LIST

```
LIST [VARIABLES={ALL**   }]
                {varlist}

  [/CASES=[FROM {1**}] [TO {eof**}] [BY {1**}]]
                {n  }      {n   }      {n  }

  [/FORMAT=[{UNNUMBERED**}]  [{WRAP**}]
            {NUMBERED    }    {SINGLE}

           [WEIGHT]]
```

**Default if subcommand is omitted.

Example:

```
LIST VARIABLES=XVAR AVAR /CASES=10 TO 100 BY 2.
```

Overview

The LIST procedure displays the values of variables for cases in the active file.

Defaults

By default, all user-defined variables are listed for all cases in the active file. The listing for each case uses as many lines as needed.

Tailoring

You can specify a list of variables and request that the listing be numbered. You can also limit the listing to a particular sequence of cases. In addition, you can list values of system variables for each case in the active file and limit each case listing to a single line.

Syntax

- The minimum specification is simply the command keyword.
- All subcommands are optional.
- Subcommands can be specified in any order and are separated by slashes.

Operations

- LIST is a procedure and causes the data to be read.
- LIST uses the dictionary print formats (see Universals: Formats).
- If a value is longer than the format, the decimal portion will be rounded. If that fails, asterisks (*) are printed.
- The LIST display uses the width specified on SET.
- If a long string variable cannot be listed within the page width, it is truncated.
- Values of the variables listed for a case are always separated by at least one blank.
- System-missing values are listed as a period (.).
- For case listings that fit on one line, the column width for each variable is determined by the length of the variable name or the format, whichever is greater. If the variable names do not fit on one line, they are printed vertically.
- If case listings require more than one line, they are wrapped. LIST displays a table illustrating the location of the variables in the listing and prints the name of the first variable in each line at the beginning of the line.
- Each execution of LIST begins at the top of a new page.
- System variables will not be printed unless explicitly requested.

Limitations

- Maximum 1 each of the VARIABLES, CASES, and FORMAT subcommands.

Example

```
DATA LIST FILE='SURVEY.DAT'
/XVAR 1 YVAR 10-15 ZVAR 3-9(2) /3 AVAR 25-30(A).
LIST VARIABLES=XVAR AVAR /CASES=10 TO 100 BY 2.
```

- This example produces a list of every second case starting with Case 10 and stopping at 100 for variables XVAR and AVAR.

VARIABLES Subcommand

By default, all user-defined variables in the active file are listed. The optional VARIABLES subcommand allows you to limit the listing to variables you specify or to specify the default explicitly with keyword ALL.

- Variables named on VARIABLES must already exist.
- Variables are listed in the order they are named.
- If a variable is named more than once it is listed more than once.
- You cannot specify the VARIABLES subcommand without specifications.

ALL *List all user-defined variables.* Variables are listed in the order in which they appear in the active file. This is the default if the VARIABLES subcommand is omitted.

CASES Subcommand

By default, all cases in the active file are listed. Use the CASES subcommand to limit the number and pattern of cases listed.

- If you omit CASES or include it with no specifications, all cases in the file are listed.
- Defaults that you do not change remain in effect.

The following may be specified on CASES:

FROM n *The case number of the first case to be listed.* The default is 1.

TO n *Upper limit on the cases to be listed.* The default is the end of the active file. CASES 100 is interpreted as CASES TO 100.

BY n *Increment used to choose cases for listing.* The default is 1.

FORMAT Subcommand

The default display does not number cases and uses more than one line per case if necessary. Use the optional FORMAT subcommand to change the defaults.

- If you omit the FORMAT subcommand or use it without any specifications, the default display is produced.
- Defaults that you do not change remain in effect.

The following specifications are available for FORMAT:

NUMBERED *Include the sequence number of each case in the listing.* The sequence number is displayed to the left of the values listed. The default is UNNUMBERED.

SINGLE *Limit each case listing to one line.* If the variables requested do not fit on a single line, LIST is not executed. The default is WRAP.

WEIGHT *List the value of the case's weight in the active file.*

MEANS

```
MEANS [TABLES=] {varlist} BY varlist [BY varlist ...] [/varlist...]
                {ALL    }

      [/OPTIONS=option numbers]

      [/STATISTICS={statistic numbers}]
                   {ALL               }
```

Options:

1 Include user-missing values
2 Exclude cases with user-missing dependent values
3 Suppress all labels
5 Suppress group counts
6 Display group sums
7 Suppress group standard deviations
8 Suppress value labels
9 Suppress independent variable names
10 Suppress independent variable values
11 Suppress group means
12 Display group variances

Statistics:

1 One-way analysis of variance
2 Test of linearity

Example:

```
MEANS TABLES=YVAR1 TO YVAR5 BY XVAR
/OPTIONS=9
/STATISTICS=1.
```

Overview

MEANS displays means, standard deviations, and group counts for a dependent variable within groups defined by one or more independent variables. Other SPSS/PC procedures that display univariate statistics are FREQUENCIES and DESCRIPTIVE.

Defaults

By default, MEANS displays means, standard deviations, and number of cases. The default table is labeled with the variable name and label of the dependent and independent variables. Groups are labeled with the variable name, variable label, values, and value labels of the independent variables. Cases that have missing values on any variables that define a table are excluded from the statistics calculated for that table.

Tailoring

Display Format. You can suppress the printing of all variable and value labels, value labels only, names of independent variables, and values of independent variables.

Statistical Display. Statistical display options allow you to display group sums and variances or to suppress group counts, group standard deviations, and group means. A one-way analysis of variance and a test of linearity are also available.

Missing Values. You can include cases with user-missing values on the dependent or independent variables. Alternatively, you can include cases with user-missing values on the independent variable only.

Syntax

- The minimum specification is a TABLES subcommand with a tables list. The actual keyword TABLES may be omitted.
- The minimum tables list specifies a dependent variable, the keyword BY, and an independent variable.
- Subcommands are separated by slashes.

Operations

- MEANS is a procedure and causes the data to be read.
- MEANS displays requested univariate statistics for the population as a whole and for each value of the first independent variable defined for the table in addition to statistics for groups.
- If an independent variable is a long string, only the short-string portion is used to identify groups in the analysis.

- Specifying a string variable as a dependent variable on any tables list stops execution of the MEANS procedure.
- Statistics are displayed with four decimal places where possible.
- The display uses the width defined on the SET command. If the statistics requested cannot fit within the available width, the command is not executed. You can use the OPTIONS subcommand to tailor the display to fit within the defined width.

Limitations

- The number of variables allowed per MEANS command is the same as the system limit.
- Maximum 250 tables can be produced.
- Maximum 30 TABLES subcommands per MEANS command.
- Maximum 1 each of the OPTIONS and STATISTICS subcommands.
- Maximum 6 dimensions (5 BY keywords) per TABLES subcommand.
- Maximum 200 value labels are displayed on any single table.

Example

```
MEANS TABLES=YVAR1 TO YVAR5 BY XVAR
/OPTIONS=9
/STATISTICS=1.
```

- In this example, the TABLES subcommand specifies that YVAR1 through YVAR5 are the dependent variables. XVAR is the independent variable.
- Assuming variables YVAR2, YVAR3, and YVAR4 lie between between YVAR1 and YVAR5 on the active file, five tables are produced: YVAR1 by XVAR; YVAR2 by XVAR; YVAR3 by XVAR, and so on.
- Option 9 suppresses the printing of variable name XVAR.
- Statistic 1 requests one-way analysis of variance tables of YVAR1 through YVAR5 by XVAR.

Example

```
MEA VARA BY VARB BY VARC/VAR1 VAR2 BY VAR3 VAR4 BY VAR5.
```

- This command contains two TABLES subcommands that omit the optional TABLES keyword.
- The first tables list requests one table. Statistics are produced for VARA within groups defined by each coinbination of values of VARB and VARC.
- The second tables list requests four tables: VAR1 by VAR3 by VAR5; VAR1 by VAR4 by VAR5; VAR2 by VAR3 by VAR5; and VAR2 by VAR4 by VAR5.
- This example takes advantage of spelling permitted by three-character truncation of keywords.

TABLES Subcommand

The TABLES subcommand specifies a tables list. The actual keyword TABLES may be omitted.

- The dependent variable is named first and must be numeric. The independent variables follow the BY keyword and can be numeric or string.
- You can specify more than one dependent variable in a tables list.
- Each use of the keyword BY in a tables list adds a dimension to the tables requested and introduces a new order of control among the independent variables.
- You can specify more than one independent variable in each dimension of a tables list.
- A table is built for each dependent variable by each combination of independent variables across dimensions.
- Each combination of values of the independent variables defined for a table defines a group.
- The order in which independent variables are displayed is the same as the order in which they are named. The values of the first independent variable defined

for the table appear in the left-most column of the table and change most slowly in the definition of groups.

- You can use keyword ALL in each dimension to refer to all user-defined variables.
- You can specify multiple TABLES subcommands on a single MEANS command. The slash between the subcommands is required; the keyword is not.

Display Format

By default, MEANS displays the variable name and variable labels of both independent and dependent variables at the beginning of each table. Within the table, groups defined by the independent variables are identified by variable name, values, and value labels. Specify the following on the OPTIONS subcommand to change these defaults:

Option 3 *Suppress all labels.* No variable or value labels are displayed for either the independent or dependent variables.

Option 8 *Suppress value labels.* No value labels are printed for the independent variables.

Option 9 *Suppress independent variable names.*

Option 10 *Suppress independent variable values.*

Statistical Display

By default, MEANS displays means, standard deviations, and number of cases. Statistics are displayed for groups and for the entire population. Use option and statistic numbers on the OPTIONS and STATISTICS subcommands to change these defaults.

Option 5 *Suppress group counts.* The number of cases in each group is not displayed.

Option 6 *Display group sums.*

Option 7 *Suppress group standard deviations.*

Option 11 *Suppress group means.*

Option 12 *Display group variances.*

Statistic 1 *One-way analysis of variance including eta and eta².* The analysis of variance is performed for the first independent variable defined for the table only.

Statistic 2 *Test of linearity.* Produces a one-way analysis of variance in which the between-groups sum of squares is subdivided into linear and nonlinear components. Pearson's r and r^2 are displayed as part of the test of linearity. The analysis of variance is performed for the first independent variable defined for the table only, and the test of linearity is ignored if the independent variable is a string.

ALL *Display all statistics.* Produces the same display as Statistic 2. Specify ALL on the STATISTICS subcommand.

Example

```
MEANS TABLES=INCOME81 BY AGECAT BY SEX
/STATISTICS=1.
```

- This example requests statistics for INCOME81 for groups defined by values of SEX within AGECAT.
- Statistic 1 requests an analysis of variance of INCOME81 by AGECAT.

Missing Values

By default, MEANS excludes cases that have missing values for any variables that define a table. You change the handling of missing values by specifying the following on the OPTIONS subcommand:

Option 1 *Include cases with user-missing values.* Cases with user-missing values on the independent or dependent variable are included.

Option 2 *Exclude cases with user-missing dependent values.* Cases with user-missing values for the dependent variable are excluded. Cases with user-missing values for independent variables are included.

MISSING VALUE

```
MISSING VALUE  {varlist}(value) [[/]varlist ...]
               {ALL     }
```

Example:

```
MISSING VALUE XVAR (8)/ YVAR ZVAR (0)/ AVAR ('      ').
```

Overview

The MISSING VALUE command declares values for numeric and short string variables as missing. These values can then be treated specially in data transformations, statistical calculations, and case selection.

Syntax

- The minimum specification is a single variable followed by a value in parentheses.
- Each variable can have only one user-missing value.
- You can declare the same value as missing for more than one variable by specifying a variable list followed by the value in parentheses.
- You can declare different values as missing for different variables by specifying separate variable lists for each value. The slash between different specifications is optional.
- The missing-value specification must correspond to the variable type (numeric or string).
- You cannot assign missing values to long strings or system variables.
- Variable lists must have either all numeric or all string variables.
- Missing values for short string variables must be enclosed in apostrophes or quotation marks (see Universals: Strings).
- Keyword ALL can be used to refer to all user-defined variables in the active file provided the variables are either all numeric or all string.
- The TO keyword can be used to refer to consecutive variables on the active file.
- The missing-value specification for short string variables must include any leading or trailing blanks.
- More than one MISSING VALUE command can be specified per session.

Operations

- MISSING VALUE is a transformation command and is executed when the data are read for the next procedure.
- If a variable is mentioned more than once on one or more MISSING VALUE commands before a procedure, only the last specification is used.
- Missing-value specifications can be changed between procedures. New declarations replace previous ones.
- A variable list followed by no value specification deletes all missing-value declarations for those variables.
- Missing-value declarations are saved in system files (see SAVE) and portable files (see EXPORT).

Example

```
MISSING VALUE XVAR (8)/ YVAR ZVAR (0)/ AVAR ('      ').
```

- Value 8 is declared missing for numeric variable XVAR.
- Value 0 is missing for numeric variables YVAR and ZVAR.
- Blanks are declared missing for string variable AVAR. AVAR is five columns wide.

Example

```
MIS VAL NAME1 TO NAME7 ('    ')/LIKE1 TO DLIKE7 (0).
```

- Blanks are declared missing for the variables between and including NAME1 and NAME7. All these variables must be string.
- The value 0 is declared missing for the variables between and including LIKE1 and DLIKE7. All variables in this list must be numeric.
- This example takes advantage of spelling permitted by three-character truncation of keywords.

N

```
N n [ESTIMATED]
```

Example:

```
N 100.
```

Overview

The N command limits the number of cases in the active file to the first *n* cases. You can also use N with keyword ESTIMATED to provide SPSS/PC with information about the estimated number of cases in your data file. SPSS/PC uses the estimate to optimize use of memory. In this case, the number of cases processed is *not* limited to the estimated number.

Syntax

- The number of cases must be a positive integer.
- The N command keyword and the integer must be separated by at least one space.
- N can be entered at any point in an SPSS/PC session and can be used more than once.
- The keyword ESTIMATED is specified after the integer estimate of the number of cases.

Operations

- N is a transformation command and is executed when the data are read for the next procedure.
- N limits the number of cases analyzed by all subsequent procedures.
- Without the N command, SPSS/PC processes all cases.
- N controls the building of cases, not the reading of individual data records.
- SPSS/PC stops processing input data when N is reached. If keyword ESTIMATED is used, all input data is processed.
- Any SAMPLE, PROCESS IF, or SELECT IF commands are executed before cases are counted toward the limit specified on N.
- If N is specified more than once in an SPSS/PC job, each N command reduces or maintains the number of cases in the active file only if a new active file is created by data transformations. Otherwise, the eliminated cases are still on the active file and can be restored for subsequent procedures using the N command with a larger number.
- You cannot increase the size of the active file by specifying a value for N greater than the original number of cases.
- The keyword ESTIMATED allows SPSS/PC to optimize the allocation of memory.

Example

```
N 100.
```

- This example limits the number of cases in the active file to the first 100 cases.

Example

```
DATA LIST FILE='INVENT.DAT'/ITEM1 TO ITEM30 1-60.
N 400 ESTIMATED.
FREQ VAR=ITEM24.
N 23.
SELECT IF (ITEM11 EQ 8).
LIST VAR=ITEM1, ITEM12 TO ITEM18, ITEM24.
```

- The DATA LIST command defines 30 variables in the file INVENT.DAT in the current directory.
- The N command with keyword ESTIMATED tells SPSS/PC to allocate memory for processing approximately 400 cases.
- The FREQUENCIES procedure produces a frequency table for ITEM24 that includes all cases.

- The second N command limits the number of cases in the active file to 23 after selecting cases that have a value of 8 for variable ITEM11, as specified on the subsequent SELECT IF command.
- The SELECT IF command causes a new active file to be created. Therefore, cases beyond the 23rd case with a value of 8 for variable ITEM11 are permanently lost to the session.
- LIST produces a listing of the values of ITEM1, ITEM24, and all variables between and including ITEM12 and ITEM18 for the 23 cases in the active file.

NPAR TESTS

```
NPAR TESTS  [CHISQUARE=varlist [(lo,hi)]]

  [/EXPECTED={EQUAL**      }]
             {f1,f2, ...fn}

  [/K-S ({UNIFORM [,lo,hi]})=varlist]
        {NORMAL [,m,sd]   }
        {POISSON [,m]     }

  [/RUNS ({MEAN  })=varlist]
         {MEDIAN}
         {MODE  }
         {value }

  [/BINOMIAL [({.5})]=varlist [({value1,value2})]]
             {p   }           {value         }

  [/MCNEMAR=varlist] [WITH varlist]

  [/SIGN=varlist] [WITH varlist]

  [/WILCOXON=varlist] [WITH varlist ]

  [/COCHRAN=varlist]

  [/FRIEDMAN=varlist]

  [/KENDALL=varlist]

  [/MEDIAN [(value)]=varlist BY var (value1,value2)]

  [/M-W=varlist BY var (value1,value2)]

  [/K-S=varlist BY var (value1,value2)]

  [/W-W=varlist BY var (value1,value2)]

  [/MOSES[(n)]=varlist BY var (value1,value2)]

  [/K-W=varlist BY var (value1,value2)]

  [/OPTIONS=option numbers]

  [/STATISTICS=statistic numbers]
```

**Default if subcommand is omitted.

Options:

1 Include user-missing values
2 Exclude missing values listwise
3 Sequential pairing of variables for two related samples
4 Random sampling

Statistics:

1 Mean, maximum, minimum, standard deviation, and count
2 Quartiles and count

Example:

```
NPAR TESTS K-S(UNIFORM)=AVAR/ K-S(NORMAL,0,1)=BVAR.
```

Overview

Procedure NPAR TESTS is a collection of nonparametric tests that make minimal assumptions about the underlying distribution of data. All of these tests are described in Siegel (1956).

The tests available in NPAR TESTS can be grouped into three broad categories, based on how the data are organized: one-sample tests, related-samples tests, and independent-samples tests. A one-sample test analyzes one variable. A test for related samples compares two or more variables for the same set of cases. An independent-samples test analyzes one variable grouped by categories of another variable.

The one-sample tests available in procedure NPAR TESTS are

- BINOMIAL.
- CHISQUARE.
- K-S (Kolmogorov-Smirnov).
- RUNS.

Tests for two related samples are

- MCNEMAR.
- SIGN.
- WILCOXON.

Tests for *k* related samples are

- COCHRAN.
- FRIEDMAN.
- KENDALL.

Tests for two independent samples are

- M-W (Mann-Whitney).
- K-S (Kolmogorov-Smirnov).
- W-W (Wald-Wolfowitz).
- MOSES.

And the tests for *k* independent samples are

- K-W (Kruskal-Wallis).
- MEDIAN.

Tests are described below in alphabetical order.

Defaults There are no default tests; each test must be requested by its subcommand keyword. By default, cases with missing values are deleted on a test-by-test basis within subcommands.

Tailoring **Statistical Display.** In addition to the tests, you can request univariate statistics, quartiles, and counts for all variables named on the command. You can also control the pairing of variables in two-related-samples tests.

Random Sampling. NPAR TESTS must store cases in memory for tests that use ranks. You can use random sampling when there is not enough space to store all cases.

Missing Values. You can include cases with user-missing values in all tests. Optionally, you can exclude cases with missing values for any variable named on any subcommand from all tests.

Syntax

- The minimum specification is a single test subcommand and its arguments.
- Each test subcommand specifies a test and a list of variables to be tested. Some tests require additional specifications. CHISQUARE has an optional subcommand.
- The OPTIONS and STATISTICS subcommands are optional. Each can be specified only once per NPAR TESTS command.
- You can request any or all tests, and you can specify a test subcommand more than once on a single NPAR TESTS command.
- Subcommands must be separated by slashes.

- You can use keyword ALL in any variable list to refer to all user-defined variables in the active file.
- Keyword WITH controls pairing of variables in two-related-samples tests.
- Keyword BY introduces the grouping variable in two- and *k*-independent-samples tests.

Operations

- NPAR TESTS is a procedure and causes the data to be read.
- The display always uses narrow format.
- Specifying a string variable on any subcommand will stop execution of NPAR TESTS.
- When ALL is used, requests for tests of variables with themselves are ignored and a warning is printed.

Limitations

- The amount of memory required is directly proportional to the number of cases being analyzed.
- Maximum 1 each OPTIONS and STATISTICS subcommands.
- Maximum 100 subcommands per NPAR TESTS command.
- The maximum number of variables is the same as the system limit.
- The maximum range of values on the CHISQUARE subcommand is 200.

BINOMIAL Subcommand

```
NPAR TESTS BINOMIAL [({.5})]=varlist[({value,value})]
                    {p  }           {value      }
```

BINOMIAL tests whether the observed distribution of a dichotomous variable is the same as that expected from a specified binomial distribution. By default, each variable named is assumed to have only two values, and the distribution of each variable named is compared to a binomial distribution with *p* (the proportion of cases expected in the first category) equal to .5. The default display includes the number of valid cases in each group, the test proportion, and the two-tailed probability of the observed proportion.

Syntax

- The minimum specification is a list of variables to be tested.
- To change the default .5 test proportion, specify a value in parentheses immediately after the BINOMIAL subcommand keyword.
- A single value in parentheses following the variable list is used as a cutting point. Cases with values equal to or less than the cutting point form the first category; the remaining cases form the second.
- If two values appear in parentheses after the variable list, cases with values equal to the first value form the first category and cases with values equal to the second value form the second category.
- If no values are specified, the variables must be dichotomous.

Operations

- The proportion observed in the first category is compared to the test proportion. Then, the probability of the observed proportion occurring given the test proportion and a binomial distribution is computed.
- If the test proportion is the default (.5), a two-tailed probability is displayed. For any other test proportion, a one-tailed probability is displayed.
- A test statistic is calculated for each variable.

Example

```
NPAR TESTS BINOMIAL(.667)=YVAR(0,1).
```

- This example requests the one-tailed probability that, for cases having value 0 or 1 for YVAR, the proportion with value 0 is greater or less than .667.

CHISQUARE Subcommand

```
NPAR TESTS CHISQUARE=varlist [(lo,hi)] [/EXPECTED={EQUAL**      }]
                                                  {f1,f2,... fn}
```

The CHISQUARE (alias CHI-SQUARE) one-sample test tabulates a variable into categories and computes a chi-square statistic based on the differences between observed and expected frequencies. By default, equal frequencies are

expected in each category. The display includes the frequency distribution, expected frequencies, residuals, chi-square, degrees of freedom, and probability.

Syntax

- The minimum specification is a list of variables to be tested.
- Optionally, you can specify a value range in parentheses following the variable list.
- You can also specify expected proportions with the EXPECTED subcommand.
- If you use the EXPECTED subcommand to specify unequal expected frequencies, you must specify a value greater than 0 for each observed category of the variable or the keyword EQUAL.
- The expected frequencies are specified in ascending order of category value.
- You can use the notation *n***f* to indicate that frequency *f* is expected for *n* consecutive categories.
- Specifying keyword EQUAL on the EXPECTED subcommand has the same effect as omitting the EXPECTED subcommand.
- EXPECTED applies to all variables named on the CHISQUARE subcommand.
- Use multiple CHISQUARE and EXPECTED subcommands to specify different expected proportions for variables.
- You can request CHISQUARE with its alias CHI-SQUARE.

Operations

- If no range is specified for the variables to be tested, each distinct value encountered defines a category.
- If a range is specified, integer-valued categories are established for each value within the range. Noninteger values are truncated before classification. Cases with values outside the specified range are excluded.
- EXPECTED values are interpreted as proportions, not absolute values. Values are summed, and then each value is divided by the total to calculate the proportion of cases expected in the corresponding category.
- A test statistic is calculated for each variable named.

Example

```
NPAR TESTS CHISQUARE=AVAR (1,5)/ EXPECTED= 12, 3*16, 18.
```

- This example requests the chi-square test for values 1 through 5 of variable AVAR.
- The observed frequencies for variable AVAR are compared with the hypothetical distribution of 12/78 occurrences of value 1; 16/78 occurrences each of values 2, 3, and 4; and 18/78 occurrences of value 5.

COCHRAN Subcommand

```
NPAR TESTS COCHRAN=varlist
```

COCHRAN calculates Cochran's *Q*, which tests whether the distribution of values of *k* dichotomous variables is the same for all the variables. The display shows the frequency distribution for each variable, degrees of freedom, and probability.

Syntax

- The minimum specification is a list of two variables.
- Variables must be dichotomous and must be coded with the same two values.

Operations

- A 2 × *k* contingency table (category vs. variable) is constructed for dichotomous variables and the proportions for each variable are computed.
- Cochran's *Q* statistic has approximately a chi-square distribution.
- A single test comparing all variables is performed.

Example

```
NPAR TESTS COCHRAN=RVAR1 TO RVAR3.
```

- This example tests whether the distribution of values 0 and 1 for RVAR1, RVAR2, and RVAR3 is the same.

FRIEDMAN Subcommand

```
NPAR TESTS FRIEDMAN=varlist
```

FRIEDMAN tests whether *k* related samples have been drawn from the same population. The display shows the mean rank for each variable, number of valid cases, chi-square, degrees of freedom, and probability.

Syntax

- The minimum specification is a list of two variables.
- Variables should be at least at the ordinal level of measurement.

Operations

- The values of *k* variables are ranked from 1 to *k* for each case and the mean rank is calculated for each variable over all cases.
- The test statistic has approximately a chi-square distribution.
- A single test statistic comparing all variables is calculated.

Example

```
NPAR TESTS FRIEDMAN=SVAR TVAR UVAR
/STATISTICS = 1.
```

- This example tests variables SVAR, TVAR, and UVAR, and requests univariate statistics for all three.

K-S Subcommand (One Sample)

```
NPAR TESTS K-S({NORMAL [mean stddev]})=varlist
              {POISSON [mean]      }
              {UNIFORM [min,max]   }
```

The K-S (alias KOLMOGOROV-SMIRNOV) one-sample test compares the cumulative distribution function for a variable with a uniform, normal, or Poisson distribution and tests whether the distributions are homogeneous. The parameters of the test distribution can be specified; the defaults are the observed parameters. The display shows the number of valid cases, parameters of the test distribution, most-extreme absolute, positive, and negative differences, K-S *Z*, and two-tailed probability for each variable.

Syntax

- The minimum specification is a distribution keyword and a list of variables. The distribution keywords are

NORMAL *Normal distribution.* Default parameters are the observed mean and standard deviation.

POISSON *Poisson distribution.* The default parameter is the observed mean.

UNIFORM *Uniform distribution.* Default parameters are the observed minimum and maximum values.

- The distribution keyword and its optional parameters must be enclosed within parentheses.
- The distribution keyword must be separated from its parameters by blanks or commas.
- You can request K-S with its alias KOLMOGOROV-SMIRNOV.

Operations

- The Kolmogorov-Smirnov *Z* is computed from the largest difference in absolute value between the observed and test distribution functions.
- The K-S probability levels assume that the test distribution is specified entirely in advance. The distribution of the test statistic and resulting probabilities change when the parameters of the test distribution are estimated from the sample. No correction is made.
- For a mean of 100,000 or larger, a normal approximation to the Poisson distribution is used.
- A test statistic is calculated for each variable.

Example

```
NPAR TESTS K-S(UNIFORM)=AVAR/ K-S(NORMAL,0,1)=BVAR.
```

- The first K-S subcommand compares the distribution of AVAR with a uniform distribution that has the same range as AVAR.
- The second K-S subcommand compares the distribution of BVAR with a normal distribution with a mean of 0 and a standard deviation of 1.

K-S Subcommand (Two Sample)

```
NPAR TESTS K-S=varlist BY variable(value1,value2)
```

K-S (alias KOLMOGOROV-SMIRNOV) tests whether the distribution of a variable is the same in two independent samples defined by a grouping variable. The test is sensitive to any difference in median, dispersion, skewness, and so forth, between the two distributions. The display shows the count in each group, the largest absolute, positive, and negative differences between the two groups, K-S *Z*, and the two-tailed probability for each variable.

Syntax

- The minimum specification is a test variable, the keyword BY, a grouping variable, and a pair of values in parentheses.
- The test variable should be at least at the ordinal level of measurement.
- Cases with the first value form one group and cases with the second value form the other. The order in which values are specified determines which difference is the largest positive and which is the largest negative.
- You can request K-S with its alias KOLMOGOROV-SMIRNOV.

Operations

- The observed cumulative distributions for both groups and the maximum positive, negative, and absolute differences are computed.
- Cases with values other than those specified for the grouping variable are excluded.
- A test statistic is calculated for each variable named before BY.

Example

```
NPAR TESTS K-S=YVAR1 YVAR2 BY NVAR(0,1).
```

- This example specifies two tests. The first compares the distribution of YVAR1 for cases with value 0 for NVAR with the distribution of YVAR1 for cases with value 1 for NVAR.
- A parallel test is calculated for YVAR2.

K-W Subcommand

```
NPAR TESTS K-W=varlist BY variable(value1,value2)
```

K-W (alias KRUSKAL-WALLIS) tests whether *k* independent samples defined by a grouping variable are from the same population. The display shows the number of valid cases, mean rank of the variable in each group, chi-square, probability, and chi-square and probability after correcting for ties.

Syntax

- The minimum specification is a test variable, the keyword BY, a grouping variable, and a pair of values in parentheses.
- Every value in the range defined by the pair of values forms a group.
- You can request K-W with its alias KRUSKAL-WALLIS.

Operations

- Cases from the *k* groups are ranked in a single series and the rank sum for each group is computed.
- Kruskal-Wallis *H* has approximately a chi-square distribution.
- Cases with values other than those specified for the control variable are excluded.
- A test statistic is calculated for each variable named before BY.

Example

```
NPAR TESTS K-W=YVAR BY IVAR(0,4).
```

- This example tests YVAR for groups defined by values 0 through 4 of IVAR.

KENDALL Subcommand

```
NPAR TESTS KENDALL=varlist
```

KENDALL tests whether *k* related samples are from the same population. *W* is a measure of agreement among judges where each case is one judge's rating of several entities (variables). The display includes the mean rank for each variable, valid count, *W*, chi-square, degrees of freedom, and probability.

Syntax

- The minimum specification is a list of two variables.

Operations

- The values of the *k* variables are ranked from 1 to *k* for each case and the mean rank is calculated for each variable over all cases.

- Kendall's *W* and a corresponding chi-square statistic are calculated, correcting for ties.
- *W* ranges between 0 (no agreement) and 1 (complete agreement).
- A single test statistic is calculated for all variables.

Example

```
DATA LIST /XVAR1 TO XVAR5 1-10.
BEGIN DATA.
2 5 4 5 1
3 3 4 5 3
3 4 4 6 2
2 4 3 6 2
END DATA.
NPAR TESTS KENDALL=ALL.
```

- This example tests four judges (cases) on five entities (variables XVAR1 through XVAR5).

M-W Subcommand

```
NPAR TESTS M-W=varlist BY variable(value1,value2)
```

M-W (alias MANN-WHITNEY) compares two independent samples defined by a grouping variable on a single test variable. The test statistic uses the rank of each case to test whether the groups are drawn from the same population. The display shows the mean rank of the variable within each group, the valid count for each group, the Mann-Whitney *U*, Wilcoxon *W* (the rank sum of the smaller group), the two-tailed probability of *U* (or *W*), the *Z* statistic, and the two-tailed probability of *Z* corrected for ties.

Syntax

- The minimum specification is a test variable, the keyword BY, a grouping variable, and a pair of values in parentheses.
- Cases with the first value form one group and cases with the second value form the other. Order is unimportant.
- You can request M-W with its alias MANN-WHITNEY.

Operations

- Cases are ranked in order of increasing size, and test statistic *U*—the number of times a score from Group 1 precedes a score from Group 2—is computed.
- For fewer than than 30 cases, an exact significance level is computed.
- For more than 30 cases, *U* is transformed into a normally distributed *Z* statistic.
- Cases with values other than those specified for the grouping variable are excluded.
- A test statistic is calculated for each variable named before BY.

Example

```
NPAR TESTS M-W=YVAR BY XVAR(1,2).
```

- This example tests YVAR based on the two groups defined by values 1 and 2 of XVAR.

MCNEMAR Subcommand

```
NPAR TESTS MCNEMAR=varlist [WITH varlist]
[/OPTIONS=3]
```

MCNEMAR tests whether the changes in proportions are the same for pairs of dichotomous variables. The display shows the 2 × 2 contingency table, number of valid cases, and two-tailed probability for each pair of variables.

Syntax

- The minimum specification is a list of two variables.
- Variables must be dichotomous and must have the same two values.
- Without keyword WITH, each variable pair in the list is tested.
- With keyword WITH, each variable before WITH is tested with each variable following WITH.
- With Option 3 and no WITH, the first variable is paired with the second, the second with the third, the third with the fourth, and so on.
- With Option 3 and WITH, the first variable before WITH is paired with the first variable after WITH, the second variable before WITH with the second variable after WITH, and so on.
- With Option 3 and WITH, the number of variables specified before and after WITH must be the same.

Operations

- A 2 × 2 table is constructed for each pair of dichotomous variables and a chi-square statistic is computed for cases having different values for the two variables.
- If fewer than 10 cases change values from the first variable to the second variable, the binomial distribution is used to compute the probability.

Example

```
NPAR TESTS MCNEMAR=YVAR1 YVAR2 YVAR3.
```

- This example performs the MCNEMAR test on variable pairs YVAR1 and YVAR2, YVAR1 and YVAR3, and YVAR2 and YVAR3.

MEDIAN Subcommand

```
NPAR TESTS MEDIAN [(value)]=varlist BY variable(value1,value2)
```

MEDIAN tests whether the median of a variable is the same in *k* independent samples defined by a grouping variable. For each variable, the display shows a table of the number of cases greater than and less than or equal to the median in each category of the grouping variable, the median, chi-square, degrees of freedom, and probability. By default, the median tested is calculated from all cases included in the test.

Syntax

- The minimum specification is a single test variable, the keyword BY, a grouping variable, and two values in parentheses.
- If the first grouping value is less than the second, every value in the range defined by the pair of values forms a group and a *k*-sample test is performed.
- If the first value is greater than the second, two groups are formed using the two values and a two-sample test is performed.
- To override the default median, specify a median value in parentheses following the MEDIAN subcommand keyword.

Operations

- A 2 × *k* contingency table is contructed with counts of the number of cases greater than the median and less than or equal to the median for the *k* groups.
- For more than 30 cases, a chi-square statistic is computed.
- For 30 or fewer cases, Fisher's exact procedure (one-tailed) is used instead of chi-square.
- For a two-sample test, cases with values other than the two specified are excluded.
- A test statistic is calculated for each variable named before BY.

Example

```
NPAR TESTS MEDIAN(8.4)=YVAR BY XVAR(1,2)/ MEDIAN=YVAR BY XVAR(1,2
    /MEDIAN=YVAR BY ZVAR(1,4)/ MEDIAN=YVAR BY ZVAR(4,1).
```

- The first two MEDIAN subcommands test variable YVAR grouped by values 1 and 2 of variable XVAR. The first test specifies a median of 8.4 and the second uses the observed median.
- The third MEDIAN subcommand requests a four-samples test, dividing the sample into four groups based on values 1, 2, 3, and 4 of variable ZVAR.
- The last MEDIAN subcommand requests a two-samples test, grouping cases based on values 1 and 4 of ZVAR and ignoring all other cases.

MOSES Subcommand

```
NPAR TESTS MOSES[(n)]=varlist BY variable(value1,value2)
```

The MOSES test of extreme reactions tests whether the range of an ordinal variable is the same in a control group and a comparison group defined by a grouping variable. For each variable tested, the display includes the count in the two groups, the number of outliers removed, the span of the control group before and after outliers are removed, and the one-tailed probability of the span with and without outliers. By default, 5% of the cases are trimmed from each end of the range of the control group to remove outliers.

Syntax

- The minimum specification is a test variable, the keyword BY, a grouping variable, and two values in parentheses.
- The test variable must be at least at the ordinal level of measurement.
- The first value of the grouping variable defines the control group, and the second value defines the comparison group.

- You can override the default 5% of cases to be trimmed from each end of the control group by specifying a value in parentheses following the subcommand keyword MOSES. This value represents an actual number of cases, not a percentage.

Operations

- Scores from the groups are arranged in a single ascending sequence.
- The span of the control group is computed as the number of cases in the sequence containing the lowest and highest control score.
- No adjustments are made for tied cases.
- Cases with values other than those specified for the grouping variable are excluded.
- A test statistic is calculated for each variable named before BY.

Example

```
NPAR TESTS MOSES=YVAR BY NVAR(0,1)/ MOSES=YVAR BY NVAR(1,0).
```

- The first MOSES subcommand tests YVAR using value 0 of NVAR to define the control group and value 1 for the experimental group. The second MOSES subcommand reverses the experimental and control groups.

RUNS Subcommand

```
NPAR TESTS RUNS({MEAN  })=varlist
               {MEDIAN}
               {MODE  }
               {value }
```

RUNS tests whether the sequence of values of a dichotomized variable is random. The display includes the test value (cutting point used to dichotomize the variable tested), number of runs, number of cases below the cutting point, number of cases equal to or greater than the cutting point, and test statistic *Z* with its one-tailed probability.

Syntax

- The minimum specification is a cutting point in parentheses followed by a test variable.
- The cutting point can be specified by an exact value or one of the keywords MEAN, MEDIAN, or MODE.

Operations

- All variables tested are treated as dichotomous: values less than the cutting point form one category, and values equal to or greater than the cutting point form the other category.
- A test statistic is calculated for each variable named.

Example

```
NPAR TESTS RUNS(MEDIAN)=XVAR/ RUNS(24.5)=XVAR/ RUNS(1)=ZVAR.
```

- This example performs three runs tests. The first two test variable XVAR, first using the median and and then using 24.5 as the cutting point.
- The third test is for variable ZVAR, with value 1 specified as the cutting point.

SIGN Subcommand

```
NPAR TESTS SIGN=varlist [WITH varlist]
[/OPTION=3]
```

SIGN tests whether the number of positive and negative differences between two paired ordinal variables in a two-related-samples test is equal. The display includes the number of positive differences, number of negative differences, number of ties, and the two-tailed binomial probability.

Syntax

- The minimum specification is a list of two variables.
- Variables should be at least at the ordinal level of measurement.
- Without keyword WITH, each variable in the list is paired with every other variable in the list.
- With keyword WITH, each variable before WITH is paired with each variable after WITH.
- With Option 3 and no WITH, the first variable is paired with the second, the second with the third, the third with the fourth, and so on.
- With Option 3 and WITH, the first variable before WITH is paired with the first variable after WITH, the second variable before WITH with the second variable after WITH, and so on.

- With Option 3 and WITH, the number of variables specified before and after WITH must be the same.

Operations

- The positive and negative differences between the pair of variables are counted. Ties are ignored.
- The probability is taken from the binomial distribution if 25 or fewer differences are observed. Otherwise, the probability comes from the *Z* distribution.
- Under the null hypothesis for large sample sizes, *Z* is approximately normally distributed with a mean of 0 and a variance of 1.

Example

```
NPAR TESTS SIGN=NVAR1,MVAR1 WITH NVAR2,MVAR2
/OPTION=3.
```

- In this example, NVAR1 is tested with NVAR2, and MVAR1 is tested with MVAR2.

W-W Subcommand

```
NPAR TESTS W-W=varlist BY variable(value1,value2)
```

W-W (alias WALD-WOLFOWITZ) tests whether the distribution of a variable is the same in two independent samples. A runs test is performed with group membership as the criterion. The display includes the number of valid cases in each group, the number of runs, *Z*, and the one-tailed probability of *Z*. If ties are present, the minimum and maximum number of ties possible, their *Z* statistics, and one-tailed probabilities are displayed.

Syntax

- The minimum specification is a single test variable, the keyword BY, a grouping variable, and two values in parentheses.
- Cases with the first value form one group and cases with the second value form the other. Order is unimportant.
- You can request W-W with its alias WALD-WOLFOWITZ.

Operations

- Cases are combined from both groups and ranked from lowest to highest.
- A runs test is performed using group membership as the criterion.
- For ties involving cases from both groups, both the minimum and maximum number of runs possible are calculated.
- For a sample size of 30 or less, the exact one-tailed probability is calculated.
- For a sample size greater than 30, the normal approximation is used.
- Cases with values other than those specified for the grouping variable are excluded.
- Test statistics are calculated for each variable named before BY.

Example

```
NPAR TESTS W-W=YVAR BY NVAR(0,1).
```

- This example ranks cases from lowest of highest based on their values for YVAR. A runs test on the group variable is done.

WILCOXON Subcommand

```
NPAR TESTS WILCOXON=varlist [WITH varlist]
[/OPTION=3]
```

WILCOXON tests the hypothesis that there are no differences between two paired populations of ordered-metric scores. The test takes into account the magnitude of the differences between two paired variables. The display includes the number of positve and negative differences and their respective means, the number of ties, the valid count, *Z*, and the probability of *Z*.

Syntax

- The minimum specification is a list of two variables.
- Without keyword WITH, each variable is paired with every other variable in the list.
- With keyword WITH, each variable before WITH is paired with each variable after WITH.
- With Option 3 and no WITH, the first variable is paired with the second, the second with the third, the third with the fourth, and so on.

- With Option 3 and WITH, the first variable before WITH is paired with the first variable after WITH, the second variable before WITH with the second variable after WITH, and so on.
- With Option 3 and WITH, the number of variables specified before and after WITH must be the same.

Operations

- The differences between the pair of variables are counted, the absolute differences ranked, the positive and negative ranks summed, and the test statistic *Z* computed from the positive and negative rank sums.
- Under the null hypothesis for large sample sizes, *Z* is approximately normally distributed with a mean of 0 and a variance of 1.

Example

```
NPAR TESTS WILCOXON=VARA VARB VARC
/OPTIONS=3.
```

- This example pairs VARA and VARB and then VARB and VARC.

Statistical Display

The following options and statistics are available in NPAR TESTS. Option 3 can be specified only for two-related-samples tests (MCNEMAR, SIGN, and WILCOXON).

Option 3 *Sequential pairing of variables for two related samples.* With a simple variable list, sequential pairs are tested. With keyword WITH, the first variable before WITH is tested with the first variable after WITH, and so on. The variable lists on both sides of WITH must have the same number of variables.

Statistic 1 *Mean, maximum, minimum, standard deviation, and count.*

Statistic 2 *Quartiles and count.* Values corresponding to the 25th, 50th, and 75th percentiles and the number of valid cases are displayed.

Example

```
NPAR TESTS MCNEMAR=XVAR YVAR ZVAR
/OPTION=3.
```

- This example tests XVAR with YVAR and YVAR with ZVAR, but not XVAR with ZVAR, as is the case without Option 3.

Random Sampling

Option 4 *Random sampling.* Use if there is insufficient memory. Ignored for the RUNS subcommand.

Missing Values

By default, cases with missing values are deleted on a test-by-test basis. For subcommands specifying several tests, each test is evaluated separately.

Option 1 *Include user-missing values.* Cases with user-defined missing values are included in all tests requested on the NPAR TESTS command.

Option 2 *Exclude missing values listwise.* Cases missing on any variable named on any subcommand are excluded from all tests.

References

Siegel, S. *Nonparametric Statistics for the Behavioral Sciences.* New York: McGraw-Hill, 1956.

ONEWAY

```
ONEWAY [VARIABLES=]varlist BY varname(min,max)

   [/POLYNOMIAL=n]

   [/CONTRAST=coefficient list] [/CONTRAST=...]

   [/RANGES={SNK           }] [RANGES=...]
            {BTUKEY        }
            {TUKEY         }
            {ranges values}
            {LSD           } [({.05  })]
            {DUNCAN        }   {alpha}
            {MODLSD        }
            {SCHEFFE       }

   [/OPTIONS=option numbers]

   [/STATISTICS={statistic numbers}]
                {ALL               }
```

Options:

1 Include user-missing values
2 Exclude missing values listwise
3 Suppress variable labels
4 Write matrix of counts, means, and standard deviations
6 Use value labels as group labels
7 Read matrix of counts, means, and standard deviations
8 Read matrix of counts, means, pooled variance, and degrees of freedom
10 Harmonic mean of all group sizes as sample sizes in range tests

Statistics:

1 Group descriptive statistics
2 Fixed- and random-effects statistics
3 Homogeneity-of-variance tests

Example:

```
ONEWAY VARIABLES=YVAR BY XVAR(1,4).
```

Overview

Procedure ONEWAY produces a one-way analysis of variance for an interval-level dependent variable by one numeric independent variable that defines the groups for the analysis. Other SPSS/PC procedures that perform analysis of variance are MEANS and ANOVA. Some tests not included in ANOVA are available as options in ONEWAY.

Defaults

By default, ONEWAY produces a labeled table for each dependent variable by the independent variable. The table contains the between and within groups sums of squares, mean squares, and degrees of freedom. The *F* ratio and the probability of *F* for the test are displayed. Cases that are missing on both variables used in each test are excluded from the calculation of the test statistics.

Tailoring

Trends, Contrasts, and Ranges. You can partition the between-groups sums of squares into linear, quadratic, cubic, and higher-order trend components. You can specify up to 10 contrasts to be tested with the *t* statistic. You can also specify seven different range tests for comparisons of all possible pairs of group means.

Display Format. You can suppress the display of variable labels. You can also label groups with the value labels of the independent variable.

Statistical Display. In addition to the default display, you can obtain means, standard deviations, and other descriptive statistics for each group. Fixed- and random-effects statistics as well as several tests for homogeneity of variance are also available. The harmonic mean of all group sizes can be used as the sample size for each group in range tests.

Writing and Reading Matrices. ONEWAY can write out a matrix of group frequencies, means, and standard deviations for use in subsequent analyses. ONEWAY also reads matrix materials consisting of group frequencies, means, pooled variance, and degrees of freedom for the pooled variance.

Missing Values. You can include cases with user-missing values in the analysis or omit cases with missing values for any variable in the analysis list from all calculations.

Syntax

- The minimum specification is the VARIABLES subcommand with a single analysis list. The actual keyword VARIABLES may be omitted.
- The minimum analysis list specifies a dependent variable, the keyword BY, an independent variable, and the minimum and maximum values of the independent variable in parentheses.
- The VARIABLES subcommand must be specified first. Other subcommands can be specified in any order.
- Subcommands must be separated by a slash.

Operations

- ONEWAY is a procedure and causes the data to be read.
- Noninteger values for the independent variable are truncated.
- Cases with values outside the range specified for the independent variable are omitted from the analysis.
- Specifying a string variable as an independent or dependent variable stops execution of ONEWAY.
- The display uses the width defined on SET.
- If SPSS/PC encounters more than one each of the POLYNOMIAL, OPTIONS, or STATISTICS subcommands, it uses the last one of each type.

Limitations

- Maximum 100 dependent variables.
- Maximum 1 independent variable.
- The number of categories for the independent variable is unlimited.
- Maximum 1 VARIABLES subcommand.
- Maximum 1 POLYNOMIAL subcommand.
- Maximum 10 CONTRAST subcommands.
- Maximum 10 RANGES subcommands.
- Maximum 1 OPTIONS subcommand.
- Maximum 1 STATISTICS subcommand.
- Contrasts tests are not performed if the range of values for the independent variable exceeds 50.
- Range tests are not performed if there are more than 50 nonempty categories.
- Range tests are not performed on less than 3 groups.

Example

```
ONEWAY VARIABLES=YVAR BY XVAR(1,4).
```

- This example names YVAR as the dependent variable and XVAR as the independent variable with a minimum value of 1 and a maximum value of 4.

VARIABLES Subcommand

The VARIABLES subcommand specifies the analysis list. The actual keyword VARIABLES may be omitted.

- An analysis list specifies a dependent variable list, the keyword BY, an independent variable, and the minimum and maximum values of the independent variable in parentheses.
- There can be only one VARIABLES subcommand and it must be specified before any of the optional subcommands.
- All variables named must be numeric.
- The minimum and maximum values of the independent variable must be separated by a comma or a space and enclosed in parentheses. These values must be integer.

POLYNOMIAL Subcommand

The POLYNOMIAL subcommand partitions the between-groups sums of squares into linear, cubic, quadratic, or higher-order trend components. The display is an expanded analysis of variance table that provides the degrees of freedom, sums of squares, mean square, F, and probability of F for each partition.

- The value specified on POLYNOMIAL denotes the highest-degree polynomial to be used.
- The polynomial value must be a positive integer less than or equal to 5. If the polynomial specified is greater than the number of groups, the highest-degree polynomial possible is assumed.
- With balanced designs, ONEWAY computes the sums of squares for each order polynomial from weighted polynomial contrasts, using the group code as the metric. These contrasts are orthogonal.
- With unbalanced designs and equal spacing between groups, ONEWAY also computes sums of squares using the unweighted polynomial contrasts. These contrasts are not orthogonal.
- The deviation sums of squares are always calculated from the weighted sums of squares (Speed, 1976).
- Only one POLYNOMIAL subcommand can be specified per ONEWAY command.

Example

```
ONEWAY VARIABLES=WELL BY EDUC6 (1,6)
/POLYNOMIAL=2.
```

- This example requests an analysis of variance of WELL by EDUC6 with second-order (quadratic) polynomial contrasts.

CONTRAST Subcommand

Use the CONTRAST subcommand to specify a priori contrasts to be tested by the t statistic. Contrasts are specified as a vector of coefficients, where each coefficient corresponds to a category of the independent variable. The display for each contrast list is the value of the contrast, the standard error of the contrast, the t statistic, the degrees of freedom for t, and the two-tailed probability of t. Both pooled- and separate-variance estimates are displayed.

- A contrast must be specified or implied for every group in the range specified for the independent variable, even if the group is empty. If the number of contrast values is less than the number of groups, contrast values of 0 are assumed for the remaining groups.
- Only one set of contrast coefficients can be specified per CONTRAST subcommand. Additional contrasts on a single CONTRAST subcommand are ignored.
- You can use the notation $n*c$ to indicate that coefficient c is repeated n times.
- Coefficients are assigned to empty and nonempty groups defined by ascending integer values of the independent variable.
- Trailing coefficients of 0 do not need to be expressed.
- A warning is issued when sets of contrasts do not sum to 0.

Example

```
ONEWAY VARIABLES=YVAR BY XVAR(1,4)
/CONTRAST = -1 -1 1 1
/CONTRAST = -1 0 0 1
/CONTRAST = -1 0 .5 .5.
```

- The first CONTRAST subcommand contrasts the combination of the first two groups with the combination of the last two groups.
- The second CONTRAST subcommand contrasts the first group with the last group.
- The third CONTRAST subcommand contrasts the first group with the combination of the third and fourth groups.

Example

```
ONEWAY VARIABLES=YVAR BY XVAR(1,4)
/CONTRAST = -1 1 2*0
/CONTRAST = -1 1 0 0
/CONTRAST = -1 1.
```

- All three CONTRAST subcommands specify the the same contrast coefficients for a four-group analysis. The first group is contrasted with the second group in all three cases.
- The first CONTRAST uses the *n***c* notation and the last CONTRAST omits the trailing zero coefficients.

RANGES Subcommand

Each RANGES subcommand specifies one of seven different tests for multiple comparisons between means. The RANGES display always includes multiple comparisons between all groups. Nonempty group means are sorted in ascending order, with asterisks indicating significantly different groups. In addition, homogeneous subsets are calculated for balanced designs. The means of the groups included within a subset are not significantly different.

- By default, the range tests use sample sizes of the two groups being compared. This is equivalent to using the harmonic mean of the sample size of the two groups. You can use Option 10 to change this default.
- The default alpha for all tests is .05. For some tests, you can specify a different alpha.

The tests available on the RANGES subcommand are

LSD(p) *Least-significant difference.* Alpha can be specified between 0 and 1. The default is .05.

DUNCAN(p) *Multiple range test.* Alpha can be specified as .01, .05, and .10 only. The default is .05. DUNCAN uses .01 if the alpha specified is less than .05; .05 if the alpha specified is greater than or equal to .05 but less than .10; .10 if the alpha specified is greater than or equal to .10; and .05 if no alpha is specified.

SNK *Student-Newman-Keuls.* Alpha is .05.

BTUKEY *Tukey's alternate procedure.* Alpha is .05.

TUKEY *Honestly significant difference.* Alpha is .05.

MODLSD(p) *Modified LSD.* Alpha can be specified between 0 and 1. The default is .05.

SCHEFFE(p) *Scheffé's test.* Alpha can be specified between 0 and 1. The default is .05.

Alternatively, you can use any other type of range by specifying range values.

- Specify the range values separated by commas or blanks.
- Up to $k-1$ range values can be specified in ascending order, where k is the number of groups and where the range value times the standard error of the combined subset is the critical value.
- If less than $k-1$ values are specified, the last value specified is used for the remaining range values.
- You can use the notation *n***r* to indicate that the range *r* is repeated *n* times.
- To use a single critical value for all subsets, specify one range value.

Example

```
ONEWAY VARIABLES=WELL BY EDUC6 (1,6)
/RANGES=SNK
/RANGES=SCHEFFE (.01).
```

- This example requests two different range tests. The first uses the Student-Newman-Keuls test and the second uses Scheffé's test with an alpha of .01.

Example

```
ONEWAY VARIABLES=WELL BY EDUC (1,6)
/RANGES=2.81, 3.34, 3.65, 3.88, 4.05.
```

- The RANGES subcommand specifies five range values.

Harmonic Means

By default, range tests use the harmonic mean of the sizes of the two groups being compared. Use Option 10 on the OPTIONS subcommand to change this default.

Option 10 *Use the harmonic mean of all group sizes as the sample size for each group in range tests.* If Option 10 is used for unbalanced designs, ONEWAY determines homogeneous subsets for all range tests.

Display Format

The default display labels groups as GRP1, GRP2, and so forth, and includes variable labels. You can change these defaults by specifying the following on the OPTIONS subcommand:

Option 3 *Suppress variable labels.*

Option 6 *Use value labels for group labels.* Use the first eight characters from the value labels of the independent variable as group labels.

Statistical Display

By default, ONEWAY displays the between- and within-groups sums of squares, mean squares, degrees of freedom, the *F* ratio, and the probability of *F* for the test. You can obtain additional statistics by specifying the following on the STATISTICS subcommand:

Statistic 1 *Group descriptive statistics.* Displays the count, mean, standard deviation, standard error, minimum, maximum, and 95% confidence interval for each group for each dependent variable.

Statistic 2 *Fixed- and random-effects statistics.* Displays the standard deviation, standard error, and 95% confidence interval for the fixed-effects model and the standard error, 95% confidence interval, and estimate of between-component variance for the random-effects model.

Statistic 3 *Homogeneity-of-variance tests.* Displays Cochran's *C*, Bartlett-Box *F*, and Hartley's *F* max.

ALL *All statistics.*

Writing Matrices

ONEWAY writes matrix materials that it can read in subsequent analyses.

- To write matrix materials, specify Option 4 on the OPTIONS subcommand.
- Matrix materials are written to the results file named on the SET command (the default is SPSS.PRC).
- If the results file is not empty when ONEWAY is executed with Option 4, the contents of the file are overwritten. Use the SET command to specify a different results file.

Option 4 *Write a matrix that is a vector of counts, means, and standard deviations.* For each dependent variable, Option 4 writes a vector of group frequencies, followed by a vector of group means and a vector of group standard deviations. Vectors are written 80 characters per line with each vector beginning on a new line. The format for the frequencies vector is F10.2. The format for the means and standard deviation vectors is F10.4. There is thus a maximum of eight values per line.

Example

```
SET RESULTS='WELL.MAT'.
DATA LIST FILE='GSS80.DAT'/
WELL 2-3 EDUC 4-5.
RECODE EDUC (0 THRU 8=1) (9 10 11=2) (12=3) (13 14 15=4)
  (16=5) (17 THRU 20=6).
ONEWAY VARIABLES=WELL BY EDUC (1,6)
/OPTIONS=4.
```

- The SET command defines file GSS80.MAT in the current directory as the results file.
- Option 4 writes group counts, means, and standard deviations for WELL by EDUC to the results file using the format supplied by ONEWAY.

Reading Matrices

You can read matrix materials in fixed or free format by specifying Option 7 or 8 on the OPTIONS subcommand. The general conventions for matrix materials are described in DATA LIST: Matrix Materials.

- If you specify Option 7 or 8 in ONEWAY, you must first use a DATA LIST command specifying matrix materials.
- All variables named on the ONEWAY analysis list must be named on the DATA LIST command.
- The analysis list must be the same as the analysis list that was used when the matrix was written.
- There must be an entry in the vectors of counts, means, and standard deviations for each group. Entries should be in ascending order of the values of the independent variable.
- Each user-generated vector must begin on a new line or record and can be entered in either fixed or freefield format.

Option 7 *Read a matrix that is a vector of counts, means, and standard deviations.* ONEWAY expects a vector of counts for each group, followed by a vector of group means and a vector of group standard deviations like those written by Option 4.

Option 8 *Read a matrix that is a vector of counts, means, and pooled variance and degrees of freedom.* ONEWAY expects a vector of counts for each group, followed by a vector of means, followed by the pooled variance (within-groups mean square) and degrees of freedom for the pooled variance. If the degrees of freedom are omitted, then $n-k$ groups are assumed, where n is the number of cases and k is the number of groups. Statistics 1, 2, and 3, and the separate variance estimate for contrasts named on the CONTRAST command are not available.

Example

```
DATA LIST FREE MATRIX
/WELL EDUC.
BEGIN DATA.
65 95 181 82 40 37
2.6462 2.7737 4.1796 4.5610 4.6625 5.2297
6.2699
494
END DATA.
ONEWAY VARIABLES=WELL BY EDUC(1,6)
/OPTIONS=8.
```

- The DATA LIST command specifies matrix input in freefield format.
- The counts for the six analysis groups are on the first line of matrix input, the means for the six groups are on the second line, the within-groups mean square is on the third line, and the within-groups degrees of freedom are on the fourth line.
- Each vector to be read begins on a new line.
- Option 8 requests that matrix materials consisting of vectors of counts, means, pooled variance, and degress of freedom be read.

Missing Values

By default, cases with missing values on either the independent or dependent variable are excluded from the test. You can change the handling of cases with missing values by specifying one of the following on the OPTIONS subcommand:

Option 1 *Include cases with user-missing values.* Cases with user-missing values are included the analyses.

Option 2 *Exclude cases with missing values listwise.* Cases that have missing values for any of the variables named in the analysis list are excluded from all analyses.

References

Speed, M. F. Response Curves in the One Way Classification with Unequal Numbers of Observations per Cell. *Proceedings of the Statistical Computing Section, American Statistical Association,* 1976.

PLOT

```
PLOT [MISSING=[{PLOTWISE**}] [INCLUDE]]
               {LISTWISE  }

 [/HSIZE={38**}]  [/VSIZE={16**}]
         {n   }           {n   }

 [/CUTPOINT={EVERY({1**})}]
                   {n   }
            {value list  }

 [/SYMBOLS={ALPHANUMERIC**                      }]
           {NUMERIC                             }
           {'symbols'[,'overplot symbols'}      }
           {X'hexsymbs'[,'overplot hexsymbs']}

 [/HORIZONTAL=['title'] [STANDARDIZE] [REFERENCE(vector)]]
                [MIN(min)] [MAX(max)] [UNIFORM]

 [/VERTICAL=['title'] [STANDARDIZE] [REFERENCE(vector)]]
              [MIN(min)] [MAX(max)] [UNIFORM]

 [/FORMAT={DEFAULT**      }]
          {CONTOUR[({10})]}
          {        {n }   }
          {OVERLAY        }
          {REGRESSION     }

 [/TITLE='title']

 /PLOT={varlist} [WITH varlist [(PAIR)]] [BY varname] [:varlist...
       {ALL    }

 [/PLOT=...]
```

**Default if subcommand is omitted.

Example:

```
PLOT FORMAT=OVERLAY/ SYMBOLS='MD'/ VSIZE=12/ HSIZE=60
/TITLE='Marriage and Divorce Rates'
/VERTICAL='Rates per 1000 population'
/HORIZONTAL='Year' REFERENCE (1918, 1945) MIN (1900) MAX (1983)
/PLOT=MARRATE DIVRATE WITH YEAR.
```

Overview

Procedure PLOT produces two-dimensional line-printer plots, including simple bivariate scatterplots, scatterplots with a control variable, contour plots, and overlay plots. You can also request bivariate regression statistics. You can choose plot symbols from over 255 ASCII graphic characters.

Defaults

By default, PLOT produces bivariate scatterplots within the page size specified on the SET command. Plots include all cases with valid values for both variables in the plot, with each symbol representing the count at that display position. The vertical and horizontal axes are labeled with variable labels. Default plot titles use either the names of the variables or type of plot requested.

Tailoring

Types of Plots. You can introduce a control variable for bivariate scatterplots or request regression plots with or without a control variable, contour plots, or overlay plots.

Plot Format. You can specify a title for the plot, and you can scale and label the horizontal and vertical axes. You can request reference lines and plot standardized variables. You can also control the plot size and specify plotting symbols and the frequency they represent.

Missing Values. You can include cases with user-missing values and request that cases with missing values be deleted listwise from all plots named on the PLOT subcommand.

Syntax

- The minimum specification is the PLOT subcommand with a variable or variable list for the vertical (Y) axis, the keyword WITH, and a variable or variable list for the horizontal (X) axis.
- The PLOT subcommand can be specified more than once on a PLOT command.
- Subcommands MISSING, VSIZE, HSIZE, CUTPOINT, and SYMBOLS apply to all plots requested and can be specified only once within any PLOT command. They can be specified anywhere before the final PLOT subcommand.
- Subcommands HORIZONTAL, VERTICAL, FORMAT, and TITLE can be specified more than once and apply only to the following PLOT subcommand.
- PLOT must be the last subcommand specified.
- Subcommands must be separated by slashes.
- PLOT subcommands cannot be used without specifications.

Operations

- PLOT is a procedure and causes the data to be read.
- The default plot frame size depends on the page size specified on SET.
- A longer page length can produce longer plots within the same width. A wider page does not produce a wider plot unless the page length is changed accordingly.

Limitations

- No limitation on the number of plots requested.
- No limitation on the number of variables on a PLOT command.
- Maximum 20 overlay plots per FORMAT subcommand.
- Maximum 1 control variable per PLOT subcommand.
- Maximum 60 characters per TITLE subcommand.
- Maximum 36 symbols per SYMBOLS subcommand.
- Maximum 35 cutpoints per CUTPOINT subcommand.
- Maximum 10 reference points on each HORIZONTAL and VERTICAL subcommand.
- Maximum 40 characters per label on each HORIZONTAL and VERTICAL subcommand.
- Maximum 35 contour levels for each CONTOUR plot.

Example

```
PLOT FORMAT=OVERLAY/ SYMBOLS='MD'/ VSIZE=12 HSIZE=60
/TITLE='Marriage and Divorce Rates'
/VERTICAL='Rates per 1000 population'
/HORIZONTAL='Year' REFERENCE (1918, 1945) MIN (1900) MAX (1983)
/PLOT=MARRATE DIVRATE WITH YEAR.
```

- This example produces an overlay plot of marriage and divorce rates by year.
- SYMBOLS selects the symbols M and D, respectively, for the two plots.
- VSIZE and HSIZE limit the vertical and horizontal axes to 12 lines and 60 columns, respectively.
- TITLE specifies a plot title, and VERTICAL provides a title for the vertical axis.
- HORIZONTAL provides a title for the horizontal axis. REFERENCE, MIN, and MAX provide reference lines at values 1918 and 1945 and minumum and maximum scale values on the horizontal axis.

PLOT Subcommand

The PLOT subcommand names the variables to be plotted on each axis. The PLOT subcommand can also name a control or contour variable.

- PLOT is the only required subcommand.

- Each plot list first specifies a list of variables to be plotted on the vertical axis, then the keyword WITH, and then a list of variables to be plotted on the horizontal axis.
- By default, PLOT creates separate plots for each combination of variables listed on the left side of WITH with variables on the right.
- Use semicolons to separate multiple plot lists on a single PLOT subcommand.
- Keyword ALL can be used to refer to all user-defined variables.
- Use keyword BY followed by a variable name to specify a control or contour variable.
- Only one control variable can be specified on any plot list.
- If a control variable is specified, PLOT uses the first character of the control variable's value label as the plot symbol. If value labels have not been specified, the first character of the value is used. The symbol $ indicates that more than one control value occurs in that display position.

You can also request special pairing of variables with the following keyword:

(PAIR) *Plot corresponding pairs of variables.* The first variable before WITH is plotted against the first variable after WITH, and so on.

Example

```
PLOT PLOT=MARRATE WITH YEAR AGE;
BIRTHS DEATHS WITH INCOME1 INCOME2 (PAIR);
DIVRATE WITH AGE BY YEAR.
```

- This PLOT subcommand contains three plot lists. The first requests a plot of MARRATE with YEAR and of MARRATE with AGE.
- The second uses the keyword (PAIR) to request two plots: BIRTHS with INCOME1 and DEATHS with INCOME2.
- The third requests a plot of DIVRATE with AGE using YEAR as a control variable. The value labels for YEAR will be used as plotting characters in this control plot.

FORMAT Subcommand

The FORMAT subcommand controls the type of plot produced.

- FORMAT can be specified only once before each PLOT subcommand and applies only to plots requested on that PLOT subcommand.
- If the subcommand is omitted or DEFAULT is specified, bivariate scatterplots are displayed.
- Only one keyword can be specified on each FORMAT subcommand.

The available keywords are

DEFAULT *Bivariate scatterplot.* When there is no control variable on the plot list, each symbol represents the case count at that plot position. When a control variable is specified, each symbol represents the first character of the value label of the control variable.

OVERLAY *Overlay plots.* All bivariate plots on the next PLOT subcommand appear in one plot frame. PLOT selects a unique symbol for each plot to be overlaid, plus a symbol to represent multiple plot points in one display position.

CONTOUR(n) *Contour plot with* n *levels.* Contour plots use a continuous variable as the control variable and *n* successive symbols to represent lowest to highest levels of the variable. Specify the control variable after BY on the PLOT subcommand. This variable is recoded into *n* equal-width intervals. If the levels specification is omitted, the default of 10 is used; the maximum is 35. When more than one level of the contour variable occurs at the same plot position, PLOT displays the value of the highest level.

REGRESSION *Regression of vertical-axis variable on horizontal-axis variable.* The regression-line intercepts on each axis are marked with the letter R. When there is no control variable, each symbol

represents the frequency of cases at that plot position. If a control variable is specified, regression statistics are pooled over all categories and each symbol represents the first character of the value labels of the control variable.

Plot Symbols

A wide range of alphabetical, numeric, and special ASCII graphic characters are available for use as PLOT symbols. Two subcommands control the display of symbols: the SYMBOLS subcommand controls the choice of plot symbols, and the CUTPOINT subcommand controls the frequencies represented by a symbol. SYMBOLS and CUTPOINT can each be specified only once and apply to all plots requested in a PLOT command. If you have more than one FORMAT subcommand within a PLOT command, the meaning of the plotting symbols can vary. The operation of SYMBOLS and CUTPOINT for each FORMAT specification is summarized below.

- *DEFAULT or REGRESSION plot, no control.* Each symbol represents the frequency of cases. Controlled by SYMBOLS and CUTPOINT.
- *DEFAULT or REGRESSION plot, control.* Each symbol represents one value of the control variable. SYMBOLS and CUTPOINT do not apply. The plot symbol is the first character of the control variable's value label or the first character of the actual value if no VALUE LABELS have been declared; the uniqueness of these symbols is not checked.
- *OVERLAY.* Each symbol represents one of the overlaid plots. SYMBOLS is applicable; CUTPOINT is not.
- *CONTOUR.* Each symbol represents one level of the contour variable. SYMBOLS is applicable; CUTPOINT is not.

CUTPOINT Subcommand

By default, each frequency in a frequency plot is represented by a different plot symbol, and successive plotting symbols represent an interval width of 1. Use the CUTPOINT subcommand to alter the categories represented by plot symbols for bivariate and regression plots.

- CUTPOINT can be specified only once and applies to all frequency plots on the PLOT command.
- If the subcommand is omitted, the default interval of width 1 is in effect.
- Only one specification can be given on CUTPOINT.

The following specifications are available:

EVERY(n) *Frequency intervals of width* n. Each plot symbol represents the specified frequency interval. The default is an interval width of 1.

(value list) *Each value defines a cutpoint.* Successive plot symbols are assigned to each cutpoint. Up to 35 cutpoints can be specified. Specify values separated by blanks or commas.

Example

```
PLOT CUTPOINT=EVERY(2)/ PLOT=YVAR WITH XVAR.
PLOT CUTPOINT=(5,10,20)/ PLOT=YVAR WITH XVAR.
```

- In the first PLOT command, 1 or 2 cases on a display position are represented by a 1; 3 or 4 cases by a 2, and so forth.
- In the second PLOT command, 1 to 5 cases on a display position are represented by a 1; 6 to 10 cases by a 2; 11 to 20 cases by a 3; and 21 or more cases by a 4.

SYMBOLS Subcommand

The SYMBOLS subcommand defines the plotting symbols for bivariate scatterplots and bivariate regression, overlay, and contour plots. Successive symbols represent increasing frequencies in scatterplots or regression plots, successive subplots in overlay plots, and successive intervals in contour plots.

- SYMBOLS can be specified only once and applies to all plots requested except control plots.
- If the subcommand is omitted, the default alphanumeric symbol set is used.

• If SYMBOLS is specified, a table defining the plotting symbols is displayed.

The following specifications are available for SYMBOLS:

ALPHANUMERIC *Alphanumeric plotting symbols.* Includes the characters 1 through 9, A through Z, and *. Thirty-six or more cases at a position are represented by a *. This is the default.

NUMERIC *Numeric plotting symbols.* Includes the characters 1 through 9 and *. Ten or more cases at a plot position are represented by a *.

'symbols'[,'ovprnt'] *List of plot symbols.* Up to 36 symbols can be specified. Symbols are specified without any intervening blanks or commas. If specified, the list of overprint symbols is separated from the first symbol list by a comma or space. Indicate hexadecimal symbols on either list by specifying X before the list and enclosing the list in apostrophes. You can also specify any special ASCII graphic characters available on your terminal. See the SET command and your DOS manual for further reference. When overprint symbols are displayed on the screen, only the symbols from the first list will be displayed.

Example

```
PLOT CUTPOINTS=EVERY(5)/SYMBOLS='.+O',' X'
/PLOT=YVAR BY XVAR.
```

• This example uses a period (.) to represent 5 or fewer cases at one point, a plus sign (+) to represent 6 to 10 cases at the same position, and a symbol overprinting O and X to represent 11 or more cases at one position. Note the leading blanks in the list of overprint symbols.

VSIZE and HSIZE Subcommands

Use the VSIZE and HSIZE subcommands to specify the vertical and horizontal frame size for the plot, respectively.

• VSIZE and HISIZE can each be used only once per PLOT command and apply to all plots requested.

• The default size of a plot depends on the current page size. If the SET command defines a width of 79 and a length of 24, the default plot width is 38 columns and the default plot length is 16 rows.

• VSIZE and HSIZE each use a single integer as their only specification.

• VSIZE and HSIZE values do not include display lines for the plot frame itself or for auxiliary information such as titles, axis scale numbers, regression statistics, or symbol table.

• If VSIZE is greater than the length specified on SET, the symbol table and other information are displayed on another page.

• If HSIZE is greater than the width specified on SET, the plot wraps on the screen if there is insufficient width for the plot.

VERTICAL and HORIZONTAL Subcommands

The VERTICAL and HORIZONTAL subcommands control labeling and scaling for the vertical and horizontal axes, respectively.

• VERTICAL and HORIZONTAL can each be specified once before each PLOT subcommand.

• VERTICAL and HORIZONTAL apply only to plots requested by the next PLOT subcommand.

• If VERTICAL and HORIZONTAL are omitted, all defaults are in effect. If VERTICAL and HORIZONTAL are included, only those defaults explictly altered are changed.

The following keywords are available for both VERTICAL and HORIZONTAL:

'label' *Label for axis.* The label can contain up to 40 characters. A label that will not fit in the plot frame is truncated. The

default is the variable label or the variable name if no variable label has been declared.

MIN (n) *Minimum axis value.* If you specify a minimum value greater than the observed minimum value, some points will not be included in the plot. The default is the minimum observed value.

MAX (n) *Maximum axis value.* PLOT may extend the scale value slightly in order to display integer scale values of equal width. The default is the maximum observed value.

UNIFORM *Uniform values on axis.* All plots on the PLOT subcommand will have the same value scale on the axis. You imply a uniform scale when you specify both MIN and MAX. If you specify UNIFORM, PLOT determines the minimum and maximum observed values across all variables on the PLOT subcommand.

REFERENCE(values) *Values at which reference lines will be drawn.* Specify the values separated by blanks or commas. The default is no reference lines.

STANDARDIZE *Standardize variables on the axes.* Standardized variables are useful for overlay plots of variables with different scales. The default is to plot observed values.

TITLE Subcommand

Use the TITLE subcommand to label plots.

- TITLE can be specified once before each PLOT subcommand.
- TITLE applies to all plots named on the next PLOT subcommand.
- The default title is either the names of the variables in a bivariate plot or the type of plot requested on FORMAT.
- The title can be up to 60 characters long.
- The rules for specifying titles follow the usual conventions for strings (see Universals: Strings).
- The title is truncated if it exceeds the width specified on the HSIZE subcommand.

MISSING Subcommand

Use the MISSING subcommand to control the handling of cases with missing values.

- The MISSING subcommand can be specified only once on each PLOT command and applies to all plots.
- By default, cases with system-missing or user-missing values for any variables in a plot are omitted from that plot.
- The LISTWISE and PLOTWISE keywords are alternatives. Either one may also be specified with INCLUDE.

The following keywords are available for the MISSING subcommand:

PLOTWISE *Delete cases with missing values plotwise.* Cases with missing values for any variable in a plot are not included in that plot. In overlay plots, PLOTWISE applies separately to each overlaid plot in the frame, not to the full list specified on the PLOT subcommand. This is the default if the MISSING subcommand is omitted.

LISTWISE *Delete cases with missing values listwise.* Cases with missing values for any variable named on the PLOT subcommand are deleted from all plots specified on that PLOT subcommand.

INCLUDE *Include cases with user-missing values.* Only cases with system-missing values are excluded according to the missing-value treatment specified (PLOTWISE, LISTWISE, or the default).

PROCESS IF

```
PROCESS IF [(]variable {EQ} value[)]
                       {= }
```

Relational Operators:

EQ or =

Example:

```
PROCESS IF (AVAR EQ 50)
```

Overview

The PROCESS IF transformation temporarily designates cases for inclusion in the next procedure.

Syntax

- The specification is a simple logical expression that can be evaluated as true or false (see Universals: Logical Expressions).
- Parentheses enclosing the logical expression are optional.
- The only relational operator permitted is EQ (or =). No other relational operators are allowed.
- Only numeric or short string variables can be specified. Long string variables may not be used.
- String values must be enclosed in apostrophes or quotation marks.
- String values must match the length of the short string being tested.
- PROCESS IF can be entered anywhere in an SPSS/PC session, except between BEGIN DATA and END DATA.

Operations

- PROCESS IF is a transformation and is executed when the data are read for the next procedure.
- PROCESS IF temporarily designates cases for inclusion in the next procedure.
- If the logical expression is true, the case is processed. If it is false or missing, the case is not processed in the next procedure.
- If multiple PROCESS IF commands are entered before a procedure, only the last one is in effect.
- PROCESS IF has no effect on the first procedure unless an active file has been created (e.g., via BEGIN DATA, GET, or IMPORT).

Limitations

- PROCESS IF can take only one simple expression.

Example

```
PROCESS IF (AVAR EQ 50).
```

- In the next procedure, only cases for which variable AVAR is equal to 50 are included.

Example

```
PROCESS IF (SEX EQ 'MALE  ')
```

- SEX is a six-column left-justified variable. The trailing blanks must be included in the value specification.
- The next procedure uses cases in which the value of SEX is MALE with two trailing blanks.

RECODE

Numeric Recodes:

```
RECODE varlist (value list=value)...(value list=value)
       [/varlist...]
```

Input keywords available for numeric recodes are:

```
LO  LOWEST  HI  HIGHEST  THRU  MISSING  SYSMIS  ELSE
```

Output keywords available for numeric recodes are:

```
SYSMIS
```

String Recodes:

```
RECODE varlist ('string',['string'...]='string')
```

Input keywords available for string recodes are:

```
ELSE
```

Examples:

```
RECODE IVAR1 TO IVAR3 (0=1) (1=0) (2,3=-1) (9=9) (ELSE=SYSMIS).
RECODE STRINGVAR ('A','B','C'='A')('D','E','F'='B').
```

Overview

The RECODE command changes the coding scheme of an existing numeric or short string variable on a value-by-value basis or for a range of values.

Syntax

- The variable(s) to be recoded must already exist.
- Each set of value specifications is enclosed in parentheses.
- Input values are specified first (to the left of the equals sign), followed by a single output value (to the right of the equals sign).
- More than one input value can be recoded to a single output value.
- Values in the value lists must be separated by blanks or commas.
- Only one output value per set of input values is allowed.
- Input values that are not mentioned remain unchanged.
- ELSE specifies all input values not previously mentioned.
- The equals sign preceding the output value is required.
- You can recode more than one variable using the same value specifications by specifying a list of variables before the value specifications.
- More than one variable can be recoded differently in one command by specifying the variable and the value specifications for each transformation, separated by a slash.

Numeric Variables

- THRU specifies a value range, inclusive of specified end values.
- LOWEST and HIGHEST (LO and HI) specify the lowest and highest values encountered in the data.
- LOWEST and HIGHEST include user-missing values.
- LOWEST does not include the system-missing value.
- ELSE includes all input values not already specified but does not include the system-missing value.
- MISSING specifies user- and system-missing values for recoding.
- MISSING is an input specification only.
- SYSMIS specifies the system-missing value and can be used as both an input and output value.

String Variables

- Only short string variables can be recoded. Long strings cannot be recoded.
- Values must be enclosed in apostrophes or quotation marks.

- Blanks are significant characters.
- Input and output values must be specified according to the format width of the variable (see Universals: Strings).

Operations

- RECODE is a transformation and is executed when the data are read for the next procedure.
- Recode value specifications are scanned left-to-right.
- A value is recoded only once per RECODE command.
- Invalid syntax stops processing of the RECODE command and any recoding of variables named on the command.

Numeric Variables

- Blank fields for numeric variables are handled according to SET BLANKS prior to recoding.
- When you recode a value that was previously defined as missing on the MISSING VALUE command, the new value is treated as valid, not missing.
- The output value from MISSING is not missing but is the new, recoded value.

String Variables

- If the input or output value is shorter or longer than the format width defined for that variable, it is an error.

Limitations

- You can recode (and count using the COUNT command) approximately 400 values prior to a data pass.
- Invalid specifications on the RECODE command that result in errors stop all processing of the RECODE command.

Example

```
RECODE IVAR1 TO IVAR3 (0=1) (1=0) (2,3=-1) (9=9) (ELSE=SYSMIS)/
   QVAR(1 THRU 5=1)(6 THRU 10=2)(11 THRU HI=3)(ELSE=0).
```

- Values for the list of numeric variables between and including IVAR1 and IVAR3 are changed: input values 0 and 1 are switched to 1 and 0, respectively; 2 and 3 become −1; 9 remains 9; and any other value is changed to the system-missing value.
- Values for variable QVAR are also changed: input values 1 through 5 become 1; 6 through 10 become 2; 11 through the highest value become 3; and any other value becomes 0.

Example

```
RECODE STRINGVAR ('A','B','C'='A')('D','E','F'='B').
RECODE PET ('IGUANA', 'SNAKE ' = 'WILD  ').
```

- Values A, B, and C are changed to value A. Values D, E, and F are changed to value B.
- Values IGUANA and SNAKE are changed to value WILD. The variable PET has a format width of six characters. Thus, values SNAKE and WILD include trailing blanks to total six characters.
- Note that each string value is enclosed within apostrophes.