SAS Publishing

TECHNICAL REPORT

SAS/QC® Software: Changes and Enhancements, Release 8.2

The Power to Know™

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/QC® Software: Changes and Enhancements, Release 8.2*, Cary, NC: SAS Institute Inc., 2001

SAS/QC® Software: Changes and Enhancements, Release 8.2

ISBN 1-58025-868-9

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

1st printing, January 2001

Table of Contents

Chapter 1
The CAPABILITY Procedure

Chapter Table of Contents

Chapter 1
The CAPABILITY Procedure

Overview

You can specify several new options to enhance histograms and comparative histograms produced by the CAPABILITY procedure. In both the COMPHISTOGRAM and HISTOGRAM statements you can specify:

- the BARLABEL= option to display labels over histogram bars
- the CLIPSPEC= option to clip histogram bars at the specification limits
- the ENDPOINTS= option to label histogram endpoints instead of midpoints

In the COMPHISTOGRAM statement you can specify the CTEXTSIDE= and CTEXTTOP= options to control the colors of row and column labels. In the HISTOGRAM statement you can use the NMIDPOINTS= or NENDPOINTS= option to specify the number of histogram interval midpoints or endpoints.

The table produced by the SPECIALINDICES option in the PROC CAPABILITY statement now contains additional specialized process capability indices. The new indices are S_{jkp}, C_{pp}, C''_{pp}, C_{pg}, C_{pq}, C_p^W, C_{pk}^W, C_{pm}^W, C_{pc}, $C_p(u, v)$, and $C_p(v)$.

Syntax

PROC CAPABILITY Statement

You can assign parameter values for several of the specialized capability indices calculated when you specify the SPECIALINDICES option.

SPECIALINDICES <(*parameter-list***)>**

requests a table of specialized process capability indices. These indices are k, Boyles' modified C_{pm} (also denoted as C_{pm+}), C_{jkp}, $C_{pm}(a)$, $C_p(5.15)$, $C_{pk}(5.15)$, C_{pmk}, Wright's C_s, Boyles' S_{jkp}, C_{pp}, C''_{pp}, C_{pg}, C_{pq}, C_p^W, C_{pk}^W, C_{pm}^W, C_{pc}, and Vännmann's $C_p(u, v)$ and $C_p(v)$. You can provide values for the parameters a for $C_{pm}(a)$, u and v for $C_p(u, v)$ and $C_p(v)$, and for the γ multiplier for C_s by specifying the following options in parentheses after the SPECIALINDICES option.

CPMA=*value*

specifies the *value* of the parameter a for the capability index $C_{pm}(a)$ described in Section 3.7 of Kotz and Johnson (1993). The *value* must be positive. The default *value* is 0.5. The existing CPMA= option in the PROC CAPABILITY statement is considered obsolete but still works.

CPU=*value*
specifies the *value* of the parameter u for Vännmann's capability index $C_p(u, v)$. The *value* must be greater than or equal to zero. The default *value* is zero.

CPV=*value*
specifies the *value* of the parameter v for Vännmann's capability indices $C_p(u, v)$ and $C_p(v)$. The *value* must be greater than or equal to zero. The default *value* is 4.

CSGAMMA=*value*
specifies the *value* of the γ multiplier suggested by Chen and Kotz (1996) for Wright's capability index C_s. The *value* must be greater than zero. The default *value* is 1.

COMPHISTOGRAM Statement

BARLABEL=COUNT | PERCENT | PROPORTION
displays labels above the histogram bars. You can specify BARLABEL=COUNT to display the number of observations associated with a given bar, BARLABEL=PERCENT to display the percentage of observations, and BARLABEL=PROPORTION to display the proportion of observations.

CLIPSPEC=CLIP | NOFILL
specifies that histogram bars are to be clipped at the upper and lower specification limit lines when there are no observations outside the specification limits. The bar intersecting the lower specification limit is clipped if there are no observations less than the lower limit; the bar intersecting the upper specification limit is clipped if there are no observations greater than the upper limit. If you specify CLIPSPEC=CLIP, histogram bars are truncated at the specification limits. If you specify CLIPSPEC=NOFILL, the portion of a filled histogram bar outside the specification limit is left unfilled. Specifying CLIPSPEC=NOFILL when histogram bars are not filled has no effect.

CTEXTSIDE=*color*
specifies the color of the row labels displayed along the left side of a comparative histogram. By default, the CTEXT= color is used for these labels.

CTEXTTOP=*color*
specifies the color of the column labels displayed along the top of a comparative histogram. By default, the CTEXT= color is used for these labels.

ENDPOINTS=*value-list* **| KEY | UNIFORM**
specifies that comparative histogram interval endpoints, rather than midpoints, are to be aligned with horizontal axis tick marks, and specifies how the endpoints are to be determined. The method you specify is used for all process variables analyzed with the COMPHISTOGRAM statement.

If you specify ENDPOINTS=*value-list*, the *values* must be listed in increasing order and must be evenly spaced. The number of endpoint values is one greater than the number of intervals in the resulting comparative histogram. The first value is the lower bound of the first interval and the last value is the upper bound of the last interval. If the range of the *values* does not cover the range of the data and any specification limits (LSL and USL) that are given, the list is extended in both directions as necessary to cover these values.

If you specify ENDPOINTS=KEY, the procedure first determines the endpoints for the data in the key cell. The initial number of intervals is based on the number of observations in the key cell using the method of Terrell and Scott (1985). The endpoint list for the key cell is then extended in both directions as necessary until it spans the data in the remaining cells. If the key cell contains no observations, the method of determining intervals reverts to ENDPOINTS=UNIFORM.

If you specify ENDPOINTS=UNIFORM, the procedure determines the endpoints using all the observations as if there were no cells. In other words, the number of endpoints is based on the total sample size using the method of Terrell and Scott (1985).

HISTOGRAM Statement

ENDPOINTS
ENDPOINTS=*value-list*

specifies that histogram interval endpoints, rather than midpoints, are to be aligned with horizontal axis tick marks. If you specify ENDPOINTS=*value-list*, the *values* must be listed in increasing order and must be evenly spaced. The number of endpoint values is one greater than the number of intervals in the resulting histogram. The first value is the lower bound of the first interval and the last value is the upper bound of the last interval. If the range of the *values* does not cover the range of the data and any specification limits (LSL and USL) that are given, the list is extended in both directions as necessary to cover these values.

If you specify ENDPOINTS, the procedure determines the number of intervals using the method of Terrell and Scott (1985), with interval boundaries at horizontal axis tick mark values.

NENDPOINTS=*value*

specifies the number of endpoints for histogram intervals, which is equal to the number of intervals plus one. If you specify the NENDPOINTS= option you do not need to specify the ENDPOINTS option.

NMIDPOINTS=*value*

specifies the number of midpoints for histogram intervals.

Details

Specialized Capability Indices

This section describes new specialized capability indices that are computed when you specify the SPECIALINDICES option on the PROC CAPABILITY statement.

The Index S_{jkp}

Boyles (1994) proposed a smooth version of C_{jkp} defined as

$$S_{jkp} = S\left(\frac{\text{USL} - T}{\sqrt{2E_{X>T}[(X-T)^2]}}, \frac{T - \text{LSL}}{\sqrt{2E_{X<T}[(X-T)^2]}}\right)$$

The CAPABILITY procedure estimates S_{jkp} as

$$\widehat{S}_{jkp} = S\left(\frac{\text{USL} - T}{\sqrt{2\sum_{X_i>T}(X_i-T)^2/n}}, \frac{T - \text{LSL}}{\sqrt{2\sum_{X_i<T}(X_i-T)^2/n}}\right)$$

where $S(x,y) = \Phi^{-1}[\{\Phi(x) + \Phi(y)\}/2]/3$.

The Index C_{pp}

Chen (1998) devised a process incapability index based on the C^*_{pm} index. The first term measures *inaccuracy* and the second measures *imprecision*. The C_{pp} index is estimated as

$$\widehat{C}_{pp} = \left(\frac{\bar{X} - T}{d^*/3}\right)^2 + \left(\frac{s}{d^*/3}\right)^2$$

where $d^* = \min(\text{USL} - T, T - \text{LSL})$.

The Index C''_{pp}

The index C_{pp} does not handle asymmetric tolerances well, as discussed by Kotz and Lovelace (1998). To address that shortcoming, Chen (1998) defined the index C''_{pp}, which is estimated by

$$\widehat{C}''_{pp} = \left(\frac{\widehat{A}}{d^*/3}\right)^2 + \left(\frac{s}{d^*/3}\right)$$

where

$$\widehat{A} = \max\left\{\frac{(\bar{X} - T)d}{T - \text{LSL}}, \frac{(T - \bar{X})d}{\text{USL} - T}\right\}$$

and $d = (USL - LSL)/2$.

The Index C_{pg}

Marcucci and Beazley (1988) defined the index

$$C_{pg} = \frac{1}{C_{pm}^2}$$

which is estimated as

$$\widehat{C}_{pg} = \frac{1}{\widehat{C}_{pm}^2}$$

The Index C_{pq}

Gupta and Kotz (1997) introduced the index C_{pq}, which is estimated by

$$\widehat{C}_{pq} = \widehat{C}_p \left[1 - \frac{1}{2} \left(\frac{\bar{X} - T}{s} \right)^2 \right]$$

The Index C_p^W

Bai and Choi (1997) defined the index

$$C_p^W = \frac{C_p}{\sqrt{1 + |1 - 2P_x|}}$$

where $P_x = \Pr(X \leq \mu)$. It is estimated by

$$\widehat{C}_p^W = \frac{\widehat{C}_p}{\sqrt{1 + |1 - 2\widehat{P}_x|}}$$

where $\widehat{P}_x$ is the fraction of observations less than or equal to $\bar{X}$. For more information on C_p^W, see Kotz and Lovelace (1998).

The Index C_{pk}^W

Bai and Choi (1997) also proposed the index

$$C_{pk}^W = \min \left\{ \frac{\text{USL} - \mu}{3\sigma\sqrt{2P_x}}, \frac{\mu - \text{LSL}}{3\sigma\sqrt{2(1 - P_x)}} \right\}$$

It is estimated by

$$\widehat{C}_{pk}^W = \min \left\{ \frac{\text{USL} - \bar{X}}{3s\sqrt{2\widehat{P}_x}}, \frac{\bar{X} - \text{LSL}}{3s\sqrt{2(1 - \widehat{P}_x)}} \right\}$$

where $\widehat{P}_x$ is the fraction of observations less than or equal to $\bar{X}$. For more information on C_{pk}^W, see Kotz and Lovelace (1998).

The Index C_{pm}^W

The index C_{pm}^W, also introduced by Bai and Choi (1997), is defined as

$$C_{pm}^{W} = \frac{C_{pm}}{\sqrt{1 + |1 - 2P_T|}}$$

where $P_T = \Pr(X \leq T)$. It is estimated by

$$\widehat{C}_{pm}^{W} = \frac{\widehat{C}_{pm}}{\sqrt{1 + |1 - 2\widehat{P}_T|}}$$

where $\widehat{P}_T$ is the fraction of observations less than or equal to T. For more information on C_{pm}^{W}, see Kotz and Lovelace (1998).

The Index C_{pc}

Luceño (1996) proposed the index

$$C_{pc} = \frac{\text{USL} - \text{LSL}}{6\sqrt{\frac{\pi}{2}E|X - M|}}$$

where $M = (\text{USL} + \text{LSL})/2$. It is estimated by

$$\widehat{C}_{pc} = \frac{\text{USL} - \text{LSL}}{6\sqrt{\frac{\pi}{2}c}}$$

where

$$c = \frac{1}{n}\sum_{i=1}^{n}|X_i - M|$$

Vännmann's Index $C_p(u, v)$

Vännmann (1995) introduced the generalized index $C_p(u, v)$, which reduces to the following capability indices given appropriate choices of u and v:

- $C_p(0, 0) = C_p$
- $C_p(0, 1) = C_{pk}$
- $C_p(1, 0) = C_{pm}$
- $C_p(1, 1) = C_{pmk}$

$C_p(u, v)$ is defined as

$$C_p(u, v) = \frac{d - u|\mu - M|}{3\sqrt{\sigma^2 + v(\mu - T)^2}}$$

and estimated by

$$\widehat{C}_p(u, v) = \frac{d - u|\bar{X} - M|}{3\sqrt{(\frac{n-1}{n})s^2 + v(\bar{X} - T)^2}}$$

You can specify u with the SPECIALINDICES(CPU=) option and v with the SPECIALINDICES(CPV=) option. By default, $u = 0$ and $v = 4$.

Vännmann's Index $C_p(v)$

Vännmann (1997) also proposed the index $C_p(v)$, which is equivalent to $C_p(u,v)$ with $u = 1$. It is estimated as

$$\widehat{C}_p(v) = \frac{d - |\bar{X} - M|}{3\sqrt{(\frac{n-1}{n})s^2 + v(\bar{X} - T)^2}}$$

You can specify v with the SPECIALINDICES(CPV=) option. By default, $v = 4$.

The Modified Index C_s

Chen and Kotz (1996) proposed a modification to Wright's C_s index which introduces a multiplier, $\gamma > 0$, and is estimated as

$$\widehat{C}_s = \frac{(\mathrm{USL} - \mathrm{LSL})/2 - |\bar{\mathrm{X}} - \mathrm{m}|}{3\sqrt{\left(\frac{\mathrm{n}-1}{\mathrm{n}}\right)\mathrm{s}^2 + (\bar{\mathrm{X}} - \mathrm{T})^2 + \gamma|\mathrm{c}_4\mathrm{s}^2\mathrm{b}_3|}}$$

You can specify a value for γ with the SPECIALINDICES(CSGAMMA=) option.

Histogram Interval Midpoints and Endpoints

By default, histogram intervals are centered on the major tick marks of the horizontal axis. The major tick mark values are the *midpoints* of the histogram intervals. The midpoints are evenly spaced, and the difference between consecutive midpoints is used as the width of the histogram bars. The CAPABILITY procedure determines the number of intervals using the algorithm described in Terrell and Scott (1985).

You can specify the number of intervals and their midpoint values with the MIDPOINTS=*value-list* option in a HISTOGRAM or COMPHISTOGRAM statement. In a COMPHISTOGRAM statement, you can specify MIDPOINTS=KEY or MIDPOINTS=UNIFORM. If you specify MIDPOINTS=KEY, the procedure first determines the midpoints for the data in the key cell using the method of Terrell and Scott (1985). It then extends the midpoint list for the key cell in both directions as necessary until it spans the data in the remaining cells. If you specify MIDPOINTS=UNIFORM, the procedure determines the midpoints using all the observations as if there were no cells.

In a HISTOGRAM statement you can specify the number of histogram intervals using the NMIDPOINTS= option. The CAPABILITY procedure then determines the midpoint values.

You can specify the ENDPOINTS option in a HISTOGRAM statement to produce a histogram whose interval boundaries, or *endpoints*, are aligned with major tick marks on the horizontal axis. The procedure uses the method of Terrell and Scott (1985) to determine the number of histogram intervals. The number of major tick marks on the horizontal axis of the resulting histogram is the number of intervals plus one. The least major tick mark value is the lower bound of the first interval and the greatest value is the upper bound of the last interval.

You can use the ENDPOINTS=*value-list* option in a HISTOGRAM or COMPHISTOGRAM statement to specify the number of interval endpoints and their values. The number of values is one greater than the number of intervals on the resulting histogram. You can specify ENDPOINTS=KEY or ENDPOINTS=UNIFORM in a COMPHISTOGRAM statement. The number of intervals is determined just as with MIDPOINTS=KEY or MIDPOINTS=UNIFORM, but the interval boundary values will not be the same, in general.

You can use the NENDPOINTS= option in a HISTOGRAM statement to align interval endpoints with major tick marks and to specify the number of endpoints. The procedure determines the endpoint values. If you specify the ENDPOINTS= or NENDPOINTS= option, you need not specify the ENDPOINTS option.

The ENDPOINTS=, MIDPOINTS=, NENDPOINTS=, and NMIDPOINTS= options are mutually exclusive. The MIDPOINTS= option takes precedence over the NMIDPOINTS= option, and the ENDPOINTS= options takes precedence over the NENDPOINTS= option. The ENDPOINTS= and NENDPOINTS= options override either MIDPOINTS= or NMIDPOINTS=.

Identifying Histogram Intervals

When you specify the OUTHISTOGRAM= or MIDPERCENTS option in a HISTOGRAM or COMPHISTOGRAM statement, the resulting data set or table identifies histogram intervals by their midpoint values by default. The _MIDPT_ variable identifies intervals in an OUTHISTOGRAM= data set. When the ENDPOINTS, ENDPOINTS=, or NENDPOINTS= option is specified, interval endpoints are labeled. Each interval has two endpoints, but is identified in a data set or table by one endpoint value. By default, the left endpoint is used and an OUTHISTOGRAM= data set contains the variable _MINPT_. If the RTINCLUDE option is specified, right endpoints are used to identify intervals and the OUTHISTGRAM= data set variable is _MAXPT_.

Examples

Example 1.1. Clipping Histogram Bars

A semiconductor manufacturer produces printed circuit boards that are sampled to determine whether the thickness of their copper plating lies between a lower specification limit of 3.45 mils and an upper specification limit of 3.55 mils. The plating process is assumed to be in statistical control. The plating thicknesses of 100 boards are saved in a data set named trans, created by the following statements:

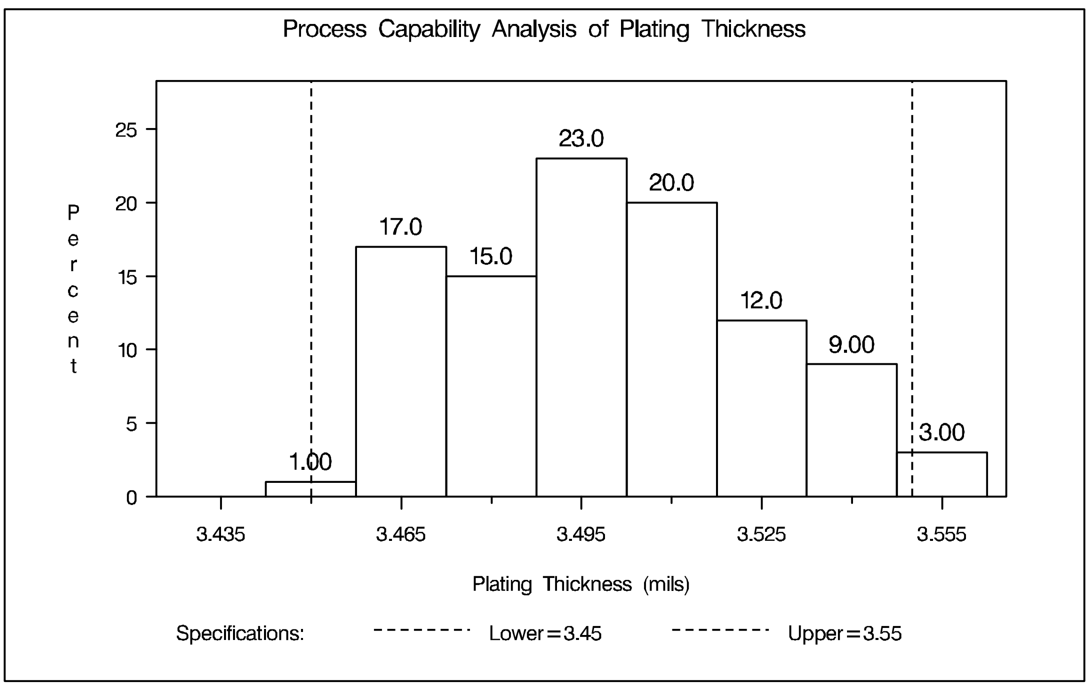
```
data trans;
   input thick @@;
   label thick = 'Plating Thickness (mils)';
   datalines;
3.468 3.488 3.509 3.506 3.481 3.492 3.478 3.546 3.502 3.512
3.490 3.482 3.498 3.519 3.504 3.469 3.497 3.495 3.508 3.523
3.449 3.488 3.463 3.500 3.549 3.525 3.461 3.489 3.514 3.470
```

```
3.461 3.506 3.464 3.489 3.524 3.531 3.501 3.495 3.543 3.510
3.481 3.497 3.461 3.513 3.528 3.496 3.533 3.480 3.516 3.476
3.512 3.550 3.481 3.541 3.549 3.531 3.468 3.494 3.522 3.520
3.505 3.523 3.475 3.470 3.507 3.536 3.528 3.477 3.536 3.491
3.510 3.461 3.531 3.502 3.491 3.506 3.539 3.513 3.496 3.539
3.469 3.481 3.515 3.535 3.460 3.475 3.488 3.515 3.484 3.482
3.517 3.483 3.467 3.467 3.502 3.471 3.516 3.474 3.500 3.466
;
```

The following statements create the histogram shown in Output 1.1.1. Note that the BARLABEL=PERCENT option displays a label above each histogram bar indicating the percentage of observations represented by that bar.

```
title 'Process Capability Analysis of Plating Thickness';
proc capability data=trans noprint;
   spec lsl = 3.45 llsl = 2 clsl = black
        usl = 3.55 lusl = 2 cusl = black;
   histogram thick / cframe   = ligr
                     cfill    = blue
                     barlabel = percent;
run;
```

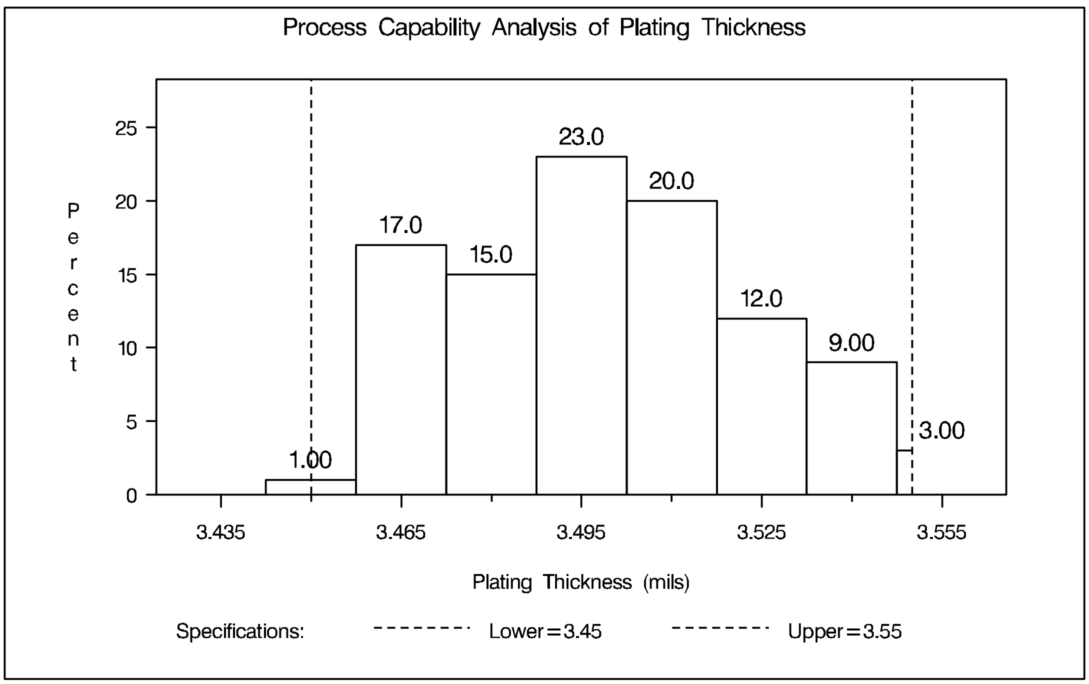
Output 1.1.1. Histogram with Bar Labels



You can use the CLIPSPEC= option to clip histogram bars at the specification limits. A histogram bar intersected by a specification limit is clipped only when no observations lie outside that limit, so clipped bars indicate that all observations are within specifications. These statements create the histogram shown in Output 1.1.2:

```
title 'Process Capability Analysis of Plating Thickness';
proc capability data=trans noprint;
   spec lsl = 3.45 llsl = 2 clsl = black
        usl = 3.55 lusl = 2 cusl = black;
   histogram thick / cframe   = ligr
                     cfill    = blue
                     barlabel = percent
                     clipspec = clip;
run;
```

Output 1.1.2. Histogram with Bar Clipping



The maximum observation in the trans data set is 3.55, which is equal to the upper specification limit (USL). Therefore, the rightmost histogram bar in Output 1.1.2 is clipped at the USL, because no observations lie outside that limit. The lower specification limit (LSL) intersects the leftmost histogram bar. However, the minimum observation is 3.449, which is less than the LSL of 3.45. Therefore, the leftmost histogram bar is not clipped.

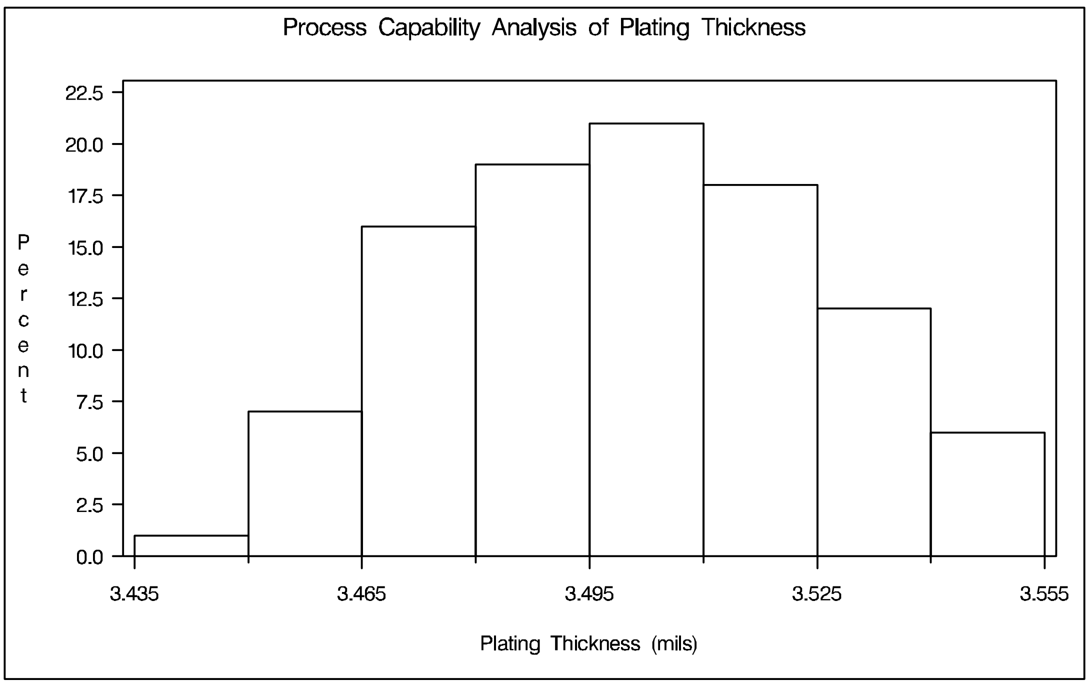
Example 1.2. Labeling Histogram Endpoints

This example uses the trans data set from the preceding example.

By default, major tick marks on the horizontal axis of a histogram correspond to midpoints of the histogram intervals (see Output 1.1.1). You can use the ENDPOINTS= and NENDPOINTS= options to create histograms and comparative histograms whose interval boundaries, or endpoints, are aligned with major tick marks. The following statements create the histogram shown in Output 1.2.1:

```
title 'Process Capability Analysis of Plating Thickness';
proc capability data=trans noprint;
   spec lsl = 3.45 llsl = 2 clsl = black
        usl = 3.55 lusl = 2 cusl = black;
   histogram thick / cframe = ligr
                     cfill  = blue
                     endpoints;
run;
```

Output 1.2.1. Histogram with Endpoints Labeled

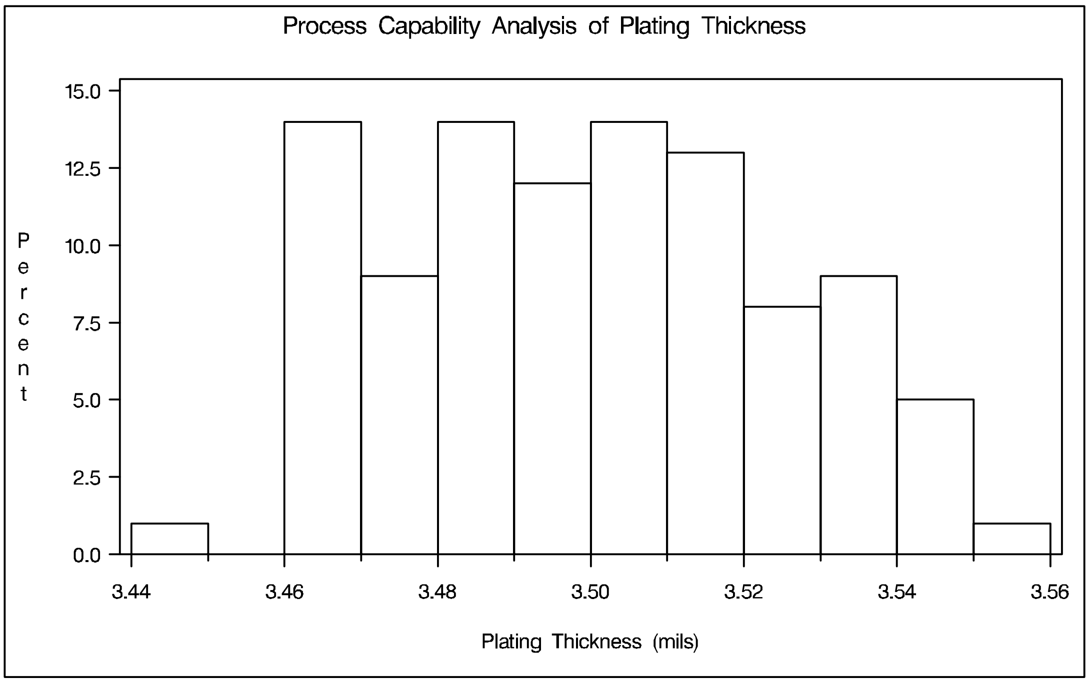


The tick mark values in Output 1.2.1 are identical to those in Output 1.1.1, but the histogram intervals lie between the tick marks. Because their intervals have different boundaries, the histograms in Output 1.1.1 and Output 1.2.1 look somewhat different. Note that a histogram produced with the ENDPOINTS option will not *necessarily* have the same tick mark values on its horizontal axis as a histogram of the same data produced without the ENDPOINTS option.

You can use the NMIDPOINTS= or the NENDPOINTS= option to specify the number of interval midpoints or endpoints in a histogram. The following statements create the histogram shown in Output 1.2.2:

```
title 'Process Capability Analysis of Plating Thickness';
proc capability data=trans noprint;
   spec lsl = 3.45 llsl = 2 clsl = black
        usl = 3.55 lusl = 2 cusl = black;
   histogram thick / cframe     = ligr
                     cfill      = blue
                     nendpoints = 13;
run;
```

Output 1.2.2. Histogram with Number of Endpoints Specified



References

Bai, D. S. and Choi, I. S. (1997), "Process Capability Indices for Skewed Populations," *Manuscript*, Korean Advanced Institute of Science and Technology, Taejon, Korea.

Boyles, R. A. (1994), "Process Capability with Asymmetric Tolerances," *Communication and Statistics, Part B–Simulation and Computation*, 23(3), 615–643.

Chen, H. F. and Kotz, S. (1996), "An Asymptotic Distribution of Wright's Process Capability Index Sensitive to Skewness," *Journal of Statistical Computation and Simulation*, 55, 147–158.

Chen, K. S. (1998), "Incapability Index with Asymmetric Tolerances," *Statistica Sinica*, 8, 253–262.

Gupta, A. K. and Kotz, S. (1997), "A New Process Capability Index," *Metrika*, 45, 213–224.

Kotz, S. and Johnson, N. L. (1993), *Process Capability Indices*, London: Chapman & Hall.

Kotz, S. and Lovelace, C. R. (1998), *Process Capability Indices in Theory and Practice*, London: Arnold.

Luceño, A. (1996), "A Process Capability Index with Reliable Confidence Intervals," *Communications in Statistics – Simulation*, 25(1), 235–245.

Marcucci, M. O. and Beazley, C. F. (1988), "Capability Indices: Process Performance Measures," *Transactions of ASQC Congress*, 516–523.

Terrell, G. R. and Scott, D. W. (1985), "Oversmoothed Nonparametric Density Estimates," *Journal of the American Statistical Association*, 80, 209–214.

Vännmann, K. (1995), "A Unified Approach to Capability Indices," *Statistica Sinica*, 5(2), 805–820.

Vännmann, K. (1997), "A General Class of Capability Indices in the Case of Asymmetric Tolerances," *Communications in Statistics – Theory and Methods*, 26(8), 2049–2072.

Chapter 2
The RELIABILITY Procedure

Chapter Table of Contents

Chapter 2
The RELIABILITY Procedure

Overview

The experimental TRELIABILITY procedure in Release 8.1 of the SAS System has now been incorporated into the RELIABILITY procedure.

The following are the major changes and enhancements available in the RELIABILITY procedure:

- You can specify the logarithm of the distribution scale parameter as a function of explanatory variables in regression models.
- You can construct simultaneous confidence bands on probability plots.
- The default confidence intervals on probability plots are pointwise asymptotic normal parametric intervals on cumulative failure probability. The default intervals in PROC RELIABILITY are pointwise parametric intervals on distribution percentiles.
- Multiple overlaying symbols on plots are labeled with the count of points instead of being jittered as in PROC RELIABILITY.
- You can plot nonlinear relationships on relation plots.
- You can construct relation plots with two independent variables.

Syntax

The following statements and statement options are now available in the RELIABILITY procedure.

LOGSCALE Statement

LOGSCALE <*effect-list*> < **/** *options* >**;**

LOGSCALE is a new statement in the RELIABILITY procedure. You use the LOGSCALE statement to model the logarithm of the distribution scale parameter as a function of explanatory variables. A MODEL statement must be present to specify the model for the distribution location parameter. *effect-list* is a list of variables in the input data set representing the values of the independent variables in the model for each observation, and combinations of variables representing interaction terms. It can contain any variables or combination of variables in the input data set. It can contain the same variables as the MODEL statement, or it can contain different variables. The variables in the *effect-list* can be any combination of indicator variables named in a CLASS statement as well as continuous variables. The coefficients of the explanatory variables are estimated by maximum likelihood.

The following *options* are available for the LOGSCALE statement.

Table 2.1. LOGSCALE Statement Options

Option	Option Description
INITIAL=*number list*	specifies initial values for log-scale regression parameters other than the location, or intercept term
INTERCEPT=*number* < INTINIT >	specifies initial or fixed value of the intercept parameter, depending on whether INTINIT is present

MCFPLOT Statement

The following options are new or can take additional values.

Table 2.2. MCFPLOT Statement Options

Option	Option Description
PLOTSYMBOL=*symbol* | (*symbol list*)	symbols representing events in an MCF plot
PLOTCOLOR=*color* | (*color list*)	colors of symbols representing events in an MCF plot

MODEL Statement

The following are new options in the MODEL statement.

Table 2.3. New MODEL Statement Options

Option	Option Description
RELATION=ARRHENIUS | ARRHENIUS2 | POWER | LOGISTIC RELATION=(ARRHENIUS | ARRHENIUS2 | POWER LOGISTIC < , > ARRHENIUS | ARRHENIUS2 | POWER | LOGISTIC)	the *logistic* function, defined as $T(x) = \log\left(\frac{x}{1-x}\right)$, has been added as an optional transformation of the independent variable in a regression model. The logistic transformation is useful when the variable is naturally between 0 and 1. Values of 0, 1, and values outside 0 and 1 are treated as missing values.

PROBPLOT Statement

The following options are new or can take additional values.

Table 2.4. PROBPLOT Statement Options

Option	Option Description
CFIT=*color* | (*color list*)	color for fit lines and confidence curves in a probability plot
LFIT=*linetype* | (*linetype list)*)	line styles for fit lines and confidence curves in a probability plot. The *linetype list* is a list of numbers from 1 to 46 representing different linetypes, and can be separated by blanks or commas or can be a list in the form n_1 to n_2 $<$by n_3 $>$.
NOPPOS	suppresses plotting of symbols for failures in a probability plot
NPINTERVALS=POINTWIASE | SIMULTANEOUS	type of nonparametric confidence interval displayed in a probability plot
PINTERVALS=PROBABILITY | PERCENTILES | LIKELIHOOD	type of parametric pointwise confidence interval displayed in a probability plot. The default type is PROBABILITY, pointwise confidence intervals on cumulative failure probability.
PPOSSYMBOL=*symbol* | (*symbol list*)	symbols representing failures on a probability plot
PPOSCOLOR=*color* | (*color list*)	colors of symbols representing failures on a probability plot
SHOWMULTIPLES	display the count for multiple overlaying symbols

RELATIONPLOT Statement

The following options are new or can take additional values.

Table 2.5. RELATIONPLOT Statement Options

Option	Option Description
CFIT=*color* | (*color list*)	color for fit lines and confidence curves in a probability plot
CPLOTFIT=*color* | (*color list*)	colors for percentile lines
FITTYPE=	specifies method of estimating distribution parameters
LSYX	-least squares fit to the probability plot. The probability axis is the dependent variable.
| LSXY	-least squares fit to the probability plot. The lifetime axis is the dependent variable.
| MLE	-maximum likelihood (default)

Table 2.5. RELATIONPLOT Statement Options (continued)

Option	**Option Description**
| MODEL	-use the fit from the preceding MODEL statement
| NONE	-no fit is computed
| REGRESSION	-use the fit from the preceding MODEL statement. Non-linear relations and percentiles from models using two independent variables can be plotted.
| WEIBAYES	-Weibayes method
LFIT=*linetype* | (*linetype list)*)	line styles for fit lines and confidence curves in a probability plot. The *linetype list* is a list of numbers from 1 to 46 representing different linetypes, and can be separated by blanks or commas or can be a list in the form n_1 to n_2 $<$by n_3 $>$.
LPLOTFIT=*linetype* | (*linetype list)*)	line styles for percentile lines. *linetype list* is a list of numbers representing different linetypes, and can be separated by blanks or commas or can be a list in the form n_1 to n_2 $<$by n_3 $>$.
NOPPOS	suppresses plotting of symbols for failures in a probability plot
PINTERVALS=PROBABILITY | PERCENTILES | LIKELIHOOD	type of parametric pointwise confidence interval displayed in a probability plot. The default type is PROBABILITY, pointwise confidence intervals on cumulative failure probability.
RCENCOLOR=*color* | (*color list*)	colors for the symbols representing uncensored, right censored, and left censored observations in a relation plot
RCENSYMBOL=*symbol* | (*symbol list*)	symbols representing right censored and left censored observations in a relation plot. The *symbol* is one of the symbol names (plus, star, square, diamond, triangle, hash, paw, point, dot, circle) or a letter (A–Z).
RELATION=ARRHENIUS | ARRHENIUS2 | POWER | LOGISTIC)	the *logistic* function, defined as $T(x) = \log\left(\frac{x}{1-x}\right)$, has been added as an optional transformation of the stress axis. The logistic transformation is useful when the variable is naturally between 0 and 1. Values of 0, 1, and values outside 0 and 1 are not valid.
SHOWMULTIPLES	display the count for multiple overlaying symbols

Table 2.5. RELATIONPLOT Statement Options (continued)

Option	**Option Description**
variable=number list	allows plots of percentiles from a regression model when two independent variables are used in a MODEL statement *effect list*. The FIT=REGRESSION option must be used with this option. Percentile plots are created for each value of the independent *variable* in the *number list*. *number list* is a list of numeric values separated by blanks or commas, or in the form of a list n_1 to n_2 <by n_3 >.

Examples

The examples in this section illustrate some of the new features in the RELIABILITY procedure.

Example 2.1. Regression Model with Non-Constant Scale

Nelson (1990, p. 272) and Meeker and Escobar (1998, p. 439) analyzed data from a strain-controlled fatigue test on 26 specimens of a type of superalloy. The following SAS statements create a SAS data set containing for each specimen the level of pseudo-stress (PSTRESS), the number of cycles (in thousands) (KCYCLES) until failure or removal from the test, and a variable to indicate whether a specimen failed (F) or was right censored (C) (STATUS):

```
data alloy;
   input pstress kcycles status$ @@;
   cen = ( status = 'C' );
   datalines;
80.3  211.629  F    99.8   43.331  F
80.6  200.027  F   100.1   12.076  F
80.8   57.923  C   100.5   13.181  F
84.3  155.000  F   113.0   18.067  F
85.2   13.949  F   114.8   21.300  F
85.6  112.968  C   116.4   15.616  F
85.8  152.680  F   118.0   13.030  F
86.4  156.725  F   118.4    8.489  F
86.7  138.114  C   118.6   12.434  F
87.2   56.723  F   120.4    9.750  F
87.3  121.075  F   142.5   11.865  F
89.7  122.372  C   144.5    6.705  F
91.3  112.002  F   145.9    5.733  F
;
run;
```

The following statements use PROC RELIABILITY to fit a Weibull regression model with the number of cycles to failure as the response variable. The data set=RESIDS contains standardized residuals created with the ODS OUTPUT statement. The MODEL statement specifies a model quadratic in the log of pseudo-stress for the extreme value location parameter. The quadratic model in pseudo-stress PSTRESS is specified in the MODEL statement, and the RELATION=POW option specifies that the log transformation be applied to PSTRESS in the MODEL statement and the LOGSCALE statement. The LOGSCALE statement specifies the log of the scale parameter as a linear function of the log of PSTRESS. The RPLOT statement specifies a plot of the data and the fitted regresion model versus the variable PSTRESS. The FIT=REGRESSION option specifies plotting the regression model fitted with the preceding MODEL statement. The RELATION=POW option specifies using a log stress axis. The PLOTFIT option specifies plotting the 10th, 50th, and 90th percentiles of the regression model at each stress level. The SLOWER, SUPPER, and LUPPER options control limits on the stress and lifetime axes:

```
ods output ModObstats = Resids;
proc reliability data = alloy;
   distribution Weibull;
   model kcycles*cen(1) = pstress pstress*pstress / relation = pow;
   logscale pstress;
   rplot kcycles*cen(1) = pstress /  fit=regression
                                     relation = pow
                                     plotfit  10 50 90
                                     slower=60 supper=160
                                     lupper=500;

   label pstress = "Pseudo-Stress";
   label kcycles = "Thousands of Cycles";
run;
```

Output 2.1.1 displays the parameter estimates from the fitted regression model. Parameter estimates for both the model for the location parameter and the scale paramater models are shown. Standard errors and confidence limits for all parameter estimates are included.

Output 2.1.1. Parameter Estimates for Fitted Regression Model

The RELIABILITY Procedure

Weibull Parameter Estimates

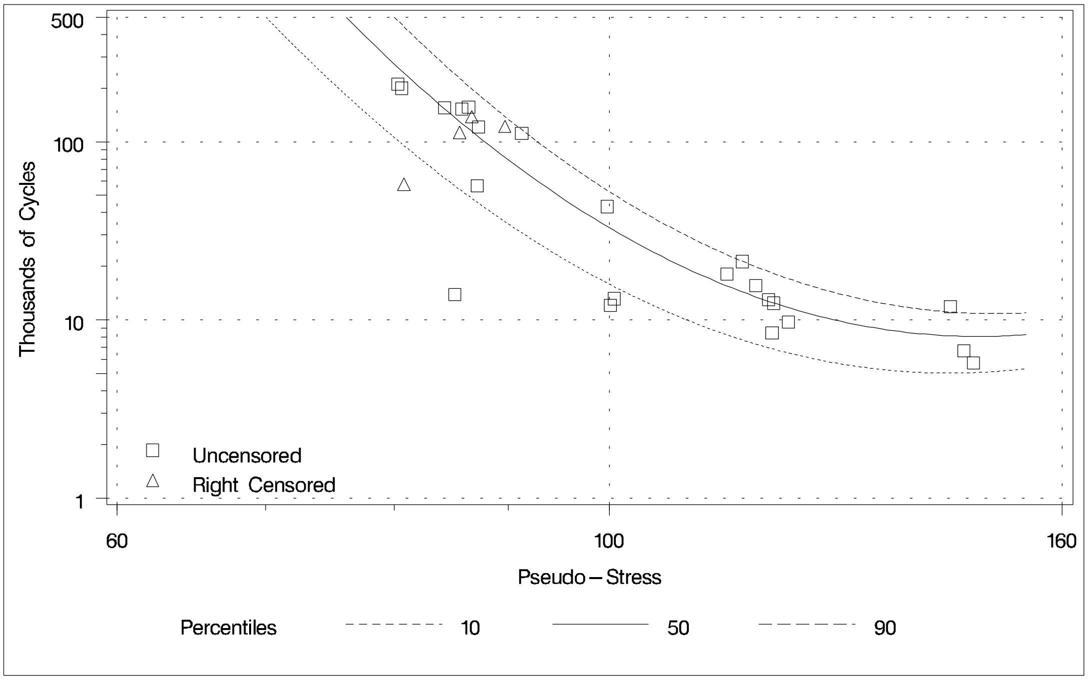
Parameter	Estimate	Standard Error	Asymptotic Normal 95% Confidence Limits Lower	Upper
Intercept	243.1681	58.0666	129.3596	356.9766
pstress	-96.5240	24.7075	-144.9498	-48.0983
pstress*pstress	9.6653	2.6247	4.5210	14.8095

Log-Scale Parameter Estimates

Parameter	Estimate	Standard Error	Asymptotic Normal 95% Confidence Limits Lower	Upper
Intercept	4.4666	4.0724	-3.5152	12.4485
pstress	-1.1757	0.8731	-2.8870	0.5355

Output 2.1.2 displays the plot of the data and fitted regression model.

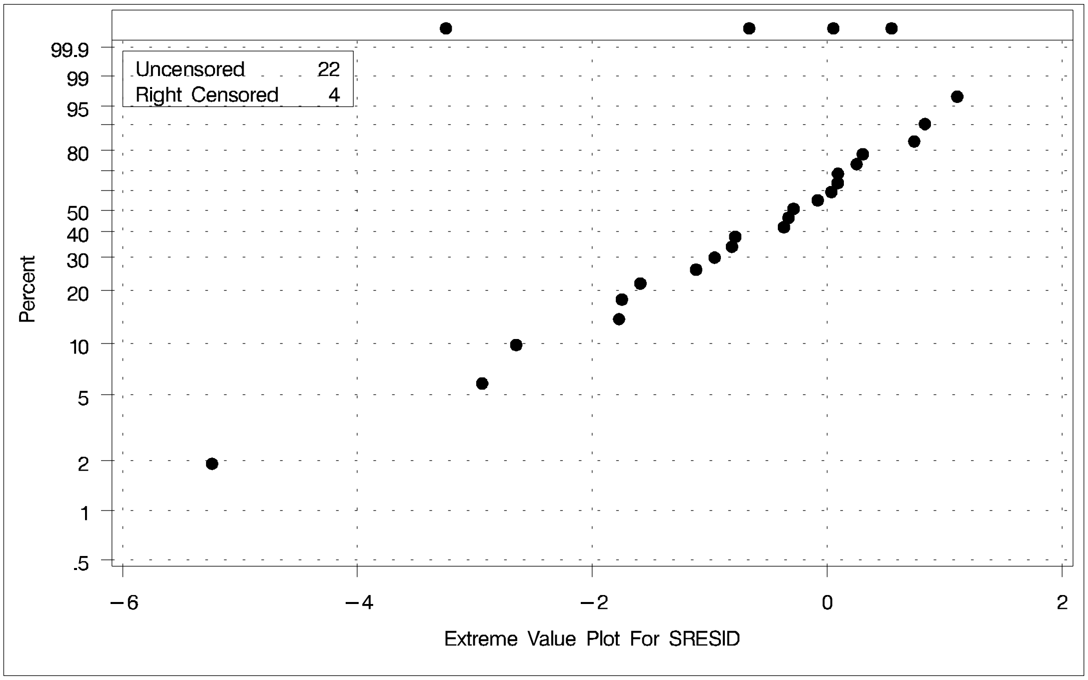
Output 2.1.2. Superalloy Fatigue Data with Fitted Regression Model



The following SAS statements create an extreme values probability plot of standardized residuals from the regression model shown in Output 2.1.3:

```
proc reliability data = Resids;
   distribution ev;
   pplot sresid*cen(1) / nofit;
run;
```

Output 2.1.3. Residuals for Superalloy Fatigue Data Regression Model



Example 2.2. Regression Model with Two Independent Variables

Meeker and Escobar (1998, p. 447) analyze data from an accelerated life test on the lifetimes of glass capacitors as a function of operating voltage and temperature. The following SAS statements create a SAS data set containing the data. There are four lifetimes for each of eight combinations and four censored observations after the fourth failure for each combination:

```
data glass;
   input Temp Voltage @;
   do i = 1 to 4;
      Cen = 0;
      input Hours @;
      output;
   end;
   do i = 1 to 4;
      Cen = 1;
      output;
   end;
   datalines;
170 200  439  904  1092  1105
170 250  572  690   904  1090
170 300  315  315   439   628
170 350  258  258   347   588
180 200  959 1065  1065  1087
180 250  216  315   455   473
180 300  241  315   332   380
180 350  241  241   435   455
;
```

The following statements use PROC RELIABILITY to analyze the capacitor data. The MODEL statement fits a regression model with TEMP and VOLTAGE as independent variables. Parameter estimates from the fitted regression model are shown in Output 2.2.1. An interaction term between TEMP and VOLTAGE is included. The PPLOT statement creates a Weibull probability plot shown in Output 2.2.2 with all temperature-voltage combinations overlaid on the same plot. The regression model fit is also plotted. The RPLOT statement creates the plot shown in Output 2.2.3 of the data and Weibull distribution percentiles from the regression model as a function of voltage for values of temperature of 150, 170, and 180:

```
proc reliability data = glass;
   distribution Weibull;
   model Hours*Cen(1) = temp voltage temp * voltage;

   pplot Hours*Cen(1) = ( temp voltage ) / fit = model
                                           overlay
                                           noconf
                                           lupper = 2000
                                           lfit = ( 1 to 8 );

   rplot Hours*Cen(1) = voltage / fit = regression
                                  plotfit
                                  temp = 150, 170, 180;
run;
```

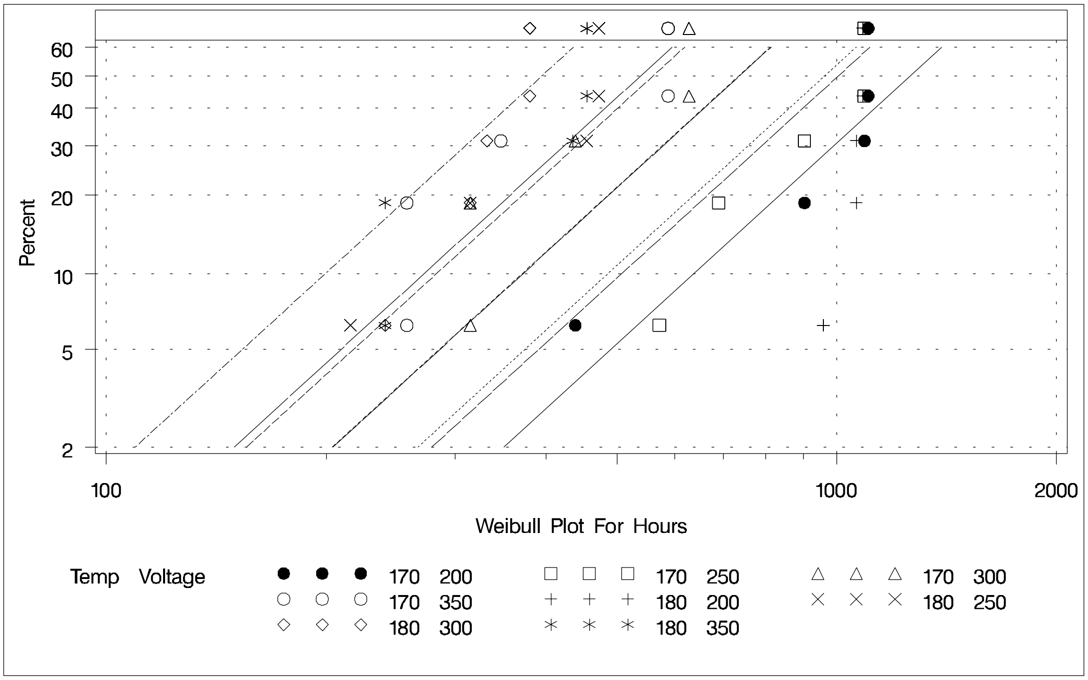
Output 2.2.1. Parameter Estimates for Fitted Regression Model

The RELIABILITY Procedure

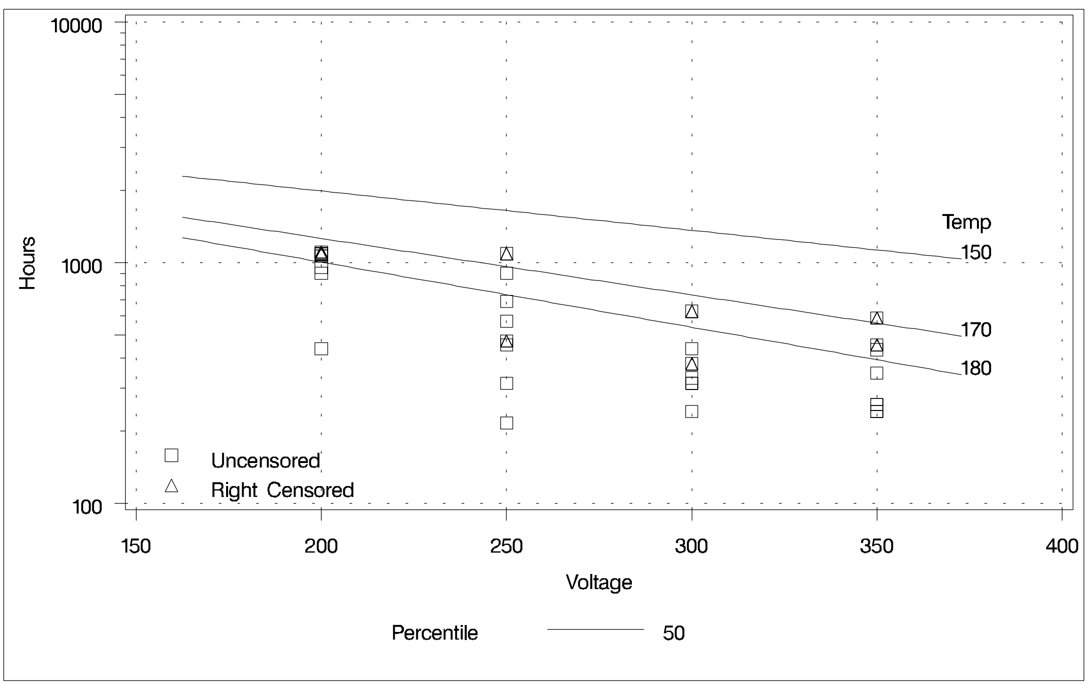
Weibull Parameter Estimates

Parameter	Estimate	Standard Error	Asymptotic Normal 95% Confidence Limits Lower	Upper
Intercept	9.4135	10.5402	-11.2449	30.0719
Temp	-0.0062	0.0598	-0.1235	0.1110
Voltage	0.0086	0.0374	-0.0648	0.0820
Temp*Voltage	-0.0001	0.0002	-0.0005	0.0003
EV Scale	0.3624	0.0553	0.2687	0.4887
Weibull Shape	2.7593	0.4210	2.0461	3.7209

Output 2.2.2. Probability Plot for Glass Capacitor Regression Model



Output 2.2.3. Plot of Data and Fitted Weibull Percentiles for Glass Capacitor Regression Model



References

Meeker, W.Q. and Escobar, L.A. (1998), *Statistical Methods for Reliability Data*, New York: John Wiley & Sons.

Nelson, W. (1990), *Accelerated Testing: Statistical Models, Test Plans, and Data Analyses*, New York: John Wiley & Sons.

Chapter 3

The SHEWHART Procedure

Chapter Table of Contents

Chapter 3
The SHEWHART Procedure

Overview

You can now specify variables from the input data set to be plotted as overlays on control charts. You can control the appearance of the symbols used to plot the overlay variables and the lines connecting the plotted points. You can also specify uniform resource locators (URLs) to be associated with overlay points. When graphics output is directed to HTML, these URLs can provide links to other Web pages.

Using new LIMITS= data set variables, you can vary the appearance of a control chart from one phase to another:

- The _CFRAME_ variable specifies the color of the area enclosed by the axes and frame within a phase.
- The _CINFILL_ variable specifies the color of the area inside the upper and lower control limits within a phase.
- The _CLIMITS_ variable specifies the color for the control limits and central line within a phase.
- The _WLIMITS_ variable specifies the width for the control limits and central line within a phase.
- The _LLMITS_ variable specifies the line type for the control limits within a phase.

The new BOXSTYLE=POINTSSCHEMATIC option in the BOXCHART statement combines the POINTS and SCHEMATIC box styles. The MRRESTART option in the IRCHART statement restarts the moving range computation when a missing value is encountered. The TESTRESET= and TEST2RESET= options reset tests for special causes on the primary and secondary control charts at specified subgroups.

Syntax

BOXSTYLE=POINTSSCHEMATIC

requests a schematic box-and-whisker plot representing each subgroup sample, overlaid with points plotting all the values in the subgroup. This box style combines the POINTS and SCHEMATIC box styles. By default, a square plotting symbol is used to plot the values. You can specify a symbol with the IDSYMBOL= option. You can specify the color of the symbols with the IDCOLOR= option (the default color is the color specified with the CBOXES= option or the second color in the device color list). This option is available only in the BOXCHART statement.

CCOVERLAY=(*color-list***)**
specifies colors for the line segments connecting points on primary chart overlays. Colors in the CCOVERLAY= list are matched with variables in the corresponding positions in the OVERLAY= list. By default, points are connected by line segments of the same color as the plotted points. You can specify the value NONE to suppress the line segments connecting points on an overlay.

CCOVERLAY2=(*color-list***)**
specifies colors for the line segments connecting points on secondary chart overlays. Colors in the CCOVERLAY2= list are matched with variables in the corresponding positions in the OVERLAY2= list. By default, points are connected by line segments of the same color as the plotted points. You can specify the value NONE to suppress the line segments connecting points on an overlay.

COVERLAY=(*color-list***)**
specifies the colors used to plot overlay variables on the primary control chart. Colors in the COVERLAY= list are matched with variables in the corresponding positions in the OVERLAY= list.

COVERLAY2=(*color-list***)**
specifies the colors used to plot overlay variables on a secondary control chart. Colors in the COVERLAY2= list are matched with variables in the corresponding positions in the OVERLAY2= list.

COVERLAYCLIP=(*color***)**
specifies the color used to plot clipped values on overlay plots. The same color is used for clipped points on primary and secondary chart overlays.

LOVERLAY=(*linetypes***)**
specifies line types for the line segments connecting points on primary chart overlays. Line types in the LOVERLAY= list are matched with variables in the corresponding positions in the OVERLAY= list.

LOVERLAY2=(*linetypes***)**
specifies line types for the line segments connecting points on secondary chart overlays. Line types in the LOVERLAY2= list are matched with variables in the corresponding positions in the OVERLAY2= list.

MRRESTART
MRRESTART=*value*
causes the moving range computation on an individual measurements and moving ranges chart to be restarted when a missing value is encountered. Without the MRRESTART option, a missing value is simply skipped, and the moving range for the next non-missing subgroup is computed using the most recent previous non-missing values. MRRESTART restarts the moving range computation, so only the observations after the missing value are used in subsequent moving range computations. You can specify MRRESTART to restart the moving range computation on any missing value, and MRRESTART=*value* to restart only on a particular missing value. For example, MRRESTART=R will restart the computation only when the missing value ".R" is encountered. This option is valid only in the IRCHART statement.

NOOVERLAYLEGEND
suppresses the legend for overlay variables that is displayed by default when the OVERLAY= or OVERLAY2= option is specified.

OVERLAY=(*variable-list***)**
specifies variables to be plotted as overlays on the primary control chart. A point is plotted for each overlay variable at each subgroup for which it has a non-missing value. The value of a particular overlay variable should be the same for each observation in the input data set with a given value of the subgroup variable.

OVERLAY2=(*variable-list***)**
specifies variables to be plotted as overlays on a secondary control chart. A point is plotted for each overlay variable at each subgroup for which it has a non-missing value. The value of a particular overlay variable should be the same for each observation in the input data set with a given value of the subgroup variable.

OVERLAY2HTML=(*variable-list***)**
specifies variables whose values are URLs to be associated with points on secondary chart overlays. These URLs are associated with points on an overlay plot when graphics output is directed into HTML. Variables in the OVERLAY2HTML= list are matched with variables in the corresponding positions in the OVERLAY2= list. The value of the OVERLAY2HTML= variable should be the same for each observation with a given value of the subgroup variable.

OVERLAY2SYM=(*symbol-list***)**
specifies symbols used to plot points on secondary chart overlays. Symbols in the OVERLAY2SYM= list are matched with variables in the corresponding positions in the OVERLAY2= list.

OVERLAY2SYMHT=(*value-list***)**
specifies the heights of symbols used to plot points on secondary chart overlays. Heights in the OVERLAY2SYMHT= list are matched with variables in the corresponding positions in the OVERLAY2= list.

OVERLAYCLIPSYM=(*symbol***)**
specifies the symbol used to plot clipped values on overlay plots. The same symbol is used on primary and secondary chart overlays.

OVERLAYCLIPSYMHT=(*value***)**
specifies the height of the symbol used to plot clipped values on overlay plots. The same height is used on primary and secondary chart overlays.

OVERLAYLEGLAB='*label***'**
specifies the label displayed to the left of the overlay legend. The label can be up to 16 characters long and must be enclosed in quotes.

OVERLAYHTML=(*variable-list***)**
specifies variables whose values are URLs to be associated with points on primary chart overlays. These URLs are associated with points on an overlay plot when graphics output is directed into HTML. Variables in the OVERLAYHTML= list are matched with variables in the corresponding positions in the OVERLAY= list. The value of the OVERLAYHTML= variable should be the same for each observation

with a given value of the subgroup variable.

OVERLAYSYM=(*symbol-list***)**
specifies symbols used to plot points on primary chart overlays. Symbols in the OVERLAYSYM= list are matched with variables in the corresponding positions in the OVERLAY= list.

OVERLAYSYMHT=(*value-list***)**
specifies the heights of symbols used to plot points on primary chart overlays. Heights in the OVERLAYSYMHT= list are matched with variables in the corresponding positions in the OVERLAY= list.

TESTRESET=*variable*
enables tests for special causes to be reset on the primary control chart. The specified variable must be a character variable of length 8, or length 16 if customized tests are requested. The variable values have the same format as those of the _TESTS_ variable in a TABLE= data set. A test that is flagged by the TESTRESET= variable value for a given subgroup is reset starting with that subgroup. That means that a positive result for the test can include the given subgroup only if it is the first subgroup in the pattern. For example, the value “12345678” for the TESTRESET= variable will reset all standard tests for special causes. This option is available in all chart statements.

TEST2RESET=*variable*
enables tests for special causes to be reset on a secondary chart. The specified variable must be a character variable of length 8, or length 16 if customized tests are requested. The variable values have the same format as those of the _TESTS_ variable in a TABLE= data set. A test that is flagged by the TEST2RESET= value for a given subgroup is reset starting with that subgroup. That means a positive result for the test can include the given subgroup only if it is the first subgroup in the pattern. For example, the value “12345678” for the TEST2RESET= variable will reset all standard tests for special causes. This option is available with the MRCHART, RCHART, SCHART, XRCHART, and XSCHART statements.

WOVERLAY=(*value-list***)**
specifies the widths in pixels of the line segments connecting points on primary chart overlays. Widths in the WOVERLAY= list are matched with variables in the corresponding positions in the OVERLAY= list.

WOVERLAY2=(*value-list***)**
specifies the widths in pixels of the line segments connecting points on secondary chart overlays. Widths in the WOVERLAY2= list are matched with variables in the corresponding positions in the OVERLAY2= list.

Details

OVERLAY= and Related Options

Values specified with the COVERLAY=, OVERLAYSYM=, and OVERLAYSYMHT= options are matched with variables in the corresponding positions in the OVERLAY= variable list. Values specified with the COVERLAY2=, OVERLAY2SYM=, and OVERLAY2SYMHT= options are matched with variables in the corresponding positions in the OVERLAY2= variable list. When one of these options is not specified, the associated values are taken from the SYMBOL definitions currently in effect. If an option is specified but lists fewer values than the number of overlay variables, the overlays for the remaining unmatched variables use values from the SYMBOL definitions. SYMBOL1 is used to plot the process variable, so the SYMBOL definitions associated by default with the overlay plots start with SYMBOL2.

The vertical axis of a control chart is scaled to accommodate data for the process variable and all the points on any overlays. If the range of overlay variable values is much greater than the range of process variable values, the control chart may be compressed and difficult to read.

LIMITS= Data Set Extensions

You can specify the value NONE for the _CLIMITS_ variable in a LIMITS= data set to suppress display of the control limits and central line within a phase. You can specify the values NONE or EMPTY for the _CFRAME_ or _CINFILL_ variables to leave the corresponding areas uncolored within a phase.

A value specified with a chart statement option takes precedence over values of the corresponding LIMITS= data set variable. For example, if you specify CINFILL=BLUE in a chart statement, the area between the control limits will be blue for all phases on the chart, overriding any values of the _CINFILL_ variable from the LIMITS= data set.

Examples

Example 3.1. Using BOXSTYLE=POINTSSCHEMATIC

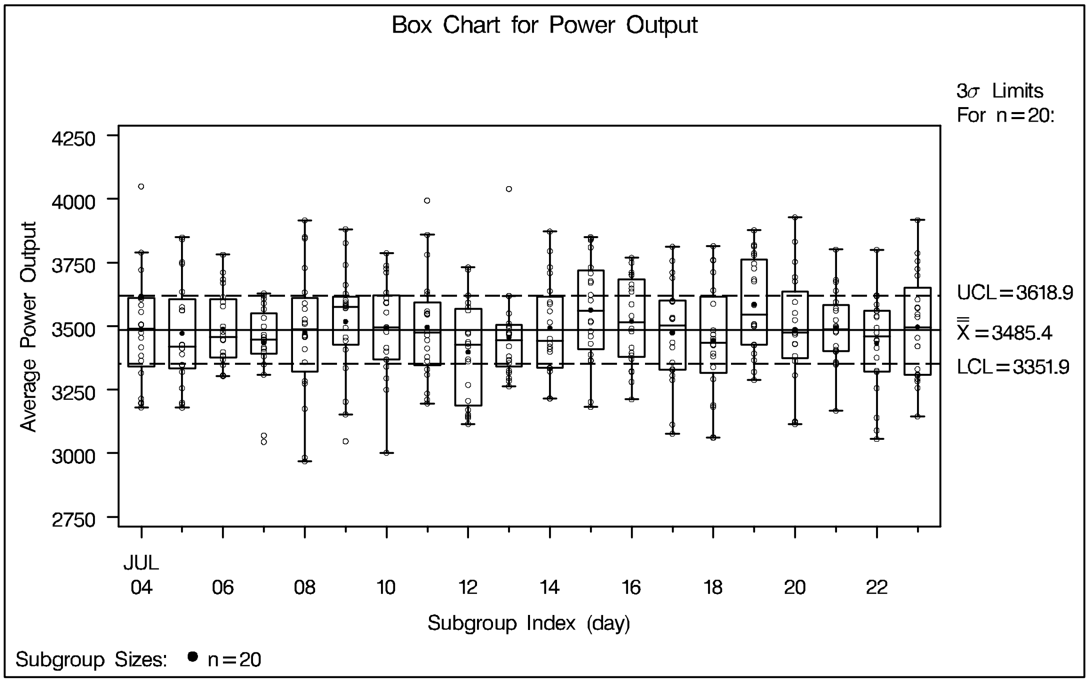
A petroleum company uses a turbine to heat water into steam that is pumped into the ground to make oil more viscous and easier to extract. This process occurs 20 times daily, and the amount of power (in kilowatts) used to heat the water to the desired temperature is recorded. The following statements create a SAS data set that contains the power output measurements for 20 days:

```
data turbine;
   informat day date7.;
   format day date5.;
   label kwatts='Average Power Output';
   input day @;
   do i=1 to 10;
      input kwatts @;
      output;
      end;
   drop i;
datalines;
04JUL94 3196 3507 4050 3215 3583 3617 3789 3180 3505 3454
04JUL94 3417 3199 3613 3384 3475 3316 3556 3607 3364 3721
05JUL94 3390 3562 3413 3193 3635 3179 3348 3199 3413 3562
05JUL94 3428 3320 3745 3426 3849 3256 3841 3575 3752 3347
06JUL94 3478 3465 3445 3383 3684 3304 3398 3578 3348 3369
06JUL94 3670 3614 3307 3595 3448 3304 3385 3499 3781 3711
.
.
.
23JUL94 3421 3787 3454 3699 3307 3917 3292 3310 3283 3536
23JUL94 3756 3145 3571 3331 3725 3605 3547 3421 3257 3574
;
```

You can use the BOXSTYLE=POINTSSCHEMATIC option to create a box chart with all observations in the turbine data set plotted. The following statements create the box chart shown in Output 3.1.1:

```
title 'Box Chart for Power Output';
symbol v=dot c=salmon h=0.4;
proc shewhart data=turbine;
   boxchart kwatts*day / cframe   = vligb
                         cboxes   = dagr
                         cboxfill = ywh
                         cinfill  = ligr
                         boxstyle = pointsschematic
                         idsymbol = circle
                         idcolor  = vigb;
run;
```

Output 3.1.1. Box Chart with BOXSTYLE=POINTSSCHEMATIC



Example 3.2. Using Overlays

This example uses the SHEWHART procedure to do analysis of means (ANOM) for rate data. It is based on an example from Rodriguez (1996). That example is in ANOMP in the SAS/QC sample library.

A health care system uses ANOM to compare cesarean section rates for a set of medical groups. The following statements create a SAS data set named csection:

```
data csection;
   length id $ 2;
   input id csect94 vag94 csect95 vag95;
   total94  = csect94 + vag94;
   total95  = csect95 + vag95;
   label id = 'Medical Group Identification Number';
datalines;
1A      163     907     150     773
1K       55     314      45     253
1B       52     179      34     136
1D       19     128      18     114
3I       21      98      20      86
3M       15      81      12      93
1E       15      52      10      67
1N        6      43      19      55
1Q       12      67       7      62
3H        7      65      11      54
1R        4      43      11      38
1H        5      32       9      39
3J        7      12       7      13
1C       13      42       8      35
3B        3      33       6      37
1M        6       8       4      25
3C        6      27       5      23
1O        8      26       4      23
1J        6      18       6      16
1T        1       3       3      19
3E        2      12       4      14
1G        1       7       4      11
3D        7      22       4       9
3G        2       5       1      10
1L        2       4       2       8
1I        1       3       1       7
1P       16      65       0       3
1F        0       1       0       3
1S        1       2       1       2
;

data csection;
   set csection end=eof;
   if eof then call symput('ngroups', left( put( _n_, 4. )));
   rate94 = csect94 / total94;
run;
```

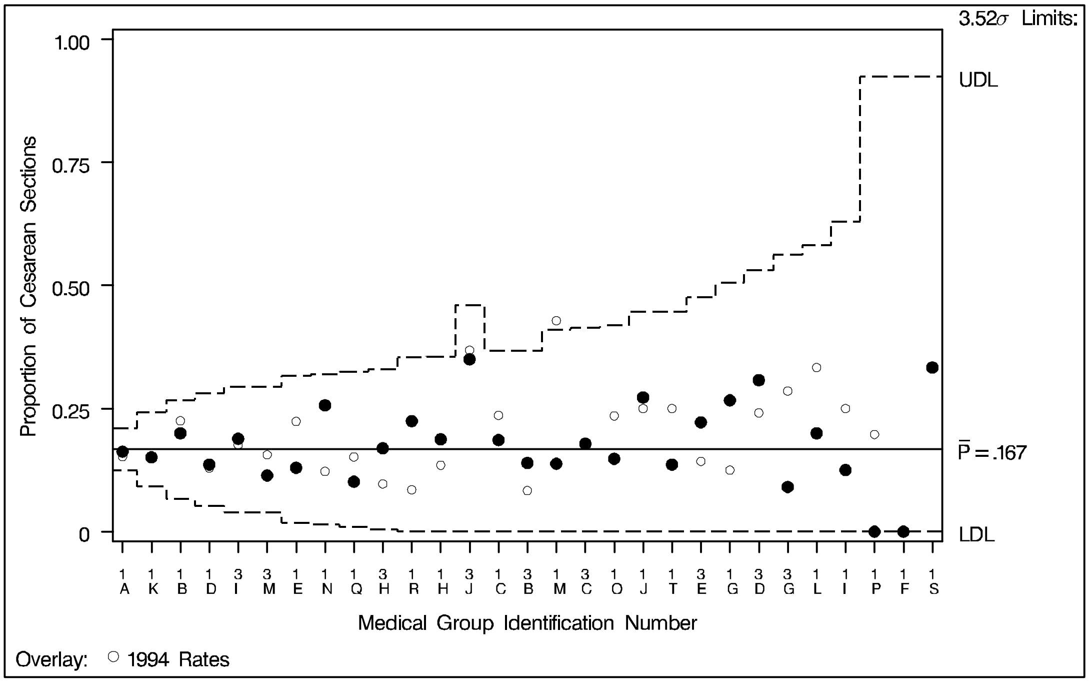
The variable id identifies the medical groups. The variable csect95 provides the number of c-sections for each group during 1995, and the variable total95 provides the total number of deliveries for each group. The variables csect94 and total94 provide similar counts for 1994.

The original example used an annotate data set to overlay rates from 1994 on the ANOM chart of 1995 rates. Now you can use OVERLAY= and related options to produce overlaid plots. The following statements create the ANOM chart shown in Output 3.2.1:

```
%anomsig( 0.01, &ngroups );
title 'Proportion of C-Sections:'
      ' 1994 and 1995';
symbol v=dot c=black h=2.4pct;
proc shewhart data=csection;
   pchart csect95*id / subgroupn=total95
                       noconnect
                       turnhlabels
                       nolegend
                       lcllabel = 'LDL'
                       ucllabel = 'UDL'
                       overlay = (rate94)
                       coverlay = (red)
                       ccoverlay = (none)
                       overlaysym = (circle)
                       overlaysymht = (2.4)
                       haxis = axis1
                       sigmas=&sigmult;
axis1 value = ( h=2.2 pct ) ;
label csect95 = 'Proportion of Cesarean Sections';
label rate94 = '1994 Rates';
run;
```

The anomsig macro computes the appropriate multiple of σ for the SIGMAS= option. It is defined in ANOMSIG in the SAS/QC sample library.

Output 3.2.1. ANOM Chart with Overlay



Example 3.3. Using LIMITS= Data Set Extensions

The following statements create the HISTORY= data set flange, containing means and ranges of flange width measurements for subgroups of size five:

```
data flange;
   input day date7. sample 10-11 _phase_ $14-23
         flwidthx flwidthr flwidthn;
   format   day date7.;
datalines;
08FEB90   6  Production  0.97360  0.06247  5
09FEB90   7  Production  1.00486  0.11478  5
10FEB90   8  Production  1.00251  0.13537  5
11FEB90   9  Production  0.95509  0.08378  5
12FEB90  10  Production  1.00348  0.09993  5
15FEB90  11  Production  1.02566  0.06766  5
16FEB90  12  Production  0.97053  0.07608  5
17FEB90  13  Production  0.94713  0.10170  5
18FEB90  14  Production  1.00377  0.04875  5
19FEB90  15  Production  0.99604  0.08242  5
22FEB90  16  Change 1    0.99218  0.09787  5
23FEB90  17  Change 1    0.99526  0.02017  5
24FEB90  18  Change 1    1.02235  0.10541  5
25FEB90  19  Change 1    0.99950  0.11476  5
26FEB90  20  Change 1    0.99271  0.05395  5
01MAR90  21  Change 1    0.98695  0.03833  5
02MAR90  22  Change 1    1.00969  0.06183  5
03MAR90  23  Change 1    0.98791  0.05836  5
04MAR90  24  Change 1    1.00170  0.05243  5
05MAR90  25  Change 1    1.00412  0.04815  5
08MAR90  26  Change 2    1.00261  0.05604  5
09MAR90  27  Change 2    0.99553  0.02818  5
10MAR90  28  Change 2    1.01463  0.05558  5
11MAR90  29  Change 2    0.99812  0.03648  5
12MAR90  30  Change 2    1.00047  0.04309  5
15MAR90  31  Change 2    0.99714  0.03689  5
16MAR90  32  Change 2    0.98642  0.04809  5
17MAR90  33  Change 2    0.98891  0.07777  5
18MAR90  34  Change 2    1.00087  0.06409  5
19MAR90  35  Change 2    1.00863  0.02649  5
;
run;
```

This DATA step creates the LIMITS= data set flanlim:

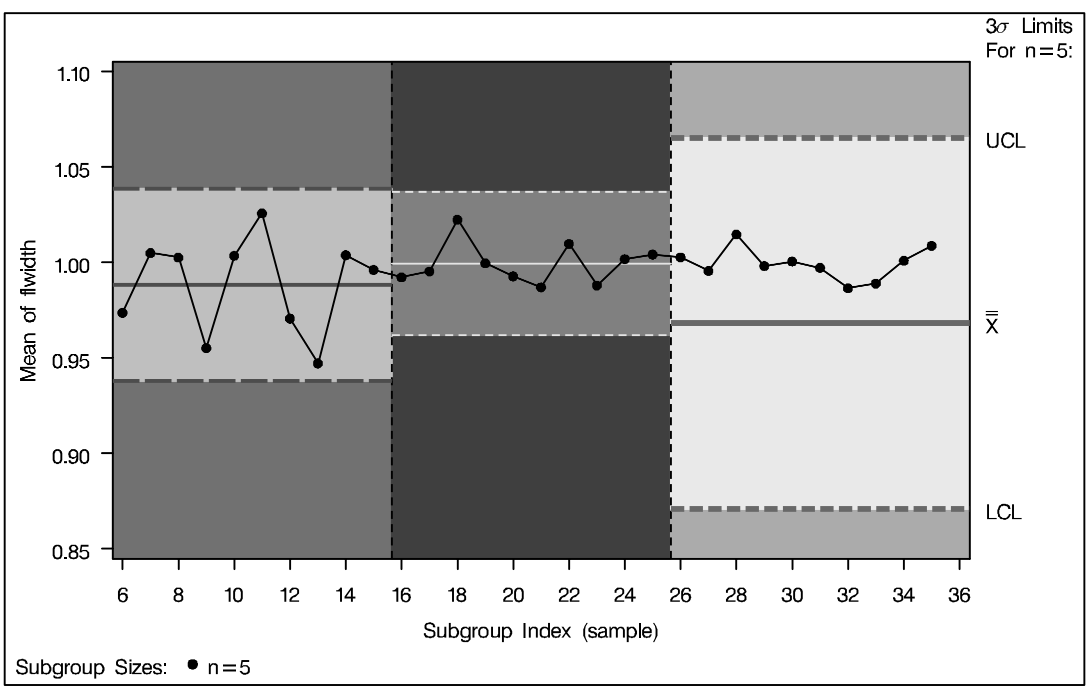
```
data flanlim;
   input _index_ $1-10 _lclx_ _mean_ _uclx_ _r_ _uclr_ _stddev_ @;
   input _cframe_ $ _cinfill_ $ _climits_ $ _wlimits_ _llimits_;
   _var_    = 'FLWIDTH ';
   _subgrp_ = 'SAMPLE  ';
   _type_   = 'ESTIMATE';
   _limitn_ = 5;
```

```
      _sigmas_ = 3;
      _alpha_  = .0026998;
      _lclr_   = 0;
   datalines;
   Production 0.9379 0.9883 1.0386 0.0873 0.1846 0.03753
              steel ligr red 2 4
   Change 1   0.9617 0.9992 1.0368 0.0651 0.1377 0.02800
              vigb vlib yellow 1 2
   Change 2   0.8709 0.9680 1.0652 0.1684 0.3561 0.07241
              vligb ywh vibg 3 3
   ;
   run;
```

The variables _CFRAME_, _CINFILL_, _CLIMITS_, _WLIMITS_, and _LLIMITS_ specify colors, line widths, and line types for displaying three different phases. For example, in the 'Production' phase the specified frame color is steel, the area between the control limits is light gray, the control limits are red, the control limits width is two pixels, and the control limits line style is 4. The following statements produce the chart shown in Output 3.3.1:

```
symbol1 v=dot c=black h=0.4;
proc shewhart history=flange limits=flanlim;
   xchart flwidth*sample /
      readphase = all
      readindex = all
      phaselegend;
run;
```

Output 3.3.1. $\bar{X}$ Chart with Variations in Display of Phases



References

Rodriguez, R. N. (1996), "Health Care Applications of Statistical Process Control: Examples Using the SAS System," *SAS Users Group International: Proceedings of the Twenty-First Annual Conference*, 1381–1396.

Chapter 4

The ADX Interface for Design and Analysis of Experiments

Chapter Table of Contents

Chapter 4

The ADX Interface for Design and Analysis of Experiments

Overview

The ADX Interface is a point-and-click interface for the design and analysis of experiments. Designed for the desktop environment of PCs and workstations, the Interface is intended primarily for engineers and researchers who require a guided interface for every step in the experimental process.

The ADX Interface has been enhanced in several ways. Mixture designs have been extended to include process variables, an import design wizard has been added, and the Design Details window now enables you to view more details of the design.

Process Variables

The ADX interface now enables you to specify process variables for mixture designs. A process variable is a factor in an experiment that is not part of the mixture but whose levels, when varied, could affect the blending properties of the mixture ingredients.

Import Design Wizard

An import design wizard has been added to enable external design data to be imported into ADX. The external data can be stored in a SAS data set or other file type, including Microsoft® Excel. Two-level designs, response surface designs, mixed-level designs, mixture designs, and split-plot designs can be imported.

Design Details

The Design Details window now enables you to view two additional details of the basic design. Variance dispersion plots, for Resolution 5 two-level and response surface designs, provide a variance graph to evaluate a design. For mixture designs, the mixture design points can be viewed graphically.

Subject Index

Syntax Index

Your Turn

If you have comments or suggestions about *SAS/QC® Software: Changes and Enhancements, Release 8.2*, please send them to us on a photocopy of this page or send us electronic mail.

For comments about this book, please return the photocopy to

SAS Institute Inc.
SAS Publishing
SAS Campus Drive
Cary, NC 27513
E-mail: yourturn@sas.com

For suggestions about the software, please return the photocopy to

SAS Institute Inc.
Technical Support Division
SAS Campus Drive
Cary, NC 27513
E-mail: suggest@sas.com

*Welcome * Bienvenue * Willkommen * Yohkoso * Bienvenido*

SAS® Institute Publishing Is Easy to Reach

Visit our Web page located at www.sas.com/pubs

You will find product and service details, including

- **sample chapters**
- **tables of contents**
- **author biographies**
- **book reviews**

Learn about

- **regional user-group conferences**
- **trade-show sites and dates**
- **authoring opportunities**
- **custom textbooks**

Explore all the services that SAS Institute Publishing has to offer!

Your Listserv Subscription Automatically Brings the News to You

Do you want to be among the first to learn about the latest books and services available from SAS Institute Publishing? Subscribe to our listserv **newdocnews-l** and, once each month, you will automatically receive a description of the newest books and which environments or operating systems and SAS release(s) that each book addresses.

To subscribe,

1. Send an e-mail message to **listserv@vm.sas.com**.
2. Leave the "Subject" line blank.
3. Use the following text for your message:

   subscribe NEWDOCNEWS-L ***your-first-name your-last-name***

   For example: subscribe NEWDOCNEWS-L John Doe

Create Customized Textbooks Quickly, Easily, and Affordably

SelecText® offers instructors at U.S. colleges and universities a way to create custom textbooks for courses that teach students how to use SAS software.

For more information, see our Web page at **www.sas.com/selectext**, or contact our SelecText coordinators by sending e-mail to **selectext@sas.com**.

You're Invited to Publish with SAS Institute's User Publishing Program

If you enjoy writing about SAS software and how to use it, the User Publishing Program at SAS Institute offers a variety of publishing options. We are actively recruiting authors to publish books, articles, and sample code. Do you find the idea of writing a book or an article by yourself a little intimidating? Consider writing with a co-author. Keep in mind that you will receive complete editorial and publishing support, access to our users, technical advice and assistance, and competitive royalties. Please contact us for an author packet. E-mail us at **sasbbu@sas.com** or call 919-677-8000, then press 1-6479. See the SAS Institute Publishing Web page at **www.sas.com/pubs** for complete information.

See *Observations*®, Our Online Technical Journal

Feature articles from *Observations*®: *The Technical Journal for SAS® Software Users* are now available online at **www.sas.com/obs**. Take a look at what your fellow SAS software users and SAS Institute experts have to tell you. You may decide that you, too, have information to share. If you are interested in writing for *Observations*, send e-mail to **sasbbu@sas.com** or call 919-677-8000, then press 1-6479.

Book Discount Offered at SAS Public Training Courses!

When you attend one of our SAS Public Training Courses at any of our regional Training Centers in the U.S., you will receive a 15% discount on book orders that you place during the course. Take advantage of this offer at the next course you attend!

SAS Institute
SAS Campus Drive
Cary, NC 27513-2414
Fax 919-677-4444

E-mail: sasbook@sas.com
Web page: www.sas.com/pubs
To order books, call Fulfillment Services at 800-727-3228*
For other SAS Institute business, call 919-677-8000*

* **Note:** Customers outside the U.S. should contact their local SAS office.

The Power to Know™

SAS Publishing