REGRESSION: Residuals

```
REGRESSION  VARIABLES=varlist/ DEPENDENT=varname/ METHOD=method
  [/RESIDUALS=[DEFAULTS] [DURBIN]

       [OUTLIERS({ZRESID     })] [ID (varname)]
                 {tempvarlist}

       [NORMPROB({ZRESID     })] [HISTOGRAM({ZRESID     })]
                 {tempvarlist}                {tempvarlist}

       [SIZE({SMALL})] [{SEPARATE}]]
             {LARGE}   {POOLED  }

  [/CASEWISE=[DEFAULTS] [{OUTLIERS({3     })}]]
                         {        {value} }
                         {ALL             }

              [PLOT({ZRESID })]  [{DEPENDENT PRED RESID}]]
                    {tempvar}    {tempvarlist          }

  [/SCATTERPLOT=(varname,*tempvarname)...  [SIZE({SMALL})]]
                                                {LARGE}

  [/PARTIALPLOT=[{ALL                 }] [SIZE({SMALL})]]
                 {varname,varname,...}        {LARGE}
```

Temporary residual variables are:

```
PRED ADJPRED SRESID MAHAL RESID ZPRED SDRESID
COOK DRESID ZRESID SEPRED LEVER
```

Example:

```
REGRESSION VARIABLES=SAVINGS INCOME POP15 POP75 WIDTH=132
/DEPENDENT=SAVINGS
/METHOD=ENTER
/RESIDUALS
/CASEWISE
/SCATTERPLOT (*ZRESID *ZPRED)
/PARTIALPLOT.
```

Overview

REGRESSION automatically calculates predicted values and residuals and several statistics based on these. The entire distribution of these statistics can be examined and outliers can be identified. The automatic variables are available for analysis within REGRESSION by means of casewise plots, scatterplots, and partial plots of the variables with independent variables in the analysis. In addition, you can specify any of the residuals subcommands to obtain statistics on the predicted values, residuals, and their standardized versions.

The residuals subcommands follow the last METHOD subcommand of the equation for which residuals analysis is requested. Statistics are based on this final equation.

Defaults

All residuals analysis subcommands are optional but most have active defaults that can be requested by including the subcommand without any further specifications. These active defaults are described by subcommand below.

Tailoring

You can specify which residuals statistics are presented, which are plotted, and the size of plots.

Syntax

- The optional residuals subcommands RESIDUALS, CASEWISE, SCATTERPLOT, and PARTIALPLOT must follow the last METHOD subcommand for the equation.
- Residuals subcommands can be specified in any order.
- Residuals subcommands affect only the equation they follow.
- The residuals subcommands cannot be specified if matrix input is used.

Operations

- All calculations and plots requested on the RESIDUALS subcommand are based on the regression equation produced as a result of the last method specified.
- The temporary variables PRED, RESID, ZPRED, and ZRESID are calculated and descriptive statistics are printed for these variables whenever any residuals subcommand is specified. Referring to any of the other eight temporary variables (or specifying DEPENDENT on CASEWISE) causes these other eight temporary variables to be calculated.
- Predicted values and statistics based on predicted values are calulated for every observation that has valid values for all variables in the equation.
- Residuals and statistics based on residuals are calculated for all obervations that have a valid predicted value and a valid value for the dependent variable.
- The missing-values option in effect affects the calculation of residuals and predicted values.
- The amount of information displayed in a casewise plot is limited by the display width (see REGRESSION).
- The widest page allows a maximum of eight variables in a casewise plot.
- No residuals or predictors are generated for cases deleted from the active file with the PROCESS IF, SELECT IF, or SAMPLE commands.
- All variables are standardized before plotting. If the unstandardized version of a variable is requested, the standardized version is plotted.
- For each analysis, REGRESSION can calculate 12 temporary variables:

PRED *Unstandardized predicted values.*
RESID *Unstandardized residuals.*
DRESID *Deleted residuals.*
ADJPRED *Adjusted predicted values.*
ZPRED *Standardized predicted values.*
ZRESID *Standardized residuals.*
SRESID *Studentized residuals.*
SDRESID *Studentized deleted residuals.* (See Hoaglin & Welsch, 1978.)
SEPRED *Standard errors of the predicted values.*
MAHAL *Mahalanobis' distances.*
COOK *Cook's distances.* (See Cook, 1977.)
LEVER *Leverage values.* (See Velleman & Welsch, 1981.)

Limitations

- If there is not enough storage available to assemble the requested plots, a warning is printed. Small plots are displayed and some plots may be deleted.

Example

```
REGRESSION VARIABLES=SAVINGS INCOME POP15 POP75/WIDTH=132
/DEPENDENT=SAVINGS
/METHOD=ENTER
/RESIDUALS
/CASEWISE
/SCATTERPLOT (*ZRESID *ZPRED)
/PARTIALPLOT.
```

- This REGRESSION command requests a single equation in which SAVINGS is the dependent variable and INCOME, POP15, and POP75 are independent variables.
- The RESIDUALS subcommand requests the default residuals output.
- Because residuals processing has been requested, statistics for predicted values, residuals, and standardized versions of predicted values and residuals are displayed.
- The CASEWISE subcommand requests a casewise plot of ZRESID of cases for which the absolute value of ZRESID is greater than 3. Values of the dependent variable, predicted value, and residual are listed for each case.

- The SCATTERPLOT subcommand requests a small plot of the standardized predicted value by the standardized residual.
- The PARTIALPLOT subcommand requests small partial residual plots of SAVINGS by POP75, SAVINGS by POP15, and SAVINGS by INCOME.

RESIDUALS Subcommand

The RESIDUALS subcommand controls the display and labeling of summary information on outliers as well as the display of the Durbin-Watson statistic and histograms and normal probability plots for the temporary variables.

- The RESIDUALS subcommand without specifications displays a histogram of the standardized residuals, a normal probability plot of the standardized residuals, the values of $CASENUM and ZRESID for the 10 cases with the largest absolute value of ZRESID, and the Durbin-Watson test statistic. The size of both plots is large.
- If any keyword specifications are given for RESIDUALS, only the displays requested are produced.

DEFAULTS *SIZE(LARGE), DURBIN, NORMPROB(ZRESID), HISTOGRAM(ZRESID), and OUTLIERS(ZRESID).* This is the default if the RESIDUALS subcommand is included without specifications.

SIZE(plotsize) *Plot sizes.* The plot size can be SMALL or LARGE. The default is large plots if the display width is at least 120 (see REGRESSION) and the page length is at least 57 (see SET). Four small histograms or normal probability plots can be displayed on a single page if the width is 132 (see REGRESSION) and the page length is 59.

HISTOGRAM(tempvars) *Histogram of the standardized temporary variable or variables.* The default is ZRESID. The other temporary variables for which histograms are available are PRED, RESID, ZPRED, DRESID, ADJPRED, SRESID, and SDRESID.

NORMPROB(tempvars) *Normal probability (P-P) plot of standardized values.* The default is ZRESID. The other temporary variables available for normal probability plots are PRED, RESID, ZPRED, DRESID, ADJPRED, SRESID, and SDRESID.

OUTLIERS(tempvars) *The 10 worst outliers based on absolute values of the specified temporary variables.* The default is ZRESID. The listing includes the value of $CASENUM and of the temporary variables for the 10 cases. The other temporary variables available for OUTLIERS are RESID, SRESID, SDRESID, DRESID, MAHAL, and COOK.

DURBIN *Durbin-Watson test statistic.*

ID(varname) *The case identifier on outlier plots.* Any variable on the active file can be named. ID also labels the list of cases produced by CASEWISE.

POOLED *Display pooled plots and statistics using all cases in the active file when the SELECT subcommand is in effect* (see REGRESSION). The alternative to POOLED is the default keyword SEPARATE, which requests separate reporting of residuals statistics and plots for selected and unselected cases.

Example

```
/RESID=DEFAULT ID(SVAR)
```

- DEFAULT implies the default residuals statistics: Durbin-Watson statistic, a normal probability plot and histogram of ZRESID, and an outlier plot of ZRESID.

- Descriptive statistics for ZRESID, RESID, PRED, and ZPRED are automatically displayed.
- ID(SVAR) names SVAR the case identifier on outlier plots. If the CASEWISE subcommand is also included, SVAR is used to label cases in the casewise plot.
- This example takes advantage of spelling permitted by three-character truncation of keywords.

CASEWISE Subcommand

The CASEWISE subcommand identifies a variable for casewise plotting (PLOT) and controls the selection of cases for plotting (OUTLIERS or ALL). CASEWISE can also be used to specify variables to be listed for each case next to the plot.

- The CASEWISE subcommand without specifications displays a casewise plot of ZRESID for cases for which the absolute value of ZRESID is at least 3. By default the values of the case sequence number, DEPENDENT, PRED, and RESID are listed next to the plot entry for each case.
- Only those defaults specifically altered are changed.

DEFAULTS *OUTLIERS(3), PLOT(ZRESID), DEPENDENT, PRED, and RESID.* This is the default if the CASEWISE subcommand is included without specifications.

OUTLIERS(value) *Plot only cases for which the absolute standardized value of the plotted variable is at least as large as the value given.* The default value is 3. The alternative to a casewise plot of outliers is a plot of all cases (keyword ALL). The keyword OUTLIERS is ignored if the keyword ALL is also present.

ALL *Include all cases in the casewise plot.* Alternative to the OUTLIERS keyword.

PLOT(tempvar) *Plot the standardized values of the temporary variable in the casewise plot.* The default temporary variable is ZRESID. Other variables that can be plotted are RESID, DRESID, SRESID, and SDRESID.

temp varlist *Display the values of these variables for each case next to its casewise plot entry.* The default variables are DEPENDENT (the dependent variable), PRED, and RESID. Other variables that can be named are DRESID, ADJPRED, ZPRED, ZRESID, SRESID, SDRESID, SEPRED, MAHALANOBIS, COOK, and LEVER. If an ID variable is specified on the RESIDUALS subcommand, the ID variable is also listed if the width is sufficient.

Example

```
/CASEWISE=DEFAULT ALL SRE MAH COOK SDR
```

- This example requests a casewise plot of the standardized residuals for all cases.
- The dependent variable and the temporary variables PRED, RESID, SRESID, MAHAL, COOK, and SDRESID are also listed for all cases.
- This example takes advantage of spelling permitted by three-character truncation of keywords.

SCATTERPLOT Subcommand

The SCATTERPLOT subcommand names pairs of variables for scatterplots and controls the size of the plots.

- The minimum specification for SCATTERPLOT is a pair of variables in parentheses.
- There are no default specifications for SCATTERPLOT.
- The first variable named in each set of parentheses is plotted along the vertical axis, and the second variable is plotted along the horizontal axis.
- Specify as many pairs of variables in parentheses as you want.

- Plotting symbols are used to represent multiple points occurring at the same print position.
- Specify the temporary variable names with a asterisk prefix to distinguish temporary from user-defined variables.
- All scatterplots are standardized. That is, specifying *RESID is the same as specifying *ZRESID, and *PRED is the same as *ZPRED.

(varname,varname) *Plot the variables specified.* Available temporary variables are PRED, RESID, ZPRED, ZRESID, DRESID, ADJPRED, SRESID, and SDRESID, along with any variable on the VARIABLES subcommand.

SIZE(plotsize) *Plot sizes.* The plot size can be either SMALL or LARGE. The default is always small. Four small scatterplots can be displayed on a single page if the width is at least 120 (see REGRESSION) and the page length is at least 57 (see SET).

Example

```
/SCATTERPLOT (*RES,*PRE)(*RES,YVAR)
```

- This example specifies two scatterplots: residuals against predicted values and residuals against the values of the variable YVAR.
- This example takes advantage of spelling permitted by three-character truncation of keywords.

PARTIALPLOT Subcommand

Use the PARTIALPLOT subcommand to request partial residual plots and to control the size of the plots. Partial residual plots are scatterplots of the residuals of the dependent variable and an independent variable when both of these variables are regressed on the rest of the independent variables.

- If the PARTIALPLOT subcommand is included without specifications, a partial residual plot is produced for every independent variable in the equation.
- All plots are standardized.
- Plots are displayed in descending order of the standard error of *B*.

varlist *List of variables to be plotted.* Any variable entered into the equation can be named. At least two independent variables must be in the equation for partial residual plots to be produced. The default is every independent variable in the equation. You can request the defaults with keyword ALL.

SIZE(plotsize) *Plot sizes.* The plotsize can be either SMALL or LARGE. The default is always small. Four small partialplots can be displayed on a single page if the width is at least 120 (see REGRESSION) and the page length is at least 57 (see SET).

Example

```
REGRESSION VARS=YVAR TO ZVAR
/DEP=YVAR/METH=ENTER
/RESID=DEFAULTS/PARTIAL.
```

- A partial residual plot is produced for every variable from YVAR to ZVAR that is entered into the equation.
- This example takes advantage of spelling permitted by three-character truncation of keywords.

References

Cook, R. D. Detection of influential observations in linear regression. *Technometrics* 19 (1977) 15–18.

Hoaglin, D. C., and R. E. Welsch. The hat matrix in regression and ANOVA. *American Statistician* 32 (1978) 17–22.

Velleman, P. F., and R., E. Welsch. Efficient computing of regression diagnostics. *American Statistician* 35 (1981) 234–42.

REPORT

REPORT

```
[FORMAT=[TSPACE({1})] [CHDSPACE({1})] [BRKSPACE({1})] [FTSPACE({1})]
               {n}              {n}              {n}              {n}

      [LENGTH({SET length})] [MARGINS({SET width})]
              {n,n        }             {n,n      }

      [{NOLIST   }] [SUMSPACE({1})] [MISSING {'.'}]]
       {LIST[(n)]}             {n}           {'s'}

[ 'STRING=stringname ([varname] [(width)] [(BLANK)]
        ['string']...)[stringname...]]

/VARIABLES={var       } [({VALUE})] ['col head'] [(width)]
           {var TO var}   {LABEL}
                          {DUMMY}

           [(OFFSET({0}))] [var...]
                    {n}

[/MISSING={VAR             }]
          {NONE            }
          {LIST(varlist{1})}
                       {n}

[/TITLE='line1' 'line2'...] [/FOOTNOTE='line1' 'line2'...]

      or                    or

[/LTITLE='line1' 'line2'...] [/LFOOTNOTE='line1' 'line2'...]
[/CTITLE='line1' 'line2'...] [/CFOOTNOTE='line1' 'line2'...]
[/RTITLE='line1' 'line2'...] [/RFOOTNOTE='line1' 'line2'...]

/BREAK=varlist [({VALUE})] ['col head'] [(width)]
                {LABEL}

      [OFFSET({0})] [({NOTOTAL})] [({NONAME})] [({SKIP({1  })})]
              {n}     {TOTAL  }     {NAME  }          {n  }
                                                     {PAGE }
```

or

```
/BREAK=(NOBREAK) [(width)] [(OFFSET({0}))] [(SKIP({1}))]
                                    {n}            {n}

[/SUMMARY=function...['summary title'][(break col #)]

          [SKIP({0})]]
                {n}
```

or

```
[/SUMMARY=PREVIOUS[(n)]]
```

where function is

```
aggregate [(varname[(d)][({PLAIN  })] [varname...])]
                         {DOLLAR }
                         {COMMA  }

or

composite(agg(varname)...)[(report col[(d)][({PLAIN  })])]
                                            {DOLLAR }
                                            {COMMA  }
```

Aggregate Functions:

VALIDN	VARIANCE	PCLT(n)
SUM	KURTOSIS	PCIN(min,max)
MIN	SKEWNESS	ABFREQ(min,max)
MAX	MEDIAN(min,max)	RELFREQ(min,max)
MEAN	MODE(min,max)	
STDEV	PCGT(n)	

Composite Functions:

DIVIDE(agg(varname) agg(varname) [factor])
PCT(agg(varname) agg(varname))
SUBTRACT(agg(varname) agg(varname))
ADD(agg(varname) agg(varname)...)
GREAT(agg(varname) agg(varname)...)
LEAST(agg(varname) agg(varname)...)
AVERAGE(agg(varname) agg(varname)...)
MULTIPLY(agg(varname) agg(varname)...)

Example:

```
REPORT FORMAT=LIST
/VARIABLES=PRODUCT (LABEL) ' ' 'Retail' 'Products'
          SALES 'Annual' 'Sales' '1981'
/BREAK=DEPT 'Department' (LABEL)
/SUMMARY=VALIDN (PRODUCT) MEAN (SALES).
```

Overview

The REPORT procedure produces both case listings and summary statistics and gives you considerable control over the appearance of the output. REPORT calculates all the univariate statistics available in DESCRIPTIVES and the statistics and subpopulation means available in MEANS. In addition, REPORT calculates statistics not directly available in any other SPSS/PC procedure, such as computations involving aggregated statistics.

REPORT provides complete report format defaults or lets you customize column widths, titles, footnotes, spacing, and other elements. Because REPORT is so flexible and the output has so many components, it is often efficient to preview report output using a small number of cases until you find the format that best meets your needs.

Defaults

A listing report with or without subgroup classification requires the FORMAT, VARIABLES, and BREAK subcommands. Required subcommands for a report with summary statistics are VARIABLES, BREAK, and SUMMARY. By default, column heads are variable labels or variable names if no variable labels have been specified. Default column widths are 8 columns for short string and numeric variables, 20 columns for display of value labels, and the dictionary width of long string variables. Intercolumn spacing adjusts automatically, using a minimum of 1 and a maximum of 4 spaces between columns. By default, the calculation of report statistics excludes cases with user-missing values, and missing-value indicators are ignored for variables named on BREAK.

Tailoring

Display Format. REPORT provides full report format defaults and offers you optional control over page length, vertical spacing between different types of information, margins and column widths, page titles and footnotes, and labels for statistics. The maximum width and length of the report are controlled by specifications on the SET command.

Use the FORMAT subcommand to control how the report is laid out on a page and whether case listings are displayed. The STRING subcommand concatenates variables to create temporary variables that can be referenced on the VARIABLES or BREAK subcommands. The VARIABLES subcommand names the report variables used to compute statistics and controls the titles, width, and contents of report columns. The BREAK subcommand specifies the variables that define groups and controls the titles, width, and contents of break columns. The SUMMARY subcommand specifies statistics and controls the titles and spacing of summary lines. Additional subcommands control the specification and placement of multiple-line titles and footnotes.

Statistical Display. The statistical display is controlled by the SUMMARY subcommand. Statistics can be calculated for each category of a break variable and for the group as a whole. Available statistics include mean, variance, standard deviation, skewness, kurtosis, sum, minimum and maximum value, mode, median, and percentages. Composite functions perform arithmetic operations using two or more summary statistics calculated on single variables.

Missing Values. You can override the default to include user-missing values in report statistics and listings with the MISSING subcommand. You can also define a missing-value symbol to represent missing data on the FORMAT subcommand.

Syntax

- The minimum specification for REPORT is the VARIABLES and BREAK subcommands, plus either FORMAT=LIST or the SUMMARY subcommand.
- If used, FORMAT must be the first subcommand specified.
- Title and footnote subcommands can appear anywhere after FORMAT except between BREAK and SUMMARY.
- STRING must precede VARIABLES.
- VARIABLES must precede BREAK.
- MISSING must follow VARIABLES and precede the first BREAK.
- SUMMARY must immediately follow the BREAK for which summary statistics are requested.
- Only one each of the FORMAT, STRING, VARIABLES, and MISSING subcommands are allowed.
- Only one TITLE or one each of RTITLE, CTITLE, and LTITLE subcommands are allowed.
- Only one FOOTNOTE or one each of RFOOTNOTE, CFOOTNOTE, and LFOOTNOTE subcommands are allowed.
- Multiple BREAK and SUMMARY subcommands are allowed.
- Multiple SUMMARY subcommands are allowed per BREAK.
- To suppress subgroup classification when LIST is specified on FORMAT, specify (NOBREAK) on BREAK.
- To obtain summary statistics but no case listing or subgroup classification, specify NOLIST on FORMAT and (NOBREAK) on BREAK.
- The keywords on REPORT subcommands have default specifications that are in effect if the keyword is not specified. Specify keywords only when you wish to change a default.
- Keywords are enclosed in parentheses if the subcommand takes variable names as arguments.
- Subcommands are separated from each other by slashes.

Operations

- REPORT is a procedure and causes the data to be read.
- REPORT processes cases sequentially. When the value of a break variable changes, REPORT displays a statistical summary for cases processed since the last set of summary statistics was displayed.
- The file must be sorted in order on the break variable or variables.
- The maximum width and page length of the report are the width and page length specified on the SET command.
- If the column is not wide enough to display numeric values, REPORT first rounds decimal digits, then converts to scientific notation if possible, and then displays asterisks. String variables that are wider than the column are truncated.
- The format used to display values in case listings is controlled by the dictionary format of the variable. Each statistical function in REPORT has a default format.

Limitations

- Maximum 1 each of the FORMAT, STRING, VARIABLES, and MISSING subcommands are allowed.
- Maximum 1 TITLE or 1 each of RTITLE, CTITLE, and LTITLE subcommands are allowed.
- Maximum 1 FOOTNOTE or one each of RFOOTNOTE, CFOOTNOTE, and LFOOTNOTE subcommands are allowed.
- Titles and footnotes can be no wider than the number of characters in an SPSS/PC command line, minus 2 characters for the apostrophes or quotes that enclose the string.
- String variables created on the STRING subcommand can be no wider than the system page width.
- There is no fixed limit on the number of BREAK and SUMMARY subcommands. The page width limits the number of variables displayed and thereby limits the number of break variables. The limit of 10 variables on SORT CASES also acts as a limit on the number of breaks.
- Maximum 10 dummy variables per VARIABLES subcommand.
- Maximum 50 strings per STRING subcommand.
- Maximum 20 MODE and MEDIAN requests per SUMMARY subcommand.
- Maximum 20 PCGT, PCLT, and PCIN requests per SUMMARY subcommand.
- The number of report variables that can be specified depends upon the width of the report, the width of the variable columns, and the number of BREAK subcommands.
- The)PAGE function can occur only once in either the title or footnote.
- Workspace is required to store all labeling information, frequency counts if summaries request MEDIAN or MODE, strings, and computed summary statistics.
- Memory requirements significantly increase if MEDIAN or MODE is requested with variables having a wide range of values. The amount of workspace required is 20 + 8*(max −min +1) bytes per variable per function per break.
- If TOTAL is in effect, workspace requirements are almost doubled.
- Memory requirements also increase if value labels are displayed for variables with many value labels. The amount of workspace required is 4 + 24*n labels per variable.

Example

```
SORT CASES BY DEPT.
REPORT FORMAT=LIST
/VARIABLES=PRODUCT (LABEL) ' ' 'Retail' 'Products'
           SALES 'Annual' 'Sales' '1981'
/BREAK=DEPT 'Department' (LABEL)
/SUMMARY=VALIDN (PRODUCT) MEAN (SALES) 'No.Sold,Mean Sales'
```

- This report is a listing of products and sales by department. A summary of the total number of products sold and the average sales by department is also produced.
- Cases are first sorted by DEPT to ensure that cases are appropriately grouped for the calculation of statistics.
- FORMAT requests a report that includes a listing of cases within each break group.
- VARIABLES specifies PRODUCT and SALES as the report variables and requests that the value labels identifying products be displayed. Three-line column headings are provided for each report column. The first line of the column head is blank for the variable PRODUCT.
- BREAK identifies DEPT as the break variable and provides a one-line column title for the break column. (LABEL) specifies that the value label be displayed instead of the value itself.
- SUMMARY requests the calculation of the valid number of cases for PRODUCT and the mean of SALES for each value of DEPT. A title is provided for the summary line to override the default title, VALIDN.

FORMAT Subcommand

```
FORMAT=[NOLIST|LIST[(n)] ] [LENGTH[l,r)]
       [MARGINS(t,b)] [TSPACE(n)]
       [HDSPACE(n)] [BRKSPACE(n)]
       [SUMSPACE(n)] [FTSPACE(n)] [MISSING 's']
```

The optional FORMAT subcommand controls the overall width and length of the report and vertical spacing.

- Defaults not explicitly changed remain in effect.
- Keyword specifications and their arguments can be named in any order.

The following can be specified on FORMAT:

NOLIST—LIST[(n)] *Listing of individual cases.* List the values of all variables on VARIABLES for each case. The optional *n* indicates that a blank line be inserted after each *n* cases; the default is not to insert blank lines. Values for cases are listed using the default formats for the variables. The default is the alternative NOLIST, which requests that no case listing be produced.

LENGTH(t,b) *The top and bottom lines of the report.* The value for the bottom line cannot be greater than the system page length. The system page length is controlled by SET. By default, the top of the report begins at line 1 and the bottom of the report is the last line of the system page length.

MARGINS(l,r) *The columns for the left and right margins.* The right column cannot be beyond the width specified on SET. The system page width is controlled by the SET command. By default, the left margin is print column 1 and the right margin is the rightmost print column of the system page width.

TSPACE(n) *The number of blank lines between the report title and the column heads.* The default is 1.

CHDSPACE(n) *The number of blank lines beneath the longest column head.* The default is 1.

BRKSPACE(n) *The number of blank lines between the break head and the next line.* The next line is a case if LIST is in effect or the first summary line if NOLIST is in effect. The default is 1. BRKSPACE(−1) places the first summary statistic or the first case listing on the same line as the break value. When a summary line is placed on the same line as the break value, the summary title is suppressed.

SUMSPACE(n) *The number of blank lines between the last summary line at the lower break and the first summary line at the higher break when they break simultaneously.* SUMSPACE also controls spacing between the last case listed and the first summary line if LIST is in effect. The default is 1.

FTSPACE(n) *The minimum number of blank lines between the last listing on the page and the footnote.* The default is 1.

MISSING 's' *Missing-value symbol.* The symbol can be only 1 character and is used to represent both system- and user-missing values. The default is a period (.).

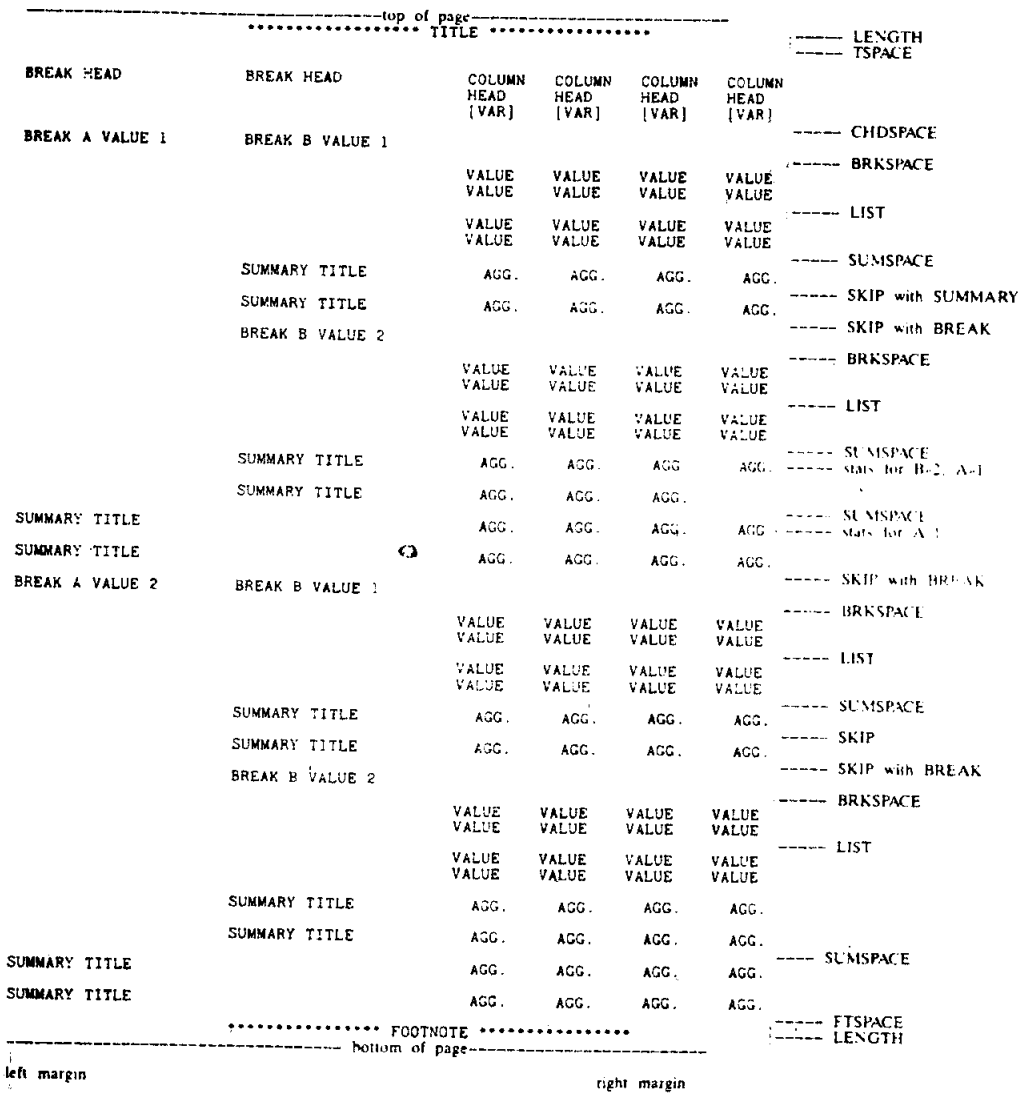
Example

```
FORMAT=LIST MARGINS(1,60) LENGTH(5,30) BRKSPACE(-1) MISSING ('*')
```

- This FORMAT subcommand requests a case listing, defines a new page size smaller than the system page size, requests that case listings begin on the line on which the break head is displayed, and specifies an asterisk as the missing-value symbol.

Page Layout This figure below displays the complete page layout and subcommand specifications used to control the basic structure of the report.

Page layout for REPORT



VARIABLES Subcommand

```
/VARIABLES=var|var TO var [(VALUE|LABEL|DUMMY)] ['col head']
          [(width)] [(OFFSET(n))] [varname...]
```

The required VARIABLES subcommand names the variables to be listed and summarized in the report. Optionally, you can use VARIABLES to control column titles, column widths, and the contents of report columns.

- The minimum VARIABLES specification is a list of variables. These are the report variables.
- Each report variable defines a report column. Each report column can be thought of as having the name of the variable that defines it.
- Variables are assigned to columns in the order in which they are named on the VARIABLES subcommand.

- To refer to a report column as a location in a composite function, use the name of the variable that is assigned to that column.
- Optional specifications can be given in any order following the variable name to which they apply.
- Optional specifications apply only to the immediately preceding variable or list of variables implied by the TO keyword.
- If an optional specification is omitted, the default is in effect.
- The value of a variable or an aggregate statistic calculated on the variable is displayed in the report column with the name of the variable.
- The number of variables that can be named is limited by the system page width.
- Variables named on the BREAK subcommand may also be named on VARIABLES.

(VALUE|LABEL|DUMMY) *Contents of the report column assigned to the variable.* If no specification is given, the keyword (VALUE) is in effect. (VALUE) specifies that values of the variable be displayed in the column. The alternative keyword (LABEL) displays value labels if value labels are defined, and values otherwise. (VALUE) and (LABEL) have no effect unless LIST has been specified on the FORMAT subcommand.

Use (DUMMY) to define a report column for a variable that does not exist in the active file. Such dummy variables are used to control spacing or to reserve space for statistics computed upon other variables. Do not name an existing SPSS/PC variable as a dummy variable.

Contents are left-justified in the column for string variables or when (LABEL) is in effect; otherwise column contents are right-justified.

'column head' *Title used for the report column assigned to the variable.* Specify multiple-line titles by enclosing each line in a set of apostrophes (') or quotes (") following the conventions for strings (see Universals: Strings). Separate the specifications for title lines with at least one blank.

If no column title is specified, the default column title is the variable label or the variable name if no variable label has been specified.

Default column titles wrap for as many lines as are required to display the entire label. User-specified column titles that exceed the column width are truncated.

Titles are left-justified for string variables or when (LABEL) is in effect; otherwise titles are right-justified.

(width) *Width for the report column.* If no width is specified, the default width is 20 if (LABEL) is in effect, 8 if (DUMMY) is in effect, or the width of the string for variables created by the STRING subcommand. The maximum width is the dictionary format width and 8 otherwise (see Universals: Formats, and FORMATS).

(OFFSET(n)) *Offset the contents of the report column* n *columns.* Contents are offset from the left for string variables or when (LABEL) is in effect, and from the right otherwise. The default is 0.

Example

```
/VARIABLES=V1 TO V3(LABEL) (15)
   V4 V5 (LABEL)(OFFSET (2))(10)
   SEP1 (DUMMY) (2) ''
   V6 'Results using' "Lieben's Method" 'of Calculation'(15)
```

- The variables from V1 through V3 each have report columns that are 15 print columns wide. Value labels are displayed for these variables in the case listing.
- Variable V4 is assigned a report column with the default width and values are listed in the case listing.
- Value labels are displayed for variable V5. The column for variable V5 is 10 columns wide and column contents are offset two spaces from the left.
- SEP1 is a dummy variable. The column assigned to SEP1 is two columns wide and there is at least one blank column on each side of SEP1. Thus, there are at least 4 blank columns between the columns for V5 and V6.
- SEP1 is given a null title to override the default column title SEP1.
- V6 is given a three-line title. The column width 15 is assigned to V6 to ensure that the column title is not truncated.

STRING Subcommand

```
/STRING=stringname ([varname] [(width)] [(BLANK)]
        ['string']...[varname]...)

        [stringname...]
```

The optional STRING subcommand concatenates variables and user-specified strings into temporary string variables that exist only within REPORT.

- The minimum specification is a name for the string variable followed by a variable name or a user-specified string enclosed in parentheses.
- The name assigned to the string variable must be unique.
- Any combination of string variables, numeric variables, and user-specified strings can be used enclosed in parentheses to define the string.
- The keyword TO cannot be used within the parentheses to specify an implied variable list.
- More than one string variable can be defined on the STRING subcommand.
- If a variable within the parentheses has a missing value, the string has a system-missing value, and the missing-value symbol is used for that value in case listings.
- A string variable defined in REPORT cannot exceed the system page width.
- String variables defined on the STRING subcommand can be used on the VARIABLES or BREAK subcommand.

(width) *Column width within the string of the preceding variable.* The default width is the dictionary width of the variable.

If the width specified is less than required by the value, asterisks are displayed for numeric variables and string variables are truncated on the right. If the width exceeds the width of a value, values of numeric variables are padded with zeros on the left and values of string variables are padded with blanks on the right.

The maximum width for numeric variables within the string definition is 16. The maximum width for a string variable is the system page width.

(BLANK) *Left-pad values of the preceding numeric variable with blanks.* If the specification is omitted, the default is to left-pad values of numeric variables with zeros. If a numeric variable has a DOLLAR or COMMA format, it is automatically left-padded with blanks.

'string' *A user-specified string.* Any combination of characters can be specified enclosed in apostrophes or quotes.

Example

```
/STRING=JOB1(AVAR VARN)
        JOB2(AVAR(2) VARN(3))
        JOB3(AVAR(2) VARN(BLANK) (4))
```

- This STRING subcommand defines four string variables to be used within the report.
- Assume that AVAR is a string variable read from a four-column field using the FIXED keyword on DATA LIST, and that VARN is a computed numeric variable with the default format of 8 columns with 2 implied decimal places.
- If a case has the value 'KJ ' for AVAR and the value 241 for VARN, JOB1 displays the value KJ 00241.00, JOB2 the value KJ241, and JOB3 the value KJ 241.

Example

```
/STRING=SOCSEC(S1 '-' S2 '-' S3)
```

- This STRING subcommand concatenates the three variables S1, S2, and S3, which each contain a portion of the social security number.
- Hyphens are inserted between the portions when the values of SOCSEC are displayed.
- This example assumes that the variables S1, S2, and S3 were read from three-column, two-column, and four-column fields respectively, using the FIXED keyword on DATA LIST. These variables have format widths of 3, 2, and 4 columns, and are not left-padded with zeros.

BREAK Subcommand: Subgroup Breaks

```
BREAK=varlist [(VALUE|LABEL)]['col title'] [(width)]
      [(OFFSET(n))] [(NOTOTAL|TOTAL)] [(NONAME|NAME)]
      [(SKIP(n)|PAGE)]
```

The required BREAK subcommand specifies the variables that define the subgroups for the report display. BREAK also allows you to control the titles, width, and contents of break columns and to begin a new page for each level of the break variable.

- The minimum specification is a variable name.
- A break occurs when any one of the break variables named on BREAK changes value. Cases must be sorted by the values of all BREAK variables on all BREAK subcommands.
- Optional specifications can be given in any order following the last variable named.
- Optional specifications apply to all variables in the break column and to the break column as a whole.
- If an optional specification is omitted, the default is in effect.
- If more than one variable is specified on a BREAK subcommand, a single break column is used. The first variable named changes most slowly. The default column width is the longest of the default widths for any of the break variables. The value or value label for each variable is displayed on a separate line in the order in which the variables are named on BREAK.
- Multiple BREAK subcommands can be used.
- A break column is reserved for each BREAK subcommand.
- When you use more than one BREAK subcommand, specify (TOTAL) only on the first BREAK to get a summary for the entire file. Otherwise, redundant statistics may be displayed.
- Missing-value specifications are ignored for variables named on the BREAK subcommand. There is one break category for system-missing values and one for user-missing values. The values are displayed by the missing-value symbol controlled by the FORMAT subcommand.
- The BREAK subcommand must precede the SUMMARY subcommand that defines the summary line for the break.
- Variables named on BREAK can also be named on VARIABLES.

The following can be specfied on BREAK:

(VALUE|LABEL) *Contents of the break column.* If no specification is given, (VALUE) is in effect. (VALUE) specifies that values of the break variables be displayed in the column. The alternative

keyword (LABEL) displays value labels if value labels have been defined, and values otherwise. The value is displayed only once for each break change and is not repeated at the top of the page in a multiple-page break group.

Contents are left-justified in the column for string variables or when (LABEL) is in effect; otherwise column contents are right-justified.

'column head' *Title used for the break column.* Specify multiple-line titles by enclosing each line in a set of apostrophes or quotes following the conventions for strings (see Universals: Strings). Separate the specifications for title lines with at least one blank.

The default title is the variable label of the break variable if there is one and the variable name of the break variable otherwise. If the break column is defined by more than one variable, the label or name of the first variable is used.

Default column titles wrap for as many lines as required to display the entire label. User-specified column titles that exceed the column width are truncated.

Titles are left-justified for string variables when (LABEL) is in effect; otherwise titles are right-justified.

(width) *Column width for the break column.* The default width is 20 when (LABEL) is in effect or the width of the string for variables created with the STRING subcommand. Otherwise, the width is the maximum of 8 or the dictionary format width. The width is 0 if (NOBREAK) is specified.

(OFFSET(n)) *Offset the contents of the break column n columns.* Contents are offset from the left for string variables or when (LABEL) is in effect and from the right otherwise. The default is 0.

(TOTAL | NOTOTAL) *With (TOTAL), summary statistics are displayed when all values of the break variables on the subcommand have been cycled through.* If (TOTAL) is specified for the first BREAK, statistics are reported for the entire report. (NOTOTAL) is the default and displays only summary statistics for each break.

(SKIP(n) | PAGE) *The vertical spacing between the last summary line for a break and the next break.* If (SKIP(n)) is specified, each break begins following *n* blank lines. The default is 1. If (PAGE) is specified, each break begins on a new page.

(NAME | NONAME) *Display the name of the break variable alongside each value or value label of the break variable.* (NAME) requires 10 print columns (the maximum eight-character length of SPSS/PC variable names plus two parentheses) in addition to the columns needed to display break values or value labels. (NAME) is ignored if the break-column width is insufficient. If the default keyword (NONAME) is specified, the name of the break variable is omitted.

Example

```
SORT CASES DIVISION BRANCH DEPT.
REPORT FORMAT=MARGINS (1,70) BRKSPACE(-1)

/VARIABLES=SPACE(DUMMY) ' ' (4)
          SALES 'Annual' 'Sales' '1981' (15) (OFFSET(2))
          EXPENSES 'Annual' 'Expenses' '1981' (15) (OFFSET(2))

/BREAK=DIVISION (LABEL)
       BRANCH (LABEL) (10) (TOTAL) (OFFSET(1))
  /SUMMARY=MEAN

/BREAK=DEPT 'Department' (LABEL) (10)
  /SUMMARY=MEAN.
```

- This example creates a report that breaks on three variables. BRANCH breaks within values of DIVISION, and DEPT breaks within values of BRANCH.
- FORMAT sets margins to a maximum of 70 columns and requests that the summary line be displayed on the same report line as the break values. Because the default NOLIST has not been overridden, only summary statistics are displayed.
- VARIABLES defines three report columns, each occupied by a report variable: SPACE, SALES, and EXPENSES.
- The variable SPACE is a dummy variable that exists only within REPORT. It has a null title and a width of 4 print columns. It is used as a space holder to separate the break columns from the report columns.
- SALES is given a three-line title and a width of 15 print columns. The values of SALES are offset 2 print columns from the right.
- EXPENSES is the third report variable and has the same width and offset specifications as SALES.
- The leftmost column in the report is reserved for the first two break variables, DIVISION and BRANCH. Value labels are displayed for both, the break column is 10 print columns wide, and the value labels are offset 1 print column from the left. Any value label more than 9 characters long is truncated. The default column title is used. (TOTAL) requests that the BRANCH summary line be displayed when all values of DIVISION and BRANCH have been cycled through, that is, at the end of the report.
- The summary line for the first BREAK subcommand consists of the mean of each report variable, displayed in its own report column. This line is displayed each time the value of DIVISION or BRANCH changes.
- The third break variable, DEPT, occupies the second column from the left in the report. The break column is 10 print columns wide and has a one-line title. The first 10 characters of the value labels are displayed in the break column.
- The second SUMMARY subcommand displays the mean for each report variable when the value of DEPT changes.

BREAK Subcommand: No Break Variables

```
/BREAK=(NOBREAK) [(width)] [(OFFSET(n))] [(SKIP(n))]
```

To obtain listing reports without break variables, specify (NOBREAK) on the BREAK subcommand. Column widths can be specified on BREAK to reserve room for summary titles.

- The minimum specification on BREAK for a report without breaks is the keyword (NOBREAK).
- Optionally, the (width), (SKIP), and (OFFSET) specifications can be used.

The defaults for optional keyword specifications depend on whether or not you have specified a column width on the BREAK subcommand with (NOBREAK). When a column width is specified, the following are in effect:

- Summary titles are left-justified in the break column. Use (OFFSET(n)) on BREAK to offset them from the left.
- BRKSPACE(n) on FORMAT is ignored.
- (SKIP(n)) on BREAK is ignored.
- SKIP(n) on SUMMARY operates as usual.

When a column width is not specified, the following are in effect:

- Summary titles are left-justified before the summary line.
- BRKSPACE(n) on FORMAT controls the spacing between the extra line for the summary title and the following summary line. BRKSPACE(−1) places the summary title and the statistic for the leftmost report variable on the same line.
- (SKIP(n)) on BREAK controls the spacing between a prior summary line and the next extra line for the summary title.
- SKIP(n) on SUMMARY is ignored.

Example

```
SORT CASES BY DEPT.
REPORT FORMAT=LIST /VAR=VAR1 TO VAR3
/BREAK=(NOBREAK) (10) (OFFSET(2))
/SUMMARY=SUM 'Totals'.
```

- The FORMAT subcommand requests a case listing.
- Because the file has been sorted by DEPT, the case listing are in order by DEPT.
- The VARIABLES subcommand defines three report variables; a report column is reserved for each, and each has default titles and spacing.
- The width specification on the BREAK subcommand reserves 10 print positions for the title of the summary line. Without this specification, there would be no room to place the title "Totals," because (NOBREAK) is also requested. The title is offset two columns from the left in the break column.
- At the end of the case listing, the sum of each report variable is displayed at the bottom of its column.

SUMMARY Subcommand

```
/SUMMARY={function...['summary title'][(break col #)] [SKIP(n)]}
         {PREVIOUS[(n)]                                        }
```

where function is

```
aggregate [(varname[(d)][(PLAIN DOLLAR COMMA)]|varname...])]
```

or

```
composite(agg(varname)...)[(report col[(d)][,PLAIN DOLLAR COMMA)]
```

The SUMMARY subcommand calculates a wide range of aggregate and composite statistics. Each SUMMARY subcommand following a BREAK subcommand specifies a new summary line.

- The minimum specification is an aggregate function or a composite function and its arguments. This must be the first specification on SUMMARY.
- Each function has its own default report-column location for displaying its result.
- The default report-column location can be altered only for composite functions.
- The default format can be altered for any function. Format specifications in an aggregate modify only the immediately preceding result.
- The default summary title is the keyword of the first function named on the SUMMARY subcommand. The default location of the summary title is the column of the break variable to which the summary applies. Both the title and the default break-column location can be altered.
- Use the SKIP keyword to insert blank lines when more than one summary line is requested for a break.
- Summary statistics requested on one SUMMARY subcommand are displayed on the same summary report line.
- SUMMARY subcommands apply only to the preceding BREAK subcommand. If there is no SUMMARY sucommand after a BREAK subcommand, no statistics are displayed for that break level.
- Specify the keyword PREVIOUS to use the summary specifications from a previous BREAK subcommand for the current BREAK subcommand.
- A SUMMARY subcommand must be specified unless NOLIST is specified on the FORMAT subcommand.
- More than one function can be specified on SUMMARY as long as you do not attempt to place two results in the same column.
- An implicit or explicit attempt to place the result of two or more functions in the same report column stops execution of REPORT. Use multiple SUMMARY subcommands to place results of more than one function in the same report column.
- Summary lines can combine any composite functions and aggregrate functions (except ABFREQ and RELFREQ).

Aggregate Functions Use the aggregate functions to request descriptive statistics on report variables.

- If no variable names are given as arguments to an aggregate function, the statistic is calculated for all variables on the VARIABLES subcommand, that is, for all report variables.
- To request an aggregate function for a subset of the report variables, specify the list of report variables in parentheses after the function keyword.
- All variables must have been named on the VARIABLES subcommand.
- The keyword TO cannot be used to specify a list of variables for an aggregate function.
- The result of an aggregate function is always displayed in the report column reserved for the variable on which the function was calculated.
- Specify multiple SUMMARY subcommands to use several aggregate functions on the same report variable. The results are displayed on different summary lines.
- The aggregate functions ABFREQ and RELFREQ have special display formats and cannot be placed on the same summary line with other aggregate or composite functions.
- Aggregate functions use only cases with valid values.

VALIDN	*Valid number of cases.* This is the only function that operates on string variables.
SUM	*Sum of values.*
MIN	*Minimum value encountered.*
MAX	*Maximum value encountered.*
MEAN	*Mean.*
STDEV	*Standard deviation.*
VARIANCE	*Variance.*
KURTOSIS	*Kurtosis.*
SKEWNESS	*Skewness.*
MEDIAN(min,max)	*Median value for values within the range.* MEDIAN sets up integer-valued bins for counting all values in the specified range. Noninteger values are truncated when the median is calculated.
MODE(min,max)	*Modal value for values within the range.* MODE sets up integer-valued bins for counting all values in the specified range. Noninteger values are truncated when the mode is calculated.
PCGT(n)	*Percentage of cases with values greater than specified value.*
PCLT(n)	*Percentage of cases with values less than specified value.*
PCIN(min,max)	*Percentage of cases within the inclusive value range specified.*
ABFREQ(min,max)	*Frequency counts for values within the inclusive range.* ABFREQ sets up integer-valued bins for counting all values in the specified range. Noninteger values are truncated when the frequency is computed. ABFREQ cannot be mixed with other aggregate statistics on a summary line.
RELFREQ(min,max)	*Percentages for values within the inclusive range.* RELFREQ sets up integer-valued bins for counting all values in the specified range. Noninteger values are truncated when the frequency is computed. RELFREQ cannot be mixed with other aggregate statistics on a summary line.

Example

```
SORT CASES BY BVAR AVAR.
REPORT FORMAT=LIST/ VARIABLES=XVAR YVAR ZVAR

/BREAK=BVAR
  /SUMMARY=SUM
  /SUMMARY=MEAN (XVAR YVAR ZVAR)
  /SUMMARY=VALIDN(XVAR)

/BREAK=AVAR
  /SUMMARY=PREVIOUS.
```

- The FORMAT subcommand requests a case listing, and the VARIABLES subcommand establishes a report column for variables XVAR, YVAR, and ZVAR. The report columns have default widths and titles.
- Both break variables, BVAR and AVAR, have the default width and titles.
- Every time the value of BVAR changes, three summary lines are displayed. The first line contains the sums of XVAR, YVAR, and ZVAR. The second line contains the means of all three variables. The third line displays the number of valid cases for XVAR in the report column for XVAR.
- Every time the value of AVAR changes within each value of BVAR, the three summary lines requested for BVAR are displayed. These summary lines are based on cases with the current value of BVAR that also have the current value of AVAR.

Example

```
SORT CASES BY DEPT.
REPORT VARIABLES=WAGE BONUS TENURE
 BREAK=DEPT (23)
 SUM=SUM(WAGE BONUS) MEAN(TENURE) 'Sum Income, Mean Tenure
```

- The SUMMARY subcommand defines a summary line consisting of the sums of WAGE and BONUS and the mean of TENURE. The result of each aggregate function is displayed in the report column of the variable on which the function is calculated.
- A title is assigned to the summary line. The break column is 23 columns wide to accommodate the right-justified summary-line title.

Composite Functions

A composite function operates on simple aggregate statistics and their arguments to produce a single result. Use composite functions to place a summary statistic in a column other than that of the report variable on which it was calculated, to manipulate variables not named on the VARIABLES subcommand, or to obtain statistics based on aggregated statistics.

- Composite functions can be computed upon constants and any variable in the active file.
- The following aggregate functions can also be arguments to composite functions: VALIDN, SUM, MIN, MAX, MEAN, STDEV, VARIANCE, KURTOSIS, and SKEWNESS. When used within composite functions, aggregate functions can have only one variable as an argument.
- By default, the results of a composite function are placed in the report column of the first variable named on the composite function that is also named on the VARIABLES subcommand.
- A composite function and its arguments cannot be separated from each other by other SUMMARY specifications.
- The result of a composite function can be placed in any report column, including columns of dummy or string variables, by specifying a target column.
- To specify a target column, enclose the variable name of a report column in parentheses after the composite function and its arguments.
- The format for the result of a composite function can be specified with a format specification in parentheses after the name of the target column and within the parentheses that enclose the target-column specification.

DIVIDE(agg() agg() [factor])	*Divide the first argument by the second and multiply by the optional factor.*
MULTIPLY (agg() ... agg())	*Multiply the arguments.*
PCT(agg() agg())	*Percentage of the first argument over the second.*
SUBTRACT(agg()agg())	*Subtract the second argument from the first argument.*
ADD(agg() ... agg())	*Add the arguments.*
GREAT(agg() ... agg())	*Give the maximum of the arguments.*
LEAST(agg() ... agg())	*Give the minimum of the arguments.*
AVERAGE(agg() ... agg())	*Give the average of the arguments.*

Example

```
SORT CASES BY DEPT.
REPORT FORMAT=BRKSPACE(-1)
/VARIABLES=WAGE BONUS SPACE1 (DUMMY) '' BNFT1 BNFT2 SPACE2
(DUMMY)''
/BREAK=DEPT (LABEL)
  /SUMMARY=MEAN(WAGE BONUS BNFT1 BNFT2)
           ADD(VALIDN(WAGE)) (SPACE2)

  /SUMMARY=ADD(SUM(WAGE) SUM(BONUS))
           ADD(SUM(BNFT1) SUM(BNFT2)) 'Totals' SKIP(1)

  /SUMMARY=DIVIDE(MEAN(WAGE) MEAN(BONUS)) (SPACE1 (COMMA)(2))
           DIVIDE(MEAN(BNFT1) MEAN(BNFT2)) (SPACE2 (COMMA)(2)) 'Ratios'
SKIP(1).
```

- The VARIABLES subcommand defines six report columns. The columns called WAGE, BONUS, BNFT1, and BNFT2 contain aggregate statistics based on those variables. The variables SPACE1 and SPACE2 are dummy variables that are created for use as space holders. Each column is given a blank title to suppress the default column head.
- The first SUMMARY computes the means of the variables WAGE, BONUS, BNFT1 and BNFT2. Because BRKSPACE=-1, this summary line will be placed on the same report line as the break value and will have no summary title. The means are displayed in the report column for each variable. It also computes the valid number of cases for WAGE inside a composite function so that the result can be placed in the column SPACE2.
- The second SUMMARY adds the sum of WAGE to the sum of BONUS. Since no location is specified, the result is displayed in the WAGE column. The sum of BNFT1 is added to the sum of BNFT2 and the result placed in the BNFT1 column. One line is skipped before the summary line requested by this SUMMARY command is displayed.
- The third summary line divides the mean of WAGE by the mean of BONUS and places the result in SPACE1. The ratio of the mean of BNFT1 to the mean of BNFT2 is displayed in the SPACE2 column. Because locations are specified, formats can also be given. The results are displayed with commas and 2 decimal places. One line is skipped before the summary line requested by this SUMMARY command is displayed.

SUMMARY Titles

By default, the title of a summary line is the keyword of the first function requested on the summary line.

- You can specify a summary title enclosed in apostrophes or quotes following the conventions for strings (see Universals: Strings).
- The summary title must be specified after the first function and its arguments and cannot separate any function from its arguments.
- A summary title can be only one line long.
- A summary title wider than the break column is truncated.

- Only one summary title applies per summary line. If more than one is specified, the last is used.
- The summary title is left- or right-justified depending upon whether the break title is left- or right-justified.
- The default location for the summary title is the column of the BREAK variable to which the SUMMARY applies.
- With multiple breaks, you can override the default placement of the title by specifying the break-column number in which you want the summary title to be displayed in parentheses following the title. You can override the default by specifying the break-column number in which you want the summary title to be displayed in parentheses following the title.
- In a (NOBREAK) report with no break-column width specified on BREAK, REPORT displays the summary title above the summary line at the left margin.

SUMMARY Print Formats

All functions have default formats that are used to print results. You can override these defaults by specifying a format keyword or the number of decimal places.

- Formats can be specified for the results of aggregate functions only if the variable is explicitly named as an argument.
- Formats can be specified for the results of composite functions only if a report-column location for the result is also specified.
- If the report column is wide enough, SUM, MEAN, STDEV, MIN, MAX, MEDIAN, and MODE use DOLLAR or COMMA format if one has been declared for a variable on the FORMATS command.
- A format specification must be enclosed in parentheses.
- For aggregate functions, one or both format specifications are placed after the variable name within the parentheses that enclose the variable name.
- For composite functions, one or both format specifications are placed after the variable name of the target column within the parentheses that enclose the target column.
- If the column is not wide enough to display the decimal digits for a given function, REPORT displays fewer decimal places. If the column is not wide enough to display the integer portion of the number, REPORT adopts scientific notation and then displays asterisks.
- An exact value of 0 is displayed with one 0 to the left of the decimal point and as many 0 digits to the right as specified by the format. A very small number is displayed without a zero digit to the left of the decimal point, except with DOLLAR and COMMA formats.

(DOLLAR) *Display the value using DOLLAR format.*

(COMMA) *Display the value using COMMA format.*

(PLAIN) *Override DOLLAR or COMMA dictionary formats.* PLAIN is the default for all functions except MEAN, STDEV, MIN, MAX, MEDIAN, and MODE. For these functions, the default is the dictionary format.

Example

```
/SUMMARY=MEAN(INCOME (DOLLAR)(2))
         ADD(SUM(INCOME)SUM(WEALTH) (WEALTH(DOLLAR)(2))
```

- SUMMARY displays the mean of INCOME with a dollar sign, commas, and two decimal digits. The format can be specified because INCOME is specified as an argument to the MEAN function. The result is displayed in the INCOME column.
- The sums of INCOME and WEALTH are added and the result is displayed with a dollar sign, commas, and 2 decimal digits. The format can be specified because an explicit location is given for the results of ADD. The result is displayed in the WEALTH column.

Default print formats for functions

Function	Format	Function	Format
SUM	Dictionary print format	PCIN	2
MEAN	Dictionary print format + 2	PCLT	2
STDEV	Dictionary print format + 2	PCGT	2
MIN	Dictionary print format	PCT	2
MAX	Dictionary print format	KURTOSIS	3
VARIANCE	Column width − 1	SKEWNESS	3
DIVIDE	Column width − 1	MODE	0
MULTIPLY	Column width − 1	MEDIAN	1
SUBTRACT	Column width − 1	VALIDN	0
ADD	Column width − 1	ABFREQ	0
GREAT	Column width − 1	RELFREQ	2
LEAST	Column width − 1		
AVERAGE	Column width − 1		

Other SUMMARY Keywords

Spacing between multiple summary lines for a single break and references to previously defined summary lines are controlled by the following keywords:

SKIP(n) *Blank lines before the summary line.* SKIP is not enclosed in parentheses. The default is 0. SKIP on the first SUMMARY subcommand for a BREAK skips the specified lines after skipping the number of lines specified for BRKSPACE on FORMAT.

PREVIOUS(n) *Use the SUMMARY subcommands for the nth BREAK.* If no specification is given in parentheses, PREVIOUS points to the set of SUMMARY subcommands for the previous BREAK. If an integer specification is given, the SUMMARY subcommands from the *n*th BREAK are used.

No other specification can be used on SUMMARY with PREVIOUS. For a multiple-break report for which you want the same sets of summaries, specify SUMMARY subcommands for the higher BREAK subcommand and keyword PREVIOUS for lower breaks.

Title and Footnote Subcommands

```
/TITLE ='title'/ (Centered head)
/FOOTNOTE='title'/ (Centered foot)
```

or

```
/LTITLE='title' (Left-justified head)
/RTITLE='title' (Right-justified head)
/CTITLE='title' (Centered head)
/LFOOTNOTE='title' (Left-justified foot)
/RFOOTNOTE='title' (Right-justified foot)
/CFOOTNOTE='title' (Centered foot)
```

- The title and footnote subcommands are optional and can be placed anywhere after the FORMAT subcommand except between a BREAK and its SUMMARY subcommands.
- Title subcommands should be specified before any footnote subcommands.
- The default REPORT title is the title specified on the TITLE command. If there is no TITLE command specified in your SPSS/PC session, the default REPORT title is the first line of the SPSS/PC header.
- A title or footnote is specified by providing a string in apostrophes or quotes on one of the title or footnote subcommands.
- If the title or footnote is more than one line, enclose each line in apostrophes or quotes and separate the specifications for each line by at least one blank.
- Titles are displayed beginning in the first line of the system page.
- Footnotes end in the last line of the system page.

- Centered titles and footnotes are centered within the report page width.
- TITLE cannot be used with CTITLE, LTITLE, or RTITLE, and FOOTNOTE cannot be used with CFOOTNOTE, LFOOTNOTE, or RFOOTNOTE.
- Only one of each title and footnote subcommand is allowed.
- Titles and footnotes are repeated on each page of a multiple-page report.
- If the total width requested for the combined titles or foonotes for a line exceeds the page width, execution of REPORT stops.
- The maximum number of characters in a title or footnote is the number of characters that can fit on a command line, minus 2 characters for the apostrophes or quotes that enclose the string.

Two function keywords can be used in a title or footnote:

)PAGE *Display the page number right-justified in a five-character field.*

)DATE *Display the current date in the form* dd/mmm/yy *right-justified in a nine-character field.*

Example

```
/LTITLE='PERSONNEL REPORT' 'PREPARED ON )DATE'
/RTITLE='Page )PAGE'
```

- This LTITLE subcommand provides a two-line title. The second line of this title contains the date on which the report was processed. This title is displayed left-justified at the top of each page of the report.
- The RTITLE subcommand displays the page number right-justified at the top of each page.

MISSING Subcommand

```
/MISSING=VAR|NONE|LIST [(varlist [n])]
```

Use the optional MISSING subcommand to control the handling of cases with missing values on the VARIABLES and SUMMARY subcommands. By default, cases with missing values are included in case listings but are excluded from the calculation of functions on a function-by-function basis.

- MISSING specifications apply to variables named on VARIABLES and SUMMARY as well as to strings created with the STRING subcommand.
- The character used to indicate missing values is controlled by the FORMAT subcommand.

The following keywords are available for the MISSING subcommand:

VAR *Missing values are treated separately for each variable.* Missing values are displayed in case listings but are not included in summary statistics. This is the default.

NONE *User-missing-value indicators are ignored.* Applies to all variables named on the VARIABLES subcommand.

LIST[(varlist[n])] *Eliminates any case with the specified number of missing values among the specified list of variables.* If no *n* is specified, the default is 1. If no variables are specified, all variables named on the VARIABLES subcommand are assumed.

Example

```
/MISSING= LIST (XVAR,YVAR,ZVAR 2)
```

- Any case with a missing value for two or more of the variables XVAR, YVAR, and ZVAR are omitted from the report.

SAMPLE

```
SAMPLE {sampling fraction          }
       {sample size FROM file size}
```

Example:

```
SAMPLE .25.
```

Overview

The SAMPLE transformation temporarily draws a random sample of cases for processing in the next procedure. Sampling is based on a pseudo-random-number generator.

Syntax

- A decimal value between 0 and 1 selects an approximate percentage of cases.
- A positive integer value less than the file size followed by keyword FROM and the file size selects an exact-sized random sample.
- SAMPLE can be entered anywhere in an SPSS/PC session, except between BEGIN DATA and END DATA.

Operations

- SAMPLE is a transformation and is executed when the data are read for the next procedure.
- By default, the initial seed value is randomly assigned. Use the SEED subcommand on SET to assign a specific seed value.
- A proportional sample (a sample based on a decimal value) usually does not produce the exact proportion specified.
- If the number (*n*) following FROM is less than the actual file size, the sample is drawn only from the first *n* cases.
- If the number following FROM is greater than the actual file size, SPSS/PC will sample an equivalent proportion of cases from the active file (see example below).
- If SAMPLE follows SELECT IF or PROCESS IF, it samples only cases selected by SELECT IF or PROCESS IF.
- If SAMPLE precedes SELECT IF or PROCESS IF, cases are selected from the sample.
- If more than one SAMPLE is specified before a procedure, only the last SAMPLE command is executed.
- If the same SAMPLE specification is used for different procedures in the same session or in different sessions, the two samples may be different because of the random assignment of the seed value. To obtain the same sample for different procedures, use the SET command specifying the same seed value before entering the SAMPLE command.

Example

```
SAMPLE .25.
```

- This command samples approximately 25% of the cases in the active file.

Example

```
SAMPLE 500 FROM 3420.
```

- In this example, the active file must have 3,420 cases or more to obtain a random sample of exactly 500 cases.
- If the file contains less than 3420 cases, approximately 14% of the cases are sampled.
- If the file contains more than 3420 cases, a random sample of 500 cases would be drawn from the first 3420 cases.

SAVE

```
SAVE [OUTFILE={SPSS.SYS**}] [/DROP=varlist]
              {'filename'}
```

**Default if subcommand is omitted.

Example:

```
SAVE DROP V1 TO V20.
```

Overview

SAVE produces an SPSS/PC system file. The system file includes all data and a data dictionary with variable and value labels (if specified), missing-value flags, and print formats for each variable. System files are read with the GET command.

Defaults

SAVE writes all variables in your active file to the file SPSS.SYS in the current directory.

Tailoring

You can direct the system file to a file of your choosing. Also, you can save only a subset of variables on the system file.

Syntax

- The minimum specification is simply the command keyword.
- Subcommands, if used, are separated by an optional slash.

Operations

- The system file created by SAVE includes the system variables $CASENUM, $DATE, and $WEIGHT and their values.
- The system file dictionary is arranged in the same order as the active file.
- System files are binary files designed to be read and written by SPSS PC only and cannot be edited.
- The active file is still available for SPSS/PC transformations and procedures after a SAVE file is created.
- To conserve space, SAVE automatically compresses small integer values.

Example

```
SAVE DROP=V1 TO V3.
```

- The system file is written to SPSS.SYS in the current directory.
- Any variables between and including V1 and V3 on the active file are excluded from the system file.

OUTFILE Subcommand

Use OUTFILE to direct the system file to a file other than the default SPSS.SYS.

- The only specification on OUTFILE is the name of the file.
- The file specification must be enclosed in parentheses.
- You can save a system file in another directory by specifying a fully qualified filename (see Universals: Files).

Example

```
SAV OUT='\NSDIR\SALDATA.SYS'.
```

- The complete active file is written to file SALDATA.SYS in directory NSDIR.
- This example takes advantage of spelling permitted by three-character truncation of keywords.

DROP Subcommand

Use the DROP subcommand to exclude a variable or list of variables from the system file. All variables not named on DROP will be in the system file. The SPSS/PC active file is not affected by specifying DROP on the SAVE command.

- You can specify variables in any order.
- You can use the TO keyword to specify a group of consecutive variables in the active file.

Example

```
SAVE OUTFILE='NEWSUM.SYS'/DROP=DEPT79 TO DEPT81, SALARY82.
```

- The system file is written to NEWSUM.SYS in the current directory.
- All variables between and including DEPT79 and DEPT81, as well as SALARY82, are excluded from the system file.
- All other variables are saved on the system file.