45  14.78  14.91
20  14.80  14.58  14.69  15.02  14.85
;
```

The following statements produce the listing of the data set CLIPS1 shown in Figure 20.1:

```
title 'The Data Set CLIPS1';
proc print data=clips1 noobs;
run;
```

```
                The Data Set CLIPS1

                   day     gap

                     1    14.76
                     1    14.82
                     1    14.88
                     1    14.83
                     1    15.23
                     2    14.95
                     2    14.91
                     2    15.09
                     2    14.99
                     2    15.13
                     .      .
                     .      .
                     .      .
                    20    14.80
                    20    14.58
                    20    14.69
                    20    15.02
                    20    14.85
```

Figure 20.1. Partial Listing of the Data Set CLIPS1

The data set CLIPS1 is said to be in “strung-out” form, since each observation contains the day and gap measurement of a single clip. The first five observations contain the gap measurements for the first day, the second five observations contain the gap measurements for the second day, and so on. Because the variable DAY classifies the observations into rational subgroups, it is referred to as the *subgroup-variable*. The variable GAP contains the gap measurements and is referred to as the *process variable* (or *process* for short).

The within-subgroup variability of the gap measurements is known to be stable. You can use an EWMA chart to determine whether the mean level is in control. The following statements create the EWMA chart shown in Figure 20.2:

```
title 'EWMA Chart for Gap Measurements';
symbol v=dot;
proc macontrol data=clips1;
   ewmachart gap*day / weight=0.3;
run;
```

This example illustrates the basic form of the EWMACHART statement. After the keyword EWMACHART, you specify the *process* to analyze (in this case, GAP) followed by an asterisk and the *subgroup-variable* (DAY). The WEIGHT= option specifies the weight parameter used to compute the EWMAs. Options such as WEIGHT= are specified after the slash (/) in the EWMACHART statement. A complete list of options is presented in the “Syntax” section on page 620. You must provide the weight parameter to create an EWMA chart. As an alternative to specifying the WEIGHT= option, you can read the weight parameter from an input data set; see “Reading Preestablished Control Limit Parameters” on page 618.

The input data set is specified with the DATA= option in the PROC MACONTROL statement.

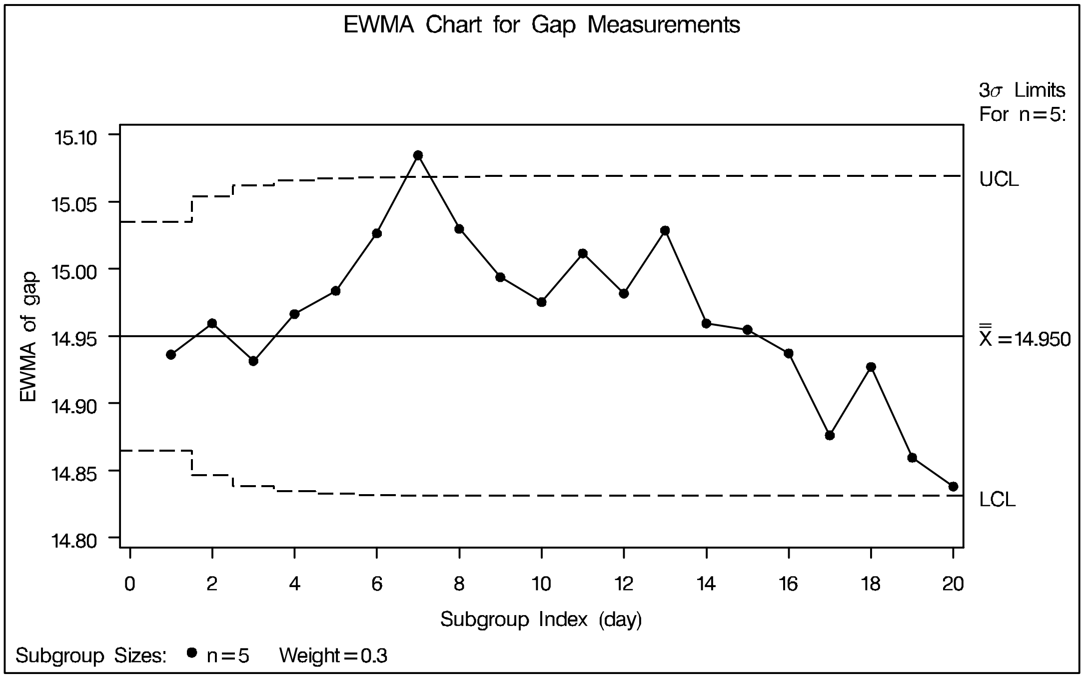


Figure 20.2. Exponentially Weighted Moving Average Chart

Each point on the chart represents the EWMA for a particular day. The EWMA E_1 plotted at DAY=1 is the weighted average of the overall mean and the subgroup mean for DAY=1. The EWMA E_2 plotted at DAY=2 is the weighted average of the EWMA E_1 and the subgroup mean for DAY=2.

$$E_1 = 0.3(14.904) + 0.7(14.952) = 14.9376\text{mm}$$
$$E_2 = 0.3(15.014) + 0.7(14.9376) = 14.9605\text{mm}$$

For succeeding days, the EWMA is the weighted average of the previous EWMA and the present subgroup mean. In the example, a weight parameter of 0.3 is used (since WEIGHT=0.3 is specified in the EWMACHART statement).

Note that the EWMA for the 7$^{\text{th}}$ day lies above the upper control limit, signaling an out-of-control process.

By default, the control limits shown are 3σ limits estimated from the data; the formulas for the limits are given in Table 20.19 on page 634.

For computational details, see “Constructing EWMA Charts” on page 633. For more details on reading from a DATA= data set, see “DATA= Data Set” on page 642.

Creating EWMA Charts from Subgroup Summary Data

See MACEW1 in the SAS/QC Sample Library

The previous example illustrates how you can create EWMA charts using raw data (process measurements). However, in many applications the data are provided as subgroup summary statistics. This example illustrates how you can use the EWMACHART statement with data of this type.

The following data set (CLIPSUM) provides the data from the preceding example in summarized form:

```
data clipsum;
   input day gapx gaps;
   gapn=5;
datalines;
 1  14.904  0.18716
 2  15.014  0.09317
 3  14.866  0.25006
 4  15.048  0.23732
 5  15.024  0.26792
 6  15.126  0.12260
 7  15.220  0.23098
 8  14.902  0.17254
 9  14.910  0.19824
10  14.932  0.24035
11  15.096  0.25618
12  14.912  0.16903
13  15.138  0.15928
14  14.798  0.26329
15  14.944  0.20876
16  14.896  0.09965
17  14.734  0.22512
18  15.046  0.24141
19  14.702  0.17880
20  14.788  0.16634
;
```

A partial listing of CLIPSUM is shown in Figure 20.3. There is exactly one observation for each subgroup (note that the subgroups are still indexed by DAY). The variable GAPX contains the subgroup means, the variable GAPS contains the subgroup standard deviations, and the variable GAPN contains the subgroup sample sizes (these are all five).

```
                    The Data Set CLIPSUM

             day     gapx       gaps     gapn

               1    14.904    0.18716      5
               2    15.014    0.09317      5
               3    14.866    0.25006      5
               .       .         .         .
               .       .         .         .
               .       .         .         .
              20    14.788    0.16634      5
```

Figure 20.3. The Summary Data Set CLIPSUM

You can read this data set by specifying it as a HISTORY= data set in the PROC MACONTROL statement, as follows:

```
title 'EWMA Chart for Gap Measurements';
proc macontrol history=clipsum lineprinter;
   ewmachart gap*day='*' / weight=0.3;
run;
```

The resulting EWMA chart is shown in Figure 20.4. Since the LINEPRINTER * option is specified in the PROC MACONTROL statement, line printer output is produced. The asterisk (*) specified in single quotes after the *subgroup-variable* indicates the character used to plot points. This character must follow an equal sign.

Note that GAP is *not* the name of a SAS variable in the data set but is, instead, the common prefix for the names of the three SAS variables GAPX, GAPS, and GAPN. The suffix characters *X*, *S*, and *N* indicate *mean*, *standard deviation*, and *sample size*, respectively. Thus, you can specify three subgroup summary variables in a HISTORY= data set with a single name (GAP), which is referred to as the *process*. The variables GAPX, GAPS, and GAPN are all required. The name DAY specified after the asterisk is the name of the *subgroup-variable*.

```
                      EWMA Chart for Gap Measurements

                                                              3 Sigma Limits
                                                              For n=5:
         ---------------------------------------------------
  15.10 +                                                   |
        |                  *                                |
        |      ===========+=+===============================| UCL
  15.05 +    ===         +  +                               |
E       |=====          *    *            *                 |
W       |              +      ++     *   + +                |
M 15.00 +              +        *+ ++ + +  +                |
A       |            +*           *    *    +               |
        |     *    *+                       *+              | =
o 14.95 +----+-++-+---------------------------+*+-----------| X = 14.950
f       |   *    *                               *    *     |
        |                                         +  + +    |
g 14.90 +                                          + +  +   |
a       |                                           *   +   |
p       |=====                                          *  |
  14.85 +    ===                                          + |
        |      ============================================*| LCL
        |                                                   |
  14.80 +                                                   |
         +----+----+----+----+----+----+----+----+----+----+
         0    2    4    6    8   10   12   14   16   18   20

                        Subgroup Index (day)

Subgroup Sizes:  * n=5   Weight=0.3
```

Figure 20.4. EWMA Chart from Summary Data

In general, a HISTORY= input data set used with the EWMACHART statement must contain the following variables:

- subgroup variable
- subgroup mean variable
- subgroup standard deviation variable
- subgroup sample size variable

*In Release 6.12 and previous releases of SAS/QC software, the keyword GRAPHICS was required in the PROC MACONTROL statement to specify that the chart be created with a graphics device. In Version 7, you can specify the LINEPRINTER option to request line printer plots.

Furthermore, the names of subgroup mean, standard deviation, and sample size variables must begin with the *process* name specified in the EWMACHART statement and end with the special suffix characters *X*, *S*, and *N*, respectively. If the names do not follow this convention, you can use the RENAME option in the PROC MACONTROL statement to rename the variables for the duration of the MACONTROL procedure step (see page 1507 for an example of the RENAME option).

In summary, the interpretation of *process* depends on the input data set.

- If raw data are read using the DATA= option (as in the previous example), *process* is the name of the SAS variable containing the process measurements.
- If summary data are read using the HISTORY= option (as in this example), *process* is the common prefix for the names of the variables containing the summary statistics.

For more information, see “HISTORY= Data Set” on page 643.

Saving Summary Statistics

See MACEW1 in the SAS/QC Sample Library

In this example, the EWMACHART statement is used to create a summary data set that can be read later by the MACONTROL procedure (as in the preceding example). The following statements read measurements from the data set CLIPS1 and create a summary data set named CLIPHIST:

```
title 'Summary Data Set for Gap Measurements';
proc macontrol data=clips1;
   ewmachart gap*day / weight     = 0.3
                       outhistory = cliphist
                       nochart;
run;
```

The OUTHISTORY= option names the output data set, and the NOCHART option suppresses the display of the chart, which would be identical to the chart in Figure 20.2.

Figure 20.5 contains a partial listing of CLIPHIST.

```
                Summary Data Set for Gap Measurements

             day     gapX       gapS       gapE     gapN

               1    14.904    0.18716    14.9362      5
               2    15.014    0.09317    14.9595      5
               3    14.866    0.25006    14.9315      5
               4    15.048    0.23732    14.9664      5
               5    15.024    0.26792    14.9837      5
               .       .         .          .         .
               .       .         .          .         .
               .       .         .          .         .
              20    14.788    0.16634    14.8381      5
```

Figure 20.5. The Summary Data Set CLIPHIST

There are five variables in the data set CLIPHIST.

- DAY contains the subgroup index.
- GAPX contains the subgroup means.
- GAPS contains the subgroup standard deviations.
- GAPE contains the subgroup exponentially weighted moving averages.
- GAPN contains the subgroup sample sizes.

Note that the summary statistic variables are named by adding the suffix characters *X*, *S*, *E*, and *N* to the *process* GAP specified in the EWMACHART statement. In other words, the variable naming convention for OUTHISTORY= data sets is the same as that for HISTORY= data sets.

For more information, see “OUTHISTORY= Data Set” on page 640.

Saving Control Limit Parameters

See MACEW1 in the SAS/QC Sample Library

You can save the control limit parameters for an EWMA chart in a SAS data set; this enables you to use these parameters with future data (see “Reading Preestablished Control Limit Parameters” on page 618) or modify the parameters with a DATA step program.

The following statements read measurements from the data set CLIPS1 (see page 610) and save the control limit parameters in a data set named CLIPLIM:

```
title 'Control Limit Parameters';
proc macontrol data=clips1;
   ewmachart gap*day / weight    = 0.3
                       outlimits = cliplim
                       nochart;
run;
```

The OUTLIMITS= option names the data set containing the control limit parameters, and the NOCHART option suppresses the display of the chart. The data set CLIPLIM is listed in Figure 20.6.

Control Limit Parameters

VAR	_SUBGRP_	_TYPE_	_LIMITN_	_ALPHA_	_SIGMAS_	_MEAN_	_STDDEV_	_WEIGHT_
gap	day	ESTIMATE	5	.002699796	3	14.95	0.21108	0.3

Figure 20.6. The Data Set CLIPLIM Containing Control Limit Information

Note that the data set CLIPLIM does not contain the actual control limits but rather the parameters required to compute the limits.

The data set contains one observation with the parameters for *process* GAP. The variable _WEIGHT_ contains the weight parameter used to compute the EWMAs. The value of _MEAN_ is an estimate of the process mean, and the value of _STDDEV_ is an estimate of the process standard deviation σ. The value of _LIMITN_ is the nominal sample size associated with the control limits, and the value of _SIGMAS_ is the multiple of σ associated with the control limits. The variables _VAR_ and _SUBGRP_ are bookkeeping variables that save the *process* and *subgroup-variable*. The variable _TYPE_ is a bookkeeping variable that indicates that the values of _MEAN_ and _STDDEV_ are estimates rather than standard values. For more information, see "OUTLIMITS= Data Set" on page 639.

You can create an output data set containing the control limits and summary statistics with the OUTTABLE= option, as illustrated by the following statements:

```
title 'Summary Statistics and Control Limits';
proc macontrol data=clips1;
   ewmachart gap*day / weight   = 0.3
                       outtable = cliptab
                       nochart;
run;
```

The data set CLIPTAB is listed in Figure 20.7.

Summary Statistics and Control Limits

VAR	day	_SIGMAS_	_LIMITN_	_WEIGHT_	_SUBN_	_SUBX_	_SUBS_	_LCLE_	_EWMA_	_MEAN_	_UCLE_	_EXLIM_
gap	1	3	5	0.3	5	14.904	0.18716	14.8650	14.9362	14.95	15.0350	
gap	2	3	5	0.3	5	15.014	0.09317	14.8463	14.9595	14.95	15.0537	
gap	3	3	5	0.3	5	14.866	0.25006	14.8383	14.9315	14.95	15.0617	
gap	4	3	5	0.3	5	15.048	0.23732	14.8345	14.9664	14.95	15.0655	
gap	5	3	5	0.3	5	15.024	0.26792	14.8327	14.9837	14.95	15.0673	
gap	6	3	5	0.3	5	15.126	0.12260	14.8319	15.0264	14.95	15.0681	
gap	7	3	5	0.3	5	15.220	0.23098	14.8314	15.0845	14.95	15.0686	UPPER
gap	8	3	5	0.3	5	14.902	0.17254	14.8312	15.0297	14.95	15.0688	
gap	9	3	5	0.3	5	14.910	0.19824	14.8311	14.9938	14.95	15.0689	
gap	10	3	5	0.3	5	14.932	0.24035	14.8311	14.9753	14.95	15.0689	
gap	11	3	5	0.3	5	15.096	0.25618	14.8311	15.0115	14.95	15.0689	
gap	12	3	5	0.3	5	14.912	0.16903	14.8310	14.9816	14.95	15.0690	
gap	13	3	5	0.3	5	15.138	0.15928	14.8310	15.0285	14.95	15.0690	
gap	14	3	5	0.3	5	14.798	0.26329	14.8310	14.9594	14.95	15.0690	
gap	15	3	5	0.3	5	14.944	0.20876	14.8310	14.9548	14.95	15.0690	
gap	16	3	5	0.3	5	14.896	0.09965	14.8310	14.9371	14.95	15.0690	
gap	17	3	5	0.3	5	14.734	0.22512	14.8310	14.8762	14.95	15.0690	
gap	18	3	5	0.3	5	15.046	0.24141	14.8310	14.9271	14.95	15.0690	
gap	19	3	5	0.3	5	14.702	0.17880	14.8310	14.8596	14.95	15.0690	
gap	20	3	5	0.3	5	14.788	0.16634	14.8310	14.8381	14.95	15.0690	

Figure 20.7. The OUTTABLE= Data Set CLIPTAB

This data set contains one observation for each subgroup sample. The variable _EWMA_ contains the EWMAs. The variables _SUBX_, _SUBS_, and _SUBN_ contain the subgroup means, subgroup standard deviations, and subgroup sample sizes, respectively. The variables _LCLE_ and _UCLE_ contain the lower and upper control limits, and the variable _MEAN_ contains the central line. The variables

VAR and DAY contain the *process* name and values of the *subgroup-variable*, respectively. For more information, see “OUTTABLE= Data Set” on page 640.

An OUTTABLE= data set can be read later as a TABLE= data set. For example, the following statements read CLIPTAB and display a EWMA chart (not shown here) identical to Figure 20.2:

```
title 'EWMA Chart for Gap Measurements';
proc macontrol table=cliptab;
   ewmachart gap*day ;
run;
```

For more information, see “TABLE= Data Set” on page 644.

Reading Preestablished Control Limit Parameters

See MACEW1 in the SAS/QC Sample Library

In the previous example, the OUTLIMITS= data set saved the control limit parameters in the data set CLIPLIM. This example shows how to apply these parameters to new data provided in the following data set:

```
data clips1a;
   label gap='Gap Measurement (mm)';
   input day @;
   do i=1 to 5;
      input gap @;
      output;
      end;
   drop i;
datalines;
21  14.86 15.01 14.67 14.67 15.07
22  14.93 14.53 15.07 15.10 14.98
23  15.27 14.90 15.12 15.10 14.80
24  15.02 15.21 14.93 15.11 15.20
25  14.90 14.81 15.26 14.57 14.94
26  14.78 15.29 15.13 14.62 14.54
27  14.78 15.15 14.61 14.92 15.07
28  14.92 15.31 14.82 14.74 15.26
29  15.11 15.04 14.61 15.09 14.68
30  15.00 15.04 14.36 15.20 14.65
31  14.99 14.76 15.18 15.04 14.82
32  14.90 14.78 15.19 15.06 15.06
33  14.95 15.10 14.86 15.27 15.22
34  15.03 14.71 14.75 14.99 15.02
35  15.38 14.94 14.68 14.77 14.83
36  14.95 15.43 14.87 14.90 15.34
37  15.18 14.94 15.32 14.74 15.29
38  14.91 15.15 15.06 14.78 15.42
39  15.34 15.34 15.41 15.36 14.96
40  15.12 14.75 15.05 14.70 14.74
;
```

The following statements create an EWMA chart for the data in CLIPS1A using the control limit parameters in CLIPLIM:

```
title 'EWMA Chart for Second Set of Gap Measurements';
symbol v=dot;
proc macontrol data=clips1a limits=cliplim;
   ewmachart gap*day;
run;
```

The chart is shown in Figure 20.8.

The LIMITS= option in the PROC MACONTROL statement specifies the data set containing the control limit parameters. By default,* this information is read from the first observation in the LIMITS= data set for which

- the value of _VAR_ matches the *process* name GAP
- the value of _SUBGRP_ matches the *subgroup-variable* name DAY

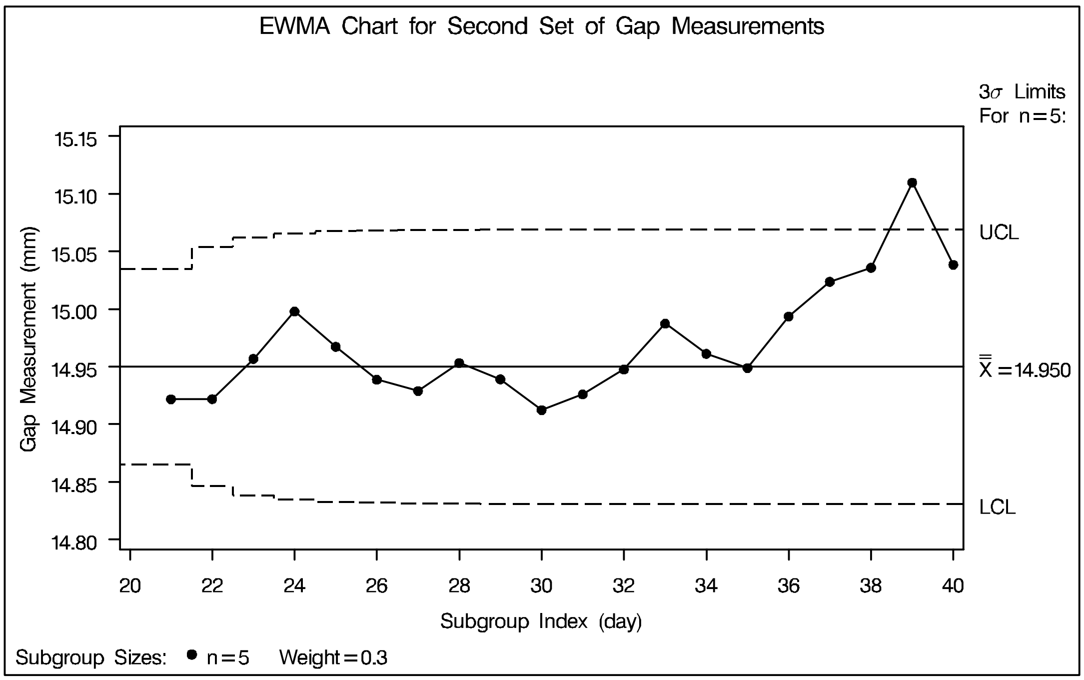


Figure 20.8. EWMA Chart Using Preestablished Control Limit Parameters

Note that the EWMA plotted for the 39th day lies above the upper control limit, signalling an out-of-control process.

In this example, the LIMITS= data set was created in a previous run of the MACONTROL procedure. You can also create a LIMITS= data set with the DATA step. See "LIMITS= Data Set" on page 643 for details concerning the variables that you must provide, and see Example 20.1 on page 649 for an illustration.

*In Release 6.09 and in earlier releases, it is also necessary to specify the READLIMITS option to read control limits from a LIMITS= data set.

Syntax

The basic syntax for the EWMACHART statement is as follows:

EWMACHART *process******subgroup-variable* **/ WEIGHT=***value* < *options* > **;**

The general form of this syntax is as follows:

EWMACHART (*processes***)***subgroup-variable* <**(** *block-variables* **)** >
< =*symbol-variable* | =**'***character***'** > **/ WEIGHT=***value* < *options* >**;**

Note that the WEIGHT= option is required unless its *value* is read from a LIMITS= data set. You can use any number of EWMACHART statements in the MACONTROL procedure. The components of the EWMACHART statement are described as follows.

process
processes
identify one or more processes to be analyzed. The specification of *process* depends on the input data set specified in the PROC MACONTROL statement.

- If raw data are read from a DATA= data set, *process* must be the name of the variable containing the raw measurements. For an example, see "Creating EWMA Charts from Raw Data" on page 610.
- If summary data are read from a HISTORY= data set, *process* must be the common prefix of the summary variables in the HISTORY= data set. For an example, see "Creating EWMA Charts from Subgroup Summary Data" on page 612.
- If summary data and control limits are read from a TABLE= data set, *process* must be the value of the variable _VAR_ in the TABLE= data set. For an example, see "Saving Control Limit Parameters" on page 616.

A *process* is required. If more than one *process* is specified, enclose the list in parentheses. For example, the following statements request distinct EWMA charts (each using a weight parameter of 0.3) for WEIGHT, LENGTH, and WIDTH:

```
proc macontrol data=measures;
   ewmachart (weight length width)*day / weight=0.3;
run;
```

subgroup-variable
is the variable that classifies the data into subgroups. The *subgroup-variable* is required. In the preceding EWMACHART statement, DAY is the subgroup variable. For details, see "Subgroup Variables" on page 1534.

block-variables
are optional variables that group the data into blocks of consecutive subgroups. The blocks are labeled in a legend, and each *block-variable* provides one level of labels in the legend. See "Displaying Stratification in Blocks of Observations" on page 1684 for an example.

symbol-variable
is an optional variable whose levels (unique values) determine the symbol marker or plotting character used to plot the EWMAs.

- If you produce a chart on a line printer, an 'A' is displayed for the points corresponding to the first level of the *symbol-variable*, a 'B' is displayed for the points corresponding to the second level, and so on.
- If you produce a chart on a graphics device, distinct symbol markers are displayed for points corresponding to the various levels of the *symbol-variable*. You can specify the symbol markers with SYMBOLn statements. See "Displaying Stratification in Levels of a Classification Variable" on page 1683 for an example.

character
specifies a plotting character for charts produced on line printers. For example, the following statements create an EWMA chart using an asterisk (*) to plot the points:

```
proc macontrol data=values;
   ewmachart length*hour='*' / weight=0.3;
run;
```

options
specify chart parameters, enhance the appearance of the chart, request additional analyses, save results in data sets, and so on. The "Summary of Options" section, which follows, lists all options by function.

Summary of Options

The following tables list the EWMACHART statement options by function. Options unique to the MACONTROL procedure are listed in Table 20.1 and Table 20.2, and they are described in detail in "Dictionary of Special Options" on page 630. Options that are common to both the MACONTROL and SHEWHART procedures are listed in Table 20.3 to Table 20.18. They are described in detail beginning on page 1613 of Part 9, "The SHEWHART Procedure."

Table 20.1. Options for Specifying Exponentially Weighted Moving Average Charts*

ALPHA=*value*	requests probability limits for control charts
ASYMPTOTIC	requests constant control limits based on asymptotic expressions
LIMITN=n|VARYING	specifies either a fixed nominal sample size (n) for control limits or allows the control limits to vary with subgroup sample size
MU0=*value*	specifies a standard (known) value μ_0 for the process mean
NOREADLIMITS	specifies that control limit parameters are not to be read from a LIMITS= data set (Release 6.10 and later releases)
READALPHA	reads _ALPHA_ instead of _SIGMAS_ from the LIMITS= data set when both variables are available
READINDEX=*'value'*	reads control limit parameters from the first observation in the LIMITS= data set where the variable _INDEX_ equals *value*
READLIMITS	reads control limit parameters from a LIMITS= data set (Release 6.09 and earlier releases)
RESET	requests that the value of the EWMA be reset after each out-of-control point
SIGMA0=*value*	specifies standard (known) value σ_0 for process standard deviation
SIGMAS=k	specifies width of control limits in terms of multiple k of standard error of plotted EWMAs
WEIGHT=*value*	specifies weight assigned to the most recent subgroup mean in the computation of the EWMA

Table 20.2. Options for Plotting Subgroup Means*

CMEANSYMBOL=*color*	specifies color for MEANSYMBOL= symbol
MEANCHAR='*character*'	specifies *character* to plot subgroup means on line printer
MEANSYMBOL= *keyword*	specifies symbol to plot subgroup means on graphics device

*The options in these tables are described in “Dictionary of Special Options” on page 630.

Table 20.3. Tabulation Options†

TABLE	creates a basic table of subgroup variable values, subgroup sample sizes, subgroup means, subgroup EWMAs, and control limits
TABLEALL	equivalent to the options TABLE, TABLECENTRAL, TABLEID, and TABLEOUT
TABLECENTRAL	augments basic table with the value of the central line
TABLEID	augments basic table with columns for ID variables
TABLEOUTLIM	augments basic table with columns indicating control limits exceeded

Note that specifying (EXCEPTIONS) after a tabulation option creates a table for exceptional points.

Table 20.4. Axis and Axis Label Options†

CAXIS=*color*	specifies color for axis lines and tick marks
CFRAME=*color*|(*color-list*)	specifies fill colors for frame for plot area
CTEXT=*color*	specifies color for tick mark values and axis labels
HAXIS=*values*|AXIS*n*	specifies major tick mark values for horizontal axis
HEIGHT=*value*	specifies height of axis label and axis legend text
HMINOR=*n*	specifies minor tick marks between major horizontal tick marks
HOFFSET=*value*	specifies length of offset at both ends of horizontal axis
INTSTART=*value*	specifies first major tick mark value for numeric horizontal axis
NOHLABEL	suppresses label for horizontal axis
NOVANGLE	requests vertical axis labels that are strung out vertically
SKIPHLABELS=*n*	specifies thinning factor for tick mark labels on horizontal axis
SPLIT='*character*'	specifies splitting character for axis labels
TURNHLABELS	requests horizontal axis labels that are strung out vertically
VAXIS=*values*|AXIS*n*	specifies major tick mark values for vertical axis on EWMA chart
VAXIS2=*values*|AXIS*n*	specifies major tick mark values for vertical axis on trend chart
VMINOR=*n*	specifies minor tick marks between major vertical tick marks
VOFFSET=*value*	specifies length of offset at both ends of vertical axis
WAXIS=*n*	specifies width of axis lines

Table 20.5. Process Mean and Standard Deviation Options†

SMETHOD=*keyword*	specifies method for estimating process standard deviation σ
TYPE=*keyword*	identifies whether parameters are estimates or standard values and specifies value of _TYPE_ in OUTLIMITS= data set

†The options in these tables are described in Chapter 46, "Dictionary of Options," of Part 9, "The SHEWHART Procedure."

Table 20.6. Grid Options†

ENDGRID	adds grid after last plotted point
GRID	adds grid to chart
LENDGRID=*linetype*	specifies line type for grid requested with the ENDGRID option
LGRID=*linetype*	specifies line type for grid requested with the GRID option
WGRID=*n*	specifies width of grid lines

Table 20.7. Reference Line Options†

CHREF=*color*	specifies color for HREF= and HREF2= lines
CVREF=*color*	specifies color for VREF= and VREF2= lines
HREF=*values*| *SAS-data-set*	specifies reference lines perpendicular to horizontal axis on EWMA chart
HREF2=*values*| *SAS-data-set*	specifies reference lines perpendicular to horizontal axis on trend chart
HREFCHAR='*character*'	specifies line character for HREF= and HREF2= lines
HREFDATA= *SAS-data-set*	specifies position of reference lines perpendicular to horizontal axis on EWMA chart
HREF2DATA= *SAS-data-set*	specifies position of reference lines perpendicular to horizontal axis on trend chart
HREFLABELS= *'label1'...'labeln'*	specifies labels for HREF= lines
HREF2LABELS= *'label1'...'labeln'*	specifies labels for HREF2= lines
HREFLABPOS=*n*	specifies position of HREFLABELS= and HREF2LABELS= labels
LHREF=*linetype*	specifies line type for HREF= and HREF2= lines
LVREF=*linetype*	specifies line type for VREF= and VREF2= lines
NOBYREF	specifies that reference line information in a data set is to be applied uniformly to charts created for all BY groups
VREF=*values*| *SAS-data-set*	specifies reference lines perpendicular to vertical axis on EWMA chart
VREF2=*values*| *SAS-data-set*	specifies reference lines perpendicular to vertical axis on trend chart
VREFCHAR='*character*'	specifies line character for VREF= and VREF2= lines
VREFLABELS= *'label1'...'labeln'*	specifies labels for VREF= lines
VREF2LABELS= *'label1'...'labeln'*	specifies labels for VREF2= lines
VREFLABPOS=*n*	specifies position of VREFLABELS= and VREF2LABELS= labels

†The options in these tables are described in Chapter 46, "Dictionary of Options," of Part 9, "The SHEWHART Procedure."

Table 20.8. Block Variable Legend Options†

BLOCKLABELPOS= *keyword*	specifies position of label for *block-variable* legend
BLOCKLABTYPE= *value*|*keyword*	specifies text size of *block-variable* legend
BLOCKPOS=*n*	specifies vertical position of *block-variable* legend
BLOCKREP	repeats identical consecutive labels in *block-variable* legend
CBLOCKLAB=*color*	specifies color for filling background in *block-variable* legend
CBLOCKVAR=*variable*| (*variables*)	specifies one or more variables whose values are colors for filling background of *block-variable* legend

Table 20.9. Options for Displaying Control Limits†

CINFILL=*color*	specifies color for area inside control limits
CLIMITS=*color*	specifies color of control limits, central line, and related labels
LCLLABEL='*label*'	specifies label for lower control limit
LIMLABSUBCHAR= '*character*'	specifies a substitution character for labels provided as quoted strings; the character is replaced with the value of the control limit
LLIMITS=*linetype*	line type for control limits
NDECIMAL=*n*	specifies number of digits to right of decimal place in default labels for control limits and central line
NOCTL	suppresses display of central line
NOLCL	suppresses display of lower control limit
NOLIMITLABEL	suppresses labels for control limits and center line
NOLIMITS	suppresses display of control limits
NOLIMITSLEGEND	suppresses legend for control limits
NOUCL	suppresses display of upper control limit
UCLLABEL='*string*'	specifies label for upper control limit
WLIMITS=*n*	width for control limits and central line
XSYMBOL='*string*'| *keyword*	specifies label for central line

Table 20.10. Options for Interactive Control Charts†

HTML=(*variable*)	specifies a variable whose values are URLs to be associated with subgroups
HTML_LEGEND= (*variable*)	specifies a variable whose values are URLs to be associated with symbols in the symbol legend
WEBOUT=*SAS-data-set*	creates an OUTTABLE= data set with additional graphics coordinate data

†The options in these tables are described in Chapter 46, "Dictionary of Options," of Part 9, "The SHEWHART Procedure."

Table 20.11. Options for Plotting and Labeling Points[†]

ALLLABEL=VALUE|(*variable*)	labels every point on EWMA chart
ALLLABEL2=VALUE|(*variable*)	labels every point on trend chart
CCONNECT=*color*	specifies color for line segments that connect points on chart
CFRAMELAB=*color*	specifies fill color for frame around labeled points
CNEEDLES=*color*	specifies color for needles that connect points to central line
CONNECTCHAR='*character*'	specifies character used to form line segments that connect points on EWMA chart
COUT=*color*	specifies color for line segments that connect points exceeding control limits
COUTFILL=*color*	specifies color for areas between connected points and control limits
LABELFONT=*font*	specifies a software font for labels requested by the ALLLABEL=, ALLLABEL2=, OUTLABEL=, and STARLABEL= options
LABELHEIGHT=*font*	specifies the height (in vertical percent screen units) for labels requested by the ALLLABEL=, ALLLABEL2=, OUTLABEL=, and STARLABEL= options
NEEDLES	connects points to central line with vertical needles
NOCONNECT	suppresses line segments that connect points on EWMA chart
NOTRENDCONNECT	suppresses line segments that connect points on trend chart
OUTLABEL=VALUE|(*variable*)	labels points exceeding control limits
SYMBOLCHARS='*characters*'	specifies characters indicating *symbol-variable*
SYMBOLLEGEND=NONE|*name*	specifies LEGEND statement for levels of *symbol-variable*
SYMBOLORDER=*keyword*	specifies order in which symbols are assigned for levels of *symbol-variable*
TURNALL	turns point labels so that they are strung out vertically

[†]The options in these tables are described in Chapter 46, "Dictionary of Options," of Part 9, "The SHEWHART Procedure."

Table 20.12. Input Data Set Options†

MISSBREAK	specifies that observations with missing values are not to be processed

Table 20.13. Output Data Set Options†

OUTHISTORY= *SAS-data-set*	creates output data set containing subgroup summary statistics
OUTINDEX='*string*'	specifies value of the variable _INDEX_ in OUTLIMITS= data set
OUTLIMITS= *SAS-data-set*	creates output data set containing control limit parameters
OUTPHASE='*string*'	specifies value of the variable _PHASE_ in OUTHISTORY= or OUTTABLE= data set
OUTTABLE= *SAS-data-set*	creates output data set containing subgroup summary statistics and control limits

Table 20.14. Plot Layout Options†

ALLN	plots EWMA for all subgroups
BILEVEL	creates control charts using half-screens and half-pages
EXCHART	creates control charts only when exceptions occur
INTERVAL=*keyword*	specifies natural time interval between consecutive subgroup positions when time, date, or datetime format is associated with a numeric subgroup variable
MAXPANELS=*n*	specifies maximum number of pages or screens for chart
NMARKERS	requests special markers for points corresponding to sample sizes not equal to nominal sample size for fixed control limits
NOCHART	suppresses creation of EWMA chart
NOFRAME	suppresses frame for plot area
NOLEGEND	suppresses legend for subgroup sample sizes
NPANELPOS=*n*	specifies number of subgroup positions per panel on each chart
REPEAT	repeats last subgroup position on panel as first subgroup position of next panel
TOTPANELS=*n*	specifies number of pages or screens to be used to display chart
TRENDVAR=*variable*| (*variable-list*)	specifies list of trend variables
YPCT1=*value*	specifies length of vertical axis on EWMA chart as a percentage of sum of lengths of vertical axes for EWMA and trend charts
ZEROSTD	displays $\bar{X}$ chart regardless of whether $\hat{\sigma} = 0$

†The options in these tables are described in Chapter 46, "Dictionary of Options," of Part 9, "The SHEWHART Procedure."

Table 20.15. Phase Options†

CPHASELEG=*color*	specifies text color for *phase* legend
OUTPHASE=’*string*’	specifies value of _PHASE_ in OUTHISTORY= data set
PHASEBREAK	disconnects last point in a *phase* from first point in next *phase*
PHASELABTYPE=*value*| *keyword*	specifies text size of *phase* legend
PHASELEGEND	displays *phase* labels in a legend across top of chart
PHASEREF	delineates *phases* with vertical reference lines
READPHASES= ALL| ’*label1*’...’*labeln*’	specifies *phases* to be read from input data set

Table 20.16. Graphical Enhancement Options†

ANNOTATE= *SAS-data-set*	specifies annotate data set that adds features to EWMA chart
ANNOTATE2= *SAS-data-set*	specifies annotate data set that adds features to trend chart
DESCRIPTION=’*string*’	specifies string that appears in the description field of PROC GREPLAY master menu for EWMA chart
FONT=*font*	specifies software font for labels and legends on chart
NAME=’*string*’	specifies name that appears in the name field of the PROC GREPLAY master menu for EWMA chart
PAGENUM=’*string*’	specifies the form of the label used in pagination
PAGENUMPOS= *keyword*	specifies the position of the page number requested with the PAGENUM= option
WTREND=*n*	specifies width of line segments connecting points on trend chart

Table 20.17. Clipping Options†

CCLIP=*color*	color for plot symbol for clipped points
CLIPCHAR=’*character*’	plot character for clipped points
CLIPFACTOR=*value*	determines extent to which extreme points are clipped
CLIPLEGEND=’*string*’	text for clipping legend
CLIPLEGPOS=*keyword*	position of clipping legend
CLIPSUBCHAR= ’*character*’	substitution character for CLIPLEGEND= text
CLIPSYMBOL=*symbol*	plot symbol for clipped points

†The options in these tables are described in Chapter 46, “Dictionary of Options,” of Part 9, “The SHEWHART Procedure.”

Table 20.18. Star Options[†]

CSTARCIRCLES=*color*	specifies color for STARCIRCLES= circles
CSTARFILL=*color*| *(variable)*	specifies color for filling stars
CSTAROUT=*color*	specifies outline color for stars exceeding inner or outer circles
CSTARS=*color*| *(variable)*	specifies color for outlines of stars
LSTARCIRCLES= *linetypes*	specifies line types for STARCIRCLES= circles
LSTARS=*linetype*| *(variable)*	specifies line types for outlines of stars requested with the STARVERTICES= option
STARBDRADIUS=*value*	specifies radius of outer bound circle for vertices of stars
STARCIRCLES=*value-list*	specifies reference circles for stars
STARINRADIUS=*value*	specifies inner radius of stars
STARLABEL=*keyword*	specifies vertices to be labeled
STARLEGEND=*keyword*	specifies style of legend for star vertices
STARLEGENDLAB= *'label'*	specifies label for STARLEGEND= legend
STAROUTRADIUS=*value*	specifies outer radius of stars
STARSPEC=*value*| *SAS-data-set*	specifies method used to standardize vertex variables
STARSTART=*value*	specifies angle for first vertex
STARTYPE=*keyword*	specifies graphical style of star
STARVERTICES= *variable*|*(variables)*	superimposes star at each point on EWMA chart
WSTARCIRCLES=*n*	specifies width of STARCIRCLES= circles
WSTARS=*n*	specifies width of STARVERTICES= stars

[†]The options in these tables are described in Chapter 46, "Dictionary of Options," of Part 9, "The SHEWHART Procedure."

Dictionary of Special Options

The marginal notes *Graphics* and *Line Printer* identify options that apply to graphics devices and line printers, respectively.

ALPHA=*value*

requests *probability limits*. If you specify ALPHA=α, the control limits are computed so that the probability is α that a single EWMA exceeds its control limits. The value of α can range between 0 and 1. This assumes that the process is in statistical control and that the data follow a normal distribution. For the equations used to compute probability limits, see "Control Limits" on page 634.

Note the following:

- As an alternative to specifying ALPHA=α, you can read α from the variable _ALPHA_ in a LIMITS= data set by specifying the READALPHA option.
- As an alternative to specifying ALPHA=α (or reading _ALPHA_ from a LIMITS= data set), you can request "$k\sigma$ control limits" by specifying SIGMAS=k (or reading _SIGMAS_ from a LIMITS= data set).

If you specify neither the ALPHA= option nor the SIGMAS= option, the procedure computes 3σ control limits by default.

ASYMPTOTIC

requests constant upper and lower control limits based on the following asymptotic expressions:

$$\text{LCL} = \overline{\overline{X}} - k\hat{\sigma}\sqrt{r/n(2-r)}$$
$$\text{UCL} = \overline{\overline{X}} + k\hat{\sigma}\sqrt{r/n(2-r)}$$

Here r is the weight parameter $(0 < r \leq 1)$, and n is the nominal sample size associated with the control limits. Substitute $\Phi^{-1}(1-\alpha/2)$ for k if you specify probability limits with the ALPHA= option. When you do not specify the ASYMPTOTIC option, the control limits are computed using the exact formulas in Table 20.19 on page 634. Use the ASYMPTOTIC option only if all the subgroup sample sizes are the same or if you specify LIMITN=n. See Example 20.2 on page 650.

CMEANSYMBOL=*color*

Graphics

specifies the *color* for the symbol requested with the MEANSYMBOL= option. The default *color* is the first color in the device color list.

LIMITN=n
LIMITN=VARYING

specifies either a fixed or varying nominal sample size for the control limits.

If you specify LIMITN=n, EWMAs are calculated and displayed only for those subgroups with a sample size equal to n, unless you also specify the ALLN option, which causes all the EWMAs to be calculated and displayed. By default (or if you specify LIMITN=VARYING), EWMAs are calculated and displayed for all subgroups, regardless of sample size.

MEANCHAR='*character*'

specifies a *character* used to plot the subgroup mean for each subgroup. By default, subgroup means are not plotted. *Line Printer*

MEANSYMBOL=*keyword*

specifies a symbol used to plot the subgroup mean for each subgroup. By default, subgroup means are not plotted. *Graphics*

MU0=*value*

specifies a known (standard) value μ_0 for the process mean μ. By default, μ is estimated from the data. See Example 20.1 on page 649.

Note: As an alternative to specifying MU0=μ_0, you can read a predetermined value for μ_0 from the variable _MEAN_ in a LIMITS= data set.

NOREADLIMITS

specifies that control limit parameters for each *process* listed in the EWMACHART statement are *not* to be read from the LIMITS= data set specified in the PROC MACONTROL statement. The NOREADLIMITS option is available only in Release 6.10 and later releases.

The following example illustrates the NOREADLIMITS option:

```
proc macontrol data=pistons limits=diamlim;
   ewmachart diameter*hour;
   ewmachart diameter*hour / noreadlimits weight=0.3;
run;
```

The first EWMACHART statement reads the control limits from the first observation in the data set DIAMLIM for which the variable _VAR_ is equal to `diameter` and the variable _SUBGRP_ is equal to `hour`. The second EWMACHART statement computes estimates of the process mean and standard deviation for the control limits from the measurements in the data set PISTONS. Note that the second EWMACHART statement is equivalent to the following statements, which would be more commonly used:

```
proc macontrol data=pistons;
   ewmachart diameter*hour / weight=0.3;
run;
```

For more information about reading control limit parameters from a LIMITS= data set, see the READLIMITS option later in this list.

READALPHA

specifies that the variable _ALPHA_, rather than the variable _SIGMAS_, is to be read from a LIMITS= data set when both variables are available in the data set. Thus the limits displayed are probability limits. If you do not specify the READALPHA option, then _SIGMAS_ is read by default.

READINDEX='*value*'

reads control limit parameters from a LIMITS= data set (specified in the PROC MACONTROL statement) for each *process* listed in the EWMACHART statement.

The control limit parameters for a particular *process* are read from the first observation in the LIMITS= data set for which

- the value of _VAR_ matches *process*
- the value of _SUBGRP_ matches the *subgroup-variable*
- the value of _INDEX_ matches *value*

The *value* can be up to 16 characters and must be enclosed in quotes.

READLIMITS

specifies that control limit parameters are to be read from a LIMITS= data set specified in the PROC MACONTROL statement. The parameters for a particular *process* are read from the first observation in the LIMITS= data set for which

- the value of _VAR_ matches *process*
- the value of _SUBGRP_ matches the *subgroup variable*

The use of the READLIMITS option depends on which release of SAS/QC software you are using.

- **In Release 6.10 and later releases, the READLIMITS option is not necessary.** To read control limits parameters as described previously, you simply specify a LIMITS= data set. However, even though the READLIMITS option is redundant, it continues to function as in earlier releases.

- **In Release 6.09 and earlier releases, you must specify the READLIMITS option to read control limits parameters as described previously.** If you specify a LIMITS= data set without specifying the READLIMITS option (or the READINDEX= option), the control limits are computed from the data and the value of the weight parameter is specified with the WEIGHT= option.

RESET

requests that the value of the EWMA be reset after each out-of-control point. Specifically, when a point exceeds the control limits, the EWMA for the next subgroup is computed as the weighted average of the subgroup mean and the overall mean. By default, the EWMAs are not reset.

SIGMA0=*value*

specifies a known (standard) value σ_0 for the process standard deviation σ. The *value* must be positive. By default, the MACONTROL procedure estimates σ from the data using the formulas given in "Methods for Estimating the Standard Deviation" on page 645.

Note: As an alternative to specifying SIGMA0=σ_0, you can read a predetermined value for σ_0 from the variable _STDDEV_ in a LIMITS= data set.

SIGMAS=*value*

specifies the width of the control limits in terms of the multiple k of the standard error of the plotted EWMAs on the chart. The value of k must be positive. By default, $k = 3$ and the control limits are 3σ limits.

WEIGHT=*value*

specifies the weight r assigned to the most recent subgroup mean in the computation of the EWMA ($0 < r \leq 1$). The WEIGHT= option is required unless you read control limit parameters from a LIMITS= data set or a TABLE= data set. See "Choosing the Value of the Weight Parameter" on page 635 for details.

Details

Constructing EWMA Charts

The following notation is used in this section:

E_i	exponentially weighted moving average for the i^{th} subgroup
r	EWMA weight parameter $(0 < r \leq 1)$
μ	process mean (expected value of the population of measurements)
σ	process standard deviation (standard deviation of the population of measurements)
x_{ij}	j^{th} measurement in i^{th} subgroup, with j =1, 2, 3, $\ldots$, n_i
n_i	sample size of i^{th} subgroup
$\overline{X}_i$	mean of measurements in i^{th} subgroup. If $n_i = 1$, then the subgroup mean reduces to the single observation in the subgroup
$\overline{\overline{X}}$	weighted average of subgroup means
$\Phi^{-1}(\cdot)$	inverse standard normal function

Plotted Points

Each point on the chart indicates the value of the exponentially weighted moving average (EWMA) for that subgroup. The EWMA for the i^{th} subgroup (E_i) is defined recursively as

$$E_i = r\overline{X}_i + (1-r)E_{i-1}\,, \quad i > 0$$

where r is a weight parameter $(0 < r \leq 1)$. Some authors (for example, Hunter 1986 and Crowder 1987a,b) use the symbol λ instead of r for the weight. You can specify the weight with the WEIGHT= option in the EWMACHART statement or with the variable _WEIGHT_ in a LIMITS= data set. If you specify a known value (μ_0) for μ, $E_0 = \mu_0$; otherwise, $E_0 = \overline{\overline{X}}$.

The preceding equation can be rewritten as

$$E_i = E_{i-1} + r(\overline{X}_i - E_{i-1})$$

which expresses the current EWMA as the previous EWMA plus the weighted error in the prediction of the current mean based on the previous EWMA.

The EWMA for the i^{th} subgroup can also be written as

$$E_i = r\sum_{j=0}^{i-1}(1-r)^j\overline{X}_{i-j} + (1-r)^i E_0$$

which expresses the EWMA as a weighted average of past subgroup means, where the weights decline exponentially, and the heaviest weight is assigned to the most recent subgroup mean.

Central Line

By default, the central line on an EWMA chart indicates an estimate for μ, which is computed as

$$\hat{\mu} = \overline{\overline{X}} = \frac{n_1\bar{X}_1 + \cdots + n_N\bar{X}_N}{n_1 + \cdots + n_N}$$

If you specify a known value (μ_0) for μ, the central line indicates the value of μ_0.

Control Limits

You can compute the limits in the following ways:

- as a specified multiple (k) of the standard error of E_i above and below the central line. The default limits are computed with $k = 3$ (these are referred to as 3σ *limits*).
- as probability limits defined in terms of α, a specified probability that E_i exceeds the limits

The following table presents the formulas for the limits:

Table 20.19. Limits for an EWMA Chart

Control Limits
LCL = lower limit = $\overline{\overline{X}} - k\hat{\sigma}r\sqrt{\sum_{j=0}^{i-1}(1-r)^{2j}/n_{i-j}}$
UCL = upper limit = $\overline{\overline{X}} + k\hat{\sigma}r\sqrt{\sum_{j=0}^{i-1}(1-r)^{2j}/n_{i-j}}$

Probability Limits
LCL = lower limit = $\overline{\overline{X}} - \Phi^{-1}(1-\alpha/2)\hat{\sigma}r\sqrt{\sum_{j=0}^{i-1}(1-r)^{2j}/n_{i-j}}$
UCL = upper limit = $\overline{\overline{X}} + \Phi^{-1}(1-\alpha/2)\hat{\sigma}r\sqrt{\sum_{j=0}^{i-1}(1-r)^{2j}/n_{i-j}}$

These formulas assume that the data are normally distributed. If standard values μ_0 and σ_0 are available for μ and σ, respectively, replace $\overline{\overline{X}}$ with μ_0 and $\hat{\sigma}$ with σ_0 in Table 20.19. Note that the limits vary with both n_i and i.

If the subgroup sample sizes are constant ($n_i = n$), the formulas for the control limits simplify to

$$\text{LCL} = \overline{\overline{X}} - k\hat{\sigma}\sqrt{r(1-(1-r)^{2i})/n(2-r)}$$
$$\text{UCL} = \overline{\overline{X}} + k\hat{\sigma}\sqrt{r(1-(1-r)^{2i})/n(2-r)}$$

Consequently, when the subgroup sample sizes are constant, the width of the control limits increases monotonically with i. For probability limits, replace k with $\Phi^{-1}(1-\alpha/2)$ in the previous equations. Refer to Roberts (1959) and Montgomery (1996).

As i becomes large, the upper and lower control limits approach constant values:

$$\text{LCL} = \overline{\overline{X}} - k\hat{\sigma}\sqrt{r/n(2-r)}$$
$$\text{UCL} = \overline{\overline{X}} + k\hat{\sigma}\sqrt{r/n(2-r)}$$

Some authors base the control limits for EWMA charts on the asymptotic expressions in the two previous equations. For asymptotic probability limits, replace k with

$\Phi^{-1}(1-\alpha/2)$ in these equations. You can display asymptotic limits by specifying the ASYMPTOTIC option.

Uniformly weighted moving average charts and exponentially weighted moving average charts have similar properties, and their asymptotic control limits are identical provided that

$$r = 2/(w+1)$$

where w is the weight factor for uniformly weighted moving average charts. Refer to Wadsworth and others (1986) and the *ASQC Glossary and Tables for Statistical Quality Control* (1983).

You can specify parameters for the EWMA limits as follows:

- Specify k with the SIGMAS= option or with the variable _SIGMAS_ in a LIMITS= data set.
- Specify α with the ALPHA= option or with the variable _ALPHA_ in a LIMITS= data set.
- Specify a constant nominal sample size $n_i \equiv n$ for the control limits with the LIMITN= option or with the variable _LIMITN_ in a LIMITS= data set.
- Specify r with the WEIGHT= option or with the variable _WEIGHT_ in a LIMITS= data set.
- Specify μ_0 with the MU0= option or with the variable _MEAN_ in a LIMITS= data set.
- Specify σ_0 with the SIGMA0= option or with the variable _STDDEV_ in a LIMITS= data set.

Choosing the Value of the Weight Parameter

Various approaches have been proposed for choosing the value of r.

- Hunter (1986) states that the choice "can be left to the judgment of the quality control analyst" and points out that the smaller the value of r, "the greater the influence of the historical data."
- Hunter (1986) also discusses a least squares procedure for estimating r from the data, **assuming an exponentially weighted moving average model for the data**. In this context, the fitted EWMA model provides a forecast of the process that is the basis for dynamic process control. You can use the ARIMA procedure in SAS/ETS software to compute the least squares estimate of r. (Refer to *SAS/ETS User's Guide* for information on PROC ARIMA.) Also see "Autocorrelation in Process Data" on page 1756.
- A number of authors have studied the design of EWMA control schemes based on average run length (ARL) computations. The ARL is the expected number of points plotted before a shift is detected. Ideally, the ARL should be short when a shift occurs, and it should be long when there is no shift (the process is in control.) The effect of r on the ARL was described by Roberts (1959), who

used simulation methods. The ARL function was approximated and tabulated by Robinson and Ho (1978), and a more general method for studying run-length distributions of EWMA charts was given by Crowder (1987a,b). Unlike Hunter (1986), these authors assume the data are independent and identically distributed; typically the normal distribution is assumed for the data, although the methods extend to nonnormal distributions. A more detailed discussion of the ARL approach follows.

Average run lengths for two-sided EWMA charts are shown in Table 20.20, which is patterned after Table 1 of Crowder (1987a,b). The ARLs were computed using the EWMAARL DATA step function (see page 1852 for details on the EWMAARL function). Note that Crowder (1987a,b) uses the notation L in place of k and the notation λ in place of r.

You can use Table 20.20 to find a combination of k and r that yields a desired ARL for an in-control process ($\delta = 0$) and for a specified shift of δ. Note that δ is assumed to be standardized; in other words, if a shift of Δ is to be detected in the process mean μ, and if σ is the process standard deviation, you should select the table entry with

$$\delta = \Delta/(\sigma/\sqrt{n})$$

where n is the subgroup sample size. Thus, δ can be regarded as the shift in the sampling distribution of the subgroup mean.

For example, suppose you want to construct an EWMA scheme with an in-control ARL of 90 and an ARL of 9 for detecting a shift of $\delta = 1$. Table 20.20 shows that the combination $r = 0.5$ and $k = 2.5$ yields an in-control ARL of 91.17 and an ARL of 8.27 for $\delta = 1$.

Crowder (1987a,b) cautions that setting the in-control ARL at a desired level does not guarantee that the probability of an early false signal is acceptable. For further details concerning the distribution of the ARL, refer to Crowder (1987a,b).

In addition to using Table 20.20 or the EWMAARL DATA step function to choose a EWMA scheme with desired average run length properties, you can use them to evaluate an existing EWMA scheme. For example, the "Getting Started" section of this chapter contains EWMA schemes with $r = 0.3$ and $k = 3$. The following statements use the EWMAARL function to compute the in-control ARL and the ARLs for shifts of $\delta = 0.25$ and $\delta = 0.5$:

```
data arlewma;
   arlin = ewmaarl(  0,0.3,3.0);
   arl1  = ewmaarl(.25,0.3,3.0);
   arl2  = ewmaarl(.50,0.3,3.0);
run;
```

The in-control ARL is 465.553, the ARL for $\delta = .25$ is 178.741, and the ARL for $\delta = .5$ is 53.1603. See Example 20.5 on page 658 for an illustration of how to use the EWMAARL function to compute average run lengths for various EWMA schemes and shifts.

Table 20.20. Average Run Lengths for Two-Sided EWMA Charts

		r (weight parameter)					
k	δ	0.05	0.10	0.25	0.50	0.75	1.00
2.0	0.00	127.53	73.28	38.56	26.45	22.88	21.98
2.0	0.25	43.94	34.49	24.83	20.12	18.86	19.13
2.0	0.50	18.97	15.53	12.74	11.89	12.34	13.70
2.0	0.75	11.64	9.36	7.62	7.29	7.86	9.21
2.0	1.00	8.38	6.62	5.24	4.91	5.26	6.25
2.0	1.25	6.56	5.13	3.96	3.59	3.76	4.40
2.0	1.50	5.41	4.20	3.19	2.80	2.84	3.24
2.0	1.75	4.62	3.57	2.68	2.29	2.26	2.49
2.0	2.00	4.04	3.12	2.32	1.95	1.88	2.00
2.0	2.25	3.61	2.78	2.06	1.70	1.61	1.67
2.0	2.50	3.26	2.52	1.85	1.51	1.42	1.45
2.0	2.75	2.99	2.32	1.69	1.37	1.29	1.29
2.0	3.00	2.76	2.16	1.55	1.26	1.19	1.19
2.0	3.25	2.56	2.03	1.43	1.18	1.13	1.12
2.0	3.50	2.39	1.93	1.32	1.12	1.08	1.07
2.0	3.75	2.26	1.83	1.24	1.08	1.05	1.04
2.0	4.00	2.15	1.73	1.17	1.05	1.03	1.02
2.5	0.00	379.09	223.35	124.18	91.17	82.49	80.52
2.5	0.25	73.98	66.59	59.66	58.33	61.07	65.77
2.5	0.50	26.63	23.63	23.28	27.16	33.26	41.49
2.5	0.75	15.41	12.95	11.96	13.96	18.05	24.61
2.5	1.00	10.79	8.75	7.52	8.27	10.57	14.92
2.5	1.25	8.31	6.60	5.39	5.52	6.75	9.46
2.5	1.50	6.78	5.31	4.18	4.03	4.65	6.30
2.5	1.75	5.75	4.46	3.43	3.14	3.43	4.41
2.5	2.00	5.00	3.86	2.92	2.57	2.67	3.24
2.5	2.25	4.43	3.42	2.56	2.18	2.17	2.49
2.5	2.50	4.00	3.07	2.29	1.90	1.83	2.00
2.5	2.75	3.64	2.80	2.08	1.69	1.59	1.67
2.5	3.00	3.36	2.57	1.91	1.52	1.41	1.45
2.5	3.25	3.12	2.39	1.77	1.39	1.29	1.29
2.5	3.50	2.92	2.24	1.64	1.28	1.19	1.19
2.5	3.75	2.74	2.13	1.52	1.20	1.13	1.12
2.5	4.00	2.58	2.04	1.42	1.13	1.08	1.07

Table 20.20. (continued)

k	δ	0.05	0.10	0.25	0.50	0.75	1.00
3.0	0.00	1383.62	842.15	502.90	397.46	374.50	370.40
3.0	0.25	133.61	144.74	171.09	208.54	245.76	281.15
3.0	0.50	37.33	37.41	48.45	75.35	110.95	155.22
3.0	0.75	19.95	17.90	20.16	31.46	50.92	81.22
3.0	1.00	13.52	11.38	11.15	15.74	25.64	43.89
3.0	1.25	10.24	8.32	7.39	9.21	14.26	24.96
3.0	1.50	8.26	6.57	5.47	6.11	8.72	14.97
3.0	1.75	6.94	5.45	4.34	4.45	5.80	9.47
3.0	2.00	6.00	4.67	3.62	3.47	4.15	6.30
3.0	2.25	5.30	4.10	3.11	2.84	3.16	4.41
3.0	2.50	4.76	3.67	2.75	2.41	2.52	3.24
3.0	2.75	4.32	3.32	2.47	2.10	2.09	2.49
3.0	3.00	3.97	3.05	2.26	1.87	1.79	2.00
3.0	3.25	3.67	2.82	2.09	1.69	1.57	1.67
3.0	3.50	3.42	2.62	1.95	1.53	1.41	1.45
3.0	3.75	3.22	2.45	1.84	1.41	1.29	1.29
3.0	4.00	3.04	2.30	1.73	1.31	1.20	1.19
3.5	0.00	12851.0	4106.4	2640.16	2227.34	2157.99	2149.34
3.5	0.25	281.09	381.29	625.78	951.18	1245.90	1502.76
3.5	0.50	53.58	64.72	123.43	267.36	468.68	723.81
3.5	0.75	25.62	25.33	38.68	88.70	182.12	334.40
3.5	1.00	16.65	14.79	17.71	35.97	78.05	160.95
3.5	1.25	12.36	10.37	10.48	17.64	37.15	81.80
3.5	1.50	9.86	8.00	7.25	10.19	19.63	43.96
3.5	1.75	8.22	6.54	5.52	6.70	11.46	24.96
3.5	2.00	7.07	5.55	4.47	4.86	7.33	14.97
3.5	2.25	6.21	4.83	3.77	3.78	5.08	9.47
3.5	2.50	5.55	4.29	3.28	3.10	3.76	6.30
3.5	2.75	5.03	3.87	2.91	2.63	2.94	4.41
3.5	3.00	4.60	3.54	2.63	2.30	2.40	3.24
3.5	3.25	4.25	3.26	2.41	2.05	2.03	2.49
3.5	3.50	3.95	3.03	2.23	1.85	1.76	2.00
3.5	3.75	3.70	2.84	2.10	1.69	1.56	1.67
3.5	4.00	3.47	2.66	1.99	1.55	1.40	1.45

Output Data Sets

OUTLIMITS= Data Set

The OUTLIMITS= data set saves the control limit parameters. The following variables can be saved:

Variable	Description
ALPHA	probability (α) of exceeding limits
INDEX	optional identifier for the control limits specified with the OUTINDEX= option
LIMITN	sample size associated with the control limits
MEAN	process mean ($\overline{\overline{X}}$ or μ_0)
SIGMAS	multiple (k) of standard error of E_i
STDDEV	process standard deviation ($\hat{\sigma}$ or σ_0)
SUBGRP	*subgroup-variable* specified in the EWMACHART statement
TYPE	type (estimate or standard value) of _MEAN_ and _STDDEV_
VAR	*process* specified in the EWMACHART statement
WEIGHT	weight (r) assigned to most recent subgroup mean in computation of EWMA

The OUTLIMITS= data set does not contain the control limits; instead, it contains control limit parameters that can be used to recompute the control limits.

Notes:

1. If the control limits vary with subgroup sample size, the special missing value *V* is assigned to the variable _LIMITN_.
2. If the limits are defined in terms of a multiple k of the standard error of E_i, the value of _ALPHA_ is computed as $\alpha = 2(1 - \Phi(k))$, where $\Phi(\cdot)$ is the standard normal distribution function.
3. If the limits are probability limits, the value of _SIGMAS_ is computed as $k = \Phi^{-1}(1 - \alpha/2)$, where Φ^{-1} is the inverse standard normal distribution function.
4. Optional BY variables are saved in the OUTLIMITS= data set.

The OUTLIMITS= data set contains one observation for each *process* specified in the EWMACHART statement.

You can use OUTLIMITS= data sets

- to keep a permanent record of the control limit parameters
- to write reports. You may prefer to use OUTTABLE= data sets for this purpose.
- as LIMITS= data sets in subsequent runs of PROC MACONTROL

For an example of an OUTLIMITS= data set, see “Saving Control Limit Parameters” on page 616.

OUTHISTORY= Data Set

The OUTHISTORY= data set saves subgroup summary statistics. The following variables can be saved:

- the *subgroup-variable*
- a subgroup mean variable named by *process* suffixed with *X*
- a subgroup standard deviation variable named by *process* suffixed with *S*
- a subgroup EWMA variable named by *process* suffixed with *E*
- a subgroup sample size variable named by *process* suffixed with *N*

Given a *process* name that contains eight characters, the procedure first shortens the name to its first four characters and its last three characters, and then it adds the suffix. For example, the procedure shortens the *process* DIAMETER to DIAMTER before adding the suffix.

Subgroup summary variables are created for each *process* specified in the EWMACHART statement. For example, consider the following statements:

```
proc macontrol data=clips;
   ewmachart (gap yldstren)*day / weight    =0.2
                                  outhistory=cliphist;
run;
```

The data set CLIPHIST would contain nine variables named DAY, GAPX, GAPS, GAPE, GAPN, YLDSRENX, YLDSRENS, YLDSRENE, and YLDSRENN.

Additionally, the following variables, if specified, are included:

- BY variables
- *block-variables*
- *symbol-variable*
- ID variables
- _PHASE_ (if the OUTPHASE= option is specified)

For an example of an OUTHISTORY= data set, see “Saving Summary Statistics” on page 615.

OUTTABLE= Data Set

The OUTTABLE= data set saves subgroup summary statistics, control limits, and related information. The following variables can be saved:

Variable	Description
ALPHA	probability (α) of exceeding control limits
EXLIM	control limit exceeded on EWMA chart
EWMA	exponentially weighted moving average
LCLE	lower control limit for EWMA
LIMITN	nominal sample size associated with the control limits
MEAN	process mean
SIGMAS	multiple (k) of the standard error associated with control limits
subgroup	values of the subgroup variable
SUBN	subgroup sample size
SUBS	subgroup standard deviation
SUBX	subgroup mean
UCLE	upper control limit for EWMA
VAR	*process* specified in the EWMACHART statement
WEIGHT	weight (r) assigned to most recent subgroup mean in computation of EWMA

In addition, the following variables, if specified, are included:

- BY variables
- *block-variables*
- ID variables
- _PHASE_ (if the READPHASES= option is specified)
- *symbol-variable*

Notes:

1. Either the variable_ALPHA_ or the variable _SIGMAS_ is saved depending on how the control limits are defined (with the ALPHA= or SIGMAS= options, respectively, or with the corresponding variables in a LIMITS= data set).

2. The variables _VAR_ and _EXLIM_ are character variables of length 8. The variable _PHASE_ is a character variable of length 16. All other variables are numeric.

For an example of an OUTTABLE= data set, see “Saving Control Limit Parameters” on page 616.

ODS Tables

The following table summarizes the ODS tables that you can request with the EWMACHART statement.

Table 20.21. ODS Tables Produced with the EWMACHART Statement

Table Name	Description	Options
EWMACHART	exponentially weighted moving average chart summary statistics	TABLE, TABLEALL, TABLEC, TABLEID, TABLEOUT
Parameters	exponentially weighted moving average parameters	TABLE, TABLEALL, TABLEC, TABLEID, TABLEOUT

Input Data Sets

DATA= Data Set

You can read raw data (process measurements) from a DATA= data set specified in the PROC MACONTROL statement. Each *process* specified in the EWMACHART statement must be a SAS variable in the DATA= data set. This variable provides measurements that must be grouped into subgroup samples indexed by the *subgroup-variable*. The *subgroup-variable*, which is specified in the EWMACHART statement, must also be a SAS variable in the DATA= data set. Each observation in a DATA= data set must contain a value for each *process* and a value for the *subgroup-variable*. If the i^{th} subgroup contains n_i items, there should be n_i consecutive observations for which the value of the *subgroup-variable* is the index of the i^{th} subgroup. For example, if each subgroup contains five items and there are 30 subgroup samples, the DATA= data set should contain 150 observations.

Other variables that can be read from a DATA= data set include

- _PHASE_ (if the READPHASES= option is specified)
- *block-variables*
- *symbol-variable*
- BY variables
- ID variables

By default, the MACONTROL procedure reads all the observations in a DATA= data set. However, if the data set includes the variable _PHASE_, you can read selected groups of observations (referred to as *phases*) with the READPHASES= option (for an example, see "Displaying Stratification in Phases" on page 1689).

For an example of a DATA= data set, see "Creating EWMA Charts from Raw Data" on page 610.

LIMITS= Data Set

You can read preestablished control limit parameters from a LIMITS= data set specified in the PROC MACONTROL statement. The LIMITS= data set used by the MACONTROL procedure does not contain the actual control limits, but rather it contains the parameters required to compute the limits. For example, the following statements read parameters from the data set PARMS:*

```
proc macontrol data=parts limits=parms;
   ewmachart gap*day;
run;
```

The LIMITS= data set can be an OUTLIMITS= data set that was created in a previous run of the MACONTROL procedure. Such data sets always contain the variables required for a LIMITS= data set; see page 639. The LIMITS= data set can also be created directly using a DATA step.

When you create a LIMITS= data set, you must provide the variable _WEIGHT_, which specifies the weight parameter used to compute the EWMAs. In addition, note the following:

- The variables _VAR_ and _SUBGRP_ are required. These must be character variables of length 8.
- The variable _INDEX_ is required if you specify the READINDEX= option. This must be a character variable of length 16.
- The variables _LIMITN_, _SIGMAS_ (or _ALPHA_), and _TYPE_ are optional, but they are recommended to maintain a complete set of control limit information. The variable _TYPE_ must be a character variable of length 8. Valid values are `ESTIMATE`, `STANDARD`, `STDMEAN`, and `STDSIGMA`.
- BY variables are required if specified with a BY statement.

Some advantages of working with a LIMITS= data set are that

- it facilitates reusing a permanently saved set of parameters
- a distinct set of parameters can be read for each *process* specified in the EWMACHART statement
- it facilitates keeping track of multiple sets of parameters that accumulate for the same *process* as the process evolves over time

For an example, see “Reading Preestablished Control Limit Parameters” on page 618.

HISTORY= Data Set

You can read subgroup summary statistics from a HISTORY= data set specified in the PROC MACONTROL statement. This allows you to reuse OUTHISTORY= data sets that have been created in previous runs of the MACONTROL, SHEWHART, or CUSUM procedures or to read output data sets created with SAS summarization procedures such as PROC MEANS.

*In Release 6.09 and earlier releases, it is necessary to specify the READLIMITS option.

A HISTORY= data set used with the EWMACHART statement must contain the following:

- the *subgroup-variable*
- a subgroup mean variable for each *process*
- a subgroup sample size variable for each *process*
- a subgroup standard deviation variable for each *process*

The names of the subgroup mean, subgroup standard deviation, and subgroup sample size variables must be the *process* name concatenated with the suffix characters X, S, and N, respectively.

For example, consider the following statements:

```
proc macontrol history=cliphist;
   ewmachart (gap diameter)*day  / weight=0.2;
run;
```

The data set CLIPHIST must include the variables DAY, GAPX, GAPS, GAPN, DIAMTERX, DIAMTERS, and DIAMTERN.

Although a subgroup EWMA variable (named by the *process* name suffixed with E) is saved in an OUTHISTORY= data set, it is not required in a HISTORY= data set, because the subgroup mean variable is sufficient to compute the EWMAs.

Note that, if you specify a *process* name that contains eight characters, the names of the summary variables must be formed from the first four characters and the last three characters of the *process* name, suffixed with the appropriate character.

Other variables that can be read from a HISTORY= data set include

- _PHASE_ (if the READPHASES= option is specified)
- *block-variables*
- *symbol-variable*
- BY variables
- ID variables

By default, the MACONTROL procedure reads all the observations in a HISTORY= data set. However, if the HISTORY= data set includes the variable _PHASE_, you can read selected groups of observations (referred to as *phases*) by specifying the READPHASES= option (see “Displaying Stratification in Phases” on page 1689 for an example).

For an example of a HISTORY= data set, see “Creating EWMA Charts from Subgroup Summary Data” on page 612.

TABLE= Data Set

You can read summary statistics and control limits from a TABLE= data set specified in the PROC MACONTROL statement. This enables you to reuse an OUTTABLE= data set created in a previous run of the MACONTROL procedure.

The following table lists the variables required in a TABLE= data set used with the EWMACHART statement:

Variable	Description
EWMA	exponentially weighted moving average
LCLE	lower control limit for EWMA
LIMITN	nominal sample size associated with the control limits
MEAN	process mean
subgroup-variable	values of the *subgroup-variable*
SUBN	subgroup sample size
SUBS	subgroup standard deviation
SUBX	subgroup mean
UCLE	upper control limit for EWMA
WEIGHT	weight (r) assigned to most recent subgroup mean in computation of EWMA

Other variables that can be read from a TABLE= data set include

- *block-variables*
- *symbol-variable*
- BY variables
- ID variables
- _PHASE_ (if the READPHASES= option is specified). This variable must be a character variable of length 16.
- _VAR_. This variable is required if more than one *process* is specified or if the data set contains information for more than one *process*. This variable must be a character variable of length 8.

For an example of a TABLE= data set, see "Saving Control Limit Parameters" on page 616.

Methods for Estimating the Standard Deviation

When control limits are computed from the input data, four methods are available for estimating the process standard deviation σ. Three methods (referred to as the default, MVLUE, and RMSDF) are available with subgrouped data. A fourth method is used if the data are individual measurements (see "Default Method for Individual Measurements" on page 646).

Default Method for Subgroup Samples

This method is the default for EWMA charts using subgrouped data. The default estimate of σ is

$$\hat{\sigma} = \frac{s_1/c_4(n_1) + \ldots + s_N/c_4(n_N)}{N}$$

where N is the number of subgroups for which $n_i \geq 2$, s_i is the sample standard deviation of the i^{th} subgroup

$$s_i = \sqrt{\frac{1}{n_i - 1} \sum_{j=1}^{n_i} (x_{ij} - \bar{X}_i)^2}$$

and

$$c_4(n_i) = \frac{\Gamma(n_i/2)\sqrt{2/(n_i-1)}}{\Gamma((n_i-1)/2)}$$

Here $\Gamma(\cdot)$ denotes the gamma function, and $\bar{X}_i$ denotes the i^{th} subgroup mean. A subgroup standard deviation s_i is included in the calculation only if $n_i \geq 2$. If the observations are normally distributed, then the expected value of s_i is $c_4(n_i)\sigma$. Thus, $\hat{\sigma}$ is the unweighted average of N unbiased estimates of σ. This method is described in the *ASTM Manual on Presentation of Data and Control Chart Analysis* (1976).

MVLUE Method for Subgroup Samples

If you specify SMETHOD=MVLUE, a minimum variance linear unbiased estimate (MVLUE) is computed for σ. Refer to Burr (1969, 1976) and Nelson (1989, 1994). The MVLUE is a weighted average of N unbiased estimates of σ of the form $s_i/c_4(n_i)$, and it is computed as

$$\hat{\sigma} = \frac{h_1 s_1/c_4(n_1) + \ldots + h_N s_N/c_4(n_N)}{h_1 + \ldots + h_N}$$

where

$$h_i = \frac{[c_4(n_i)]^2}{1 - [c_4(n_i)]^2}$$

A subgroup standard deviation s_i is included in the calculation only if $n_i \geq 2$, and N is the number of subgroups for which $n_i \geq 2$. The MVLUE assigns greater weight to estimates of σ from subgroups with larger sample sizes, and it is intended for situations where the subgroup sample sizes vary. If the subgroup sample sizes are constant, the MVLUE reduces to the default estimate.

RMSDF Method for Subgroup Samples

If you specify SMETHOD=RMSDF, a weighted root-mean-square estimate is computed for σ as follows:

$$\hat{\sigma} = \frac{\sqrt{(n_1-1)s_1^2 + \cdots + (n_N-1)s_N^2}}{c_4(n)\sqrt{n_1 + \cdots + n_N - N}}$$

The weights are the degrees of freedom $n_i - 1$. A subgroup standard deviation s_i is included in the calculation only if $n_i \geq 2$, and N is the number of subgroups for which $n_i \geq 2$.

If the unknown standard deviation σ is constant across subgroups, the root-mean-square estimate is more efficient than the minimum variance linear unbiased estimate. However, in process control applications it is generally not assumed that σ is constant, and if σ varies across subgroups, the root-mean-square estimate tends to be more inflated than the MVLUE.

Default Method for Individual Measurements

When each subgroup sample contains a single observation ($n_i \equiv 1$), the process standard deviation σ is estimated as

$$\hat{\sigma} = \sqrt{\frac{1}{2(N-1)} \sum_{i=1}^{N-1} (x_{i+1} - x_i)^2}$$

where N is the number of observations, and $x_1, x_2, \ldots, x_N$ are the individual measurements. This formula is given by Wetherill (1977), who states that the estimate of the variance is biased if the measurements are autocorrelated.

Axis Labels

You can specify axis labels by assigning labels to particular variables in the input data set, as summarized in the following table:

Axis	Input Data Set	Variable
Horizontal	all	*subgroup-variable*
Vertical	DATA=	*process*
Vertical	HISTORY=	subgroup mean variable
Vertical	TABLE=	_EWMA_

For example, the following sets of statements specify the label *EWMA of Clip Gaps* for the vertical axis and the label *Day* for the horizontal axis of the EWMA chart:

```
proc macontrol data=clips1;
   ewmachart gap*day / weight=0.3;
   label gap = 'EWMA of Clip Gaps';
   label day = 'Day';
run;


proc macontrol history=cliphist;
   ewmachart gap*day / weight=0.3;
   label gapx = 'EWMA of Clip Gaps';
   label day  = 'Day';
run;

proc macontrol table=cliptab;
   ewmachart gap*day;
   label _ewma_  = 'EWMA of Clip Gaps';
   label day     = 'Day';
run;
```

In this example, the label assignments are in effect only for the duration of the procedure step, and they temporarily override any permanent labels associated with the variables.

Missing Values

An observation read from a DATA=, HISTORY=, or TABLE= data set is not analyzed if the value of the subgroup variable is missing. For a particular process variable, an observation read from a DATA= data set is not analyzed if the value of the process variable is missing. Missing values of process variables generally lead to unequal subgroup sample sizes. For a particular process variable, an observation read from a HISTORY= or TABLE= data set is not analyzed if the values of any of the corresponding summary variables are missing.

Examples

This section provides advanced examples of the EWMACHART statement.

Example 20.1. Specifying Standard Values for the Process Mean and Process Standard Deviation

By default, the EWMACHART statement estimates the process mean (μ) and standard deviation (σ) from the data. This is illustrated in the "Getting Started" section of this chapter. However, there are applications in which standard values (μ_0 and σ_0) are available based, for instance, on previous experience or extensive sampling. You can specify these values with the MU0= and SIGMA0= options.

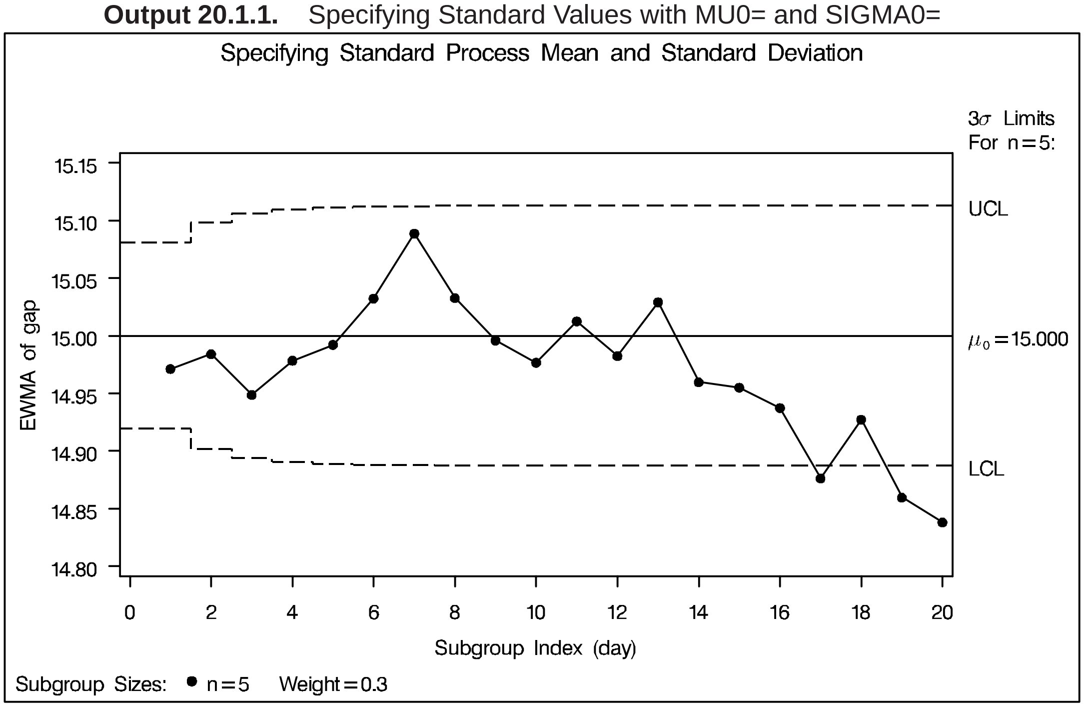
See MACEW2 in the SAS/QC Sample Library

For example, suppose it is known that the metal clip manufacturing process (introduced on page 610) has a mean of 15 and standard deviation of 0.2. The following statements specify these standard values:

```
title 'Specifying Standard Process Mean and Standard Deviation';
symbol v=dot;
proc macontrol data=clips1;
   ewmachart gap*day / mu0    = 15
                       sigma0 = 0.2
                       weight = 0.3
                       xsymbol= mu0;
run;
```

The XSYMBOL= option specifies the label for the central line. The resulting chart is shown in Output 20.1.1.

Output 20.1.1. Specifying Standard Values with MU0= and SIGMA0=



The central line and control limits are determined using μ_0 and σ_0 (see the equations in Table 20.19 on page 634). Output 20.1.1 indicates that the process is out-of-control, since the moving averages for DAY=17, DAY=19, and DAY=20 lie below the lower control limit.

You can also specify μ_0 and σ_0 with the variables _MEAN_ and _STDDEV_ in a LIMITS= data set, as illustrated by the following statements:

```
data cliplim;
   length _var_ _subgrp_ _type_ $8;
   _var_    = 'gap';
   _subgrp_ = 'day';
   _type_   = 'STANDARD';
   _limitn_ = 5;
   _mean_   = 15;
   _stddev_ = 0.2;
   _weight_ = 0.3;

proc macontrol data=clips1 limits=cliplim;
   ewmachart gap*day / xsymbol=mu0;
run;
```

The variable _WEIGHT_ is required, and its value provides the weight parameter used to compute the EWMAs. The variables _VAR_ and _SUBGRP_ are also required, and their values must match the *process* and *subgroup-variable*, respectively, specified in the EWMACHART statement. The bookkeeping variable _TYPE_ is not required, but it is recommended to indicate that the variables _MEAN_ and _STDDEV_ provide standard values rather than estimated values.

The resulting chart (not shown here) is identical to the one shown in Output 20.1.1.

Example 20.2. Displaying Limits Based on Asymptotic Values

See MACEW3 in the SAS/QC Sample Library

The upper (lower) control limits in Output 20.1.1 are monotonically increasing (decreasing). As the number of subgroups increases, the control limits approach the following asymptotic values:

$$\mathrm{LCL} = \overline{\overline{X}} - k\hat{\sigma}\sqrt{r/n(2-r)}$$

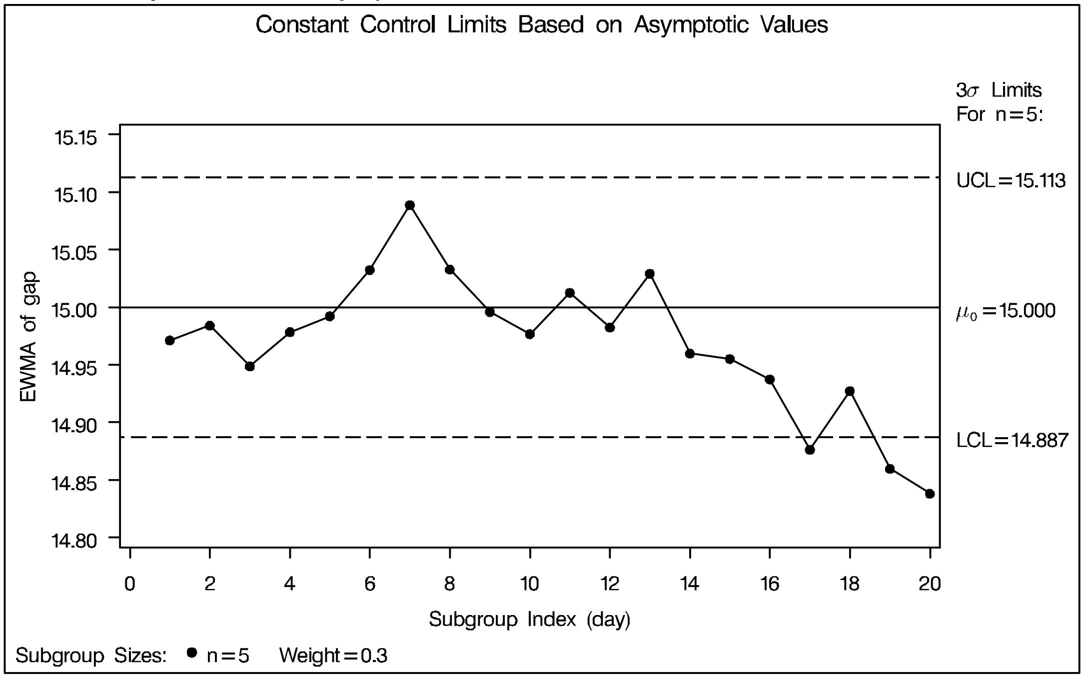
$$\mathrm{UCL} = \overline{\overline{X}} + k\hat{\sigma}\sqrt{r/n(2-r)}$$

These constant limits are displayed if you specify the ASYMPTOTIC option, as illustrated by the following statements:

```
title 'Constant Control Limits Based on Asymptotic Values';
symbol v=dot;
proc macontrol data=clips1;
   ewmachart gap*day / mu0    = 15
                       sigma0 = 0.2
                       weight = 0.3
                       xsymbol= mu0
                       asymptotic;
run;
```

The chart is shown in Output 20.2.1.

Output 20.2.1. Asymptotic Control Limits



Note that the same three points that were outside the exact limits (displayed in Output 20.1.1) fall outside the asymptotic limits. The exact limits quickly approach the asymptotic values, so only the first few subgroups have appreciably different limits.

Example 20.3. Working with Unequal Subgroup Sample Sizes

See MACEW4 in the SAS/QC Sample Library

This example contains measurements from the metal clip manufacturing process (introduced on page 610). The following statements create a SAS data set named CLIPS4, which contains additional clip gap measurements taken on a daily basis:

```
data clips4;
   input day @;
   length dayc $2.;
   informat day ddmmyy8.;
   format   day date5.;
   dayc=put(day,date5.);
   dayc=substr(dayc,1,2);
   do i=1 to 5;
      input gap @;
      output;
      end;
   drop i;
   label dayc='April';
datalines;
 1/4/86  14.93  14.65  14.87  15.11  15.18
 2/4/86  15.06  14.95  14.91  15.14  15.41
 3/4/86  14.90  14.90  14.96  15.26  15.18
 4/4/86  15.25  14.57  15.33  15.38  14.89
 7/4/86  14.68  14.63  14.72  15.32  14.86
```

```
 8/4/86  14.48  14.88  14.98  14.74  15.48
 9/4/86  14.99  15.16  15.02  15.53  14.66
10/4/86  14.88  15.44  15.04  15.10  14.89
11/4/86  15.14  15.33  14.75  15.23  14.64
14/4/86  15.46  15.30  14.92  14.58  14.68
15/4/86  15.23  14.63    .      .      .
16/4/86  15.13  15.25    .      .      .
17/4/86  15.06  15.25  15.28  15.30  15.34
18/4/86  15.22  14.77  15.12  14.82  15.29
21/4/86  14.95  14.96  14.65  14.87  14.77
22/4/86  15.01  15.11  15.11  14.79  14.88
23/4/86  14.97  15.50  14.93  15.13  15.25
24/4/86  15.23  15.21  15.31  15.07  14.97
25/4/86  15.08  14.75  14.93  15.34  14.98
28/4/86  15.07  14.86  15.42  15.47  15.24
29/4/86  15.27  15.20  14.85  15.62  14.67
30/4/86  14.97  14.73  15.09  14.98  14.46
;
```

Note that only two gap measurements were recorded on April 15 and April 16.

A partial listing of CLIPS4 is shown in Output 20.3.1. This data set contains three variables: DAY is a numeric variable that contains the date (month, day, and year) that the measurement is taken, DAYC is a character variable that contains the day the measurement is taken, and GAP is a numeric variable that contains the measurement.

Output 20.3.1. The Data Set CLIPS4

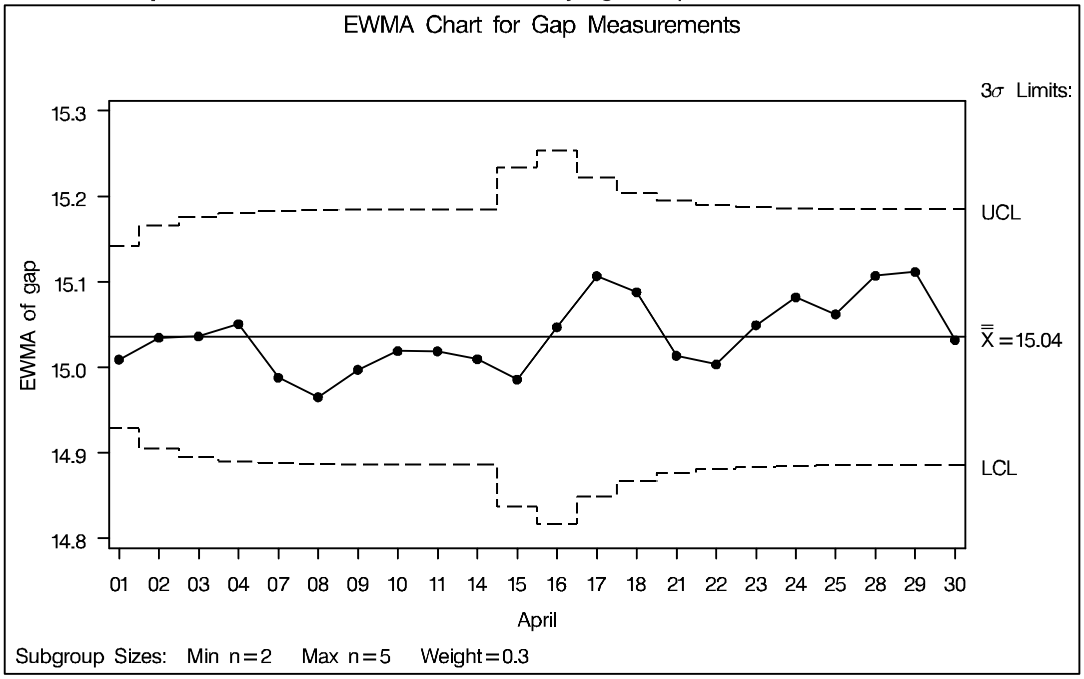
```
                The Data Set CLIPS4

               day    dayc     gap

              01APR    01     14.93
              01APR    01     14.65
              01APR    01     14.87
              01APR    01     15.11
              01APR    01     15.18
              02APR    02     15.06
              02APR    02     14.95
              02APR    02     14.91
              02APR    02     15.14
              02APR    02     15.41
                .       .       .
                .       .       .
                .       .       .
              30APR    30     14.46
```

The following statements request an EWMA chart, shown in Output 20.3.2, for these gap measurements:

```
title 'EWMA Chart for Gap Measurements';
symbol v=dot;
proc macontrol data=clips4;
   ewmachart gap*dayc / weight = 0.3;
run;
```

The character variable DAYC (rather than the numeric variable DAY) is specified as the *subgroup-variable* in the preceding EWMACHART statement. If DAY were the *subgroup-variable*, each day during April would appear on the horizontal axis, including the weekend days of April 5 and April 6 for which no measurements were taken. To avoid this problem, the *subgroup-variable* DAYC is created from DAY using the PUT and SUBSTR function. Since DAYC is a character *subgroup-variable*, a discrete axis is used for the horizontal axis, and as a result, April 5 and April 6 do not appear on the horizontal axis in Output 20.3.2. A LABEL statement is used to specify the label *April* for the horizontal axis, indicating the month that these measurements were taken.

Output 20.3.2. EWMA Chart with Varying Sample Sizes



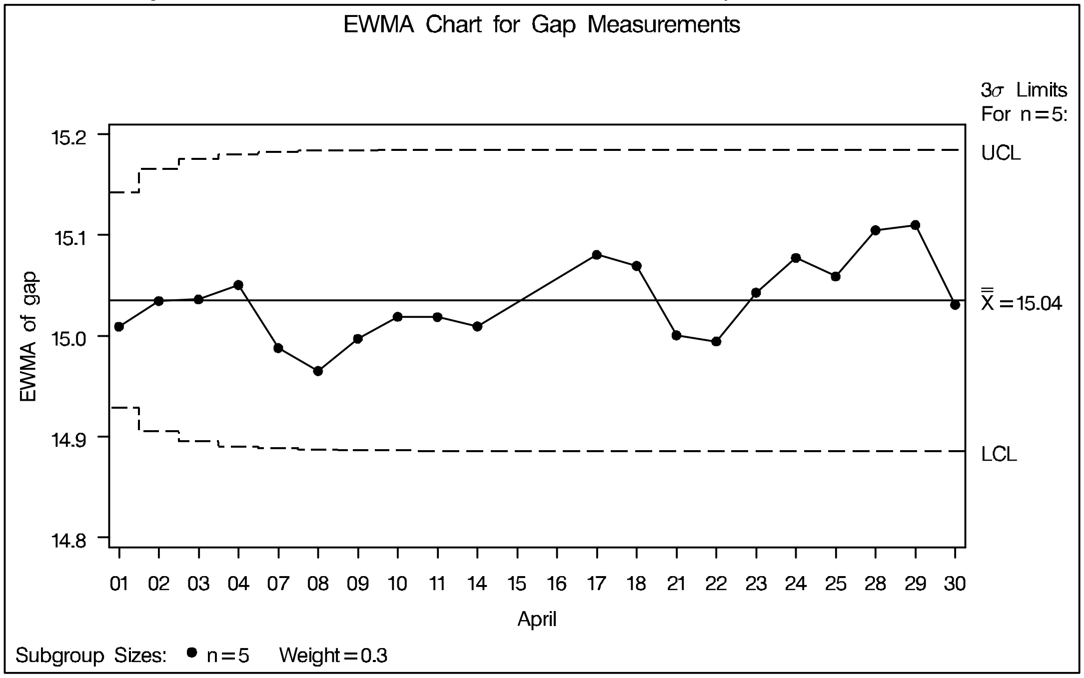
Note that the control limits vary with the subgroup sample size. The sample size legend in the lower left corner displays the minimum and maximum subgroup sample sizes.

The EWMACHART statement provides various options for working with unequal subgroup sample sizes. For example, you can use the LIMITN= option to specify a fixed (nominal) sample size for computing control limits, as illustrated by the following statements:

```
proc macontrol data=clips4;
   ewmachart gap*dayc / weight=0.3
                        limitn=5;
run;
```

The resulting chart is shown in Output 20.3.3.

Output 20.3.3. Control Limits Based on Fixed Sample Size

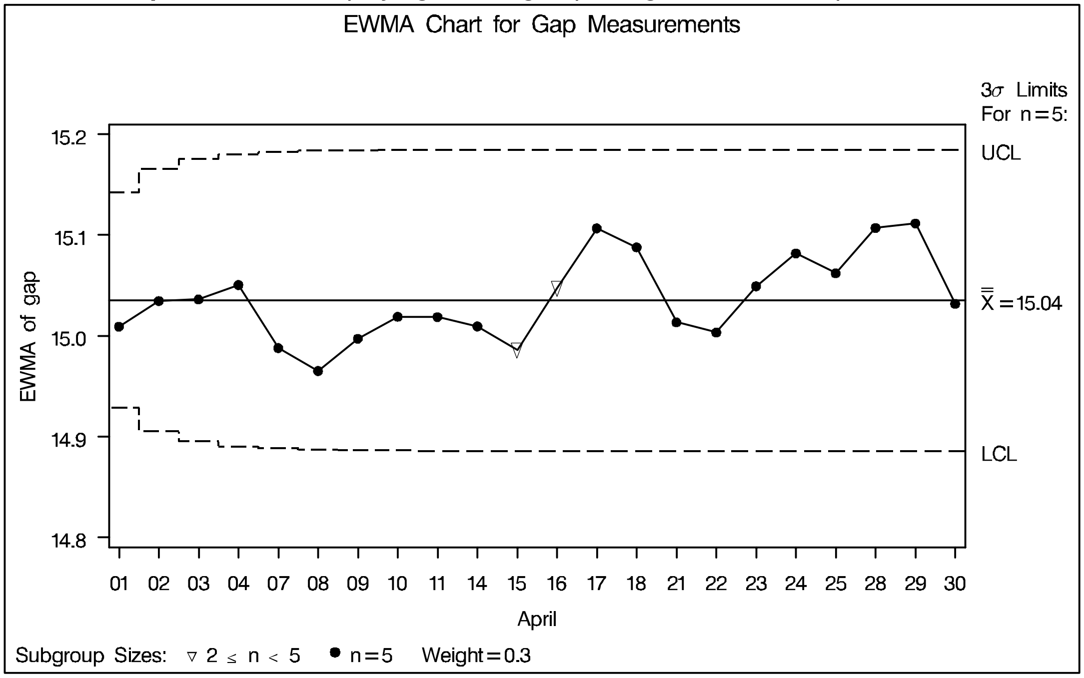


Note that the only points displayed are those corresponding to subgroups whose sample size matches the nominal sample size of five. Therefore, points are not displayed for April 15 and April 16. To plot points for all subgroups (regardless of subgroup sample size), you can specify the ALLN option, as follows:

```
proc macontrol data=clips4;
   ewmachart gap*dayc / weight=0.3
                        limitn=5
                        alln
                        nmarkers;
run;
```

The chart is shown in Output 20.3.4. The NMARKERS option requests special symbols to identify points for which the subgroup sample size differs from the nominal sample size.

Output 20.3.4. Displaying All Subgroups Regardless of Sample Size



You can use the SMETHOD= option to determine how the process standard deviation σ is to be estimated when the subgroup sample sizes vary. The default method computes $\hat{\sigma}$ as an unweighted average of subgroup estimates of σ. Specifying SMETHOD=MVLUE requests a minimum variance linear unbiased estimate (MVLUE), which assigns greater weight to estimates of σ from subgroups with larger sample sizes. Specifying SMETHOD=RMSDF requests a weighted root-mean-square estimate. If the unknown standard deviation σ is constant across subgroups, the root-mean-square estimate is more efficient than the MVLUE. For more information, see "Methods for Estimating the Standard Deviation" on page 645.

The following statements apply all three methods:

```
proc macontrol data=clips4;
   ewmachart gap*dayc / outlimits = cliplim1
                        outindex  = 'Default'
                        weight    = 0.3
                        nochart;
   ewmachart gap*dayc / smethod   = mvlue
                        outlimits = cliplim2
                        outindex  = 'MVLUE'
                        weight    = 0.3
                        nochart;
   ewmachart gap*dayc / smethod   = rmsdf
                        outlimits = cliplim3
                        outindex  = 'RMSDF'
                        weight    = 0.3
                        nochart;
run;
```

```
title 'Estimating the Process Standard Deviation';
data climits;
   set cliplim1 cliplim2 cliplim3;
run;
```

The data set CLIMITS is listed in Output 20.3.5.

Output 20.3.5. Listing of the Data Set CLIMITS

Estimating the Process Standard Deviation

VAR	_SUBGRP_	_INDEX_	_TYPE_	_LIMITN_	_ALPHA_	_SIGMAS_	_MEAN_	_STDDEV_	_WEIGHT_
gap	dayc	Default	ESTIMATE	V	.002699796	3	15.0354	0.26503	0.3
gap	dayc	MVLUE	ESTIMATE	V	.002699796	3	15.0354	0.26096	0.3
gap	dayc	RMSDF	ESTIMATE	V	.002699796	3	15.0354	0.25959	0.3

Note that the estimate of the process standard deviation (stored in the variable _STDDEV_) is slightly different depending on the estimation method. The variable _LIMITN_ is assigned the special missing value *V* in the OUTLIMITS= data set, indicating that the subgroup sample sizes vary.

Example 20.4. Displaying Individual Measurements on an EWMA Chart

See MACEW5 in the SAS/QC Sample Library

In the manufacture of automotive tires, the diameter of the steel belts inside the tire is measured. The following data set contains these measurements for 30 tires:

```
data tires;
   input sample diameter @@;
   datalines;
 1  24.05    2  23.99    3  23.95
 4  23.93    5  23.97    6  24.02
 7  24.06    8  24.10    9  23.98
10  24.03   11  23.91   12  24.06
13  24.06   14  23.96   15  23.98
16  24.06   17  24.01   18  24.00
19  23.93   20  23.92   21  24.09
22  24.11   23  24.05   24  23.98
25  23.98   26  24.06   27  24.02
28  24.06   29  23.97   30  23.96
;
```

The following statements use the IRCHART statement in the SHEWHART procedure (see Chapter 34, "IRCHART Statement," in Part 9, "The SHEWHART Procedure") to create a data set containing the control limits for individual measurements and moving range charts for DIAMETER:

```
proc shewhart data=tires;
   irchart diameter*sample / nochart
                             outlimits=tlimits;
run;
```

A listing of the data set TLIMITS is shown in Output 20.4.1.

Output 20.4.1. Listing of the Data Set TLIMITS

```
                    Control Limits for Diameter Measurements

  _VAR_      _SUBGRP_     _TYPE_     _LIMITN_       _ALPHA_     _SIGMAS_

 diameter     sample     ESTIMATE        2       .002699796        3

  _LCLI_     _MEAN_     _UCLI_     _LCLR_       _R_        _UCLR_     _STDDEV_

 23.8571    24.0083    24.1596       0      0.056897    0.18585    0.050423
```

The upper and lower control limits for the diameter measurements are 24.1596 and 23.8571, respectively.

In this example, reference lines will be used to display the control limits for the individual measurements on the EWMA chart. The following DATA step reads these control limits from TLIMITS and creates a data set named VREFDATA, which contains the reference line information:

```
data vrefdata;
   set tlimits;
   length _reflab_ $16.;
   keep _ref_ _reflab_;
   _ref_ = _lcli_; _reflab_= 'LCL for X'; output;
   _ref_ = _ucli_; _reflab_= 'UCL for X'; output;
run;
```

A listing of the data set VREFDATA is shown in Output 20.4.2.

Output 20.4.2. Listing of the Data Set VREFDATA

```
                          Reference Line Information

                             _reflab_      _ref_

                             LCL for X    23.8571
                             UCL for X    24.1596
```

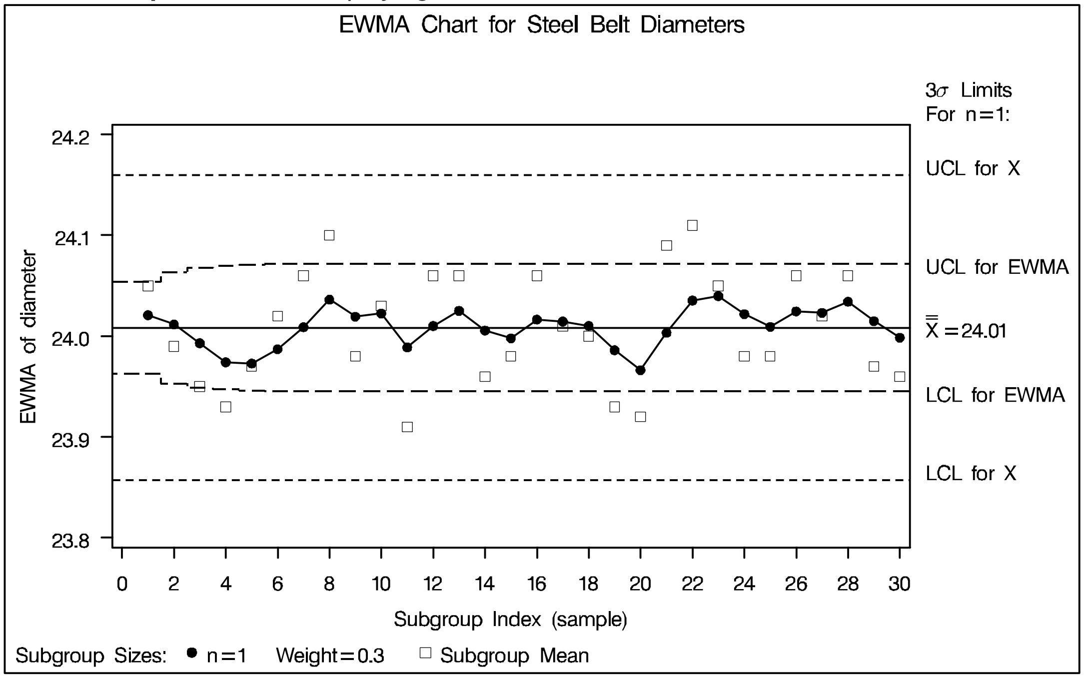
The following statements request an EWMA chart for these measurements:

```
title 'EWMA Chart for Steel Belt Diameters';
proc macontrol data=tires;
   ewmachart diameter*sample / weight     = 0.3
                               meansymbol = square
                               lcllabel   = 'LCL for EWMA'
                               ucllabel   = 'UCL for EWMA'
                               vref       = vrefdata
                               vreflabpos = 3;
run;
```

The MEANSYMBOL= option displays the individual measurements on the EWMA chart. By default, these values are not displayed. The MEANSYMBOL= option specifies the symbol used to plot the individual measurements. The VREF= option reads the reference line information from VREFDATA. The resulting chart is shown in Output 20.4.3.

Output 20.4.3 indicates that the process is in control. None of the diameter measurements (indicated by squares) exceed their control limits, and none of the EWMAs exceed their limits.

Output 20.4.3. Displaying Individual Measurements on EWMA Chart



Example 20.5. Computing Average Run Lengths

See MACEW6 in the SAS/QC Sample Library

The EWMAARL DATA step function computes the average run length for an exponentially weighted moving average (EWMA) scheme (refer to Crowder 1987a,b for details). You can use this function to design a scheme by first calculating average run lengths for a range of values for the weight and then choosing the weight that yields a desired average run length.

The following statements compute the average run lengths for shifts between 0.5 and 2 and weights between 0.25 and 1. The data set ARLS is displayed in Output 20.5.1.

```
data arls;
   do shift=.5 to 2 by .5;
      do weight=.25 to 1 by .25;
        arl=ewmaarl(shift,weight,3.0);
        output;
      end;
   end;
run;
```

```
title 'Average Run Lengths for Various Shifts and Weights';
proc print data=arls noobs;
   by shift;
run;
```

Output 20.5.1. Listing of the Data Set ARLS

```
          Average Run Lengths for Various Shifts and Weights

-------------------------------- shift=0.5 ---------------------------------

                            weight      arl

                             0.25      48.453
                             0.50      75.354
                             0.75     110.950
                             1.00     155.224


--------------------------------- shift=1 ----------------------------------

                            weight      arl

                             0.25     11.1543
                             0.50     15.7378
                             0.75     25.6391
                             1.00     43.8947


-------------------------------- shift=1.5 ---------------------------------

                            weight      arl

                             0.25      5.4697
                             0.50      6.1111
                             0.75      8.7201
                             1.00     14.9677


--------------------------------- shift=2 ----------------------------------

                            weight      arl

                             0.25     3.61677
                             0.50     3.46850
                             0.75     4.15346
                             1.00     6.30296
```

Note that when the weight is 1.0, the EWMAARL function returns the average run length for a Shewhart chart for means. For more details, see "EWMAARL Function" on page 1852.

In addition to using the EWMAARL function to design a EWMA scheme with desired average run length properties, you can use it to evaluate an existing scheme. For example, suppose you have an EWMA chart with 3σ control limits using a weight parameter of 0.3. The following DATA step computes the average run lengths for various shifts using this scheme:

```
data arlinfo;
   do shift=0 to 2 by .25;
      arl = ewmaarl(shift,0.3,3.0);
      output;
   end;
run;

title 'Average Run Lengths for EWMA Scheme (k=3 and r=0.3)';
proc print data=arlinfo noobs;
run;
```

The data set ARLINFO is displayed in Output 20.5.2.

Output 20.5.2. Listing of the Data Set ARLINFO

```
          Average Run Lengths for EWMA Scheme (k=3 and r=0.3)

                           shift      arl

                            0.00    465.553
                            0.25    178.741
                            0.50     53.160
                            0.75     21.826
                            1.00     11.699
                            1.25      7.525
                            1.50      5.447
                            1.75      4.258
                            2.00      3.506
```

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/QC® User's Guide, Version 8*, Cary, NC: SAS Institute Inc., 1999. 1994 pp.

SAS/QC® User's Guide, Version 8

Copyright © 1999 SAS Institute Inc., Cary, NC, USA.

ISBN 1–58025–493–4

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, by any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227–19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, October 1999

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute in the USA and other countries.® indicates USA registration.

IBM®, ACF/VTAM®, AIX®, APPN®, MVS/ESA®, OS/2®, OS/390®, VM/ESA®, and VTAM® are registered trademarks or trademarks of International Business Machines Corporation. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

The Institute is a private company devoted to the support and further development of its software and related services.