

Chapter 1

PROC CAPABILITY and General Statements

Overview

This chapter describes several statements that are generally used with the CAPABILITY procedure:

- The PROC CAPABILITY statement is required to invoke the CAPABILITY procedure. You can use this statement by itself to compute summary statistics.
- The VAR statement, which is optional, specifies the variables in the input data set that are to be analyzed. By default, all of the numeric variables are analyzed.
- The SPEC statement, which is optional, provides specification limits for the variables that are to be analyzed. When you use a SPEC statement, the procedure computes process capability indices in addition to summary statistics. Furthermore, the specification limits are displayed in plots created with plot statements (such as HISTOGRAM) that are described in subsequent chapters.

You can use the PROC CAPABILITY statement to request a variety of statistics for summarizing the data distribution of each analysis variable:

- sample moments
- basic measures of location and variability
- confidence intervals for the mean, standard deviation, and variance
- tests for location
- tests for normality
- trimmed and Winsorized means
- robust estimates of scale
- quantiles and related confidence intervals
- extreme observations and extreme values
- frequency counts for observations
- missing values

You can use the PROC CAPABILITY and SPEC statements together to request a variety of statistics for process capability analysis:

- percents of measurements within and outside specification limits
- confidence intervals for the probabilities of exceeding the specification limits
- standard capability indices and related confidence intervals
- tests of normality in conjunction with capability indices
- specialized capability indices

Part 1. The CAPABILITY Procedure

In addition, you can use options in the PROC CAPABILITY statement to

- specify the input data set to be analyzed
- specify an input data set containing specification limits
- specify a graphics catalog for saving graphical output
- specify rounding units for variable values
- specify the definition used to calculate percentiles
- specify the divisor used to calculate variances and standard deviations
- request that plots be produced on line printers and define special printing characters used for features
- suppress tables

You can use options in the SPEC statement to

- provide lower and upper specification limits and target values
- control the appearance of specification lines on plots
- control the appearance of the areas under a histogram outside the specification limits

Getting Started

This section introduces the PROC CAPABILITY, VAR, and SPEC statements with examples that illustrate the most commonly used options.

Computing Descriptive Statistics

The fluid weights of 100 drink cans are measured in ounces. The filling process is assumed to be in statistical control. The measurements are saved in a SAS data set named CANS.

See CAPPROC
in the SAS/QC
Sample Library

```
data cans;
  label weight = 'Fluid Weight (ounces)';
  input weight @@;
datalines;
12.07 12.02 12.00 12.01 11.98 11.96 12.04 12.05 12.01 11.97
12.03 12.03 12.00 12.04 11.96 12.02 12.06 12.00 12.02 11.91
12.05 11.98 11.91 12.01 12.06 12.02 12.05 11.90 12.07 11.98
12.02 12.11 12.00 11.99 11.95 11.98 12.05 12.00 12.10 12.04
12.06 12.04 11.99 12.06 11.99 12.07 11.96 11.97 12.00 11.97
12.09 11.99 11.95 11.99 11.99 11.96 11.94 12.03 12.09 12.03
11.99 12.00 12.05 12.04 12.05 12.01 11.97 11.93 12.00 11.97
12.13 12.07 12.00 11.96 11.99 11.97 12.05 11.94 11.99 12.02
11.95 11.99 11.91 12.06 12.03 12.06 12.05 12.04 12.03 11.98
12.05 12.05 12.11 11.96 12.00 11.96 11.96 12.00 12.01 11.98
;
```

You can use the PROC CAPABILITY and VAR statements to compute summary statistics for the weights.

```
title 'Process Capability Analysis of Fluid Weight';
proc capability data=cans normaltest;
  var weight;
run;
```

The input data set is specified with the DATA= option. The NORMALTEST option requests tests for normality. The VAR statement specifies the variables to analyze. If you omit the VAR statement, all numeric variables in the input data set are analyzed.

The descriptive statistics* for WEIGHT are shown in Figure 1.1. For instance, the average weight (labeled *Mean*) is 12.0093. The Shapiro-Wilk test statistic labeled *W* is 0.987876, and the probability of a more extreme value of *W* (labeled *Pr < W*) is 0.499. Compared to the usual cutoff value of 0.05, this probability (referred to as a *p*-value) indicates that the weights are normally distributed.

*In Version 7, the *Moments* table has been reorganized. Tables for *Basic Statistical Measures*, *Tests for Location*, and *Tests for Normality* have been added.

Part 1. The CAPABILITY Procedure

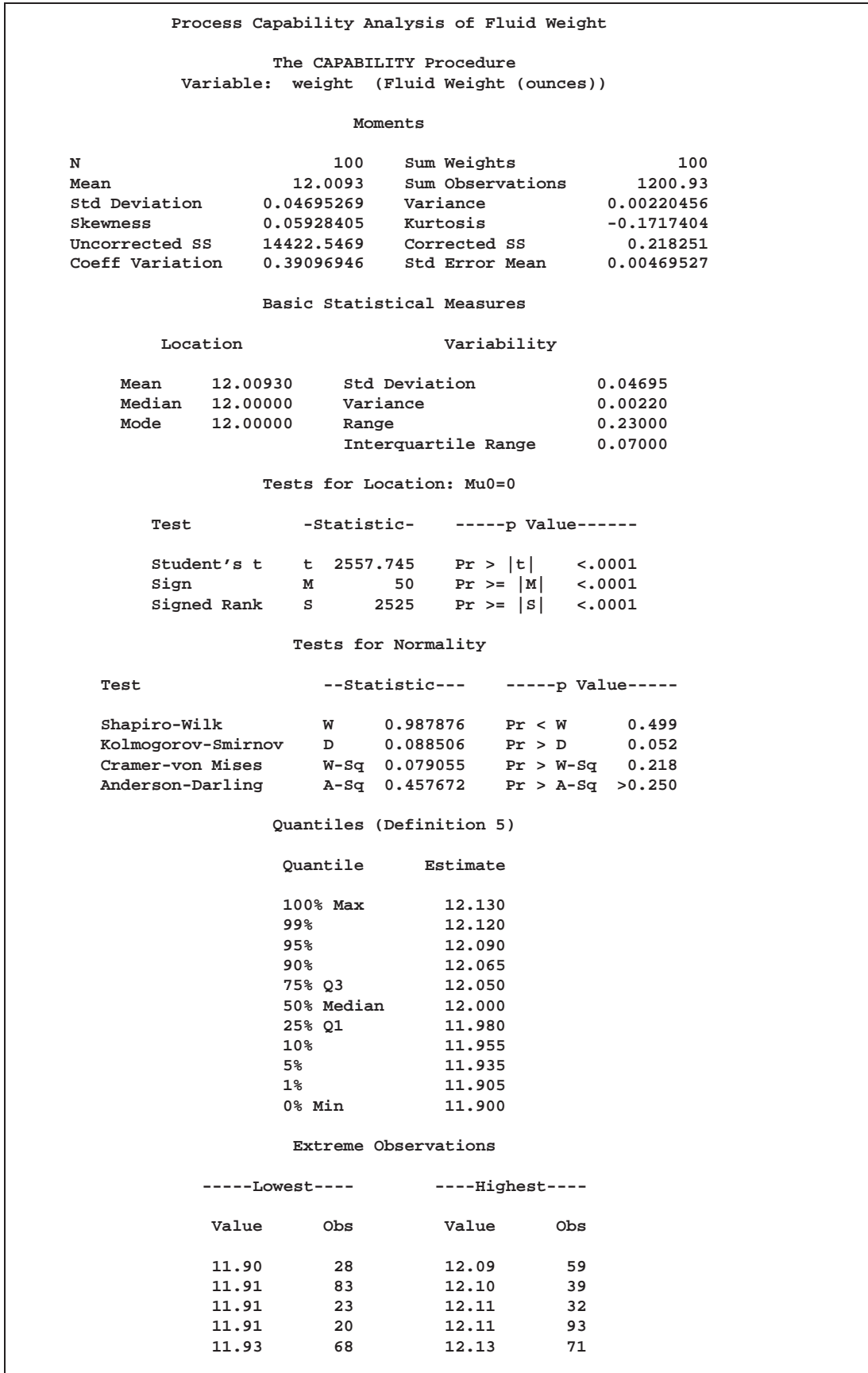


Figure 1.1. Descriptive Statistics 10

Computing Capability Indices

This example is a continuation of the previous example and shows how you can provide specification limits with a SPEC statement to request capability indices in addition to descriptive statistics.

See CAPPROC
in the SAS/QC
Sample Library

```
proc capability data=cans normaltest freq;
  spec lsl=11.95 target=12 usl=12.05;
  var weight;
run;
```

The options LSL=, TARGET=, and USL= specify the lower specification limit, target value, and upper specification limit for the weights. These statements produce the output shown in Figure 1.2 in addition to the output shown in Figure 1.1.

Specification Limits							
-----Limit-----				-----Percent-----			
Lower (LSL)	11.95000	% < LSL	7.00000				
Target	12.00000	% Between	77.00000				
Upper (USL)	12.05000	% > USL	16.00000				
Process Capability Indices							
Index	Value	95% Confidence Limits					
Cp	0.354967	0.305565	0.404288				
CPL	0.420991	0.332644	0.508117				
CPU	0.288943	0.211699	0.365112				
Cpk	0.288943	0.212210	0.365677				
Cpm	0.348203	0.301472	0.398228				
Frequency Counts							
Percents				Percents			
Value	Count	Cell	Cum	Value	Count	Cell	Cum
12	1	1.0	1.0	12	6	6.0	62.0
12	3	3.0	4.0	12	6	6.0	68.0
12	1	1.0	5.0	12	6	6.0	74.0
12	2	2.0	7.0	12	10	10.0	84.0
12	3	3.0	10.0	12	6	6.0	90.0
12	8	8.0	18.0	12	4	4.0	94.0
12	6	6.0	24.0	12	2	2.0	96.0
12	6	6.0	30.0	12	1	1.0	97.0
12	10	10.0	40.0	12	2	2.0	99.0
12	11	11.0	51.0	12	1	1.0	100.0
12	5	5.0	56.0				

Figure 1.2. Capability Indices and Frequency Table

In Figure 1.2, the table labeled *Specification Limits* lists the specification limits and target value, together with the percents of observations outside and between the limits. The table labeled *Process Capability Indices* lists estimates for the standard process capability indices* C_p , CPL , CPU , C_{pk} , and C_{pm} , along with 95% confidence

*In Release 6.12 and earlier releases, the capability index k was also included in this table; this index is now provided in the table labeled *Specialized Capability Indices*, which you can request with the SPECIALINDICES option.

Part 1. The CAPABILITY Procedure

limits. The index C_{pm} is not computed unless you specify a TARGET= value. See page 46 for formulas used to compute the indices.

If you specify more than one variable in the VAR statement, you can provide corresponding specification limits and target values by specifying lists of values for the LSL=, USL=, and TARGET= options. As an alternative to the SPEC statement, you can read specification limits and target values from a data set specified with the SPEC= option in the PROC CAPABILITY statement. This is illustrated in Example 1.1 on page 56.

The FREQ option in the PROC CAPABILITY statement requests the table labeled *Frequency Counts* in Figure 1.2.

Syntax Overview for the CAPABILITY Procedure

The following are the primary statements that control the CAPABILITY procedure:

PROC CAPABILITY *< options >* ;

VAR *variables* ;

SPEC *< options >* ;

CDFPLOT *< variables >* *< / options >* ;

COMPHISTOGRAM *< variables >* **CLASS=***(class-variables)* *< options >* ;

HISTOGRAM *< variables >* *< / options >* ;

PPLOT *< variables >* *< / options >* ;

PROBPLOT *< variables >* *< / options >* ;

QQPLOT *< variables >* *< / options >* ;

INSET *keyword-list* *< / options >* ;

INTERVALS *< variables >* *< / options >* ;

OUTPUT **<OUT=SAS-data-set>** *keyword=names* *< ... keyword=names >* ;

The PROC CAPABILITY statement invokes the procedure. The VAR statement specifies the numeric variables to be analyzed, and it is required if the OUTPUT statement is used to save summary statistics and capability indices in an output data set. If you do not use the VAR statement, all numeric variables in the data set are analyzed. The SPEC statement provides specification limits.

The plot statements (CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PPLOT, PROBPLOT, and QQPLOT) create graphical displays, and the INSET statement enhances these displays by adding a table of summary statistics directly on the graph. The INTERVALS statement computes statistical intervals. You can specify one or more of each of the plot statements, the INSET statement, the INTERVALS statement, and the OUTPUT statement. If you use a VAR statement, the variables listed in a plot statement must be a subset of the variables listed in the VAR statement.

BY, FREQ, WEIGHT, and ID Statements

In addition, you can optionally specify one of each of the following statements:

BY *variables*;

FREQ *variable*;

WEIGHT *variable*;

ID *variable*;

The **BY** statement specifies variables in the input data set that are used for BY processing. A separate analysis is done for each group of observations defined by the levels of the BY variables. The input data set must be sorted in order of the BY variables.

The **FREQ** statement names a variable that provides frequencies for each observation in the input data set. If n is the value of the **FREQ** variable for a given observation, then that observation is used n times. If the value of the **FREQ** variable is missing or is less than one, the observation is not used in the analysis. If the value is not an integer, only the integer portion is used.

The **WEIGHT** statement names a variable that provides weights for each observation in the input data set. The **CAPABILITY** procedure uses the values w_i of the **WEIGHT** variable to modify the computation of a number of summary statistics by assuming that the variance of the i th value X_i of the analysis variable is equal to σ^2/w_i , where σ is an unknown parameter. This assumption is rarely applicable in process capability analysis, and the purpose of the **WEIGHT** statement is simply to make the **CAPABILITY** procedure consistent with other data summarization procedures, such as the **UNIVARIATE** procedure.

The values of the **WEIGHT** variable do not have to be integers and are typically positive. By default, observations with non-positive or missing values of the **WEIGHT** variable are handled as follows*:

- If the value is zero, the observation is counted in the total number of observations.
- If the value is negative, it is converted to zero, and the observation is counted in the total number of observations.
- If the value is missing, the observation is excluded from the analysis.

To exclude observations that contain negative and zero weights from the analysis, specify the option **EXCLNPWGT** in the **PROC** statement. Note that most SAS/STAT procedures, such as **PROC GLM**, exclude negative and zero weights by default.

*In Release 6.12 and earlier releases, observations were used in the analysis if and only if the **WEIGHT** variable value was greater than zero.

When you specify a WEIGHT variable, the procedure uses its values, w_i , to compute weighted versions of the statistics* provided in the *Moments* table. For example, the procedure computes a weighted mean \bar{X}_w and a weighted variance s_w^2 as $\bar{X}_w = \frac{\sum_i w_i x_i}{\sum_i w_i}$ and $s_w^2 = \frac{1}{d} \sum_i w_i (x_i - \bar{X}_w)^2$ where x_i is the i th variable value. The divisor d is controlled by the VARDEF= option in the PROC CAPABILITY statement.

When you use both the WEIGHT and SPEC statements, capability indices are computed using \bar{X}_w and s_w in place of \bar{X} and s . Again, note that weighted capability indices are seldom needed in practice.

When you specify a WEIGHT statement, the procedure also computes a weighted standard error and a weighted version of Student's t test. This test is the only test of location that is provided when weights are specified.

The WEIGHT statement does not affect the determination of the mode, extreme values, extreme observations, or the number of missing values of the analysis variables. However, the weights w_i are used to compute weighted percentiles†.

The WEIGHT variable has no effect on the calculation of extreme values, and it has no effect on graphical displays produced with the plot statements.

Graphical Enhancement Statements

You can use TITLE, FOOTNOTE, and NOTE statements to enhance printed output. If you are creating plots with a graphics device, you can also use AXIS, LEGEND, PATTERN, and SYMBOL statements to enhance your plots. For details, see SAS/GRAPH documentation and the chapter for the plot statement that you are using.

*In Release 6.12 and earlier releases, weighted skewness and kurtosis were not computed.

†In Release 6.12 and earlier releases, the weights did not affect the computation of percentiles.

Syntax for the PROC CAPABILITY Statement

The syntax for the PROC CAPABILITY statement is as follows:

PROC CAPABILITY < *options* >;

The following section lists all *options*. See “Dictionary of Options” on page 19 for detailed information.

Summary of Options

The following tables list all the PROC CAPABILITY *options* by function.

Table 1.1. Input Data Set Options

ANNOTATE= <i>SAS-data-set</i>	specifies input data set containing annotation information
DATA= <i>SAS-data-set</i>	specifies input data set
EXCLNPWGT	specifies that non-positive weights are to be excluded
NOBYSPECS	specifies that specification limits in SPEC= data set are to be applied to all BY groups
SPEC= <i>SAS-data-set</i>	specifies input data set with specification limits

Table 1.2. Plotting and Graphics Options

FORMCHAR(<i>index</i>)= <i>'string'</i>	defines characters used for features on plots
GOUT= <i>graphics-catalog</i>	specifies catalog for saving graphical output
LINEPRINTER	requests line printer plots

Table 1.3. Computational Options

PCTLDEF= <i>n</i>	specifies definition used to calculate percentiles
ROUND= <i>round-units</i>	specifies units used to round variable values
VARDEF= <i>keyword</i>	specifies divisor used to calculate variances and standard deviations

Table 1.4. Data Summary Options

ALL	requests all tables
FREQ	requests frequency table
MODES	requests table of modes
NEXTROBS= <i>n</i>	requests table of <i>n</i> lowest, <i>n</i> highest observations
NEXTRVAL= <i>n</i>	requests table of <i>n</i> lowest, <i>n</i> highest values

Table 1.5. Output Options

NOPRINT	suppresses printed output
OUTTABLE= <i>SAS-data-set</i>	creates an output data set containing univariate statistics and capability indices in tabular form

Table 1.6. Hypothesis Testing Options

MU0= <i>value</i>	specifies mean for null hypothesis in tests for location
LOCCOUNT	requests table of counts used in sign test and signed rank test
NORMALTEST	performs tests for normality

Table 1.7. Robust Estimation Options

ROBUSTSCALE	requests table of robust measures of scale
TRIMMED=(<i>trimmed options</i>)	requests table of trimmed means
WINSORIZED=(<i>Winsorized options</i>)	requests table of Winsorized means

Table 1.8. TRIMMED-Options

ALPHA= <i>value</i>	specifies confidence level
TYPE=LOWER UPPER TWSIDED	specifies type of confidence limit

Table 1.9. WINSORIZED-Options

ALPHA= <i>value</i>	specifies confidence level
TYPE=LOWER UPPER TWSIDED	specifies type of confidence limit

Table 1.10. Capability Index Options

CPMA= <i>a</i>	specifies <i>a</i> for Cpm(<i>a</i>)
CHECKINDICES (<i>checkindices-options</i>)	requests test of normality in conjunction with standard indices
SPECIALINDICES	requests table of specialized indices including Boyles' Cpm, Cj _{kp} , Cpm _k , Cpm(<i>a</i>), and Wright's Cs

Part 1. The CAPABILITY Procedure

Table 1.11. CHECKINDICES-Options

TEST=SW KS AD CVM NONE	specifies test for normality (Shapiro-Wilk, Kolmogorov-Smirnov, Anderson-Darling, Cramér-von Mises, or no test)
----------------------------	---

Table 1.12. Confidence Limit Options

ALPHA= <i>value</i>	specifies level for all confidence limits
CIBASIC (<i>cibasic-options</i>)	requests confidence limits for the mean, standard deviation, variance
CIINDICES (<i>ciindices-options</i>)	specifies level and type of confidence limits for capability indices
CIPCTLDF (<i>cipctldf-options</i>)	requests distribution-free confidence limits for percentiles
CIPCTLNORMAL (<i>cipctlnormal-options</i>)	requests confidence limits for percentiles assuming normality
CIPROBEX (<i>ciprobex-options</i>)	requests confidence limits for the probability of exceeding specifications

Table 1.13. CIBASIC-Options

ALPHA= <i>value</i>	specifies confidence level
TYPE=LOWER UPPER TWO SIDED	specifies type of confidence limit

Table 1.14. CIINDICES-Options

ALPHA= <i>value</i>	specifies confidence level
TYPE=LOWER UPPER TWO SIDED	specifies type of confidence limit

Table 1.15. CIPCTLDF-Options

ALPHA= <i>value</i>	specifies confidence level
TYPE=LOWER UPPER SYMMETRIC ASYMMETRIC	specifies type of confidence limit

Table 1.16. CIPCTLNORMAL-Options

ALPHA= <i>value</i>	specifies confidence level
TYPE=LOWER UPPER TWO SIDED	specifies type of confidence limit

Table 1.17. CIPROBEX-Options

ALPHA= <i>value</i>	specifies confidence level
TYPE=LOWER UPPER TWO SIDED	specifies type of confidence limit

Dictionary of Options

The following entries provide detailed descriptions of the *options* in the PROC CAPABILITY statement. The marginal notes *Graphics* and *Line Printer* identify options that apply to graphics devices and line printers, respectively.

ALL

requests all of the tables generated by the `FREQ`, `MODES`, `NEXTRVAL=5`, `CIBASIC`, `CIPCTLDF`, and `CIPCTLNORMAL` options. If a `WEIGHT` statement is not used, the `ALL` option also requests the tables generated by the `LOCCOUNT`, `NORMALTEST`, `ROBUSTSCALE`, `TRIMMED=.25`, and `WINSORIZED=.25` options. PROC CAPABILITY uses any values that you specify with the `ALPHA=`, `MUO=`, `NEXTRVAL=`, `CIBASIC`, `CIPCTLDF`, `CIPCTLNORMAL`, `TRIMMED=`, or `WINSORIZED=` options in conjunction with the `ALL` option.

ALPHA=*value*

specifies the default confidence level for all confidence limits computed by the CAPABILITY procedure*. The coverage percent for the confidence limits is $(1 - \textit{value})100$. For example, `ALPHA=0.10` results in 90% confidence limits. The default *value* is 0.05.

Note that specialized `ALPHA=` options are available for a number of confidence interval options. For example, you can specify `CIBASIC(ALPHA=0.10)` to request a table of *Basic Confidence Limits* at the 90% level. The default *values* of these options default to the value of the general `ALPHA=` option.

ANNOTATE=*SAS-data-set***ANNO=*SAS-data-set***

specifies an input data set containing annotate variables as described in SAS/GRAPH documentation. You can use this data set to add features to plots produced on graphics devices. Use this data set only when the chart is created using a graphics device; it is ignored when the `LINEPRINTER` option is specified. Features provided in this data set are added to every plot produced in the current run of the procedure.

<i>Graphics</i>

CHECKINDICES<(<TEST = SW | KS | AD | CVM | NONE> <ALPHA=*value*>)>

specifies the test of normality used in conjunction with process capability indices that are displayed in the *Process Capability Indices* table. The tests available are Shapiro-Wilk (SW), Kolmogorov-Smirnov (KS), Anderson-Darling (AD), and Cramér-von Mises (CVM). The default test is the Shapiro-Wilk test if the sample size is less than or equal to 2000 and the Kolmogorov-Smirnov test if the sample size is greater than

*In Release 6.12 and earlier release, the level for confidence limits for capability indices was specified with the `GAMMA=` option in the `SPEC` statement. This option still works but is considered obsolete.

Part 1. The CAPABILITY Procedure

2000. If the p -value for the test is less than the cutoff probability *value* specified with the ALPHA= option*, a warning is added to the table, as illustrated in Figure 1.3. The *value* must be between zero and one, and typical values are 0.05 and 0.10. See “Tests for Normality” on page 36 for details concerning the test.

Process Capability Indices			
Index	Value	95% Confidence Limits	
Cp	0.541072	0.388938	0.692946
CPL	0.642426	0.417087	0.862984
CPU	0.439718	0.257339	0.617184
Cpk	0.439718	0.259310	0.620126

Warning: Normality is rejected for alpha = 0.05 using the Shapiro-Wilk test

Figure 1.3. Warning Message Printed with Capability Indices

CIINDICES<(<TYPE=keyword><ALPHA=value>)>

specifies the type and level of the confidence limits for standard capability indices displayed in the table labeled *Process Capability Indices*.

TYPE=keyword

specifies the type of confidence limit, where *keyword* is LOWER, UPPER, or TWOSIDED. The default value is TWOSIDED.

ALPHA=value

specifies the confidence level. The coverage percent for the confidence limits is $(1 - \textit{value})100$. For example, ALPHA=0.10 requests 90% confidence limits. The default value is 0.05.

CIBASIC<(<TYPE=keyword><ALPHA=value>)>

requests confidence limits for the mean, standard deviation, and variance based on the assumption that the data are normally distributed. With large sample sizes, this assumption is not required for confidence limits for the mean.

TYPE=keyword

specifies the type of confidence limit, where *keyword* is LOWER, UPPER, or TWOSIDED. The default value is TWOSIDED.

ALPHA=value

specifies the confidence level. The coverage percent for the confidence limits is $(1 - \textit{value})100$. For example, ALPHA=0.10 requests 90% confidence limits. The default *value* is 0.05.

CIPCTLDF<(TYPE=keyword><ALPHA=value>)>

CIQUANTDF<(TYPE=keyword><ALPHA=value>)>

requests confidence limits for quantiles using a distribution-free method. In other words, no specific parametric distribution (such as the normal) is assumed for the data. Order statistics are used to compute the confidence limits as described in Sec-

*In Release 6.12 and earlier releases, this *value* was specified with the ALPHA= option in the SPEC statement. This option still works but is considered obsolete.

tion 5.2 of Hahn and Meeker (1991). This option is not available if you specify a WEIGHT statement.

TYPE=keyword

specifies the type of confidence limit, where *keyword* is LOWER, UPPER, SYMMETRIC, or ASYMMETRIC. The default value is SYMMETRIC.

ALPHA=value

specifies the confidence level. The coverage percent for the confidence limits is $(1 - \textit{value})100$. For example, ALPHA=0.10 requests 90% confidence limits. The default *value* is 0.05.

CIPCTLNORMAL<(TYPE=keyword)<ALPHA=value>>

CIQUANTNORMAL<(TYPE=keyword)<ALPHA=value>>

requests confidence limits for quantiles based on the assumption that the data are normally distributed. The computational method is described in Section 4.4.1 of Hahn and Meeker (1991) and uses the noncentral *t* distribution as given by Odeh and Owen (1980). This option is not available if you specify a WEIGHT statement

CIPROBEX<(TYPE=keyword)<ALPHA=value>>

requests confidence limits for $Pr[X \leq \text{LSL}]$ and $Pr[X \geq \text{USL}]$, where *X* is the analysis variable, LSL is the lower specification limit, and USL is the upper specification limit. The computational method, which assumes that *X* is normally distributed, is described in Section 4.5 of Hahn and Meeker (1991) and uses the noncentral *t* distribution as given by Odeh and Owen (1980). This option is not available if you specify a WEIGHT statement

TYPE=keyword

specifies the type of confidence limit, where *keyword* is LOWER, UPPER, or TWOSIDED. The default value is TWOSIDED.

ALPHA=value

specifies the confidence level. The coverage percent for the confidence limits is $(1 - \textit{value})100$. For example, ALPHA=0.10 requests 90% confidence limits. The default value is 0.05.

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). You must specify the option SPECIALINDICES in order to request $C_{pm}(a)$, which is displayed in the table of *Specialized Capability Indices*. The *value* must be positive. The default *value* is 0.5.

DATA=SAS-data-set

specifies the input data set containing the observations to be analyzed. If the DATA= option is omitted, the procedure uses the most recently created SAS data set.

DEF=index

is an alias for the PCTLDEF= option. See the entry for the PCTLDEF= option.

EXCLNPWGT

excludes observations with non-positive weight values (zero or nonnegative) for the analysis. By default, PROC CAPABILITY treats observations with negative weights

like those with zero weights and counts them in the total number of observations. This option is applicable only if you specify a WEIGHT statement.

Line Printer

FORMCHAR(index)='string'

defines characters used for features on plots, where *index* is a number ranging from 1 to 11, and *string* is a character or hexadecimal string. The *index* identifies which features are controlled with the *string* characters, as discussed in the table that follows. If you specify the FORMCHAR= option omitting the *index*, the *string* controls all 11 features.

By default, the form character list specified with the SAS system option FORMCHAR= is used; otherwise, the default is FORMCHAR='|---|+|---'. If you print to a PC screen or your device supports the ASCII symbol set (1 or 2), the following is recommended:

formchar= ' B3 , C4 , DA , C2 , BF , C3 , C5 , B4 , C0 , C1 , D9 ' X

As an example, suppose you want to plot the data values of the empirical cumulative distribution function with asterisks (*). You can change the appropriate character using the following:

formchar(2) = ' * '

Note that the FORMCHAR= option in the PROC CAPABILITY statement allows you to temporarily override the values of the SAS system option with the same name. The values of the SAS system option are not altered by using the FORMCHAR= option in PROC CAPABILITY statement.

The features associated with values of *index* are as follows:

Value of <i>index</i>	Description of Character	Chart Feature
1	vertical bar	frame, ecdf line, HREF= lines
2	horizontal bar	frame, ecdf line, VREF= lines
3	box character (upper left)	frame, ecdf line, histogram bars
4	box character (upper middle)	histogram bars, tick marks (horizontal axis)
5	box character (upper right)	frame, histogram bars
6	box character (middle left)	histogram bars
7	box character (middle middle)	not used
8	box character (middle right)	histogram bars, tick marks (vertical axis)
9	box character (lower left)	frame
10	box character (lower middle)	histogram bars
11	box character (lower right)	frame, ecdf line

FREQ

requests a frequency table in the printed output that contains the variable values, frequencies, percentages, and cumulative percentages. See Figure 1.2 on page 11 for an example.

GOUT=graphics-catalog

specifies a graphics catalog in which to save graphics output.

Graphics

LINEPRINTER

requests that line printer charts be produced. By default, the procedure creates charts for a graphics device.

Line Printer

LOCCOUNT

requests a table with the number of observations greater than, equal to, and less than the value of MUO=. PROC CAPABILITY uses these values to construct the sign test and signed rank test. This option is not available if you specify a WEIGHT statement.

MODES**MODE**

requests a table of all possible modes. By default, when the data contains multiple modes, PROC CAPABILITY displays the lowest mode in the table of basic statistical measures. When all values are unique, PROC CAPABILITY does not produce a table of modes.

MUO=value(s)**LOCATION=value(s)**

specifies the *value* of the mean or location parameter (μ_0) in the null hypothesis for the tests summarized in the table labeled *Tests for Location: Mu0=value*. If you specify a single *value*, PROC CAPABILITY tests the same null hypothesis for all analysis variables. If you specify multiple *values*, a VAR statement is required, and PROC CAPABILITY tests a different null hypothesis for each analysis variable by matching the VAR variables with the *values* in the corresponding order. The default *value* is 0.

NEXTROBS=n

specifies the number of extreme observations in the table labeled *Extreme Observations*. The table lists the *n* lowest observations and the *n* highest observations. The default value is 5. The value of *n* must be an integer between 0 and half the number of observations. You can specify NEXTROBS=0 to suppress the table.

NEXTRVAL=n

requests the table labeled *Extreme Values* and specifies the number of extreme values in the table. The table lists the *n* lowest unique values and the *n* highest unique values. The value of *n* must be an integer between 0 and half the maximum number of observations. By default, *n* = 0 and no table is displayed.

NOPRINT

suppresses the tables of descriptive statistics and capability indices which are created by the PROC CAPABILITY statement. The NOPRINT option does not suppress the tables created by the INTERVALS or plot statements. You can use the NOPRINT options in these statements to suppress the creation of their tables.

NORMALTEST

NORMAL

requests a table of *Tests for Normality* for each of the analysis variables. The table provides test statistics and *p*-values for the Shapiro-Wilk test (provided the sample size is less than or equal to 2000), the Kolmogorov-Smirnov test, the Anderson-Darling test, and the Cramér-von Mises test. See “Tests for Normality” on page 36 for details. If specification limits are provided, the NORMALTEST option is assumed.

OUTTABLE=SAS-data-set

specifies an output data set that contains univariate statistics and capability indices arranged in tabular form. See “OUTTABLE= Data Set” on page 32 for details.

PCTLDEF=index

DEF=index

specifies one of five definitions used to calculate percentiles. The value of *index* can be 1, 2, 3, 4, or 5. See “Percentile Computations” on page 39 for details. By default, PCTLDEF=5.

ROBUSTSCALE

requests a table of robust measures of scale. These measures include the interquartile range, Gini’s mean difference, the median absolute deviation about the median (*MAD*), and two statistics proposed by Rousseeuw and Croux (1993), Q_n , and S_n . This option is not available if you specify a WEIGHT statement.

ROUND=value-list

specifies units used to round variable values. The ROUND= option reduces the number of unique values for each variable and hence reduces the memory required for temporary storage. *Values* must be greater than 0 for rounding to occur.

If you use only one *value*, the procedure uses this unit for all variables. If you use a list of values, you must also use a VAR statement. The procedure then uses the roundoff values for variables in the order given in the VAR statement. For example, the following statements specify a roundoff value of 1 for YLDSTREN and a roundoff value of 0.5 for TENSTREN.

```
proc capability round=1 0.5;
  var yldstren tenstren;
run;
```

When a variable value is midway between the two nearest rounded points, the value is rounded to the nearest even multiple of the roundoff value. For example, with a roundoff value of 1, the variable values of -2.5 , -2.2 , and -1.5 are rounded to -2 ; the values of -0.5 , 0.2 , and 0.5 are rounded to 0; and the values of 0.6 , 1.2 , and 1.4 are rounded to 1.

SPECIALINDICES

requests a table of specialized process capability indices. These indices include 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} , and Wright’s C_s . You can specify the parameter a for $C_{pm}(a)$ with the CPMA= option.

SPECS=SAS-data-set**SPEC=SAS-data-set**

specifies an input data set containing specification limits for each of the variables in the VAR statement. This option is an alternative to the SPEC statement, which also provides specification limits. See “SPEC= Data Set” on page 30 for details on SPEC= data sets, and Example 1.1 on page 56 for an example. If you use both the SPEC= option and a SPEC statement, the SPEC= option is ignored.

TRIMMED=values(s) <(TYPE=keyword)<ALPHA=value>>**TRIM=values(s) <(TYPE=keyword)<ALPHA=value>>**

requests a table of trimmed means, where each *value* specifies the number or the proportion of trimmed observations. If the *value* is the number n of trimmed observations, n must be between 0 and half the number of nonmissing observations. If the *value* is a proportion p between 0 and 0.5, the number of observations trimmed is the smallest integer greater than or equal to np , where n is the number of observations. To obtain confidence limits for the mean and the student t -test, you must use the default value of VARDEF= which is DF. The TRIMMED= option is not available if you specify a WEIGHT statement.

TYPE=keyword

specifies the type of confidence limit, where *keyword* is LOWER, UPPER, or TWOSIDED. The default value is TWOSIDED.

ALPHA=value

specifies the confidence level. The coverage percent is $(1 - \text{value})100$. For example, ALPHA=0.10 requests a 90% confidence limit. The default value is 0.05.

VARDEF=DF | N | WDF | WEIGHT | WGT

specifies the divisor used in calculating variances and standard deviations. The values and associated divisors are shown in the following table. By default, VARDEF=DF.

Value	Divisor	Formula
DF	degrees of freedom	$n - 1$
N	number of observations	n
WEIGHT WGT	sum of weight	$\sum_i w_i$
WDF	sum of weights minus one	$(\sum_i w_i) - 1$

WINSORIZED=values(s) <(TYPE=keyword)<ALPHA=value>>**WINSOR=values(s) <(TYPE=keyword)<ALPHA=value>>**

requests a table of winsorized means, where each *value* specifies the number or the proportion of winsorized observations. If the *value* is the number n of winsorized observations, n must be between 0 and half the number of nonmissing observations. If the *value* is a proportion p between 0 and 0.5, the number of observations winsorized is the smallest integer greater than or equal to np , where n is the number of observations. To obtain confidence limits for the mean and the student t -test, you must use the default value of VARDEF= which is DF. The WINSORIZED= option is not available if you specify a WEIGHT statement.

TYPE=keyword

specifies the type of confidence limit, where *keyword* is LOWER, UPPER, or TWOSIDED. The default value is TWOSIDED.

ALPHA=value

specifies the confidence level. The coverage percent is $(1 - value)100$. For example, ALPHA=0.10 results in a 90% confidence limit. The default value is 0.05.

Syntax for the SPEC Statement

The syntax for the SPEC statement is as follows:

SPEC < options >;

You can use at most one SPEC statement in the CAPABILITY procedure. When you provide specification limits and target values in a SPEC statement, the tabular output produced by the PROC CAPABILITY statement includes process capability indices as well as summary statistics. You can use the SPEC statement in conjunction with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements to add specification limit and target lines to the plots produced with these statements.

options

control features of the specification limits and target values. The “Summary of Options” section, which follows, lists all options by function. The “Dictionary of Options” section on page 27 describes each option in more detail.

Summary of Options

The following tables list the SPEC statement *options* by function. For complete descriptions see the “Dictionary of Options” on page 27.

Table 1.18. Lower Specification Limit Options

CLEFT= <i>color</i>	color of pattern used to fill area left of lower specification limit (histograms only)
CLSL= <i>color</i>	color of lower specification limit line
LLSL= <i>linetype</i>	line type of lower specification limit line
LSL= <i>value-list</i>	lower specification limit values
LSLSYMBOL= <i>'character'</i>	character used to plot lower specification limit line on line printers
PLEFT= <i>pattern</i>	pattern type used to fill area left of lower specification limit (histograms only)
WLSL= <i>n</i>	width of lower specification limit line

Table 1.19. Target Options

CTARGET= <i>color</i>	color of target line
LTARGET= <i>linetype</i>	line type of target line
TARGET= <i>value-list</i>	target value
TARGETSYMBOL= <i>'character'</i>	character used to plot target on line printers
WTARGET= <i>n</i>	width of target line

Table 1.20. Upper Specification Limit Options

CRIGHT= <i>color</i>	color of pattern used to fill area right of upper specification limit (histograms only)
CUSL= <i>color</i>	color of upper specification limit line
LUSL= <i>linetype</i>	line type of upper specification limit line
PRIGHT= <i>pattern</i>	pattern type used to fill area right of upper specification limit (histograms only)
USL= <i>value-list</i>	upper specification limit values
USLSYMBOL= <i>'character'</i>	character used to plot upper specification limit on line printers
WLSL= <i>n</i>	width of upper specification limit line

Dictionary of Options

The following entries provide detailed descriptions of options for the SPEC statement. The marginal notes *Graphics* and *Line Printer* identify options that can be used only with graphics devices and line printers, respectively.

CLEFT=*color*

specifies the color of the pattern used to fill the area under a histogram to the left of the lower specification limit. This option is applicable only when the SPEC statement is used in conjunction with a HISTOGRAM or COMPHISTOGRAM statement. See Output 1.2.1 on page 60 for an example. If a pattern is specified for this area with the PLEFT= option, the default color is the second color in the device color list. The CLEFT= option also applies to the area under a fitted curve; for an example, see Output 4.1.1 on page 171.

*Graphics***CLSL**=*color*

specifies the color of the the lower specification line displayed in plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. The default is the second color in the device color list.

*Graphics***CRIGHT**=*color*

specifies the color of the pattern used to fill the area under a histogram to the right of the upper specification limit. This option is applicable only when the SPEC statement is used in conjunction with a HISTOGRAM or COMPHISTOGRAM statement. See

Graphics

Output 1.2.1 on page 60 for an example. If a pattern is specified for this area with the PRIGHT= option, the default color is the third color in the device color list. The CRIGHT= option also applies to the area under a fitted curve; for an example, see Output 4.1.1 on page 171.

Graphics

CTARGET=*color*

specifies the color of the the target line displayed in plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. The default is the first color in the device color list.

Graphics

CUSL=*color*

specifies the color of the upper specification line displayed in plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. The default is the third color in the device color list.

Graphics

LLSL=*linetype*

specifies the line type for the lower specification line displayed in plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. See Output 1.2.1 on page 60 for an example. The default is 1, which produces a solid line.

LSL=*value-list*

specifies the lower specification limits for the variables listed in the VAR statement, or for all numeric variables in the input data set if no VAR statement is used. If you specify only one lower limit, it is used for all of the variables; otherwise, the number of limits must match the number of variables. See “Computing Capability Indices” on page 11 for an example.

Line Printer

LSLSYMBOL=*'character'*

specifies the *character* used to display the lower specification line in line printer plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. The default character is 'L'.

Graphics

LTARGET=*linetype*

specifies the line type for the target line in plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. See Output 1.2.1 on page 60 for an example. The default is 1, which produces a solid line.

Graphics

LUSL=*linetype*

specifies the line type for the upper specification line displayed in plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. See Output 1.2.1 on page 60 for an example. The default is 1, which produces a solid line.

Graphics

PLEFT=*pattern*

specifies the pattern used to fill the area under a histogram to the left of the lower specification limit. This option is applicable only when the SPEC statement is used in conjunction with a HISTOGRAM or COMPHISTOGRAM statement. For an example, see Output 1.2.1 on page 60. The PLEFT= option also applies to the area under a fitted curve; for an example, see Output 4.1.1 on page 171. If a CLEFT= color is specified, the default pattern is a solid fill.

PRIGHT=*pattern*

specifies the pattern used to fill the area under a histogram to the right of the upper specification limit. This option is applicable only when the SPEC statement is used in conjunction with a HISTOGRAM or COMPHISTOGRAM statement. For an example, see Output 1.2.1 on page 60. The PRIGHT= option also applies to the area under a fitted curve; for an example, see Output 4.1.1 on page 171. If a CRIGHT= color is specified, the default pattern is a solid fill.

Graphics

TARGET=*value-list*

specifies a target values for the variables listed in the VAR statement, or for all numeric variables in the input data set if no VAR statement is used. If you specify only one target value, it is used for all of the variables; otherwise, the number of values must match the number of variables. See “Computing Capability Indices” on page 11 for an example.

TARGETSYMBOL=*'character'***TARGETSYM=***'character'*

specifies the *character* used to display the target line in line printer plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. The default character is 'T'.

Line Printer

USL=*value-list*

specifies the upper specification limits for the variables listed in the VAR statement, or for all numeric variables in the input data set if no VAR statement is used. If you specify only one upper limit, it is used for all of the variables; otherwise, the number of limits must match the number of variables. See “Computing Capability Indices” on page 11 for an example.

USLSYMBOL=*'character'*

specifies the *character* used to display the upper specification line in line printer plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. The default character is 'U'.

Line Printer

WLSL=*n*

specifies the width in pixels of the lower specification line in plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. See Output 1.2.1 on page 60 for an illustration. The default is 1.

Graphics

WTARGET=*n*

specifies the width in pixels of the target line in plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. See Output 1.2.1 on page 60 for an illustration. The default is 1.

Graphics

WUSL=*n*

specifies the width in pixels of the upper specification line in plots created with the CDFPLOT, COMPHISTOGRAM, HISTOGRAM, PROBPLOT, and QQPLOT statements. See Output 1.2.1 on page 60 for an illustration. The default is 1.

Graphics

Details

This section provides details on the following topics:

- input data sets specified with the DATA= option, the SPEC= option, and the ANNOTATE= option
- the output data set specified with the OUTTABLE= option
- descriptive statistics
- the tests for normality requested with the NORMALTEST option
- percentile definitions controlled using the PCTLDEF= option
- robust estimators
- computing the mode
- assumptions and terminology for capability indices
- standard capability indices
- specialized capability indices

Input Data Sets

DATA= Data Set

The DATA= data set contains a set of variables that represent measurements from a process. The CAPABILITY procedure must have a DATA= data set. If you do not specify one with the DATA= option in the PROC CAPABILITY statement, the procedure uses the last data set created.

SPEC= Data Set

The SPEC= option in the PROC CAPABILITY statement identifies a SPEC= data set, which contains specification limits. This option is an alternative to using the SPEC statement. If you use both the SPEC= option and a SPEC statement, the SPEC= option is ignored. The SPEC= option is especially useful when:

- the number of variables is large
- the same specification limits are referred to in more than one analysis
- a BY statement is used
- batch processing is used

The following variables are read from a SPEC= data set:

Variable	Description
LSL	lower specification limit
TARGET	target value
USL	upper specification limit
VAR	name of the variable

You may omit either `_LSL_` or `_USL_` but not both. `_TARGET_` is optional. If the `SPEC=` data set contains both `_LSL_` and `_USL_`, you can assign missing values to `_LSL_` or `_USL_` to indicate one-sided specifications. You can assign missing values to `_TARGET_` when the variable does not use a target value. `_LSL_`, `_USL_`, and `_TARGET_` must be numeric variables. `_VAR_` must be a character variable.

If you are using the `HISTOGRAM` statement to create “clickable” histograms in HTML, you can also provide the following variables in a `SPEC=` data set:

Variable	Description
<code>_LOURL_</code>	URL associated with area to left of lower specification limit
<code>_HIURL_</code>	URL associated with area to right of upper specification limit
<code>_URL_</code>	URL associated with area between specification limits

These are character variables whose values are Uniform Resource Locators (URLs) linked to areas on a histogram. When you view the ODS HTML output with a browser, you can click on an area, and the browser will bring up the page specified by the corresponding URL.

If you use a `BY` statement, the `SPEC=` data set must also contain the `BY` variables. The `SPEC=` data set must be sorted in the same order as the `DATA=` data set. Within a `BY` group, specification limits for each variable plotted are read from the first observation where `_VAR_` matches the variable name.

See the “Examples” section on page 56 for an example of reading specification limits from a `SPEC=` data set.

ANNOTATE= Data Sets

In the `CAPABILITY` procedure, you can add features to plots by specifying `ANNOTATE=` data sets either in the `PROC CAPABILITY` statement or in individual plot statements. Depending on where you specify an `ANNOTATE=` data set, however, the information is used for all plots or only for plots produced by a given statement.

Information contained in the `ANNOTATE=` data set specified in the `PROC CAPABILITY` statement is used for all plots produced in a given `PROC` step; this is a “global” `ANNOTATE=` data set. By using this global data set, you can keep information common to all high-resolution plots in one data set.

Information contained in the `ANNOTATE=` data set specified in a plot statement is used for plots produced by that statement; this is a “local” `ANNOTATE=` data set. By using this data set, you can add statement-specific features to plots. For example, you can add different features to plots produced by the `HISTOGRAM` and `QQPLOT` statements by specifying an `ANNOTATE=` data set in each plot statement.

In addition, you can specify an `ANNOTATE=` data set in the `PROC CAPABILITY` statement and in plot statements. This allows you to add some features to all plots (those given in the data set specified in the `PROC` statement) and also add statement-specific features to plots (those given in the data set specified in the plot statement).

For complete details on the structure and content of Annotate type data sets, see SAS/GRAPH documentation.

Output Data Set

OUTTABLE= Data Set

The OUTTABLE= data set saves univariate statistics and capability indices. The following variables can be saved:

Variable	Description
CP	capability index C_p
CPK	capability index C_{pk}
CPL	capability index C_{PL}
CPM	capability index C_{pm}
CPU	capability index C_{PU}
K	capability index K
KURT	kurtosis
LSL	lower specification limit
MAX	maximum
MEAN	mean
MEDIAN	median
MIN	minimum
MODE	mode
NMISS	number of missing observations
N	number of nonmissing observations
P1	1 st percentile
P5	5 th percentile
P10	10 th percentile
P90	90 th percentile
P95	95 th percentile
P99	99 th percentile
PCTGTR	percentage of observations greater than upper specification limit
PCTLSS	percentage of observations less than lower specification limit
Q1	25 th percentile (lower quartile)
Q3	75 th percentile (upper quartile)
QRANGE	interquartile range (upper quartile minus lower quartile)
RANGE	range
SGNRNK	centered sign rank
SKEW	skewness
STD	standard deviation
SUMWGT	sum of the weights
SUM	sum
TARGET	target value
USL	upper specification limit
VARI	variance
VAR	variable name

Note: The variables _CP_, _CPK_, _CPL_, _CPM_, _CPU_, _K_, _LSL_, _PCTGTR_, _PCTLSS_, _TARGET_, and _USL_ are included if you provide specification limits.

The OUTTABLE= data set and the OUT= data set* starting on page 233 for details on the OUT= data set. contain essentially the same information. However, the structure of the OUTTABLE= data set may be more appropriate when you are computing summary statistics or capability indices for more than one process variable in the same invocation of the CAPABILITY procedure. Each observation in the OUTTABLE= data set corresponds to a different process variable, and the variables in the data set correspond to summary statistics and indices.

For example, suppose you have ten process variables (P1-P10). The following statements create an OUTTABLE= data set named TABLE, which contains summary statistics and capability indices for each of these variables:

See CAPTAB1
in the SAS/QC
Sample Library

```
proc capability data=process outtable=table noprint;
  var p1-p10;
  specs lsl= 5 10 65 35 35 5 25 25 60 15
          usl=175 275 300 450 550 200 275 425 500 525;
run;
```

The following statements create the table shown in Figure 1.4, which contains the mean, standard deviation, lower and upper specification limits, and capability index C_{pk} for each process variable:

```
proc print data=table label noobs;
  var _VAR_ MEAN_ _STD_ _LSL_ _USL_ _CPK_;
  label _VAR_='Process';
run;
```

Process	Mean	Standard Deviation	Lower Specification Limit	Upper Specification Limit	Capability Index CPK
p1	90.76	57.024	5	175	0.49242
p2	167.32	81.628	10	275	0.43972
p3	224.56	96.525	65	300	0.26052
p4	258.08	145.218	35	450	0.44053
p5	283.48	157.033	35	550	0.52745
p6	107.48	52.437	5	200	0.58814
p7	153.20	90.031	25	275	0.45096
p8	217.08	130.031	25	425	0.49239
p9	280.68	140.943	60	500	0.51870
p10	243.24	178.799	15	525	0.42551

Figure 1.4. Tabulating Results for Multiple Process Variables

*See Chapter 7, "OUTPUT Statement"

Descriptive Statistics

This section provides computational details for the descriptive statistics which are computed with the PROC CAPABILITY statement. These statistics can also be saved in the OUT= data set by specifying the keywords listed in Table 7.1 on page 240 in the OUTPUT statement.

Standard algorithms (Fisher 1973) are used to compute the moment statistics. The computational methods used by the CAPABILITY procedure are consistent with those used by other SAS procedures for calculating descriptive statistics. For details on statistics also calculated by base SAS software, see *SAS Language Reference: Dictionary*.

The following sections give specific details on several statistics calculated by the CAPABILITY procedure.

Mean

The sample mean is calculated as

$$\frac{\sum_{i=1}^n w_i x_i}{\sum_{i=1}^n w_i}$$

where n is the number of nonmissing values for a variable, x_i is the i^{th} value of the variable, and w_i is the weight associated with the i^{th} value of the variable. If there is no WEIGHT= variable, the formula reduces to $\frac{1}{n} \sum_{i=1}^n x_i$.

Sum

The sum is calculated as $\sum_{i=1}^n w_i x_i$, where n is the number of nonmissing values for a variable, x_i is the i^{th} value of the variable, and w_i is the weight associated with the i^{th} value of the variable. If there is no WEIGHT= variable, the formula reduces to $\sum_{i=1}^n x_i$.

Sum of the Weights

The sum of the weights is calculated as $\sum_{i=1}^n w_i$, where n is the number of nonmissing values for a variable and w_i is the weight associated with the i^{th} value of the variable. If there is no WEIGHT= variable, the sum of the weights is n .

Variance

The variance is calculated as

$$\frac{1}{d} \sum_{i=1}^n w_i (x_i - \bar{X}_w)^2$$

where n is the number of nonmissing values for a variable, x_i is the i^{th} value of the variable, \bar{X}_w is the weighted mean, w_i is the weight associated with the i^{th} value of the variable, and d is the divisor controlled by the VARDEF= option in the PROC CAPABILITY statement. If there is no WEIGHT= variable, the formula reduces to

$$\frac{1}{d} \sum_{i=1}^n (x_i - \bar{X}_w)^2$$

Standard Deviation

The standard deviation is calculated as

$$\sqrt{\frac{1}{d} \sum_{i=1}^n w_i (x_i - \bar{X}_w)^2}$$

where n is the number of nonmissing values for a variable, x_i is the i^{th} value of the variable, \bar{X}_w is the weighted mean, w_i is the weight associated with the i^{th} value of the variable, and d is the divisor controlled by the VARDEF= option in the PROC CAPABILITY statement. If there is no WEIGHT= variable, the formula reduces to

$$\sqrt{\frac{1}{d} \sum_{i=1}^n (x_i - \bar{X}_w)^2}$$

Skewness

The sample skewness is calculated as

$$\frac{n}{(n-1)(n-2)} \sum_{i=1}^n \left(\frac{x_i - \bar{X}}{s} \right)^3$$

where n is the number of nonmissing values for a variable and must be greater than 2, x_i is the i^{th} value of the variable, \bar{X} is the sample average, and s is the sample standard deviation.

The sample skewness can be positive or negative; it measures the asymmetry of the data distribution and estimates the theoretical skewness $\sqrt{\beta_1} = \mu_3 \mu_2^{-\frac{3}{2}}$, where μ_2 and μ_3 are the second and third central moments. Observations that are normally distributed should have a skewness near zero.

Kurtosis

The sample kurtosis is calculated as

$$\frac{n(n+1)}{(n-1)(n-2)(n-3)} \sum_{i=1}^n \left(\frac{x_i - \bar{X}}{s} \right)^4 - \frac{3(n-1)^2}{(n-2)(n-3)}$$

where $n > 3$. The sample kurtosis measures the heaviness of the tails of the data distribution. It estimates the adjusted theoretical kurtosis denoted as $\beta_2 - 3$, where $\beta_2 = \frac{\mu_4}{\mu_2^2}$, and μ_4 is the fourth central moment. Observations that are normally distributed should have a kurtosis near zero.

Coefficient of Variation (CV)

The coefficient of variation is calculated as

$$CV = \frac{100 \times s}{\bar{X}}$$

Signed Rank Statistic

The signed rank statistic S is computed as

$$S = \sum_{i: x_i > 0} r_i^+ - \frac{n(n+1)}{4}$$

where r_i^+ is the rank of $|x_i|$ after discarding values of $x_i = 0$, and n is the number of nonzero x_i values. Average ranks are used for tied values.

If $n \leq 20$, the significance of S is computed from the exact distribution of S , where the distribution is a convolution of scaled binomial distributions. When $n > 20$, the significance of S is computed by treating

$$S \sqrt{\frac{n-1}{nV - S^2}}$$

as a Student t variate with $n - 1$ degrees of freedom. V is computed as

$$V = \frac{1}{24}n(n+1)(2n+1) - \frac{1}{48} \sum t_i(t_i+1)(t_i-1)$$

where the sum is over groups tied in absolute value and where t_i is the number of values in the i^{th} group (Iman 1974, Conover 1980). The null hypothesis tested is that the mean (or median) is zero, assuming that the distribution is symmetric. Refer to Lehmann (1975).

Tests for Normality

You can use the NORMALTEST option in the PROC CAPABILITY statement to request several tests of the hypothesis that the analysis variable values are a random sample from a normal distribution. These tests, which are summarized in the table labeled *Tests for Normality*, include the following:

- Shapiro-Wilk test
- Kolmogorov-Smirnov test
- Anderson-Darling test
- Cramér-von Mises test

Tests for normality are particularly important in process capability analysis because the commonly used capability indices are difficult to interpret unless the data are at least approximately normally distributed. Furthermore, the confidence limits for capability indices displayed in the table labeled *Process Capability Indices* require the assumption of normality. Consequently, the tests of normality are always computed when you specify the SPEC statement, and a note is added to the table when the hypothesis of normality is rejected. You can specify the particular test and the significance level with the CHECKINDICES option.

Shapiro-Wilk Test

If the sample size is 2000 or less,* the procedure computes the Shapiro-Wilk statistic W (also denoted as W_n to emphasize its dependence on the sample size n). The statistic W_n is the ratio of the best estimator of the variance (based on the square of a linear combination of the order statistics) to the usual corrected sum of squares estimator of the variance. When n is greater than three, the coefficients to compute the linear combination of the order statistics are approximated by the method of Royston (1992). The statistic W_n is always greater than zero and less than or equal to one ($0 < W \leq 1$).

Small values of W lead to rejection of the null hypothesis. The method for computing the p -value (the probability of obtaining a W statistic less than or equal to the observed value) depends on n . For $n = 3$, the probability distribution of W is known and is used to determine the p -value. For $n > 4$, a normalizing transformation is computed:

$$Z_n = \begin{cases} (-\log(\gamma - \log(1 - W_n)) - \mu)/\sigma & \text{if } 4 \leq n \leq 11 \\ (\log(1 - W_n) - \mu)/\sigma & \text{if } 12 \leq n \leq 2000 \end{cases}$$

The values of σ , γ , and μ are functions of n obtained from simulation results. Large values of Z_n indicate departure from normality, and since the statistic Z_n has an approximately standard normal distribution, this distribution is used to determine the p -values for $n > 4$.

EDF Tests for Normality

The Kolmogorov-Smirnov, Anderson-Darling and Cramér-von Mises tests for normality are based on the empirical distribution function (EDF) and are often referred to as EDF tests. EDF tests for a variety of non-normal distributions are available in the HISTOGRAM statement; see the “EDF Goodness-of-Fit Tests” section on page 159 for details. For a thorough discussion of these tests, refer to D’Agostino and Stephens (1986).

The empirical distribution function is defined for a set of n independent observations X_1, \dots, X_n with a common distribution function $F(x)$. Under the null hypothesis, $F(x)$ is the normal distribution. Denote the observations ordered from smallest to largest as $X_{(1)}, \dots, X_{(n)}$. The empirical distribution function, $F_n(x)$, is defined as

$$F_n(x) = \begin{cases} 0, & x < X_{(1)} \\ \frac{i}{n}, & X_{(i)} \leq x < X_{(i+1)}, i = 1, \dots, n-1 \\ 1, & X_{(n)} \leq x \end{cases}$$

Note that $F_n(x)$ is a step function that takes a step of height $\frac{1}{n}$ at each observation. This function estimates the distribution function $F(x)$. At any value x , $F_n(x)$ is the

*In Release 6.12 and earlier releases, the CAPABILITY procedure performed a Shapiro-Wilk test for sample sizes of 2000 or smaller, and a Kolmogorov-Smirnov test otherwise. The computed value of W was used to interpolate linearly within the range of simulated critical values given in Shapiro and Wilk (1965). In Version 7, minor improvements have been made to the algorithm for the Shapiro-Wilk test, as described in this section.

Part 1. The CAPABILITY Procedure

proportion of observations less than or equal to x , while $F(x)$ is the probability of an observation less than or equal to x . EDF statistics measure the discrepancy between $F_n(x)$ and $F(x)$.

The EDF tests make use of the probability integral transformation $U = F(X)$. If $F(X)$ is the distribution function of X , the random variable U is uniformly distributed between 0 and 1. Given n observations $X_{(1)}, \dots, X_{(n)}$, the values $U_{(i)} = F(X_{(i)})$ are computed. These values are used to compute the EDF test statistics, as described in the next three sections. The CAPABILITY procedures compute the associated p -values by interpolating internal tables of probability levels similar to those given by D'Agostino and Stephens (1986).

Kolmogorov-Smirnov Test

The Kolmogorov-Smirnov statistic (D) is defined as

$$D = \sup_x |F_n(x) - F(x)|$$

The Kolmogorov-Smirnov statistic belongs to the supremum class of EDF statistics. This class of statistics is based on the largest vertical difference between $F(x)$ and $F_n(x)$.

The Kolmogorov-Smirnov statistic is computed as the maximum of D^+ and D^- , where D^+ is the largest vertical distance between the EDF and the distribution function when the EDF is greater than the distribution function, and D^- is the largest vertical distance when the EDF is less than the distribution function.

$$\begin{aligned} D^+ &= \max_i \left(\frac{i}{n} - U_{(i)} \right) \\ D^- &= \max_i \left(U_{(i)} - \frac{i-1}{n} \right) \\ D &= \max(D^+, D^-) \end{aligned}$$

PROC CAPABILITY uses a modified Komogorov D statistic to test the data against a normal distribution with mean and variance equal to the sample mean and variance.

Anderson-Darling Test

The Anderson-Darling statistic and the Cramér-von Mises statistic belong to the quadratic class of EDF statistics. This class of statistics is based on the squared difference $(F_n(x) - F(x))^2$. Quadratic statistics have the following general form:

$$Q = n \int_{-\infty}^{+\infty} (F_n(x) - F(x))^2 \psi(x) dF(x)$$

The function $\psi(x)$ weights the squared difference $(F_n(x) - F(x))^2$.

The Anderson-Darling statistic (A^2) is defined as

$$A^2 = n \int_{-\infty}^{+\infty} (F_n(x) - F(x))^2 [F(x)(1 - F(x))]^{-1} dF(x)$$

Here the weight function is $\psi(x) = [F(x)(1 - F(x))]^{-1}$.

The Anderson-Darling statistic is computed as

$$A^2 = -n - \frac{1}{n} \sum_{i=1}^n [(2i-1) \log U_{(i)} + (2n+1-2i) \log (1-U_{(i)})]$$

Cramér-von Mises Test

The Cramér-von Mises statistic (W^2) is defined as

$$W^2 = n \int_{-\infty}^{+\infty} (F_n(x) - F(x))^2 dF(x)$$

Here the weight function is $\psi(x) = 1$.

The Cramér-von Mises statistic is computed as

$$W^2 = \sum_{i=1}^n \left(U_{(i)} - \frac{2i-1}{2n} \right)^2 + \frac{1}{12n}$$

Percentile Computations

The CAPABILITY procedure automatically computes the 1st, 5th, 10th, 25th, 50th, 75th, 90th, 95th, and 99th percentiles (quantiles), as well as the minimum and maximum of each analysis variable. To compute percentiles other than these default percentiles, use the PCTLPTS= and PCTLPRE= options in the OUTPUT statement.

You can specify one of five definitions for computing the percentiles with the PCTLDEF= option. Let n be the number of nonmissing values for a variable, and let x_1, x_2, \dots, x_n represent the ordered values of the variable. Let the t^{th} percentile be y , set $p = \frac{t}{100}$, and let

$$\begin{aligned} np &= j + g && \text{when PCTLDEF}=1, 2, 3, \text{ or } 5 \\ (n+1)p &= j + g && \text{when PCTLDEF}=4 \end{aligned}$$

where j is the integer part of np , and g is the fractional part of np . Then the PCTLDEF= option defines the t^{th} percentile, y , as described in the following table:

Part 1. The CAPABILITY Procedure

PCTLDEF=	Description	Formula
1	weighted average at x_{np}	$y = (1 - g)x_j + gx_{j+1}$ where x_0 is taken to be x_1
2	observation numbered closest to np	$y = x_i$ if $g \neq \frac{1}{2}$ $y = x_j$ if $g = \frac{1}{2}$ and j is even $y = x_{j+1}$ if $g = \frac{1}{2}$ and j is odd where i is the integer part of $np + \frac{1}{2}$
3	empirical distribution function	$y = x_j$ if $g = 0$ $y = x_{j+1}$ if $g > 0$
4	weighted average aimed at $x_{(n+1)p}$	$y = (1 - g)x_j + gx_{j+1}$ where x_{n+1} is taken to be x_n
5	empirical distribution function with averaging	$y = \frac{1}{2}(x_j + x_{j+1})$ if $g = 0$ $y = x_{j+1}$ if $g > 0$

Weighted Percentiles

When you use a WEIGHT statement, the percentiles are computed differently. The 100 p th weighted percentile y is computed from the empirical distribution function with averaging

$$y = \begin{cases} \frac{1}{2}(x_i + x_{i+1}) & \text{if } \sum_{j=1}^i w_j = pW \\ x_{i+1} & \text{if } \sum_{j=1}^i w_j < pW < \sum_{j=1}^{i+1} w_j \end{cases}$$

where w_i is the weight associated with x_i , and where $W = \sum_{i=1}^n w_i$ is the sum of the weights.

Note that the PCTLDEF= option is not applicable when a WEIGHT statement is used. However, in this case, if all the weights are identical, the weighted percentiles are the same as the percentiles that would be computed without a WEIGHT statement and with PCTLDEF=5.

Confidence Limits for Percentiles

You can use the CIPCTLNORMAL option to request confidence limits for percentiles which assume the data are normally distributed. These limits are described in Section 4.4.1 of Hahn and Meeker (1991). When $0.0 < p < 0.5$, the two-sided 100(1 - α)% confidence limits for the 100 p -th percentile are

$$\begin{aligned} \text{lower limit} &= \bar{X} - g'(\alpha/2; 1 - p, n)s \\ \text{upper limit} &= \bar{X} - g'(1 - \alpha/2; p, n)s \end{aligned}$$

where n is the sample size. When $0.5 \leq p < 1.0$, the two-sided 100(1 - α)% confidence limits for the 100 p -th percentile are

$$\begin{aligned} \text{lower limit} &= \bar{X} + g'(\alpha/2; 1 - p, n)s \\ \text{upper limit} &= \bar{X} + g'(1 - \alpha/2; p, n)s \end{aligned}$$

One-sided 100(1 - α)% confidence bounds are computed by replacing $\alpha/2$ by α in

the appropriate equation above. The factor $g'(\gamma, p, n)$ is related to the noncentral t distribution and is described in Owen and Hua (1977) and Odeh and Owen (1980).

You can use the CIPCTLDF option to request confidence limits for percentiles which are distribution free (in particular, it is not necessary to assume that the data are normally distributed). These limits are described in Section 5.2 of Hahn and Meeker (1991). The two-sided $100(1 - \alpha)\%$ confidence limits for the $100p$ -th percentile are

$$\begin{aligned}\text{lower limit} &= X_{(l)} \\ \text{upper limit} &= X_{(u)}\end{aligned}$$

where $X_{(j)}$ is the j th order statistic when the data values are arranged in increasing order:

$$X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(n)}$$

The lower rank l and upper rank u are integers that are symmetric (or nearly symmetric) around $[np] + 1$ where $[np]$ is the integer part of np , and where n is the sample size. Furthermore, l and u are chosen so that $X_{(l)}$ and $X_{(u)}$ are as close to $X_{[n+1]p}$ as possible while satisfying the coverage probability requirement

$$Q(u - 1; n, p) - Q(l - 1; n, p) \geq 1 - \alpha$$

where $Q(k; n, p)$ is the cumulative binomial probability

$$Q(k; n, p) = \sum_{i=0}^k \binom{n}{i} p^i (1 - p)^{n-i}$$

In some cases, the coverage requirement cannot be met, particularly when n is small and p is near 0 or 1. To relax the requirement of symmetry, you can specify CIPCTLDF(TYPE = ASYMMETRIC). This option requests symmetric limits when the coverage requirement can be met, and asymmetric limits otherwise.

If you specify CIPCTLDF(TYPE = LOWER), a one-sided $100(1 - \alpha)\%$ lower confidence bound is computed as X_l , where l is the largest integer that satisfies the inequality

$$1 - Q(l - 1; n, p) \geq 1 - \alpha$$

with $0 < l \leq n$. Likewise, if you specify CIPCTLDF(TYPE = UPPER), a one-sided $100(1 - \alpha)\%$ lower confidence bound is computed as X_l , where l is the largest integer that satisfies the inequality

$$Q(u - 1; n, p) \geq 1 - \alpha$$

where $0 < u \leq n$.

Note that confidence limits for percentiles are not computed when a WEIGHT statement is specified.

Robust Estimators

The CAPABILITY procedure provides several methods for computing robust estimates of location and scale, which are insensitive to outliers in the data.

Winsorized Means

The k -times Winsorized mean is a robust estimator of location which is computed as

$$\bar{x}_{wk} = \frac{1}{n} \left((k+1)x_{(k+1)} + \sum_{i=k+2}^{n-k-1} x_{(i)} + (k+1)x_{(n-k)} \right)$$

where n is the number of observations, and $x_{(i)}$ is the i th order statistic when the observations are arranged in increasing order:

$$x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}$$

The Winsorized mean is the mean computed after replacing the k smallest observations with the $(k+1)$ st smallest observation, and the k largest observations with the $(k+1)$ st largest observation.

For data from a symmetric distribution, the Winsorized mean is an unbiased estimate of the population mean. However, the Winsorized mean does not have a normal distribution even if the data are normally distributed.

The Winsorized sum of squared deviations is defined as

$$s_{wk}^2 = (k+1)(x_{(k+1)} - \bar{x}_{wk})^2 + \sum_{i=k+2}^{n-k-1} (x_{(i)} - \bar{x}_{wk})^2 + (k+1)(x_{(n-k)} - \bar{x}_{wk})^2$$

A Winsorized t test is given by

$$t_{wk} = \frac{\bar{x}_{wk} - \mu_0}{\text{STDERR}(\bar{x}_{wk})}$$

where the standard error of the Winsorized mean is

$$\text{STDERR}(\bar{x}_{wk}) = \frac{n-1}{n-2k-1} \frac{s_{wk}}{\sqrt{n(n-1)}}$$

When the data are from a symmetric distribution, the distribution of t_{wk} is approximated by a Student's t distribution with $n-2k-1$ degrees of freedom. Refer to Tukey and McLaughlin (1963) and Dixon and Tukey (1968).

A $100(1-\alpha)\%$ Winsorized confidence interval for the mean has upper and lower limits

$$\bar{x}_{wk} \pm t_{1-\alpha/2} \text{STDERR}(\bar{x}_{wk})$$

where $t_{1-\alpha/2}$ is the $(1 - \alpha)/2$ 100th percentile of the Student's t distribution with $n - 2k - 1$ degrees of freedom.

Trimmed Means

The k -times trimmed mean is a robust estimator of location which is computed as

$$\bar{x}_{tk} = \frac{1}{n - 2k} \sum_{i=k+1}^{n-k} x_{(i)}$$

where n is the number of observations, and $x_{(i)}$ is the i th order statistic when the observations are arranged in increasing order:

$$x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}$$

The trimmed mean is the mean computed after the k smallest observations and the k largest observations in the sample are deleted.

For data from a symmetric distribution, the trimmed mean is an unbiased estimate of the population mean. However, the trimmed mean does not have a normal distribution even if the data are normally distributed.

A robust estimate of the variance of the trimmed mean t_{tk} can be obtained from the Winsorized sum of squared deviations; refer to Tukey and McLaughlin (1963). the corresponding trimmed t test is given by

$$t_{tk} = \frac{\bar{x}_{tk} - \mu_0}{\text{STDERR}(\bar{x}_{tk})}$$

where the standard error of the trimmed mean is

$$\text{STDERR}(\bar{x}_{tk}) = \frac{s_{tk}}{\sqrt{(n - 2k)(n - 2k - 1)}}$$

and s_{tk} is the square root of the Winsorized sum of squared deviations.

When the data are from a symmetric distribution, the distribution of t_{tk} is approximated by a Student's t distribution with $n - 2k - 1$ degrees of freedom. Refer to Tukey and McLaughlin (1963) and Dixon and Tukey (1968).

A $100(1 - \alpha)\%$ trimmed confidence interval for the mean has upper and lower limits

$$\bar{x}_{tk} \pm t_{1-\alpha/2} \text{STDERR}(\bar{x}_{tk})$$

where $t_{1-\alpha/2}$ is the $(1 - \alpha)/2$ 100th percentile of the Student's t distribution with $n - 2k - 1$ degrees of freedom.

Robust Estimates of Scale

The sample standard deviation, which is the most commonly used estimator of scale, is sensitive to outliers. Robust scale estimators, on the other hand, remain bounded

Part 1. The CAPABILITY Procedure

when a single data value is replaced by an arbitrarily large or small value. The CAPABILITY procedure computes several robust measures of scale, including the interquartile range Gini's mean difference G , the median absolute deviation about the median (MAD), Q_n , and S_n . In addition, the procedure computes estimates of the normal standard deviation σ derived from each of these measures.

The interquartile range (IQR) is simply the difference between the upper and lower quartiles. For a normal population, σ can be estimated as $\text{IQR}/1.34898$.

Gini's mean difference is computed as

$$G = \frac{1}{\binom{n}{2}} \sum_{i < j} |x_i - x_j|$$

For a normal population, the expected value of G is $2\sigma/\sqrt{\pi}$. Thus $G\sqrt{\pi}/2$ is a robust estimator of σ when the data are from a normal sample. For the normal distribution, this estimator has high efficiency relative to the usual sample standard deviation, and it is also less sensitive to the presence of outliers.

A very robust scale estimator is the MAD, the median absolute deviation from the median (Hampel, 1974), which is computed as

$$\text{MAD} = \text{med}_i(|x_i - \text{med}_j(x_j)|)$$

where the inner median, $\text{med}_j(x_j)$, is the median of the n observations, and the outer median (taken over i) is the median of the n absolute values of the deviations about the inner median. For a normal population, 1.4826MAD is an estimator of σ .

The MAD has low efficiency for normal distributions, and it may not always be appropriate for symmetric distributions. Rousseeuw and Croux (1993) proposed two statistics as alternatives to the MAD. The first is

$$S_n = 1.1926 \text{med}_i(\text{med}_j(|x_i - x_j|))$$

where the outer median (taken over i) is the median of the n medians of $|x_i - x_j|$, $j = 1, 2, \dots, n$. To reduce small-sample bias, $c_{sn}S_n$ is used to estimate σ , where c_{sn} is a correction factor; refer to Croux and Rousseeuw (1992).

The second statistic is

$$Q_n = 2.219 \{ |x_i - x_j|; i < j \}_{(k)}$$

where

$$k = \binom{h}{2}$$

and $h = [n/2] + 1$. In other words, Q_n is 2.219 times the k th order statistic of the $\binom{n}{2}$ distances between the data points. The bias-corrected statistic $c_{qn}Q_n$ is used to estimate σ , where c_{qn} is a correction factor; refer to Croux and Rousseeuw (1992).

Computing the Mode

The mode is the value that occurs most often in a set of observations. The CAPABILITY procedure counts repetitions of the actual values (or the rounded values, if you specify the ROUND= option). If a tie occurs for the most frequent value, the procedure reports the lowest mode in the table labeled *Basic Statistical Measures*. To list all possible modes, specify the MODES option in the PROC CAPABILITY statement. When no repetitions occur in the data, the procedure does not report the mode. The WEIGHT statement has no effect on the mode.

Assumptions and Terminology for Capability Indices

One of the fundamental assumptions in process capability analysis is that the process must be in statistical control. Without statistical control, the process is not predictable, the concept of a process distribution does not apply, and quantities related to the distribution, such as probabilities, percentiles, and capability indices, cannot be meaningfully estimated. Additionally, all of the standard process capability indices described in the next section require that the process distribution be normal, or at least approximately normal.

In many industries, statistical control is routinely checked with a Shewhart chart (such as an \bar{X} and R chart) before capability indices such as

$$C_{pk} = \min \left(\frac{USL - \mu}{3\sigma}, \frac{LSL - \mu}{3\sigma} \right)$$

are computed. The control chart analysis yields estimates for the process mean μ and standard deviation σ , which are based on subgrouped data and can be used to estimate C_{pk} . In particular, σ can be estimated by

$$s_R = \bar{R}/d_2$$

rather than the ungrouped sample standard deviation

$$s = \frac{1}{n-1} \sqrt{\sum_{i=1}^n n(x_i - \bar{x})^2}$$

You can use the SHEWHART procedure to carry out the control chart analysis and to compute capability indices based on s_R . On the other hand, the CAPABILITY procedure computes indices based on s .

Some industry manuals distinguish these two approaches. For instance, the ASQC/AIAG manual *Fundamental Process Control* uses the notation C_{pk} for the estimate based on s_R , and it uses the notation P_{pk} for the estimate based on s . However, assuming that the process is in control and only common cause variation is present, both s_R and s are estimates of the same parameter σ , and so there is fundamentally no difference in the two approaches*.

*Statistically, s is a more efficient estimator of σ than s_R .

Once control has been established, attention should focus on the distribution of the process measurements, and at this point there is no practical or statistical advantage to working with subgrouped measurements. In fact, the use of s is closely associated with a wide variety of methods that are highly useful for process capability analysis, including tests for normality, graphical displays such as histograms and probability plots, and confidence intervals for parameters and capability indices.

Standard Capability Indices

This section provides computational details for the standard process capability indices computed by the CAPABILITY procedure: C_p , CPL , CPU , C_{pk} , and C_{pm} .

The Index C_p

The process capability index C_p , sometimes called the “process potential index,” the “process capability ratio,” or the “inherent capability index,” is estimated as

$$\hat{C}_p = \frac{USL - LSL}{6s}$$

where USL is the upper specification limit, LSL is the lower specification limit, and s is the sample standard deviation. If you do not specify both the upper and the lower specification limits in the SPEC statement or the SPEC= data set, then C_p is assigned a missing value.

The interpretation of C_p can depend on the application, on past experience, and on local practice. However, broad guidelines for interpretation have been proposed by several authors. Ekvall and Juran (1974) classify C_p values as

- “not adequate” if $C_p < 1$
- “adequate” if $1 \leq C_p \leq 1.33$, but requiring close control as C_p approaches 1
- “more than adequate” if $C_p > 1.33$

Montgomery (1996) recommends minimum values of C_p as

- 1.33 for existing processes
- 1.50 for new processes or for existing processes when the variable is critical (for example, related to safety or strength)
- 1.67 for new processes when the variable is critical

Exact $100(1 - \alpha)\%$ lower and upper confidence limits for C_p (denoted by LCL and UCL) are computed using percentiles of the chi-square distribution, as indicated by the following equations:

$$\begin{aligned}\text{lower limit} &= \hat{C}_p \sqrt{\chi_{\alpha/2, n-1}^2 / (n-1)} \\ \text{upper limit} &= \hat{C}_p \sqrt{\chi_{1-\alpha/2, n-1}^2 / (n-1)}\end{aligned}$$

Here, $\chi_{\alpha, \nu}^2$ denotes the lower $100\alpha^{\text{th}}$ percentile of the chi-square distribution with ν degrees of freedom. Refer to Chou et al. (1990) and Kushler and Hurley (1992).

You can specify α with the ALPHA= option in the PROC CAPABILITY statement or with the CIINDICES(ALPHA=value) in the PROC CAPABILITY statement. The default value is 0.05. You can save these limits in the OUT= data set by specifying the keywords CPLCL and CPUCL in the OUTPUT statement. In addition, you can display these limits on plots produced by the CAPABILITY procedure by specifying the keywords in the INSET statement.

The Index CPL

The process capability index *CPL* is estimated as

$$\widehat{CPL} = \frac{\bar{X} - LSL}{3s}$$

where \bar{X} is the sample mean, *LSL* is the lower specification limit, and *s* is the sample standard deviation. If you do not specify the lower specification limit in the SPEC statement or the SPEC= data set, then *CPL* is assigned a missing value.

Montgomery (1996) refers to *CPL* as the “process capability ratio” in the case of one-sided lower specifications and recommends minimum values as follows:

- 1.25 for existing processes
- 1.45 for new processes or for existing processes when the variable is critical
- 1.60 for new processes when the variable is critical

Exact $100(1 - \alpha)\%$ lower and upper confidence limits for *CPL* are computed using a generalization of the method of Chou et al. (1990), who point out that the $100(1 - \alpha)$ lower confidence limit for *CPL* (denoted by CPLLCL)satisfies the equation

$$\Pr\{T_{n-1}(\delta = 3\sqrt{n}) \text{ CPLLCL} \leq 3\widehat{CPL}\sqrt{n}\} = 1 - \alpha$$

where $T_{n-1}(\delta)$ has a non-central *t* distribution with $n - 1$ degrees of freedom and noncentrality parameter δ . You can specify α with the ALPHA= option in the PROC CAPABILITY statement. The default value is 0.05. The confidence limits can be saved in an output data set by specifying the keywords CPLLCL and CPLUCL in the OUTPUT statement. In addition, you can display these limits on plots produced by the CAPABILITY procedure by specifying these keywords in the INSET statement.

The Index CPU

The process capability index *CPU* is estimated as

$$\widehat{CPU} = \frac{USL - \bar{X}}{3s}$$

where *USL* is the upper specification limit, \bar{X} is the sample mean, and *s* is the sample standard deviation. If you do not specify the upper specification limit in the SPEC statement or the SPEC= data set, then *CPU* is assigned a missing value.

Montgomery (1996) refers to *CPU* as the “process capability ratio” in the case of one-sided upper specifications and recommends minimum values that are the same as those specified previously for *CPL*.

Part 1. The CAPABILITY Procedure

Exact $100(1 - \alpha)\%$ lower and upper confidence limits for CPU are computed using a generalization of the method of Chou et al. (1990), who point out that the $100(1 - \alpha)\%$ lower confidence limit for CPU (denoted by $CPULCL$) satisfies the equation

$$\Pr\{T_{n-1}(\delta = 3\sqrt{n} \text{ CPULCL} \geq 3\text{CPU}\sqrt{n}\} = 1 - \alpha$$

where $T_{n-1}(\delta)$ has a non-central t distribution with $n - 1$ degrees of freedom and noncentrality parameter δ . You can specify α with the ALPHA= option in the PROC CAPABILITY statement. The default value is 0.05. The confidence limits can be saved in an output data set by specifying the keywords CPULCL and CPUUCL in the OUTPUT statement. In addition, you can display these limits on plots produced by the CAPABILITY procedure by specifying these keywords in the INSET statement.

The Index C_{pk}

The process capability index C_{pk} is defined as

$$C_{pk} = \frac{1}{3\sigma} \min(USL - \mu, \mu - LSL) = \min(CPU, CPL)$$

Note that the indices C_{pk} , C_p , and k are related as $C_{pk} = C_p(1 - k)$. The CAPABILITY procedure estimates C_{pk} as

$$\hat{C}_{pk} = \frac{1}{3s} \times \min(USL - \bar{X}, \bar{X} - LSL) = \min(CPU, CPL)$$

where USL is the upper specification limit, LSL is the lower specification limit, \bar{X} is the sample mean, and s is the sample standard deviation.

If you specify only the upper limit in the SPEC statement or the SPEC= data set, then C_{pk} is computed as CPU , and if you specify only the lower limit in the SPEC statement or the SPEC= data set, then C_{pk} is computed as CPL .

Bissell (1990) derived approximate two-sided 95% confidence limits for C_{pk} by assuming that the distribution of \hat{C}_{pk} is normal. Using Bissell's approach, $100(1 - \alpha)\%$ lower and upper confidence limits can be computed as

$$\begin{aligned} \text{lower limit} &= \hat{C}_{pk} \left[1 - \Phi^{-1}(1 - \alpha/2) \sqrt{\frac{1}{9n\hat{C}_{pk}^2} + \frac{1}{2(n-1)}} \right] \\ \text{upper limit} &= \hat{C}_{pk} \left[1 + \Phi^{-1}(1 - \alpha/2) \sqrt{\frac{1}{9n\hat{C}_{pk}^2} + \frac{1}{2(n-1)}} \right] \end{aligned}$$

where Φ denotes the cumulative standard normal distribution function. Kushler and Hurley (1992) concluded that Bissell's method gives reasonably accurate results.

You can specify α with the ALPHA= option in the PROC CAPABILITY statement. The default value is 0.05. These limits can be saved in an output data set by specifying the keywords CPKLCL and CPKUCL in the OUTPUT statement. In addition,

you can display these limits on plots produced by the CAPABILITY procedure by specifying these same keywords in the INSET statement.

The Index C_{pm}

The process capability index C_{pm} is intended to account for deviation from the target T in addition to variability from the mean. This index is often defined as

$$C_{pm} = \frac{USL - LSL}{6\sqrt{\sigma^2 + (\mu - T)^2}}$$

A closely related version of C_{pm} is the index

$$C_{pm}^* = \frac{\min(USL - T, T - LSL)}{3\sqrt{\sigma^2 + (\mu - T)^2}} = \frac{d - |T - m|}{3\sqrt{\sigma^2 + (\mu - T)^2}}$$

where $d = (USL - LSL)/2$ and $m = (USL + LSL)/2$. If $T = m$, then $C_{pm} = C_{pm}^*$. However, if $T \neq m$, then both indices suffer from problems of interpretation, as pointed out by Kotz and Johnson (1993), and their use should be avoided in this case.

The CAPABILITY procedure computes an estimator of C_{pm} as

$$\hat{C}_{pm} = \frac{\min(USL - T, T - LSL)}{3\sqrt{s^2 + (\bar{X} - T)^2}}$$

where s is the sample standard deviation.

If you specify only a single specification limit SL in the SPEC statement or the SPEC= data set, then C_{pm} is estimated as

$$\hat{C}_{pm} = \frac{|T - SL|}{3\sqrt{s^2 + (\bar{X} - T)^2}}$$

Boyles (1991) proposed a slightly modified point estimate for C_{pm} computed as

$$\tilde{C}_{pm} = \frac{(USL - LSL)/2}{3\sqrt{(\frac{n-1}{n})s^2 + (\bar{X} - T)^2}}$$

Boyles also suggested approximate two-sided $100(1 - \alpha)\%$ confidence limits for C_{pm} , which are computed as

$$\begin{aligned} \text{lower limit} &= \tilde{C}_{pm} \sqrt{\chi_{\alpha/2, \nu}^2 / \nu} \\ \text{upper limit} &= \tilde{C}_{pm} \sqrt{\chi_{1-\alpha/2, \nu}^2 / \nu} \end{aligned}$$

Here $\chi_{\alpha, \nu}^2$ denotes the lower $100\alpha^{\text{th}}$ percentile of the chi-square distribution with ν degrees of freedom, where ν equals

$$\frac{n(1 + (\frac{\bar{X} - T}{s})^2)}{1 + 2(\frac{\bar{X} - T}{s})^2}$$

Part 1. The CAPABILITY Procedure

You can specify α with the ALPHA= option in the PROC CAPABILITY statement. The default value is 0.05. These confidence limits can be saved in an output data set by specifying the keywords CPMLCL and CPMUCL in the OUTPUT statement. In addition, you can display these limits on plots produced by the CAPABILITY procedure by specifying these keywords in the INSET statement.

Specialized Capability Indices

This section describes a number of specialized capability indices which you can request with the SPECIALINDICES option in the PROC CAPABILITY statement.

The Index k

The process capability index k (also denoted by K) is computed as

$$k = \frac{2|m - \bar{X}|}{USL - LSL}$$

where $m = \frac{1}{2}(USL + LSL)$ is the midpoint of the specification limits, \bar{X} is the sample mean, USL is the upper specification limit, and LSL is the lower specification limit.

The formula for k used here is given by Kane (1986). Note that k is sometimes computed without taking the absolute value of $m - \bar{X}$ in the numerator. See Wadsworth *et al.* (1986).

If you do not specify the upper and lower limits in the SPEC statement or the SPEC= data set, then k is assigned a missing value.

Boyles' Index C_{pm}^+

Boyles (1992) proposed the process capability index C_{pm}^+ which is defined as

$$C_{pm}^+ = \frac{1}{3} \left[\frac{E_{X < T} [(X - T)^2]}{(T - LSL)^2} + \frac{E_{X > T} [(X - T)^2]}{(USL - T)^2} \right]^{-1/2}$$

He proposed this index as a modification of C_{pm} for use when $\mu \neq T$. The quantities

$$E_{X < T} [(X - T)^2] = E [(X - T)^2 | X < T] Pr [X < T]$$

and

$$E_{X > T} [(X - T)^2] = E [(X - T)^2 | X > T] Pr [X > T]$$

are referred to as semivariances. Kotz and Johnson (1993) point out that if $T = (LSL + USL)/2$, then $C_{pm}^+ = C_{pm}$.

Kotz and Johnson (1993) suggest that a natural estimator for C_{pm}^+ is

$$\hat{C}_{pm}^+ = \frac{1}{3} \left[\frac{1}{n} \left\{ \frac{\sum_{X_i < T} (X_i - T)^2}{(T - LSL)^2} + \frac{\sum_{X_i > T} (X_i - T)^2}{(USL - T)^2} \right\} \right]^{-1/2}$$

Note that this index is not defined when either of the specification limits is equal to the target T . Refer to Section 3.5 of Kotz and Johnson (1993) for further details.

The Index C_{jkp}

Johnson *et al.* (1992) introduced a so-called “flexible” process capability index which takes into account possible differences in variability above and below the target T . They defined this index as

$$C_{jkp} = \frac{1}{3\sqrt{2}} \min \left(\frac{\text{USL} - T}{\sqrt{E_{X>T}[(X - T)^2]}}, \frac{T - \text{LSL}}{\sqrt{E_{X<T}[(X - T)^2]}} \right)$$

where $d = (\text{USL} - \text{LSL})/2$.

A natural estimator of this index is

$$\hat{C}_{jkp} = \frac{1}{3\sqrt{2}} \min \left(\frac{\text{USL} - T}{\sqrt{\sum_{X_i>T} (X_i - T)^2/n}}, \frac{T - \text{LSL}}{\sqrt{\sum_{X_i<T} (X_i - T)^2/n}} \right)$$

For further details, refer to Section 4.4 of Kotz and Johnson (1993).

The Indices $C_{pm}(a)$

The class of capability indices $C_{pm}(a)$, indexed by the parameter a ($a > 0$) allows flexibility in choosing between the relative importance of variability and deviation of the mean from the target value T .

The class defined as

$$C_{pm}(a) = (1 - a\zeta^2)C_p$$

where $\zeta = (\mu - T)/\sigma$. The motivation for this definition is that if $|\zeta|$ is small, then

$$C_{pm} \approx (1 - \frac{1}{2}\zeta^2)C_p$$

A natural estimator of $C_{pm}(a)$ is

$$\frac{d}{3s} \hat{C}_{pm}(a) = \left\{ 1 - a \left(\frac{\bar{X} - T}{s} \right)^2 \right\}$$

where $d = (\text{USL} - \text{LSL})/2$. You can specify the value of a with the CPMA= option in the PROC CAPABILITY statement. By default, $a = 0.5$.

This index is not recommended for situation in which the target T is not equal to the midpoint of the specification limits.

For additional details, refer to Section 3.7 of Kotz and Johnson (1993).

Part 1. The CAPABILITY Procedure

The Index $C_{p(5.15)}$

Johnson *et al.* (1992) suggest the class of process capability indices defined as

$$C_{p(\theta)} = \frac{USL - LSL}{\theta\sigma}$$

where θ is chosen so that the proportion of conforming items is robust with respect to the shape of the process distribution. In particular, Kotz and Johnson (1993) recommend use of

$$C_{p(5.15)} = \frac{USL - LSL}{5.15\sigma}$$

which is estimated as

$$\hat{C}_{p(5.15)} = \frac{USL - LSL}{5.15s}$$

For details, refer to Section 4.3.2 of Kotz and Johnson (1993).

The Index $C_{pk(5.15)}$

Similarly, Kotz and Johnson (1993) recommend use of the robust capability index

$$C_{pk(5.15)} = \frac{d - |\mu - (USL + LSL)/2|}{2.575\sigma}$$

where $d = (USL - LSL)/2$. This index is estimated as

$$\hat{C}_{pk(5.15)} = \frac{d - |\bar{X} - (USL + LSL)/2|}{2.575s}$$

For details, refer to Section 4.3.2 of Kotz and Johnson (1993).

The Index C_{pmk}

Pearn *et al.* (1992) proposed the index C_{pmk}

$$C_{pmk} = \frac{(USL - LSL)/2 - |\mu - m|}{3\sqrt{\sigma^2 + (\mu - T)^2}}$$

where $m = (LCL + UCL)/2$. A natural estimator for C_{pmk} is

$$\hat{C}_{pmk} = \frac{(USL - LSL)/2 - |\bar{X} - m|}{3\sqrt{(\frac{n-1}{n})s^2 + (\bar{X} - T)^2}}$$

where $m = (USL + LSL)/2$.

For further details, refer to Section 3.6 of Kotz and Johnson (1993).

Wright's Index C_s

Wright (1995) defines the capability index

$$C_s = \frac{\min(\text{USL} - \mu, \mu - \text{LSL})}{3\sqrt{\sigma^2 + (\mu - T)^2 + \mu_3/\sigma}}$$

where $\mu_3 = E(X - \mu)^3$.

A natural estimator of C_s is

$$\hat{C}_s = \frac{(\text{USL} - \text{LSL})/2 - |\bar{X} - m|}{3\sqrt{\left(\frac{n-1}{n}\right) s^2 + (\bar{X} - T)^2 + |c_4 s^2 b_3|}}$$

where c_4 is an unbiasing constant for the sample standard deviation, and b_3 is a measure of skewness. Wright (1995) shows that C_s compares favorably with C_{pmk} even when skewness is not present, and he advocates the use of C_s for monitoring near-normal processes when loss of capability typically leads to asymmetry.

Missing Values

If a variable for which statistics are calculated has a missing value, that value is ignored in the calculation of statistics, and the missing values are tabulated separately. A missing value for one such variable does not affect the treatment of other variables in the same observation.

If the WEIGHT variable has a missing value, the observation is excluded from the analysis*. If the FREQ variable has a missing value, the observation is excluded from the analysis. If a variable in a BY or ID statement has a missing value, the procedure treats it as it would treat any other value of a BY or ID variable.

*In Release 6.12 and earlier releases, the observation was included in the analysis, and the missing WEIGHT variable value was taken to be zero.

ODS Tables

This section describes the ODS tables produced by the CAPABILITY procedure.

The following table summarizes the ODS tables that you can request with options in the PROC CAPABILITY statement.

Table 1.21. ODS Tables Produced with the PROC CAPABILITY Statement

Table Name	Description	Option
BasicIntervals	confidence intervals for mean, standard deviation, variance	CIBASIC
BasicMeasures	measures of location and variability	default
ExtremeObs	extreme observations	default
ExtremeValues	extreme values	NEXTRVAL=
Frequencies	frequencies	FREQ
LocationCounts	counts used for sign test and signed rank test	LOCCOUNTS
MissingValues	missing values	default
Modes	modes	MODES
Moments	sample moments	default
Quantiles	quantiles	default
RobustScale	robust measures of scale	ROBUSTSCALE
TestsForLocation	tests for location	default
TestsForNormality	tests for normality	NORMALTEST
TrimmedMeans	trimmed means	TRIMMED=
WinsorizedMeans	Winsorized means	WINSORIZED=

The following table summarizes the ODS tables related to capability indices that you can request with options in the PROC CAPABILITY statement when you provide specification limits with a SPEC statement or with a SPEC= data set.

Table 1.22. ODS Tables Related to Specification Limits

Table Name	Description	Option
CIProbExSpecs	confidence limits for probabilities of exceeding specifications	CIPROBEX in PROC CAPABILITY statement
Indices	standard capability indices	default
SpecialIndices	specialized capability indices	SPECIALINDICES in PROC CAPABILITY statement
Specifications	percents outside specification limits based on empirical	default

The following table summarizes the ODS tables related to fitted distributions that you can request with options in the HISTOGRAM statement.

Table 1.23. ODS Tables Produced with the HISTOGRAM Statement

Table Name	Description	Option
Bins	histogram bins	MIDPERCENTS sub-option with any distribution option, such as NORMAL(MIDPERCENTS)
FitIndices	capability indices computed from fitted distribution	INDICES sub-option with any distribution option, such as LOGNORMAL(INDICES)
FitQuantiles	quantiles of fitted distribution	any distribution option such as NORMAL
GoodnessOfFit	goodness-of-fit tests for fitted distribution	any distribution option such as NORMAL
ParameterEstimates	parameter estimates for fitted distribution	any distribution option such as NORMAL
Specifications	percents outside specification limits based on empirical and fitted distributions	any distribution option such as NORMAL

The following table summarizes the ODS tables that you can request with options in the INTERVALS statement.

Table 1.24. ODS Tables Produced with the INTERVALS Statement

Table Name	Description	Option
Intervals1	prediction interval for future observations	METHODS=1
Intervals2	prediction interval for mean	METHODS=2
Intervals3	tolerance interval for proportion of population	METHODS=3
Intervals4	confidence limits for mean	METHODS=4
Intervals5	prediction interval for standard deviation	METHODS=5
Intervals6	confidence limits for standard deviation	METHODS=6

Examples

This section provides a more advanced example of the PROC CAPABILITY statement.

Example 1.1. Reading Specification Limits

See CAPSPEC2
in the SAS/QC
Sample Library

You can specify specification limits either in the SPEC statement or in a SPEC= data set. In “Computing Capability Indices” on page 11, limits were specified in a SPEC statement. This example illustrates how to create a SPEC= data set to read specification limits with the SPEC= option in the PROC CAPABILITY statement.

Consider the drink can data presented on page 9. Suppose, in addition to the fluid weight of each drink can, the weight of the can itself is stored in a variable named CWEIGHT, and both variables are saved in a data set called CAN2. A partial listing of CAN2 follows:

Output 1.1.1. The Data Set CAN2

Obs	weight	cweight
1	12.07	1.07
2	12.02	0.86
3	12.00	1.06
.	.	.
.	.	.
.	.	.
100	11.98	1.07

The following data step creates a data set named LIMITS containing specification limits for the fluid weight and the can weight. LIMITS has 4 variables (_VAR_, _LSL_, _USL_, and _TARGET_) and 2 observations. The first observation contains the specification limit information for the variable WEIGHT, and the second contains the specification limit information for the variable CWEIGHT.

```
data limits;
  length _var_ $8;
  _var_   = 'weight';
  _lsl_   = 11.95;
  _target_ = 12;
  _usl_   = 12.05;
  output;
  _var_   = 'cweight';
  _lsl_   = 0.90;
  _target_ = 1;
  _usl_   = 1.10;
  output;
run;
```

The following statements read the specification information from the LIMITS data set into the CAPABILITY procedure using the SPEC= option. These statements print summary statistics, capability indices, and specification limit information for

WEIGHT and CWEIGHT. Figure 1.1 (page 10) and Figure 1.2 (page 11) display the output for WEIGHT. Output 1.1.2 displays the output for CWEIGHT.

```

title 'Process Capability Analysis of Drink Can Data';
proc capability data=can2 specs=limits;
  var weight cweight;
run;

```

Output 1.1.2. Printed Output for Variable CWEIGHT

Process Capability Analysis of Drink Can Data			
The CAPABILITY Procedure			
Variable: cweight (Can Weight (ounces))			
Moments			
N	100	Sum Weights	100
Mean	1.004	Sum Observations	100.4
Std Deviation	0.06330941	Variance	0.00400808
Skewness	-0.074821	Kurtosis	-0.5433858
Uncorrected SS	101.1984	Corrected SS	0.3968
Coeff Variation	6.30571767	Std Error Mean	0.00633094
Basic Statistical Measures			
Location		Variability	
Mean	1.004000	Std Deviation	0.06331
Median	1.000000	Variance	0.00401
Mode	1.040000	Range	0.29000
		Interquartile Range	0.08500
NOTE: The mode displayed is the smallest of 2 modes with a count of 8.			
Tests for Location: Mu0=0			
Test	-Statistic-	-----p Value-----	
Student's t	t 158.5862	Pr > t	<.0001
Sign	M 50	Pr >= M	<.0001
Signed Rank	S 2525	Pr >= S	<.0001
Tests for Normality			
Test	--Statistic--	-----p Value-----	
Shapiro-Wilk	W 0.987310	Pr < W	0.459
Kolmogorov-Smirnov	D 0.061410	Pr > D	>0.150
Cramer-von Mises	W-Sq 0.048175	Pr > W-Sq	>0.250
Anderson-Darling	A-Sq 0.361939	Pr > A-Sq	>0.250

Output 1.1.3. Printed Output for Variable CWEIGHT (cont.)

```

Process Capability Analysis of Drink Can Data

The CAPABILITY Procedure
Variable: cweight (Can Weight (ounces))

Moments

N              100      Sum Weights              100
Mean           1.004      Sum Observations      100.4
Std Deviation  0.06330941  Variance              0.00400808
Skewness       -0.074821   Kurtosis              -0.5433858
Uncorrected SS 101.1984    Corrected SS          0.3968
Coeff Variation 6.30571767  Std Error Mean        0.00633094

Basic Statistical Measures

Location                      Variability
Mean      1.004000    Std Deviation      0.06331
Median    1.000000    Variance           0.00401
Mode      1.040000    Range              0.29000
                                Interquartile Range 0.08500

NOTE: The mode displayed is the smallest of 2 modes with a count of 8.

Tests for Location: Mu0=0

Test          -Statistic-      -----p Value-----
Student's t   t  158.5862    Pr > |t|    <.0001
Sign          M      50    Pr >= |M|   <.0001
Signed Rank   S     2525    Pr >= |S|   <.0001

Tests for Normality

Test          --Statistic---      -----p Value-----
Shapiro-Wilk  W     0.987310    Pr < W      0.459
Kolmogorov-Smirnov D     0.061410    Pr > D      >0.150
Cramer-von Mises W-Sq  0.048175    Pr > W-Sq   >0.250
Anderson-Darling A-Sq  0.361939    Pr > A-Sq   >0.250
    
```

Example 1.2. Enhancing Reference Lines

See CAPSPEC1
in the SAS/QC
Sample Library

A telecommunications company manufactures amplifiers to be used in telephones. Each amplifier is designed to boost the input signal by 5 decibels (dB). Since it is difficult to make every amplifier's boosting power exactly 5 decibels, the company decides that amplifiers that boost the input signal between 4 and 6 decibels are acceptable. Therefore, the target value is 5 decibels, and the lower and upper specification limits are 4 and 6 decibels, respectively. The following data set contains the boosting powers of a sample of 75 amplifiers:

```

data amps;
  label decibels = 'Amplification in Decibels (dB)';
  input decibels @@;
datalines;
4.54 4.87 4.66 4.90 4.68 5.22 4.43 5.14 3.07 4.22
5.09 3.41 5.75 5.16 3.96 5.37 5.70 4.11 4.83 4.51
4.57 4.16 5.73 3.64 5.48 4.95 4.57 4.46 4.75 5.38
5.19 4.35 4.98 4.87 3.53 4.46 4.57 4.69 5.27 4.67
5.03 4.50 5.35 4.55 4.05 6.63 5.32 5.24 5.73 5.08
5.07 5.42 5.05 5.70 4.79 4.34 5.06 4.64 4.82 3.24
4.79 4.46 3.84 5.05 5.46 4.64 6.13 4.31 4.81 4.98
4.95 5.57 4.11 4.15 5.95
;

```

The SPEC statement provides several options to control the appearance of reference lines for the specification limits and the target value. The following statements use the data set AMPS to create a histogram that demonstrates some of these options:

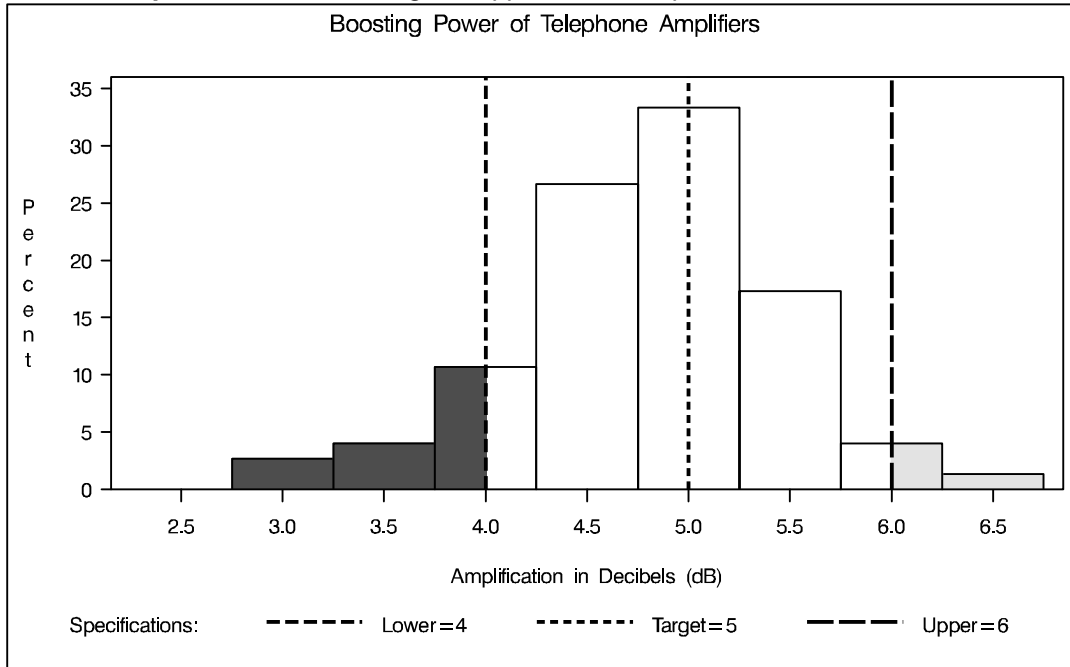
```

title 'Boosting Power of Telephone Amplifiers';
proc capability data=amps;
  spec target = 5    lsl = 4        usl = 6
    ltarget = 2    lls1 = 3        lus1 = 4
    wtarget = 3    wls1 = 3        wus1 = 3
          cleft = red   cright = yellow
          pleft = m5x45 pright = x1;
  histogram decibels / cbarline=black;
run;

```

The resulting histogram is shown in Output 1.2.1. The LTARGET=, LLSL=, and LUSL= options control the line type of the reference lines for the target, lower specification limit, and upper specification limit, respectively. Likewise, the WTARGET=, WLSL=, and WUSL= options control the line widths. The CLEFT= and PLEFT= options control the color and pattern type used to fill the area to the left of the lower specification limit. Similarly, the CRIGHT= and PRIGHT= options control the color and pattern type used to fill the area to the right of the upper specification limit.

Output 1.2.1. Controlling the Appearance of Specification Limits



Example 1.3. Displaying a Confidence Interval for Cpk

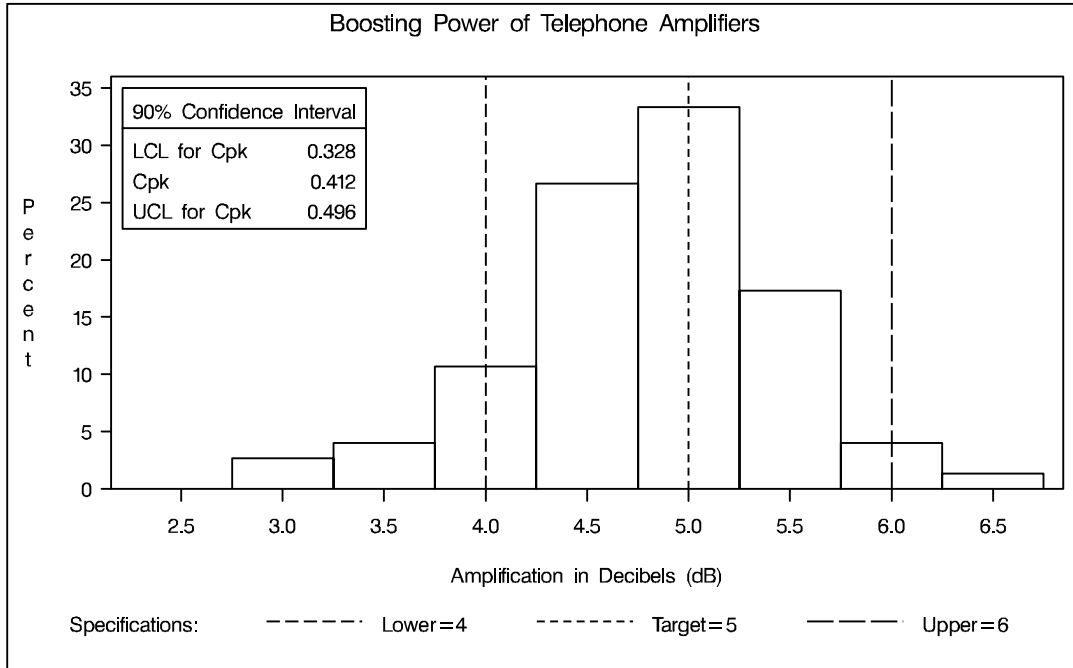
See CAPSPEC3
in the SAS/QC
Sample Library

In this example, the capability index C_{pk} is computed for the amplification data in AMPS. To examine the accuracy of this estimate, the following statements calculate a 90% confidence interval for C_{pk} , then display the interval on a histogram (shown in Output 1.3.1) using the INSET statement:

```
proc capability data=amps alpha=0.10;
  spec target = 5 lsl = 4 usl = 6
    ltarget = 2 llsl = 3 lusl = 4;
  histogram decibels;
  inset cplcll cpk cpkucl / header = '90% Confidence Interval'
    format = 6.3;
run;
```

The ALPHA= option in the PROC CAPABILITY statement controls the level of the confidence interval. In this case, the 90% confidence interval on C_{pk} is wide (from 0.328 to 0.496), indicating that the process may need adjustments in order to improve process variability. Confidence limits for capability indices can be displayed using the INSET statement (as shown in Output 1.3.1) or saved in an output data set using the OUTPUT statement. For formulas and details about capability indices, see “Specialized Capability Indices” on page 50. For more information about the INSET statement, see Chapter 5, “INSET Statement” starting on page 191.

Output 1.3.1. Confidence Interval on C_{pk}



The following statements can be used to produce a table of process capability indices including the index C_{pk} :

```
ods select indices;
proc capability data=amps alpha=0.10;
  spec target = 5 lsl = 4 usl = 6
  ltarget = 2 llsl = 3 lusl = 4;
  var decibels;
run;
```

Output 1.3.2. Process Capability Indices

Boosting Power of Telephone Amplifiers			
The CAPABILITY Procedure			
Variable: decibels (Amplification in Decibels (dB))			
Process Capability Indices			
Index	Value	90% Confidence Limits	
Cp	0.508962	0.439538	0.576922
CPL	0.411920	0.326620	0.495136
CPU	0.606004	0.501261	0.708127
Cpk	0.411920	0.327599	0.496241
Cpm	0.488674	0.425292	0.556732

Part 1. The CAPABILITY Procedure