SAS/STAT® 14.1 User’s Guide
The STDIZE Procedure
Overview: STDIZE Procedure

The STDIZE procedure standardizes one or more numeric variables in a SAS data set by subtracting a location measure and dividing by a scale measure. A variety of location and scale measures are provided, including estimates that are resistant to outliers and clustering. Some of the well-known standardization methods such as mean, median, standard deviation, range, Huber’s estimate, Tukey’s biweight estimate, and Andrew’s wave estimate are available in the STDIZE procedure.
In addition, you can multiply each standardized value by a constant and add a constant. Thus, the final output value is

\[ \text{result} = \text{add} + \text{multiply} \times \frac{\text{original} - \text{location}}{\text{scale}} \]

where

- \( \text{result} \) = final output value
- \( \text{add} \) = constant to add (ADD= option)
- \( \text{multiply} \) = constant to multiply by (MULT= option)
- \( \text{original} \) = original input value
- \( \text{location} \) = location measure
- \( \text{scale} \) = scale measure

PROC STDIZE can also find quantiles in one pass of the data, a capability that is especially useful for very large data sets. With such data sets, the UNIVARIATE procedure might have high or excessive memory or time requirements.

---

**Getting Started: STDIZE Procedure**

The following example demonstrates how you can use the STDIZE procedure to obtain location and scale measures of your data.

In the following hypothetical data set, a random sample of grade twelve students is selected from a number of coeducational schools. Each school is classified as one of two types: Urban or Rural. There are 40 observations.

The variables are id (student identification), Type (type of school attended: ‘urban’=urban area and ‘rural’=rural area), and total (total assessment scores in History, Geometry, and Chemistry).

The following DATA step creates the SAS data set TotalScores.

```sas
data TotalScores;
  title 'High School Scores Data';
  input id Type $ total @@;
datalines;
  1 rural 135 2 rural 125 3 rural 223 4 rural 224 5 rural 133
  6 rural 253 7 rural 144 8 rural 193 9 rural 152 10 rural 178
  11 rural 120 12 rural 180 13 rural 154 14 rural 184 15 rural 187
  16 rural 111 17 rural 190 18 rural 128 19 rural 110 20 rural 217
  21 urban 192 22 urban 186 23 urban 64 24 urban 159 25 urban 133
  26 urban 163 27 urban 130 28 urban 163 29 urban 189 30 urban 144
  31 urban 154 32 urban 198 33 urban 150 34 urban 151 35 urban 152
  36 urban 151 37 urban 127 38 urban 167 39 urban 170 40 urban 123
;```
Suppose you now want to standardize the total scores in different types of schools prior to any further analysis. Before standardizing the total scores, you can use the box plot from PROC BOXPLOT to summarize the total scores for both types of schools.

```sas
ods graphics on;
proc boxplot data=TotalScores;
   plot total*Type / boxstyle=schematic noserifs;
run;
```

The PLOT statement in the PROC BOXPLOT statement creates the schematic plots (without the serifs) when you specify `boxstyle=schematic noserifs`. Figure 106.1 displays a box plot for each type of school.

**Figure 106.1** Schematic Plots from PROC BOXPLOT
Inspection reveals that one urban score is a low outlier. Also, if you compare the lengths of two box plots, there seems to be twice as much dispersion for the rural scores as for the urban scores.

The following PROC UNIVARIATE statement reports the information about the extreme values of the Score variable for each type of school:

```plaintext
c proc univariate data=TotalScores;
  var total;
  by Type;
run;
```

Figure 106.2 displays the table from PROC UNIVARIATE for the lowest and highest five total scores for urban schools. The outlier (Obs = 23), marked in Figure 106.2 by the symbol ‘0’, has a score of 64.

**Figure 106.2** Table for Extreme Observations When Type=urban

<table>
<thead>
<tr>
<th>High School Scores Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>The UNIVARIATE Procedure</td>
</tr>
<tr>
<td>Variable: total</td>
</tr>
</tbody>
</table>

| Type=urban
<table>
<thead>
<tr>
<th>Extreme Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lowest</td>
</tr>
<tr>
<td>Value</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>64</td>
</tr>
<tr>
<td>123</td>
</tr>
<tr>
<td>127</td>
</tr>
<tr>
<td>130</td>
</tr>
<tr>
<td>133</td>
</tr>
</tbody>
</table>

The following PROC STDIZE procedure requests the METHOD=STD option for computing the location and scale measures:

```plaintext
c proc stdize data=totalscores method=std pstat;
  title2 'METHOD=STD';
  var total;
  by Type;
run;
```

Figure 106.3 displays the table of location and scale measures from the PROC STDIZE statement. PROC STDIZE uses the sample mean as the location measure and the sample standard deviation as the scale measure for standardizing. The PSTAT option displays a table containing these two measures.
The ratio of the scale of rural scores to the scale of urban scores is approximately 1.4 (41.96/30.07). This ratio is smaller than the dispersion ratio observed in the previous schematic plots.

The STDIZE procedure provides several location and scale measures that are resistant to outliers. The following statements invoke three different standardization methods and display the tables for the location and scale measures:

```sas
proc stdize data=totalscores method=mad pstat;
title2 'METHOD=MAD';
  var total;
  by Type;
run;

proc stdize data=totalscores method=iqr pstat;
title2 'METHOD=IQR';
  var total;
  by Type;
run;

proc stdize data=totalscores method=abw(4) pstat;
title2 'METHOD=ABW(4)';
  var total;
  by Type;
run;
```
Figure 106.4 displays the table of location and scale measures when the standardization method is median absolute deviation (MAD). The location measure is the median, and the scale measure is the median absolute deviation from the median. The ratio of the scale of rural scores to the scale of urban scores is approximately 2.06 (32.0/15.5) and is close to the dispersion ratio observed in the previous schematic plots.

**Figure 106.4 Location and Scale Measures Table When METHOD=MAD**

<table>
<thead>
<tr>
<th>Name</th>
<th>Location</th>
<th>Scale</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>total</td>
<td>166.000000</td>
<td>32.000000</td>
<td>20</td>
</tr>
</tbody>
</table>

Figure 106.5 displays the table of location and scale measures when the standardization method is IQR. The location measure is the median, and the scale measure is the interquartile range. The ratio of the scale of rural scores to the scale of urban scores is approximately 2.03 (61/30) and is, in fact, the dispersion ratio observed in the previous schematic plots.

**Figure 106.5 Location and Scale Measures Table When METHOD=IQR**

<table>
<thead>
<tr>
<th>Name</th>
<th>Location</th>
<th>Scale</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>total</td>
<td>153.000000</td>
<td>15.500000</td>
<td>20</td>
</tr>
</tbody>
</table>
Figure 106.6 displays the table of location and scale measures when the standardization method is ABW, for which the location measure is the biweight one-step M-estimate, and the scale measure is the biweight A-estimate. Note that the initial estimate for ABW is MAD. The following steps help to decide the value of the tuning constant:

1. For rural scores, the location estimate for MAD is 166.0, and the scale estimate for MAD is 32.0. The maximum of the rural scores is 253 (not shown), and the minimum is 110 (not shown). Thus, the tuning constant needs to be 3 so that it does not reject any observation that has a score between 110 to 253.

2. For urban scores, the location estimate for MAD is 153.0, and the scale estimate for MAD is 15.5. The maximum of the rural scores is 198, and the minimum (also an outlier) is 64. Thus, the tuning constant needs to be 4 so that it rejects the outlier (64) but includes the maximum (198) as an normal observation.

3. The maximum of the tuning constants, obtained in steps 1 and 2, is 4.

See Goodall (1983, Chapter 11) for details about the tuning constant. The ratio of the scale of rural scores to the scale of urban scores is approximately 2.06 (32.0/15.5). It is also close to the dispersion ratio observed in the previous schematic plots.
The preceding analysis shows that METHOD=MAD, METHOD=IQR, and METHOD=ABW all provide better dispersion ratios than METHOD=STD does.

You can recompute the standard deviation after deleting the outlier from the original data set for comparison. The following statements create a data set NoOutlier that excludes the outlier from the TotalScores data set and invoke PROC STDIZE with METHOD=STD.

```plaintext
data NoOutlier;
  set totalscores;
  if (total = 64) then delete;
run;

proc stdize data=NoOutlier method=std pstat;
  title2 'After Removing Outlier, METHOD=STD';
  var total;
  by Type;
run;
```

Figure 106.7 displays the location and scale measures after deleting the outlier. The lack of resistance of the standard deviation to outliers is clearly illustrated: if you delete the outlier, the sample standard deviation of urban scores changes from 30.07 to 22.09. The new ratio of the scale of rural scores to the scale of urban scores is approximately 1.90 (41.96/22.09).
Syntax: STDIZE Procedure

The following statements are available in the STDIZE procedure:

```
PROC STDIZE <options> ;
   BY variables ;
   FREQ variable ;
   LOCATION variables ;
   SCALE variables ;
   VAR variables ;
   WEIGHT variable ;
```

The PROC STDIZE statement is required. The BY, LOCATION, FREQ, VAR, SCALE, and WEIGHT statements are described in alphabetical order following the PROC STDIZE statement.

PROC STDIZE Statement

```
PROC STDIZE <options> ;
```

The PROC STDIZE statement invokes the STDIZE procedure. You can specify the following options in the PROC STDIZE statement. Table 106.1 summarizes the options available in the PROC STDIZE statement.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify standardization methods</td>
<td></td>
</tr>
<tr>
<td>METHOD=</td>
<td>Specifies the name of the standardization method</td>
</tr>
<tr>
<td>INITIAL=</td>
<td>Specifies the method for computing initial estimates for the A estimates</td>
</tr>
</tbody>
</table>
Table 106.1  continued  

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unstandardize variables</strong></td>
<td></td>
</tr>
<tr>
<td>UNSTD</td>
<td>Unstandardizes variables when you also specify the METHOD=IN option</td>
</tr>
<tr>
<td><strong>Process missing values</strong></td>
<td></td>
</tr>
<tr>
<td>NOMISS</td>
<td>Omits observations with any missing values from computation</td>
</tr>
<tr>
<td>MISSING=</td>
<td>Specifies the method or a numeric value for replacing missing values</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replaces missing data with zero in the standardized data</td>
</tr>
<tr>
<td>REPOONLY</td>
<td>Replaces missing data with the location measure (does not standardize the data)</td>
</tr>
<tr>
<td><strong>Specify data set details</strong></td>
<td></td>
</tr>
<tr>
<td>DATA=</td>
<td>Specifies the input data set</td>
</tr>
<tr>
<td>KEEPLEN</td>
<td>Specifies that output variables inherit the length of the analysis variable</td>
</tr>
<tr>
<td>OUT=</td>
<td>Specifies the output data set</td>
</tr>
<tr>
<td>OPREFIX=</td>
<td>Specifies that original variables appear in the OUT= data set</td>
</tr>
<tr>
<td>SPREFIX=</td>
<td>Specifies a prefix for the standardized variable names</td>
</tr>
<tr>
<td>OUTSTAT=</td>
<td>Specifies the output statistic data set</td>
</tr>
<tr>
<td><strong>Specify computational settings</strong></td>
<td></td>
</tr>
<tr>
<td>VARDEF=</td>
<td>Specifies the variances divisor</td>
</tr>
<tr>
<td>NMARKERS=</td>
<td>Specifies the number of markers when you also specify</td>
</tr>
<tr>
<td>PCTLMTD=</td>
<td>Specifies that original variables appear in the OUT= data set</td>
</tr>
<tr>
<td>PCTLMTD=</td>
<td>Specifies the method used to estimate percentiles</td>
</tr>
<tr>
<td>PCTLPTS=</td>
<td>Writes observations containing percentiles to the data set specified in the OUTSTAT= option</td>
</tr>
<tr>
<td><strong>Normalize scale estimators</strong></td>
<td></td>
</tr>
<tr>
<td>NORM</td>
<td>Normalizes the scale estimator to be consistent for the standard deviation of a normal distribution</td>
</tr>
<tr>
<td>SNORM</td>
<td>Normalizes the scale estimator to have an expectation of approximately 1 for a standard normal distribution</td>
</tr>
<tr>
<td><strong>Specify output</strong></td>
<td></td>
</tr>
<tr>
<td>PSTAT</td>
<td>Displays the location and scale measures</td>
</tr>
</tbody>
</table>

These options and their abbreviations are described (in alphabetical order) in the remainder of this section.
ADD=c
specifies a constant, c, to add to each value after standardizing and multiplying by the value you specify in the MULT= option. The default value is 0.

DATA=SAS-data-set
specifies the input data set to be standardized. If you omit the DATA= option, the most recently created data set is used.

FUZZ=c
specifies the relative fuzz factor. The default value is 1E–14. For the OUT= data set, the score is computed as follows:

\[ \text{if } |\text{result}| < m \times c \text{ then } \text{result} = 0 \]

where \( m \) is the constant specified in the MULT= option, or 1 if MULT= option is not specified.

For the OUTSTAT= data set and the location and scale table, the scale and location values are computed as follows:

\[ \text{if } \text{scale} < |\text{location}| \times c \text{ then } \text{scale} = 0 \]

Otherwise,

\[ \text{if } |\text{location}| < m \times c \text{ then } \text{location} = 0 \]

INITIAL=method
specifies the method for computing initial estimates for the A estimates (ABW, AAVE, and AHUBER). You cannot specify the following methods for initial estimates: INITIAL=ABW, INITIAL=AHUBER, INITIAL=AAVE, and INITIAL=IN. The default is INITIAL=MAD.

KEEPLEN
specifies that the standardized variables inherit the lengths of the analysis variables that PROC STDIZE uses to derive them. PROC STDIZE stores numbers in double-precision without this option.

Caution: The KEEPLEN option causes the standardized variables to permanently lose numeric precision by truncating or rounding the values. However, the precision of the output variables will match that of the input.

METHOD=name
specifies the name of the method for computing location and scale measures. Valid values for name are as follows: MEAN, MEDIAN, SUM, EUCLEN, USTD, STD, RANGE, MIDRANGE, MAXABS, IQR, MAD, ABW, AHUBER, AAVE, AGK, SPACING, L, and IN.

For details about these methods, see the descriptions in the section “Standardization Methods” on page 8709. The default is METHOD=STD.

MISSING=method | value
specifies the method (or a numeric value) for replacing missing values. If you omit the MISSING= option, the REPLACE option replaces missing values with the location measure given by the METHOD= option. Specify the MISSING= option when you want to replace missing values with a different value. You can specify any name that is valid in the METHOD= option except the name IN. The corresponding location measure is used to replace missing values.
If a numeric value is given, the value replaces missing values after standardizing the data. However, you can specify the REPONLY option with the MISSING= option to suppress standardization for cases in which you want only to replace missing values.

**MULT=c**

specifies a constant, \( c \), by which to multiply each value after standardizing. The default value is 1.

**NMARKERS=n**

specifies the number of markers used when you specify the one-pass algorithm (PCTLMTD=ONEPASS). The value \( n \) must be greater than or equal to 5. The default value is 105.

**NOMISS**

omits observations with missing values for any of the analyzed variables from calculation of the location and scale measures. If you omit the NOMISS option, all nonmissing values are used.

**NORM**

normalizes the scale estimator to be consistent for the standard deviation of a normal distribution when you specify the option METHOD=AGK, METHOD=IQR, METHOD=MAD, or METHOD=SPACING.

**OPREFIX< =o-prefix >**

specifies that the original variables should appear in the OUT= data set. You can optionally specify an equal sign and a prefix. For example, if OPREFIX=Original, then the names of the variables are OriginalVAR1, OriginalVAR2, and so on, where VAR1 and VAR2 are the original variable names. The value of OPREFIX= must be different from the value of SPREFIX=. If you specify OPREFIX, without an equal sign and a prefix, then the default prefix is null and you must specify SPREFIX=

**OUT=SAS-data-set**

specifies the name of the SAS data set created by PROC STDIZE. By default, the output data set is a copy of the DATA= data set except that the analyzed variables have been standardized. Analyzed variables are those specified in the VAR statement or, if there is no VAR statement, all numeric variables not listed in any other statement. However, you can use the OPREFIX option to request that both the original and standardized variables be included in the output data set. You can change variable names by specifying prefixes with the OPREFIX= and SPREFIX= options. See the section “Output Data Sets” on page 8714 for more information.

If you want to create a SAS data set in a permanent library, you must specify a two-level name. For more information about permanent libraries and SAS data sets, see SAS Language Reference: Concepts.

If you omit the OUT= option, PROC STDIZE creates an output data set named according to the DATA= convention.

**OUTSTAT=SAS-data-set**

specifies the name of the SAS data set containing the location and scale measures and other computed statistics. See the section “Output Data Sets” on page 8714 for more information.

**PCTLDEF=percentiles**

specifies which of five definitions is used to calculate percentiles when you specify the option PCTLMTD=ORD_STAT. By default, PCTLDEF=5. Note that the option PCTLMTD=ONEPASS implies PCTLDEF=5. See the section “Computational Methods for the PCTLDEF= Option” on page 8712 for details about percentile definition.
PROC STDIZE Statement  ➤  8705

You cannot use PCTLDEF= when you compute weighted quantiles.

PCTLMTD=ORD_STAT | ONEPASS | P2
specifies the method used to estimate percentiles. Specify the PCTLMTD=ORD_STAT option to compute the percentiles by the order statistics method.

The PCTLMTD=ONEPASS option modifies an algorithm invented by Jain and Chlamtac (1985). See the section “Computing Quantiles” on page 8712 for more details about this algorithm.

PCTLPTS=\text{n}
writes percentiles to the OUTSTAT= data set. Values of \text{n} can be any decimal number between 0 and 100, inclusive.

A requested percentile is identified by the \textit{TYPE} variable in the OUTSTAT= data set with a value of \textit{Pn}. For example, suppose you specify the option PCTLPTS=10, 30. The corresponding observations in the OUTSTAT= data set that contain the 10th and the 30th percentiles would then have values \textit{TYPE} = P10 and \textit{TYPE} = P30, respectively.

PSTAT
displays the location and scale measures.

REPLACE
replaces missing data with the value 0 in the standardized data (this value corresponds to the location measure before standardizing). To replace missing data by other values, see the preceding description of the MISSING= option. You cannot specify both the REPLACE and REPONLY options.

REONLY
replaces missing data only; PROC STDIZE does not standardize the data. Missing values are replaced with the location measure unless you also specify the MISSING=value option, in which case missing values are replaced with value. You cannot specify both the REPLACE and REPONLY options.

SNORM
normalizes the scale estimator to have an expectation of approximately 1 for a standard normal distribution when you specify the METHOD=SPACING option.

SPREFIX=\text{s-prefix}
specifies a prefix for the standardized variables. For example, if SPREFIX=Std, then the names of the standardized variables are StdVAR1, StdVAR2, and so on, where VAR1 and VAR2 are the original variable names. The value of SPREFIX= must be different from the value of OPREFIX=. The default prefix is null. If you omit the SPREFIX option, the standardized variables still appear in the OUT= data set by default and the variable names remain the same. If you want to have the variable names changed, you need to specify a prefix with SPREFIX=\text{s-prefix}.

UNSTD
UNSTDIZE
unstandardizes variables when you specify the METHOD=IN(ds) option. The location and scale measures, along with constants for addition and multiplication that the unstandardization is based on, are identified by the \textit{TYPE} variable in the ds data set.

The ds data set must have a \textit{TYPE} variable and contain the following two observations: a \textit{TYPE} = ‘LOCATION’ observation and a \textit{TYPE} = ‘SCALE’ observation. The variable \textit{TYPE} can also contain the optional observations, ‘ADD’ and ‘MULT’; if these observations are not found in the ds
data set, the constants specified in the ADD= and MULT= options (or their default values) are used for unstandardization.

See the section “OUTSTAT= Data Set” on page 8714 for details about the statistics that each value of _TYPE_ represents. The formula used for unstandardization is as follows: If the final output value from the previous standardization is calculated as

\[ \text{result} = \text{add} + \text{multiply} \times \frac{\text{original} - \text{location}}{\text{scale}} \]

The unstandardized variable is computed as

\[ \text{original} = \text{scale} \times \frac{\text{result} - \text{add}}{\text{multiply}} + \text{location} \]

**VARDEF=DF | N | WDF | WEIGHT | WGT**

specifies the divisor to be used in the calculation of variances. By default, VARDEF=DF. The values and associated divisors are as follows.

<table>
<thead>
<tr>
<th>Value</th>
<th>Divisor</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>DF</td>
<td>Degrees of freedom</td>
<td>( n - 1 )</td>
</tr>
<tr>
<td>N</td>
<td>Number of observations</td>
<td>( n )</td>
</tr>
<tr>
<td>WDF</td>
<td>Sum of weights minus 1</td>
<td>( (\sum_i w_i) - 1 )</td>
</tr>
<tr>
<td>WEIGHT</td>
<td>Sum of weights</td>
<td>( \sum_i w_i )</td>
</tr>
</tbody>
</table>

**BY Statement**

```
BY variables;
```

You can specify a BY statement with PROC STDIZE to obtain separate analyses of observations in groups that are defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in order of the BY variables. If you specify more than one BY statement, only the last one specified is used.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the NOTSORTED or DESCENDING option in the BY statement for the STDIZE procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

When you specify the option METHOD=IN(ds), the following rules are applied to BY-group processing:

- If the ds data set does not contain any of the BY variables, the entire DATA= data set is standardized by the location and scale measures (along with the constants for addition and multiplication) in the ds data set.
FREQ Statement

FREQ variable;

If one variable in the input data set represents the frequency of occurrence for other values in the observation, specify the variable name in a FREQ statement. PROC STDIZE treats the data set as if each observation appeared \( n \) times, where \( n \) is the value of the FREQ variable for the observation. Nonintegral values of the FREQ variable are truncated to the largest integer less than the FREQ value. If the FREQ variable has a value that is less than 1 or is missing, the observation is not used in the analysis.

NOTRUNCATE
NOTRUNC

specifies that frequency values are not truncated to integers.

The nonintegral values of the FREQ variable can be used for the following standardization methods: AGK, ABW, AHUBER, AWA VE, EUCLEN, IQR, L, MAD, MEAN, MEDIAN, SPACING, STD, SUM, and USTD. The nonintegral frequency values are used for the MAD, MEDIAN, or IQR method only when PCTLMTD=ORD_STAT is specified. If PCTLMTD=ONEPASS is specified, the NOTRUNCATE option is ignored.

LOCATION Statement

LOCATION variables;

The LOCATION statement specifies a list of numeric variables that contain location measures in the input data set specified by the METHOD=IN option.
**SCALE Statement**

```
SCALE variables ;
```

The SCALE statement specifies the list of numeric variables that contain scale measures in the input data set specified by the METHOD=IN option.

**VAR Statement**

```
VAR variable ;
```

The VAR statement lists numeric variables to be standardized. If you omit the VAR statement, all numeric variables not listed in the BY, FREQ, and WEIGHT statements are used.

**WEIGHT Statement**

```
WEIGHT variable ;
```

The WEIGHT statement specifies a numeric variable in the input data set with values that are used to weight each observation. Only one variable can be specified.

The WEIGHT variable values can be nonintegers. An observation is used in the analysis only if the value of the WEIGHT variable is greater than zero.

The WEIGHT variable applies only when you specify the following standardization methods: AGK, EUCLEN, IQR, L, MAD, MEAN, MEDIAN, STD, SUM, and USTD. Weights are used for the METHOD=MAD, MEDIAN, or IQR only when PCTLMTD=ORD_STAT is specified; if PCTLMTD=ONEPASS is specified, the WEIGHT statement is ignored.

PROC STDIZE uses the value of the WEIGHT variable to calculate the sample mean and sample variances:

\[
\bar{x}_w = \frac{\sum_i w_i x_i}{\sum_i w_i} \quad \text{(sample mean)}
\]

\[
us_{w}^2 = \frac{\sum_i w_i x_i^2}{d} \quad \text{(uncorrected sample variances)}
\]

\[
s_{w}^2 = \frac{\sum_i w_i (x_i - \bar{x}_w)^2}{d} \quad \text{(sample variances)}
\]

where \(w_i\) is the weight value of the \(i\)th observation, \(x_i\) is the value of the \(i\)th observation, and \(d\) is the divisor controlled by the VARDEF= option (see the VARDEF= option for details).

The following weighted statistics are defined accordingly:

- **MEAN** the weighted mean, \(\bar{x}_w\)
- **SUM** the weighted sum, \(\sum_i w_i x_i\)
- **USTD** the weighted uncorrected standard deviation, \(\sqrt{us_{w}^2}\)
STD the weighted standard deviation, $\sqrt{\frac{\sum w_i x_i^2}{\sum w_i}}$

EUCLEN the weighted Euclidean length, computed as the square root of the weighted uncorrected sum of squares:

$$\sqrt{\sum_{i} w_i x_i^2}$$

MEDIAN the weighted median. See the section “Weighted Percentiles” on page 8713 for the formulas and descriptions.

MAD the weighted median absolute deviation from the weighted median. See the section “Weighted Percentiles” on page 8713 for the formulas and descriptions.

IQR the weighted median, 25th percentile, and the 75th percentile. See the section “Weighted Percentiles” on page 8713 for the formulas and descriptions.

AGK the AGK estimate. This estimate is documented further in the ACECLUS procedure as the METHOD=COUNT option. See the discussion of the WEIGHT statement in Chapter 24, “The ACECLUS Procedure,” for information about how the WEIGHT variable is applied to the AGK estimate.

L the $L_p$ estimate. This estimate is documented further in the FASTCLUS procedure as the LEAST= option. See the discussion of the WEIGHT statement in Chapter 38, “The FASTCLUS Procedure,” for information about how the WEIGHT variable is used to compute weighted cluster means. The number of clusters is always 1.

---

**Details: STDIZE Procedure**

### Standardization Methods

The following table lists standardization methods and their corresponding location and scale measures available with the METHOD= option.

<table>
<thead>
<tr>
<th>Method</th>
<th>Location</th>
<th>Scale Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEAN</td>
<td>Mean</td>
<td>1</td>
</tr>
<tr>
<td>MEDIAN</td>
<td>Median</td>
<td>1</td>
</tr>
<tr>
<td>SUM</td>
<td>0</td>
<td>Sum</td>
</tr>
<tr>
<td>EUCLEN</td>
<td>0</td>
<td>Euclidean length</td>
</tr>
<tr>
<td>USTD</td>
<td>0</td>
<td>Standard deviation about origin</td>
</tr>
<tr>
<td>STD</td>
<td>Mean</td>
<td>Standard deviation</td>
</tr>
<tr>
<td>RANGE</td>
<td>Minimum</td>
<td>Range</td>
</tr>
<tr>
<td>MIDRANGE</td>
<td>Midrange</td>
<td>Range/2</td>
</tr>
<tr>
<td>MAXABS</td>
<td>0</td>
<td>Maximum absolute value</td>
</tr>
</tbody>
</table>
### Table 106.2 (continued)

<table>
<thead>
<tr>
<th>Method</th>
<th>Location</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>IQR</td>
<td>Median</td>
<td>Interquartile range</td>
</tr>
<tr>
<td>MAD</td>
<td>Median</td>
<td>Median absolute deviation from median</td>
</tr>
<tr>
<td>ABW(c)</td>
<td>Biweight one-step M-estimate</td>
<td>Biweight A-estimate</td>
</tr>
<tr>
<td>AHUBER(c)</td>
<td>Huber one-step M-estimate</td>
<td>Huber A-estimate</td>
</tr>
<tr>
<td>AWA VE(c)</td>
<td>Wave one-step M-estimate</td>
<td>Wave A-estimate</td>
</tr>
<tr>
<td>AGK(p)</td>
<td>Mean</td>
<td>AGK estimate (ACECLUS)</td>
</tr>
<tr>
<td>SPACING(p)</td>
<td>Mid-minimum spacing</td>
<td>Minimum spacing</td>
</tr>
<tr>
<td>L(p)</td>
<td>L(p)</td>
<td>L(p)</td>
</tr>
<tr>
<td>IN(ds)</td>
<td>Read from data set</td>
<td>Read from data set</td>
</tr>
</tbody>
</table>

For METHOD=ABW(c), METHOD=AHUBER(c), or METHOD=AWAVE(c), \( c \) is a positive numeric tuning constant.

For METHOD=AGK(p), \( p \) is a numeric constant that gives the proportion of pairs to be included in the estimation of the within-cluster variances.

For METHOD=SPACING(p), \( p \) is a numeric constant that gives the proportion of data to be contained in the spacing.

For METHOD=L(p), \( p \) is a numeric constant greater than or equal to 1 that specifies the power to which differences are to be raised in computing an L\((p)\) or Minkowski metric.

For METHOD=IN(ds), ds is the name of a SAS data set that meets either of the following two conditions:

- The data set contains a _TYPE_ variable. The observation that contains the location measure corresponds to the value _TYPE_ = ‘LOCATION’, and the observation that contains the scale measure corresponds to the value _TYPE_ = ‘SCALE’. You can also use a data set created by the OUTSTAT= option from another PROC STDIZE statement as the ds data set. See the section “Output Data Sets” on page 8714 for the contents of the OUTSTAT= data set.

- The data set contains the location and scale variables specified by the LOCATION and SCALE statements.

PROC STDIZE reads in the location and scale variables in the ds data set by first looking for the _TYPE_ variable in the ds data set. If it finds this variable, PROC STDIZE continues to search for all variables specified in the VAR statement. If it does not find the _TYPE_ variable, PROC STDIZE searches for the location variables specified in the LOCATION statement and the scale variables specified in the SCALE statement.

The variable _TYPE_ can also contain the optional observations, ‘ADD’ and ‘MULT’. If these observations are found in the ds data set, the values in the observation of _TYPE_ = ‘MULT’ are the multiplication constants, and the values in the observation of _TYPE_ = ‘ADD’ are the addition constants; otherwise, the constants specified in the ADD= and MULT= options (or their default values) are used.

For robust estimators, see Goodall (1983) and Iglewicz (1983). The MAD method has the highest breakdown point (50%), but it is somewhat inefficient. The ABW, AHUBER, and AWA VE methods provide a good compromise between breakdown and efficiency. The L\((p)\) location estimates are increasingly robust as \( p \)
drops from 2 (which corresponds to least squares, or mean estimation) to 1 (which corresponds to least absolute value, or median estimation). However, the L(p) scale estimates are not robust.

The SPACING method is robust to both outliers and clustering (Janssen et al. 1995) and is, therefore, a good choice for cluster analysis or nonparametric density estimation. The mid-minimum spacing method estimates the mode for small p. The AGK method is also robust to clustering and more efficient than the SPACING method, but it is not as robust to outliers and takes longer to compute. If you expect g clusters, the argument to METHOD=SPACING or METHOD=AGK should be \( \frac{1}{g} \) or less. The AGK method is less biased than the SPACING method for small samples. As a general guide, it is reasonable to use AGK for samples of size 100 or less and SPACING for samples of size 1,000 or more, with the treatment of intermediate sample sizes depending on the available computer resources.

### Computation of the Statistics

Formulas for statistics of METHOD=MEAN, METHOD=MEDIAN, METHOD=SUM, METHOD=USTD, METHOD=STD, METHOD=RANGE, and METHOD=IQR are given in the chapter “Elementary Statistics Procedures” (Base SAS Procedures Guide).

Note that the computations of median and upper and lower quartiles depend on the PCTLMTD= option.

The other statistics listed in Table 106.2, except for METHOD=IN, are described as follows:

- **EUCLEN**
  
  Euclidean length. 
  
  \[ \sqrt{\sum_{i=1}^{n} x_i^2}, \text{ where } x_i \text{ is the } i\text{th observation and } n \text{ is the total number of observations in the sample.} \]

- **L(p)**
  
  Minkowski metric. This metric is documented as the LEAST=p option in the PROC FASTCLUS statement of the FASTCLUS procedure (see Chapter 38, “The FASTCLUS Procedure”).

  If you specify METHOD=L(p) in the PROC STDIZE statement, your results are similar to those obtained from PROC FASTCLUS if you specify the LEAST=p option with MAXCLUS=1 (and use the default values of the MAXITER= option). The difference between the two types of calculations concerns the maximum number of iterations. In PROC STDIZE, it is a criterion for convergence on all variables; in PROC FASTCLUS, it is a criterion for convergence on a single variable.

  The location and scale measures for L(p) are output to the OUTSEED= data set in PROC FASTCLUS.

- **MIDRANGE**
  
  \[ \frac{\text{maximum} + \text{minimum}}{2} \]

- **ABW(c)**
  

- **AHUBER(c)**
  

- **AWAVE(c)**
  

- **AGK(p)**
  
  The noniterative univariate form of the estimator described by Art, Gnanadesikan, and Kettenring (1982).
The AGK estimate is documented in the section on the METHOD= option in the PROC ACECLUS statement of the ACECLUS procedure (also see the section “Background” on page 884 in Chapter 24, “The ACECLUS Procedure”). Specifying METHOD=AGK(p) in the PROC STDIZE statement is the same as specifying METHOD=COUNT and P=p in the PROC ACECLUS statement.

SPACING(p) The absolute difference between two data values. The minimum spacing for a proportion p is the minimum absolute difference between two data values that contain a proportion p of the data between them. The mid-minimum spacing is the mean of these two data values.

### Computing Quantiles

PROC STDIZE offers two methods for computing quantiles: the one-pass approach and the order-statistics approach (like that used in the UNIVARIATE procedure).

The one-pass approach used in PROC STDIZE modifies the P^2 algorithm for histograms proposed by Jain and Chlamtac (1985). The primary difference comes from the movement of markers. The one-pass method allows a marker to move to the right (or left) by more than one position (to the largest possible integer) as long as it does not result in two markers being in the same position. The modification is necessary in order to incorporate the FREQ variable.

You might obtain inaccurate results if you use the one-pass approach to estimate quantiles beyond the quartiles (that is, when you estimate quantiles < P25 or quantiles > P75). A large sample size (10,000 or more) is often required if the tail quantiles (quantiles ≤ P10 or quantiles ≥ P90) are requested. Note that, for variables with highly skewed or heavy-tailed distributions, tail quantile estimates might be inaccurate.

The order-statistics approach for estimating quantiles is faster than the one-pass method but requires that the entire data set be stored in memory. The accuracy in estimating the quantiles is comparable for both methods when the requested percentiles are between the lower and upper quartiles. The default is PCTLMTD=ORD_STAT if enough memory is available; otherwise, PCTLMTD=ONEPASS.

### Computational Methods for the PCTLDEF= Option

You can specify one of five methods for computing quantile statistics when you use the order-statistics approach (PCTLMTD=ORD_STAT); otherwise, the PCTLDEF=5 method is used when you use the one-pass approach (PCTLMTD=ONEPASS).

**Percentile Definitions**

Let n be the number of nonmissing values for a variable, and let \( x_1, x_2, \ldots, x_n \) represent the ordered values of the variable. For the \( t \)th percentile, let \( p = t / 100 \). In the following definitions numbered 1, 2, 3, and 5, let

\[
np = j + g
\]

where \( j \) is the integer part and \( g \) is the fractional part of \( np \). For definition 4, let

\[
(n + 1) p = j + g
\]

Given the preceding definitions, the \( t \)th percentile, \( y \), is defined as follows:
Computing Quantiles

PCTLDEF=1 weighted average at \( x_{np} \)

\[ y = (1-g)x_j + gx_{j+1} \]

where \( x_0 \) is taken to be \( x_1 \)

PCTLDEF=2 observation numbered closest to \( np \)

\[ y = x_i \]

where \( i \) is the integer part of \( np + 1/2 \) if \( g \neq 1/2 \). If \( g = 1/2 \), then

\[ y = x_j \text{ if } j \text{ is even}, \quad y = x_{j+1} \text{ if } j \text{ is odd} \]

PCTLDEF=3 empirical distribution function

\[ y = x_j \text{ if } g = 0 \]
\[ y = x_{j+1} \text{ if } g > 0 \]

PCTLDEF=4 weighted average aimed at \( x_{p(n+1)} \)

\[ y = (1-g)x_j + gx_{j+1} \]

where \( x_{n+1} \) is taken to be \( x_n \)

PCTLDEF=5 empirical distribution function with averaging

\[ y = (x_j + x_{j+1})/2 \text{ if } g = 0 \]
\[ y = x_{j+1} \text{ if } g > 0 \]

Weighted Percentiles

When you specify a WEIGHT statement, or specify the NOTRUNCATE option in a FREQ 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.

For PCTLMTD= ORD_STAT, the PCTLDEF= option is not applicable when a WEIGHT statement is used, or when a NOTRUNCATE option is specified in a FREQ statement. 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.

For PCTLMTD= ONEPASS, the quantile computation currently does not use any weights.
Constant Data

Constant variables are not standardized. The scale value is set to missing when the data are constant.

Missing Values

Missing values can be replaced by the location measure or by any specified constant (see the REPLACE option and the MISSING= option). You can also suppress standardization if you want only to replace missing values (see the REPONLY option).

If you specify the NOMISS option, PROC STDIZE omits observations with any missing values in the analyzed variables from computation of the location and scale measures.

Output Data Sets

OUT= Data Set

By default, the output data set is a copy of the DATA= data set except that the analyzed variables have been standardized. Analyzed variables are those specified in the VAR statement or, if there is no VAR statement, all numeric variables not listed in any other statement. However, you can use the OPREFIX option to request that both the original and standardized variables be included in the output data set. You can change variable names by specifying prefixes with the OPREFIX=o-prefix and SPREFIX=s-prefix options, but keep in mind that the two prefixes must be different. See OPREFIX and SPREFIX for more information.

OUTSTAT= Data Set

The new data set contains the following variables:

- the BY variables, if any
- _TYPE_, a character variable
- the analyzed variables

Each observation in the new data set contains a type of statistic as indicated by the _TYPE_ variable. The values of the _TYPE_ variable are as follows:

- LOCATION location measure of each variable
- SCALE scale measure of each variable
- ADD constant specified in the ADD= option. This value is the same for each variable.
- MULT constant specified in the MULT= option. This value is the same for each variable.
- N total number of nonmissing positive frequencies of each variable
Displayed Output

If you specify the PSTAT option, PROC STDIZE displays the following statistics for each variable:

- the name of the variable, Name
- the location estimate, Location
- the scale estimate, Scale
- the norm estimate, Norm (when you specify the NORM option with METHOD=AGK, METHOD=IQR, METHOD=MAD, or METHOD=SPACING or when you specify the SNORM option with METHOD=SPACING)
- sum of nonmissing positive frequencies, N
- sum of nonmissing positive weights if the WEIGHT statement is specified, Sum of Weights

ODS Table Names

PROC STDIZE assigns a name to the single table it creates. You can use this name to reference the table when using the Output Delivery System (ODS) to select a table or create an output data set. This name is listed in Table 106.3. For more information about ODS, see Chapter 20, “Using the Output Delivery System.”
Example: STDIZE Procedure

Example 106.1: Standardization of Variables in Cluster Analysis

To illustrate the effect of standardization in cluster analysis, this example uses the Fish data set described in the “Getting Started” section of Chapter 38, “The FASTCLUS Procedure.” The numbers are measurements taken on 159 fish caught from the same lake (Laengelmaevesi) near Tampere in Finland (Puranen 1917). The fish data set is available from the Sashelp library.

The species (bream, parkki, pike, perch, roach, smelt, and whitefish), weight, three different length measurements (measured from the nose of the fish to the beginning of its tail, the notch of its tail, and the end of its tail), height, and width of each fish are recorded.

A couple of new variables are created in the Fish data set: Weight3 and logLengthRatio. The weight of a fish indicates its size—a heavier pike tends to be larger than a lighter pike. To get a one-dimensional measure of the size of a fish, take the cubic root of the weight (Weight3). The variables Height, Width, Length1, Length2, and Length3 are rescaled in order to adjust for dimensionality. The logLengthRatio variable measures the tail length.

Because the new variables Weight3–logLengthRatio depend on the variable Weight, observations with missing values for Weight are not added to the data set. Consequently, there are 157 observations in the SAS data set Sashelp.Fish.

Before you perform a cluster analysis on coordinate data, it is necessary to consider scaling or transforming the variables since variables with large variances tend to have a larger effect on the resulting clusters than variables with small variances do.

This example uses three different approaches to standardize or transform the data prior to the cluster analysis. The first approach uses several standardization methods provided in the STDIZE procedure. However, since standardization is not always appropriate prior to the clustering (see Milligan and Cooper (1987) for a Monte Carlo study on various methods of variable standardization), the second approach performs the cluster analysis with no standardization. The third approach invokes the ACECLUS procedure to transform the data into a within-cluster covariance matrix.

The clustering is performed by the FASTCLUS procedure to find seven clusters. Note that the variables Length2 and Length3 are eliminated from this analysis since they both are significantly and highly correlated with the variable Length1. The correlation coefficients are 0.9958 and 0.9604, respectively. An output data set is created, and the FREQ procedure is invoked to compare the clusters with the species classification.
The DATA step is as follows:

``` SAS
    title 'Fish Measurement Data';

    data Fish;
        set sashelp.fish;
        if Weight <= 0 or Weight = . then delete;
        Weight3 = Weight ** (1/3);
        Height = Height / Weight3;
        Width = Width / Weight3;
        Length1 = Length1 / Weight3;
        Length2 = Length2 / Weight3;
        Length3 = Length3 / Weight3;
        LogLengthRatio = log(Length3 / Length1);
    run;
```

The following macro, Std, standardizes the Fish data. The macro reads a single argument, mtd, which selects the METHOD= specification to be used in PROC STDIZE.

``` SAS
/*--- macro for standardization ---*/
%macro Std(mtd);
    title2 "Data are Standardized by PROC STDIZE with METHOD= &mtd";
    proc stdize data=fish out=sdzout method=&mtd;
        var Length1 logLengthRatio Height Width Weight3;
    run;
%mend Std;
```

The following macro, FastFreq, includes a PROC FASTCLUS statement for performing cluster analysis and a PROC FREQ statement for crosstabulating species with the cluster membership information that is derived from the previous PROC FASTCLUS statement. The macro reads a single argument, ds, which selects the input data set to be used in PROC FASTCLUS.

``` SAS
/*--- macro for clustering and crosstabulating ---*/
/*--- cluster membership with species ---*/
%macro FastFreq(ds);
    proc fastclus data=&ds out=clust maxclusters=7 maxiter=100 noprint;
        var Length1 logLengthRatio Height Width Weight3;
    run;

    proc freq data=clust;
        tables species*cluster;
    run;
%mend FastFreq;
```

The following analysis (labeled ‘Approach 1’) includes 18 different methods of standardization followed by clustering. Since there is a large amount of output from this approach, only results from METHOD=STD, METHOD=RANGE, METHOD=AGK(0.14), and METHOD=SPACING(0.14) are shown. The following statements produce Output 106.1.1 through Output 106.1.4.
Chapter 106: The STDIZE Procedure

/* Approach 1: data are standardized by PROC STDIZE */

%Std(MEAN);
%FastFreq(sdzout);

%Std(MEDIAN);
%FastFreq(sdzout);

%Std(SUM);
%FastFreq(sdzout);

%Std(EUCLEN);
%FastFreq(sdzout);

%Std(USTD);
%FastFreq(sdzout);

%Std(STD);
%FastFreq(sdzout);

%Std(RANGE);
%FastFreq(sdzout);

%Std(MIDRANGE);
%FastFreq(sdzout);

%Std(MAXABS);
%FastFreq(sdzout);

%Std(IQR);
%FastFreq(sdzout);

%Std(MAD);
%FastFreq(sdzout);

%Std(AGK(.14));
%FastFreq(sdzout);

%Std(SPACING(.14));
%FastFreq(sdzout);

%Std(ABW(5));
%FastFreq(sdzout);

%Std(AWAVE(5));
%FastFreq(sdzout);

%Std(L(1));
%FastFreq(sdzout);

%Std(L(1.5));
%FastFreq(sdzout);

%Std(L(2));
%FastFreq(sdzout);
### Output 106.1.1 Data Are Standardized by PROC STDIZE with METHOD=STD

#### Fish Measurement Data
Data are Standardized by PROC STDIZE with METHOD= STD

#### The FREQ Procedure

<table>
<thead>
<tr>
<th>Frequency</th>
<th>Percent</th>
<th>Table of Species by CLUSTER</th>
<th>CLUSTER(Cluster)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Species</td>
<td>1</td>
</tr>
<tr>
<td>Row Pct</td>
<td>Col Pct</td>
<td>Bream</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Parkki</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Perch</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Pike</td>
<td>17.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Roach</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Smelt</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Whitefish</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total</td>
<td>17.00</td>
</tr>
</tbody>
</table>


### Output 106.1.2

Data Are Standardized by PROC STDIZE with METHOD=RANGE

#### Fish Measurement Data

Data are Standardized by PROC STDIZE with METHOD= RANGE

#### The FREQ Procedure

<table>
<thead>
<tr>
<th>Frequency Percent</th>
<th>Table of Species by CLUSTER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row Pct</td>
<td>CLUSTER(Cluster)</td>
</tr>
<tr>
<td>Col Pct</td>
<td>Species</td>
</tr>
<tr>
<td>Bream</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td>Parkki</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td>Perch</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td>Pike</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>10.83</td>
</tr>
<tr>
<td></td>
<td>100.00</td>
</tr>
<tr>
<td>Roach</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td>Smelt</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td>Whitefish</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
</tr>
<tr>
<td>Total</td>
<td>17</td>
</tr>
</tbody>
</table>
Output 106.1.3 Data Are Standardized by PROC STDIZE with METHOD=AGK(0.14)

Fish Measurement Data
Data are Standardized by PROC STDIZE with METHOD= AGK(.14)

The FREQ Procedure

<table>
<thead>
<tr>
<th>Frequency Percent</th>
<th>Table of Species by CLUSTER</th>
<th>CLUSTER(Cluster)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row Pct</td>
<td>Species</td>
<td>1 2 3 4 5 6 7</td>
</tr>
<tr>
<td>Col Pct</td>
<td>Bream</td>
<td>0 0 34 0 0 0 0</td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 21.66 0.00 0.00 0.00 21.66</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 100.00 0.00 0.00 0.00 0.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 100.00 0.00 0.00 0.00 0.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Parkki</td>
<td>11 0 0 0 0 0 11</td>
</tr>
<tr>
<td></td>
<td>7.01 0.00 0.00 0.00 0.00 0.00 0.00 7.01</td>
<td></td>
</tr>
<tr>
<td></td>
<td>100.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>100.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Perch</td>
<td>0 0 3 0 20 33 56</td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 1.91 0.00 12.74 21.02 35.67</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 5.36 0.00 35.71 58.93 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 13.04 0.00 86.96 94.29 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pike</td>
<td>0 0 0 17 0 0 17</td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 0.00 10.83 0.00 0.00 10.83</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 0.00 100.00 0.00 0.00 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 0.00 100.00 0.00 0.00 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Roach</td>
<td>0 0 17 0 0 2 19</td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 10.83 0.00 0.00 1.27 12.10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 89.47 0.00 0.00 10.53 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 73.91 0.00 0.00 5.71 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Smelt</td>
<td>0 14 0 0 0 0 14</td>
</tr>
<tr>
<td></td>
<td>0.00 8.92 0.00 0.00 0.00 0.00 0.00 8.92</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 100.00 0.00 0.00 0.00 0.00 0.00 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Whitefish</td>
<td>0 0 3 0 3 0 6</td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 1.91 0.00 1.91 0.00 3.82</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 50.00 0.00 50.00 0.00 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00 0.00 0.00 13.04 0.00 13.04 0.00 100.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>11 14 34 23 17 23 35 157</td>
</tr>
<tr>
<td></td>
<td>7.01 8.92 21.66 14.65 10.83 14.65 22.29 100.00</td>
<td></td>
</tr>
</tbody>
</table>
The following analysis (labeled ‘Approach 2’) applies the cluster analysis directly to the original data. The following statements produce Output 106.1.5.

/* Approach 2: data are untransformed */

title2 'Data are Untransformed';
%FastFreq(fish);
Output 106.1.5 Untransformed Data

Fish Measurement Data
Data are Untransformed

The FREQ Procedure

<table>
<thead>
<tr>
<th>Frequency</th>
<th>Species</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percent</td>
<td>Bream</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Row Pct</td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>21</td>
<td>0</td>
<td>34</td>
</tr>
<tr>
<td>Col Pct</td>
<td>8.28</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>13.38</td>
<td>21.66</td>
<td></td>
</tr>
<tr>
<td></td>
<td>38.24</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>61.76</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>44.83</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>47.73</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parkki</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.27</td>
<td>1.91</td>
<td>0.00</td>
<td>0.00</td>
<td>3.82</td>
<td>0.00</td>
<td>0.00</td>
<td>7.01</td>
<td></td>
</tr>
<tr>
<td></td>
<td>18.18</td>
<td>27.27</td>
<td>0.00</td>
<td>0.00</td>
<td>54.55</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6.90</td>
<td>18.75</td>
<td>0.00</td>
<td>0.00</td>
<td>15.38</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Perch</td>
<td>8</td>
<td>9</td>
<td>0</td>
<td>1</td>
<td>20</td>
<td>0</td>
<td>18</td>
<td>56</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.10</td>
<td>5.73</td>
<td>0.00</td>
<td>0.64</td>
<td>12.74</td>
<td>0.00</td>
<td>11.46</td>
<td>35.67</td>
<td></td>
</tr>
<tr>
<td></td>
<td>14.29</td>
<td>16.07</td>
<td>0.00</td>
<td>1.79</td>
<td>35.71</td>
<td>0.00</td>
<td>32.14</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>27.59</td>
<td>56.25</td>
<td>0.00</td>
<td>6.67</td>
<td>51.28</td>
<td>0.00</td>
<td>40.91</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pike</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>6.37</td>
<td>0.00</td>
<td>0.64</td>
<td>2.55</td>
<td>1.27</td>
<td>10.83</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>58.82</td>
<td>0.00</td>
<td>5.88</td>
<td>23.53</td>
<td>11.76</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
<td>0.00</td>
<td>2.56</td>
<td>100.00</td>
<td>4.55</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Roach</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>0</td>
<td>0</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.91</td>
<td>2.55</td>
<td>0.00</td>
<td>0.00</td>
<td>7.64</td>
<td>0.00</td>
<td>0.00</td>
<td>12.10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15.79</td>
<td>21.05</td>
<td>0.00</td>
<td>0.00</td>
<td>63.16</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.34</td>
<td>25.00</td>
<td>0.00</td>
<td>0.00</td>
<td>30.77</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smelt</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>8.92</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>8.92</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>93.33</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Whitefish</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.91</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>1.91</td>
<td>3.82</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>50.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.34</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>6.82</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>29</td>
<td>16</td>
<td>10</td>
<td>15</td>
<td>39</td>
<td>4</td>
<td>44</td>
<td>157</td>
<td></td>
</tr>
<tr>
<td></td>
<td>18.47</td>
<td>10.19</td>
<td>6.37</td>
<td>9.55</td>
<td>24.84</td>
<td>2.55</td>
<td>28.03</td>
<td>100.00</td>
<td></td>
</tr>
</tbody>
</table>

The following analysis (labeled ‘Approach 3’) transforms the original data with the ACECLUS procedure and creates a TYPE=ACE output data set that is used as an input data set for the cluster analysis. The following statements produce Output 106.1.6.

/* Approach 3: data are transformed by PROC ACECLUS */

title2 'Data are Transformed by PROC ACECLUS';
proc aceclus data=fish out=ace p=.02 noprint;
    var Length1 logLengthRatio Height Width Weight3;
run;
%FastFreq(ace);
### Output 106.1.6 Data Are Transformed by PROC ACECLUS

#### Fish Measurement Data

Data are Transformed by PROC ACECLUS

#### The FREQ Procedure

<table>
<thead>
<tr>
<th>Frequency Percent</th>
<th>Table of Species by CLUSTER CLUSTER(Cluster)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Species</td>
</tr>
<tr>
<td></td>
<td>Bream</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Parkki</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Perch</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pike</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Roach</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Smelt</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Whitefish</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 106.4 displays a table summarizing each classification results. In this table, the first column represents the standardization method, the second column represents the number of clusters that the seven species are classified into, and the third column represents the total number of observations that are misclassified.

**Table 106.4** Summary of Clustering Results

<table>
<thead>
<tr>
<th>Method of Standardization</th>
<th>Number of Clusters</th>
<th>Misclassification</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEAN</td>
<td>5</td>
<td>71</td>
</tr>
<tr>
<td>MEDIAN</td>
<td>5</td>
<td>71</td>
</tr>
<tr>
<td>SUM</td>
<td>6</td>
<td>51</td>
</tr>
<tr>
<td>EUCLLEN</td>
<td>6</td>
<td>45</td>
</tr>
<tr>
<td>USTD</td>
<td>6</td>
<td>45</td>
</tr>
<tr>
<td>STD</td>
<td>5</td>
<td>33</td>
</tr>
<tr>
<td>RANGE</td>
<td>7</td>
<td>32</td>
</tr>
<tr>
<td>MIDRANGE</td>
<td>7</td>
<td>32</td>
</tr>
<tr>
<td>MAXABS</td>
<td>7</td>
<td>26</td>
</tr>
<tr>
<td>IQR</td>
<td>5</td>
<td>28</td>
</tr>
<tr>
<td>MAD</td>
<td>4</td>
<td>35</td>
</tr>
<tr>
<td>ABW(5)</td>
<td>6</td>
<td>34</td>
</tr>
<tr>
<td>AWAVERE(5)</td>
<td>6</td>
<td>29</td>
</tr>
<tr>
<td>AGK(0.14)</td>
<td>7</td>
<td>28</td>
</tr>
<tr>
<td>SPACING(0.14)</td>
<td>7</td>
<td>25</td>
</tr>
<tr>
<td>L(1)</td>
<td>6</td>
<td>41</td>
</tr>
<tr>
<td>L(1.5)</td>
<td>5</td>
<td>33</td>
</tr>
<tr>
<td>L(2)</td>
<td>5</td>
<td>33</td>
</tr>
<tr>
<td>untransformed</td>
<td>5</td>
<td>71</td>
</tr>
<tr>
<td>PROC ACECLUS</td>
<td>5</td>
<td>71</td>
</tr>
</tbody>
</table>

Consider the results displayed in Output 106.1.1. In that analysis, the method of standardization is STD, and the number of clusters and the number of misclassifications are computed as shown in Table 106.5.

**Table 106.5** Computations of Numbers of Clusters and Misclassification When Standardization Method Is STD

<table>
<thead>
<tr>
<th>Species</th>
<th>Cluster Number</th>
<th>Misclassification in Each Species</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bream</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>Parkki</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Perch</td>
<td>7</td>
<td>29</td>
</tr>
<tr>
<td>Pike</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Roach</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>Smelt</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Whitefish</td>
<td>7</td>
<td>3</td>
</tr>
</tbody>
</table>

In Output 106.1.1, the bream species is classified as cluster 6 since all 34 bream are categorized into cluster 6 with no misclassification. A similar pattern is seen with the roach, parkki, pike, and smelt species.
For the whitefish species, two fish are categorized into cluster 2, one fish is categorized into cluster 4, and three fish are categorized into cluster 7. Because the majority of this species is categorized into cluster 7, it is recorded in Table 106.5 as being classified as cluster 7 with 3 misclassifications. A similar pattern is seen with the perch species: it is classified as cluster 7 with 29 misclassifications.

In summary, when the standardization method is STD, seven species of fish are classified into only five clusters and the total number of misclassified observations is 33.

The result of this analysis demonstrates that when variables are standardized by the STDIZE procedure with methods including RANGE, MIDRANGE, MAXABS, AGK(0.14), and SPACING(0.14), the FASTCLUS procedure produces the correct number of clusters and less misclassification than it does when other standardization methods are used. The SPACING method attains the best result, probably because the variables Length1 and Height both exhibit marked groupings (bimodality) in their distributions.

References


Subject Index

AGK estimate
STDIZE procedure, 8711

cluster analysis (STDIZE)
standardizing, 8716

clustering and scaling
STDIZE procedure, example, 8716

Euclidean length
STDIZE procedure, 8711

fractional frequencies
STDIZE procedure, 8707

Minkowski metric
STDIZE procedure, 8711

missing values
STDIZE procedure, 8703, 8705

percentiles
weighted, 8713

quantile computation
STDIZE procedure, 8694, 8712

robust
estimators (STDIZE), 8710

scaling variables
STDIZE procedure, 8716

spacing
STDIZE procedure, 8712

standardizing
cluster analysis (STDIZE), 8716
values (STDIZE), 8693

STDIZE procedure
AGK estimate, 8711
Andrew’s wave estimate, 8711
breakdown point and efficiency, 8710
comparisons of quantile computation, PCTLMTD option, 8712
computational methods, PCTLDEF option, 8712
Euclidean length, 8711
examples, 8694, 8716
final output value, 8694
formulas for statistics, 8711
fractional frequencies, 8707
fuzz factor, 8703
Huber’s estimate, 8711
initial estimates for A estimates, 8703
input data set (METHOD=IN()), 8710
methods resistant to clustering, 8711
methods resistant to outliers, 8697, 8711
Minkowski metric, 8711
missing values, 8703–8705, 8714
normalization, 8704, 8705
one-pass quantile computations, 8712
OUT= data set, 8703, 8714
output data sets, 8704, 8714
output table names, 8715
OUTSTAT= data set, 8714
quantile computation, 8694, 8712
robust estimators, 8710
spacing, 8712
standardization methods, 8693, 8709
standardization with weights, 8708
Tukey’s biweight estimate, 8699, 8711
tuning constant, 8699, 8710
unstandardization, 8705
weights, 8708

weighted percentiles, 8713
Syntax Index

ADD= option
   PROC STDIZE statement, 8703

BY statement
   STDIZE procedure, 8706

DATA= option
   PROC STDIZE statement, 8703

FREQ statement
   STDIZE procedure, 8707
FUZZ= option
   PROC STDIZE statement, 8703

INITIAL= option
   PROC STDIZE statement, 8703

KEEPLEN option
   PROC STDIZE statement, 8703

METHOD= option
   PROC STDIZE statement, 8703
MISSING= option
   PROC STDIZE statement, 8703
MULT= option
   PROC STDIZE statement, 8704

NMARKERS= option
   PROC STDIZE statement, 8704
NOMISS option
   PROC STDIZE statement, 8704
NORM option
   PROC STDIZE statement, 8704
NOTRUNCATE option
   FREQ statement (STDIZE), 8707

OPREFIX option
   PROC STDIZE statement, 8704
OUT= option
   PROC STDIZE statement, 8704
OUTSTAT= option
   PROC STDIZE statement, 8704

PCTLDEF= option
   PROC STDIZE statement, 8704
PCTLMTD option
   PROC STDIZE statement, 8705
PCTLPTS option
   PROC STDIZE statement, 8705
PSTAT option
   PROC STDIZE statement, 8705
REPLACE option
   PROC STDIZE statement, 8705
REPONLY option
   PROC STDIZE statement, 8705
SNORM option
   PROC STDIZE statement, 8705
SPREFIX option
   PROC STDIZE statement, 8705
UNSTD option
   PROC STDIZE statement, 8705

STDIZE procedure
   syntax, 8701
STDIZE procedure, BY statement, 8706
STDIZE procedure, FREQ statement, 8707
   NOTRUNCATE option, 8707
STDIZE procedure, LOCATION statement, 8707
STDIZE procedure, PROC STDIZE statement, 8701
   ADD= option, 8703
   DATA= option, 8703
   FUZZ= option, 8703
   INITIAL= option, 8703
   KEEPLEN, 8703
   METHOD= option, 8703
   MISSING= option, 8703
   MULT= option, 8704
   NMARKERS= option, 8704
   NOMISS option, 8704
   NORM option, 8704
   OPREFIX option, 8704
   OUT= option, 8704
   OUTSTAT= option, 8704
   PCTLDEF= option, 8704
   PCTLMTD option, 8705
   PCTLPTS option, 8705
   PSTAT option, 8705
   REPLACE option, 8705
   REPONLY option, 8705
   SNORM option, 8705
   SПРЕФИКС option, 8705
   UNSTD option, 8705
   VARDEF option, 8706
STDIZE procedure, SCALE statement, 8708
STDIZE procedure, VAR statement, 8708
STDIZE procedure, WGT statement, 8708

PROC STDIZE statement, see STDIZE procedure
   PROC STDIZE statement, 8705
VAR statement
   STDIZE procedure, 8708
VARDEF option
   PROC STDIZE statement, 8706

WGT statement
   STDIZE procedure, 8708