

## Tables and Graphics That Will “FREQ” You Out

Kathryn McLawhorn, SAS Institute Inc.

### ABSTRACT

Did you know that you could leverage the statistical power of the FREQ procedure and still be able to control the appearance of your output? Many people think they have to use procedures such as REPORT and TABULATE to be able to apply style options and control formats and headings for their output. However, if you pair PROC FREQ with a TEMPLATE procedure step, you can customize the appearance of your output and make enhancements to tables, such as adding colors and controlling headings.

If you are a statistician, you know the many PROC FREQ options that produce high-level statistics. But did you also know that PROC FREQ can generate a graphical representation of those statistics? PROC FREQ can generate the graphs, and then you can use ODS Graphics and the Graph Template Language (GTL) to improve the appearance of the graphs.

Written for intermediate users, this paper demonstrates how you can enhance the default output for PROC FREQ's one-way and multi-way tables by modifying colors, formats, and labels. This paper also describes the syntax for creating graphs for multiple statistics, and it uses examples to show how you can customize these graphs.

### INTRODUCTION

You can enhance PROC FREQ's tables and graphics when you use the procedure in conjunction with the TEMPLATE procedure and ODS Graphics. PROC TEMPLATE creates table templates that describe how tables are constructed for most procedures. Use the following SOURCE statement in PROC TEMPLATE to view the table-template definitions:

```
proc template;
  source Base.Freq.OneWayFreqs;
  source Base.Freq.OneWayList;
  source Base.Freq.CrossTabFreqs;
run;
```

You can modify the default structure and appearance of the tables using syntax that PROC TEMPLATE provides. By default, template modifications are stored in the Sasuser library so that they are retained from one SAS session to another. After you modify any template, you can restore the default output by using the DELETE statement in PROC TEMPLATE, as shown below:

```
proc template;
  delete Base.Freq.<table template name>;
  delete Base.Freq.Graphics.<graph template name>;
run;
```

This paper discusses the following table templates for the FREQ procedure:

- OneWayFreqs and OneWayList: Control the output for one-way tables
- CrossTabFreqs: Controls the output for  $n$ -way tables

Beginning in SAS® 9.2, PROC FREQ provides the ability to create graphs as an alternate way to display the data in the tables by using the PLOTS option in the TABLES statement. These types of graphs include frequency plots, mosaic plots, deviation plots, odds ratio plots, as well as others. You can then enhance these graphs using ODS Graphics and the Graph Template Language (GTL).

This paper provides examples that show how to enhance the table output generated by PROC FREQ. It also introduces some of the graphs that the procedure can generate, and it includes methods for

customizing the graphs. Some of the data sets used in the examples come from the Sashelp library. Other examples use created data in order to include missing values or because that data better illustrates the plot.

## ONE-WAY TABLES

The default one-way table generated by PROC FREQ always includes the following statistics: Frequency, Percent, Cumulative Frequency, and Cumulative Percent. Frequency values are displayed as integers, and percent values are displayed with two decimal places. The default placement for the variable label is centered above the table. The next sections of this paper describe how to use PROC TEMPLATE to move this label, provide alternate headings for the statistics, and change how the statistics are displayed.

### MOVE THE VARIABLE LABEL

Often, table output looks clearer if the label for the variable appears above the column of values. In order to move the label, you must set the default label to a blank string. Then, move VARLABEL to the HEADER statement of FVariable, which corresponds to the data set variable name used in the TABLES statement. The following sample code shows how to move the variable label:

```
proc template;
  edit Base.Freq.OneWayList;
  edit h1;
  /* Set text to a blank instead of VARLABEL. */
  text " ";
  end;
  edit FVariable;
  just=varjust;
  style=rowheader;
  id;
  generic;
  header=varlabel;
  end;
end;
run;

proc freq data=sashelp.cars;
  label origin="Location of Origin";
  tables origin;
run;
```

Output 1 is the resulting output.

Location of Origin	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Asia	158	36.92	158	36.92
Europe	123	28.74	281	65.65
USA	147	34.35	428	100.00

**Output 1. Moving the Variable Label**

## LABEL AND FORMAT STATISTICS

You can enhance table output by labeling the statistics. Formats can also make the values in the columns easier to read. In PROC TEMPLATE, you can modify the column by using the EDIT, HEADER, and FORMAT statements. The following sample code shows how to label and format statistics:

```
proc format;
  picture pctfmt (round) other="009.9%";
run;

proc template;
  edit Base.Freq.OneWayList;
  edit frequency;
  header="Count";
  format=comma12.;
end;
  edit cumfrequency;
  header="Cumulative Count";
  format=comma12.;
end;
  edit percent;
  header="Pct";
  format=pctfmt.;
end;
  edit cumpercent;
  format=pctfmt.;
end;
end;
run;

data cars;
  set sashelp.cars;
  wgt=1000;
run;

proc freq data=cars;
  tables origin;
  /* Add WEIGHT to demonstrate comma format */
  weight wgt;
run;
```

Output 2 is the resulting output.

Origin	Count	Pct	Cumulative Count	Cumulative Percent
Asia	158,000	36.9%	158,000	36.9%
Europe	123,000	28.7%	281,000	65.7%
USA	147,000	34.3%	428,000	100.0%

**Output 2. Labeling and Formatting Statistics**

## SUPPRESS THE FREQUENCY MISSING ROW

Missing variable values are not included in table output by default unless you specify the MISSING option in the TABLES statement in PROC FREQ. When these values are not part of the table, a line below the table displays, **Frequency Missing=*n***, where *n* is the number of missing values. Suppose you do not want this line to appear below the table. PROC TEMPLATE provides a way to suppress it by setting the text to a blank string for the F2, or footer, element of the table. The following code demonstrates how to suppress the Frequency Missing row:

```
data test;
input x;
datalines;
1
2
.
;
run;

proc template;
edit Base.Freq.OneWayFreqs;
edit f2;
text " ";
end;
end;
run;

proc freq data=test;
tables x;
run;
```

Output 3 is the resulting output.

x	Frequency	Percent	Cumulative Frequency	Cumulative Percent
1	1	50.00	1	50.00
2	1	50.00	2	100.00

**Output 3. Suppressing the Frequency Missing Row**

## MULTI-WAY TABLES

A multi-way table in PROC FREQ is a crosstabulation of rows and columns and a legend of the statistics included in the table. The statistics that are always included in the default table are Frequency, Percent, Row Percent, and Column Percent. Frequency values are displayed as integers, and percent values are displayed with two decimal places. If a variable has a label, the label is included in parentheses after the variable name by default. The table also includes a total row and a total column with frequency and overall percentage values. The next sections of this paper describe how to use PROC TEMPLATE to modify this table output.

## CHANGE THE COLOR OF CELL VALUES

If you want to highlight certain values in the table, you can add color based on the value. Use PROC FORMAT to assign the colors, and assign the format to the appropriate style element in PROC TEMPLATE. The following code shows how to add colors:

```
proc format;
  value color low-2="green"
            3-5="lightblue"
            6-high="orange";
run;

proc template;
  edit Base.Freq.CrossTabFreqs;
  cellvalue Frequency Expected Deviation CellChiSquare TotalPercent
            Percent RowPercent ColPercent CumColPercent;
  cell_style={background=color.};
  row_total_style={background=color.};
  col_total_style={background=color.};
  grand_total_style={background=color.};
end;
run;

proc freq data=sashelp.class;
  tables age*sex / norow nocol nopercent;
  where age le 13;
run;
```

Output 4 is the resulting output.

Table of Age by Sex			
Age	Sex		
Frequency	F	M	Total
11	1	1	2
12	2	3	5
13	2	1	3
Total	5	5	10

### Output 4. Changing the Color of Cell Values

## REPLACE ROW, COLUMN, AND TABLE HEADINGS

Above the default table, PROC FREQ includes the heading, "Table of <var1> by <var2>". The DEFINE HEADER statements in the code below replace the variable names with the variable labels and suppress the variable names in the table directly above the rows and columns. The DEFINE <statistic name> statements customize the statistic headings in the legend and format the percentage values in the table with one decimal place. The following code shows the modifications described above:

```

proc format;
  picture pctfmt (round) other="009.9%";
run;

proc template;
  define crosstabs Base.Freq.CrossTabFreqs;
  cellvalue frequency percent;
  define frequency;
    format=8.;
    header="Count";
  end;
  define percent;
    format=pctfmt.;
    header="Overall %";
  end;
  define header tableof;
    text "Table of " _row_label_ " by " _col_label_;
  end;
  define header rowsheader;
    text " ";
  end;
  define header colsheader;
    text " ";
  end;
  cols_header=colsheader;
  rows_header=rowsheader;
  header tableof;
end;
run;

proc freq data=sashelp.class;
  tables age*sex / norow nocol;
  where age le 13;
  label age="Years" sex="Gender";
run;

```

Output 5 is the resulting output.

Table of Years by Gender			
Count			
Overall %	F	M	Total
11	1 10.0%	1 10.0%	2 20.0%
12	2 20.0%	3 30.0%	5 50.0%
13	2 20.0%	1 10.0%	3 30.0%
Total	5 50.0%	5 50.0%	10 100.0%

### Output 5. Modifying Headings and Formatting Statistics

#### SUPPRESS THE LEGEND AND FREQUENCY MISSING ROW

A multi-way table includes a legend with the statistic names that are included in the table. Missing variable values are not included in the table output by default unless you specify the MISSING option in the TABLES statement in PROC FREQ. When these values are not part of the table, a line below the table displays, **Frequency Missing=*n***, where *n* is the number of missing values. In PROC TEMPLATE, the TableOf element suppresses the legend and the variable headings, and the Missing element suppresses the bottom row. To make these modifications, use the code below:

```
proc template;
  define crosstabs Base.Freq.CrossTabFreqs;
    header TableOf ;
    footer Missing;

  define TableOf;
    dynamic StratNum NoTitle;
    text "Table " StratNum 10. " of " _ROW_NAME_ " by " _COL_NAME_ /
      (NoTitle=0) and (StratNum>0);
    text "Table of " _ROW_NAME_ " by " _COL_NAME_ / NoTitle=0;
  end;
  define Missing;
    text " ";
  end;
end;

run;

data test;
input x y;
datalines;
1 .
1 1
1 2
```

```

2 .
2 1
2 2
;
run;

proc freq data=test;
  tables x*y / norow nocol nopercnt;
run;

```

Output 6 is the resulting output.

Table of x by y			
	1	2	Total
1	1	1	2
2	1	1	2
Total	2	2	4

### Output 6. Suppressing the Legend and Frequency Missing Row

## GRAPHS WITH ODS STATISTICAL GRAPHICS

Often a visual illustration of your data is a helpful addition or alternative to a table. The PLOTS option in the TABLES statement in PROC FREQ enables you to create a graphical representation of your tabular output. Here is the syntax for the PLOTS option:

PLOTS < (*global-plot-options*)> <=*plot-request* <(*plot-options*)>>

Plot requests identify the plots that are available, and plot options control the appearance and content of the plots. A global plot option applies to all plots; a plot option after a plot request applies to a specific plot. Some of the commonly requested plots are included in Table 1 below.

Plot Request	Required TABLES Statement Option
ALL	None
CUMFREQPLOT	One-way table request
DEVIATIONPLOT	CHISQ (one-way table)
FREQPLOT	Any table request
MOSAICPLOT	Two-way or multi-way table request
ODDSRATIOPLOT	MEASURES, OR, or RELRISK

**Table 1. Common Plot Requests**

See [Table 40.9: TABLES Statement Options](#) in “The FREQ Procedure” section of the SAS/STAT® 14.1 User’s Guide for a complete list of plot requests. The documentation also includes a table after each plot request that lists the plot options that specifically apply to that plot.



The following statement must be enabled for the graphs to be generated. This option is turned on by default beginning in SAS® 9.3 when running SAS in SAS® Display Manager:

```
ODS GRAPHICS ON;
```

The default behavior if you do not specify PLOTS= in the TABLES statement is to produce all plots that are associated with the analyses that you request, except for frequency, cumulative frequency, and mosaic plots. You must explicitly request these plots. The following values also are available:

- PLOTS=NONE: Suppresses all plots
- PLOTS(ONLY): Suppresses the default plots and requests only the plots that are specified as plot requests
- PLOTS=ALL: Requests all plots that are associated with the specified analyses

**Note:** The PLOTS= option has no effect if you use the NOPRINT option in PROC FREQ or the TABLES statement.

### CREATE ONE-WAY PLOTS

One-way plots show the data in graphical form for any one-way table request. These plots include frequency plots, cumulative frequency plots, and deviation plots. The examples below illustrate two common types of plots.

#### Cumulative Frequency Plot

The CUMFREQPLOT request creates a plot of cumulative frequencies. This plot request must be specified in the PLOTS= option because this is one of the plots that is not created by default. You can also use PLOTS=ALL to create it. The following plot options are available for a cumulative frequency plot:

- ORIENT=HORIZONTAL|VERTICAL\*
- SCALE=FREQ\*|PERCENT
- TYPE=BARCHART\*|DOTPLOT

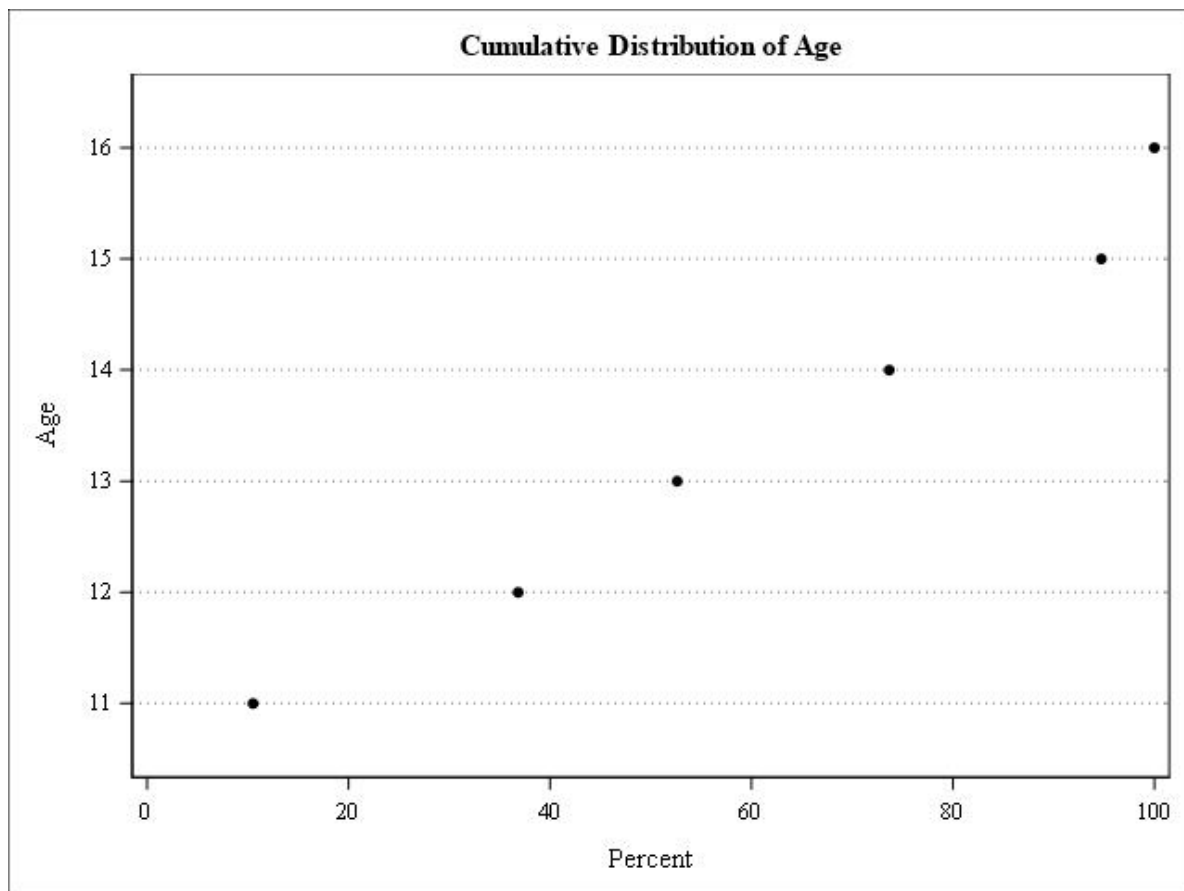
**Note:** For all plot options listed in this paper, an asterisk (\*) denotes the default.

The following code shows how to include plot options in the syntax and creates a dot plot of the cumulative percentage for the Age variable:

```
ods graphics on;

proc freq data=sashelp.class;
    tables age /plots=cumfreqplot(scale=percent type=dotplot);
run;
```

Output 7 is the resulting output.



### Output 7. Cumulative Frequency Plot

#### Deviation Plot

The DEVIATIONPLOT request creates a plot of relative deviations from expected frequencies. With this type of plot, you must specify the CHISQ option in the TABLES statement. The following plot options are available for a deviation plot:

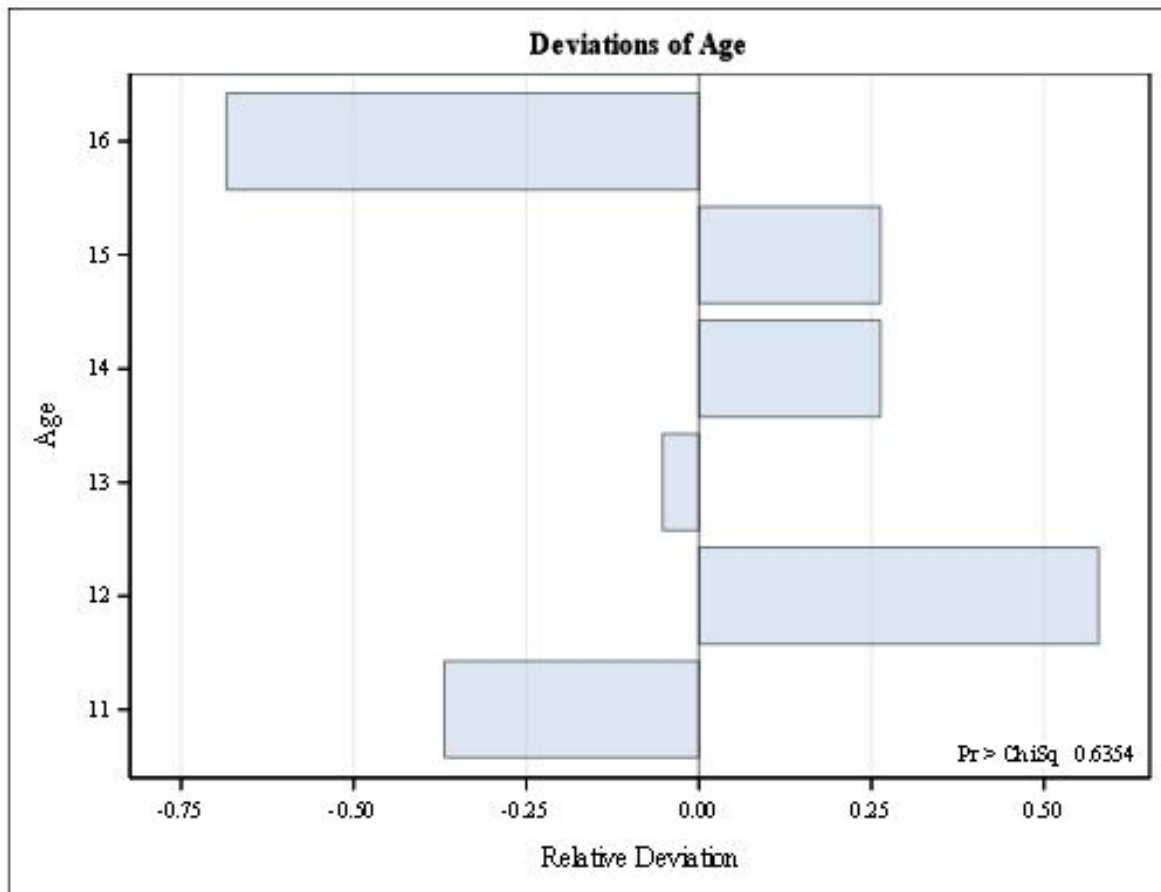
- NOSTAT
- ORIENT=HORIZONTAL|VERTICAL\*
- TYPE=BARCHART\*|DOTPLOT

The following code creates a horizontal bar chart of the relative deviation for the Age variable:

```
ods graphics on;

proc freq data=sashelp.class;
    tables age /chisq plots=deviationplot(orient=horizontal);
run;
```

Output 8 is the resulting output.



### Output 8. Deviation Plot

#### CREATE MULTI-WAY PLOTS

Multi-way plots show the data in graphical form for any  $n$ -way table request. These plots include statistical graphics such as agreement plots, kappa plots, and odds ratio plots. The examples below illustrate three common types of plots.

#### Frequency Plot

The FREQPLOT request creates a frequency plot. You must specify this plot request, or PLOTS=ALL, in the PLOTS= option. Some of the plot options that are available for a frequency plot include the following:

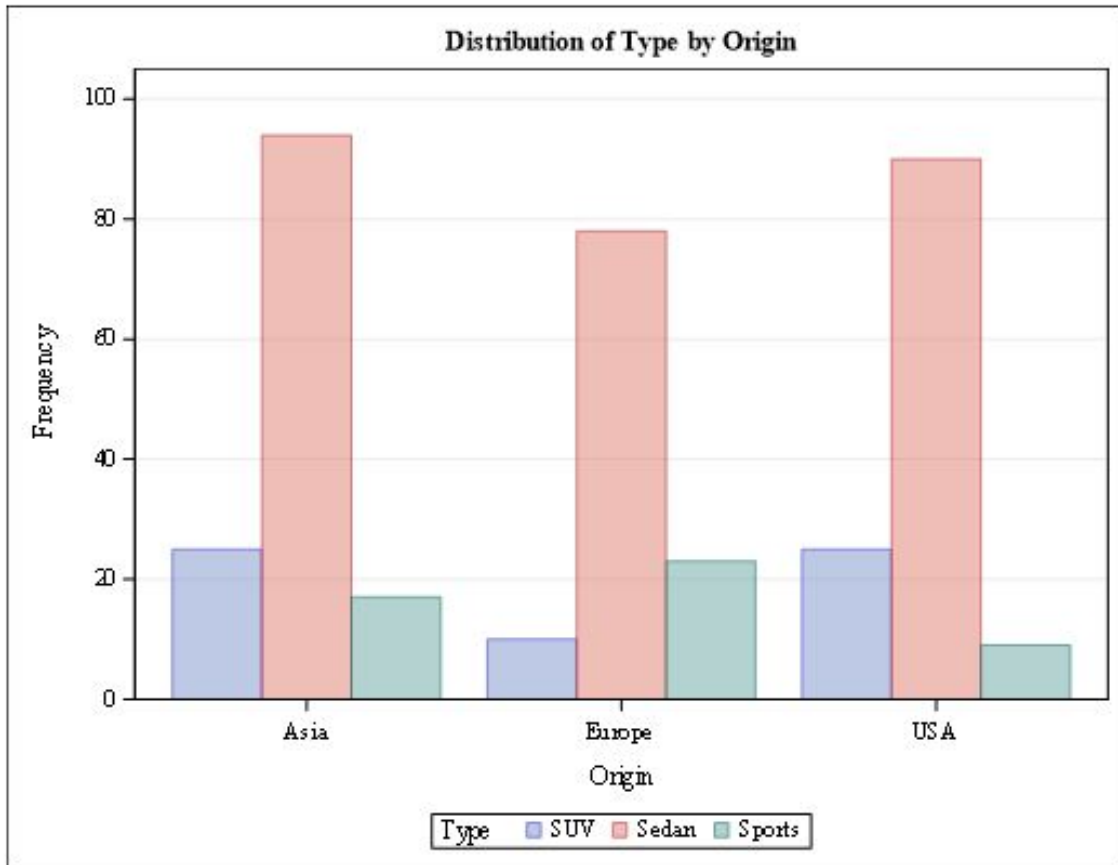
- GROUPBY=COLUMN\*|ROW
- SCALE=FREQ\*|GROUPPERCENT|LOG|PERCENT|SQRT
- TWOWAY=CLUSTER|GROUPHORIZONTAL|GROUPVERTICAL\*|STACKED

The following code creates a clustered frequency bar chart for Type by Origin:

```
ods graphics on;

proc freq data=sashelp.cars;
  tables origin*type /plots=freqplot(groupby=row twoway=cluster );
  where type in ("Sedan","SUV","Sports");
run;
```

Output 9 is the resulting output.



### Output 9. Frequency Plot

#### Mosaic Plot

The MOSAICPLOT request creates a mosaic plot that displays tiles that correspond to the crosstabulation-table cells and are proportional to the relative frequencies. In a mosaic plot, the column variable is displayed on the X axis, and the row variable is displayed on the Y axis. By default, the colors of the tiles correspond to the levels of the row variable unless you use the COLORSTAT= option. You must specify this plot request, or PLOTS=ALL, in the PLOTS= option. The following plot options are available for a mosaic plot:

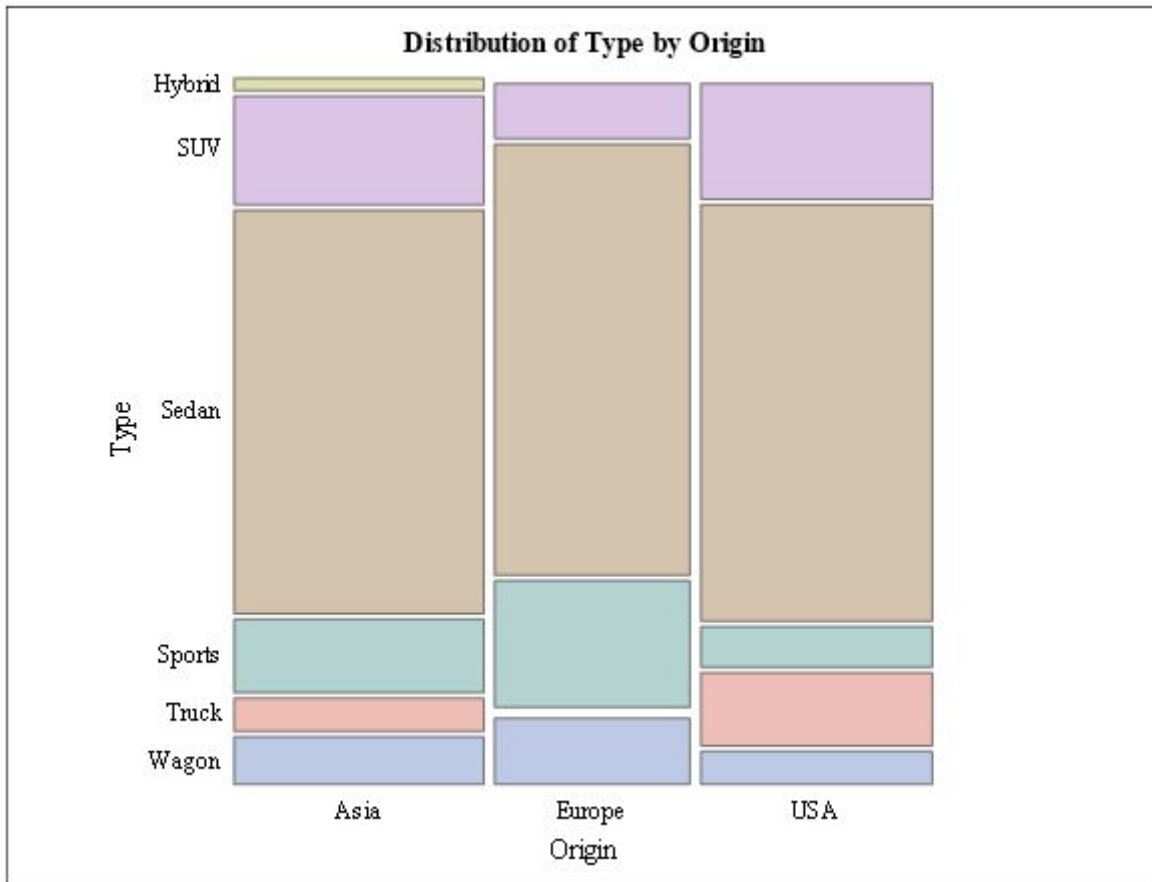
- COLORSTAT=PEARSONRES|STDRES
- SQUARE

The following code creates a mosaic plot for Type by Origin:

```
ods graphics on;

proc freq data=sashelp.cars;
  tables type*origin /plots=mosaicplot(square);
run;
```

Output 10 is the resulting output.



## Output 10. Mosaic Plot

### Odds Ratio Plot

The ODDSRATIO PLOT request creates an odds ratio plot with confidence limits when you specify the MEASURES, OR, or RELRISK option in the TABLES statement. Some of the plot options that are available for an odds ratio plot include the following:

- CLDISPLAY=BAR|LINE|LINEARROW|SERIF\*|SERIFARROW
- COLUMN=1\*|2
- LOGBASE=2|E|10
- ORDER=ASCENDING|DESCENDING

The following code creates an odds ratio plot for Treatment by Response, controlling for Gender:

```
ods graphics on;

data Migraine;
  input Gender $ Treatment $ Response $ Count @@;
  datalines;
  Female Active Better 16   Female Active Same 11
  Female Placebo Better 5   Female Placebo Same 20
  Male Active Better 12    Male Active Same 16
  Male Placebo Better 7    Male Placebo Same 19
```

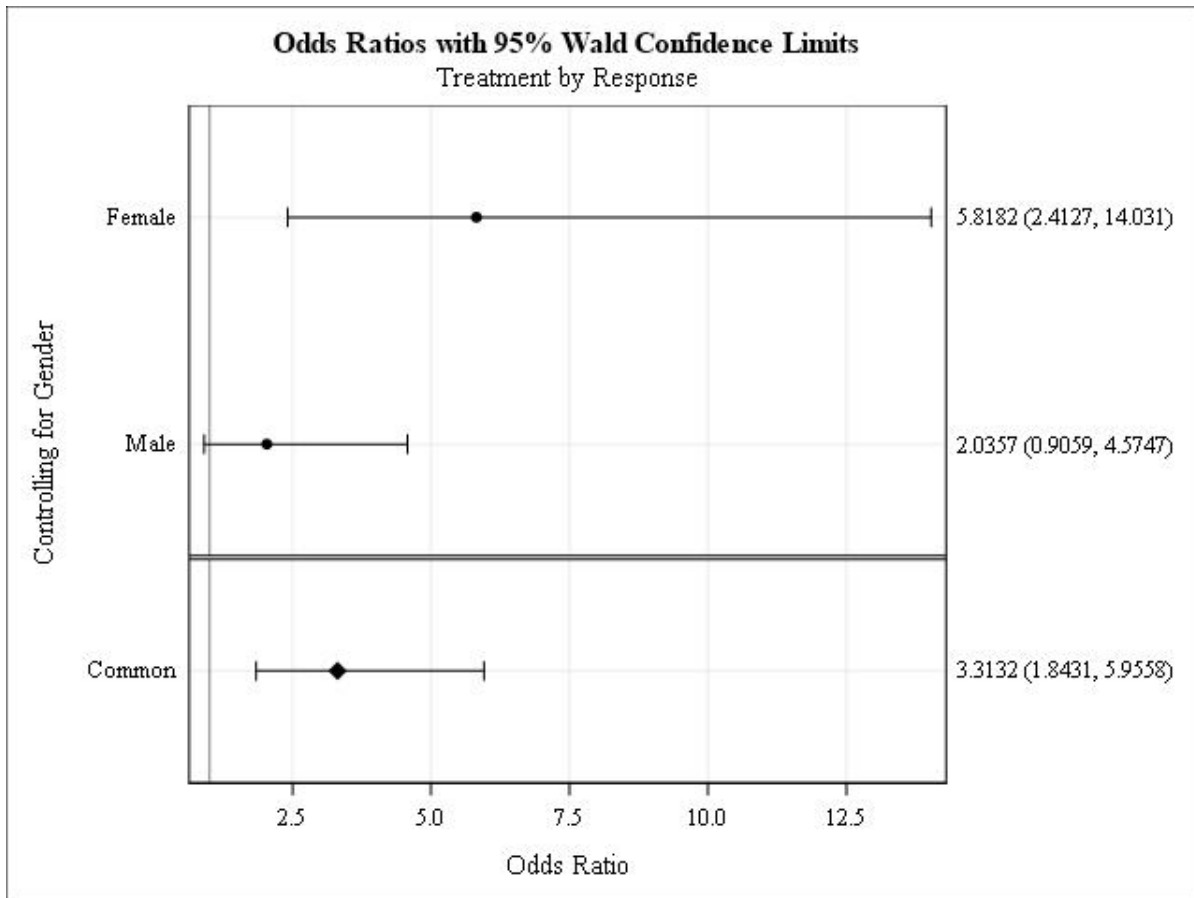
```

Female Active Better 16   Female Active Same 11
Female Placebo Better 5   Female Placebo Same 20
Male Active Better 12    Male Active Same 16
Male Placebo Better 7    Male Placebo Same 19
;
run;

proc freq data=Migraine;
  tables Gender*Treatment*Response /
    relrisk plots(only)=(oddsratioplot(stats)) cmh ;
  weight Count;
run;

```

Output 11 is the resulting output.



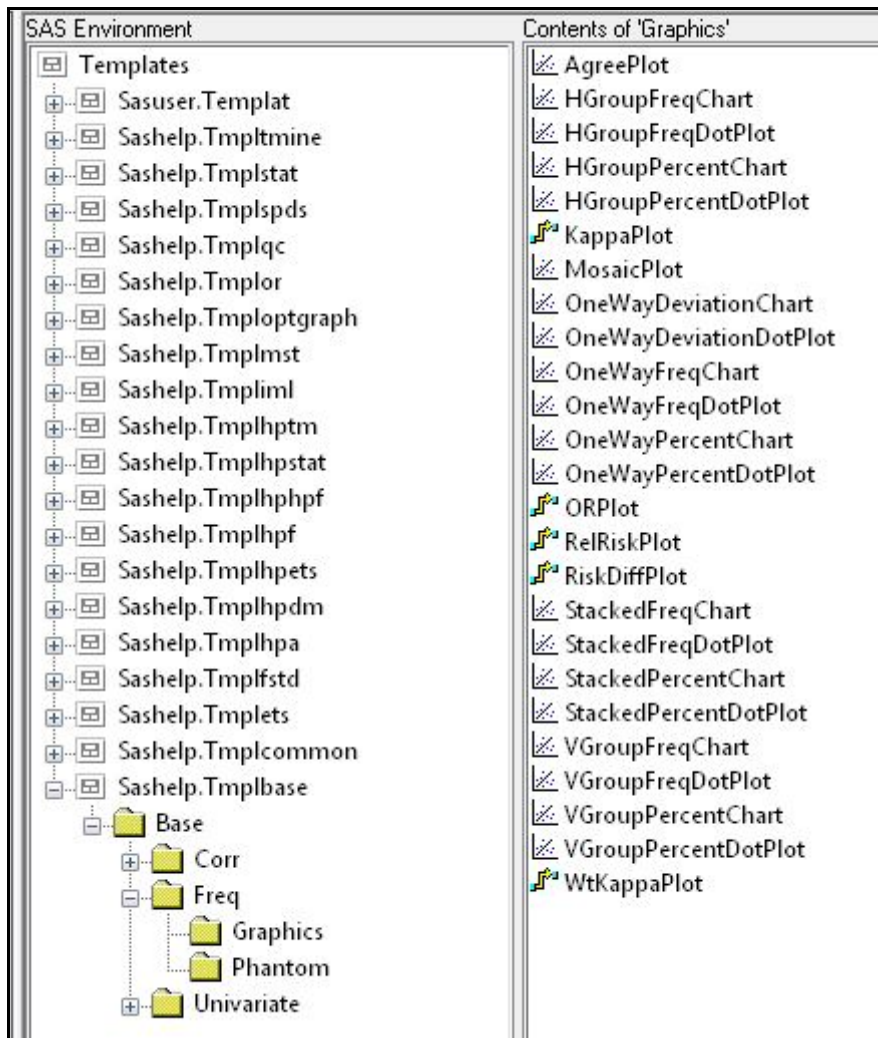
**Output 11. Odds Ratio Plot**

### MODIFY GRAPH OUTPUT

The previous examples showed some of the default graphs that are generated by the plot requests, using a few options to control the orientation, the type of graph, or the scale. Suppose you want to control colors, axis labels, or other parts of the graph. You can use PROC TEMPLATE and GTL to assist you in customizing the appearance of the output. To see the names of the graphs that PROC FREQ can create, complete these steps in your SAS session:

1. Right-click **Results** in the Results window.
2. Select **Templates**.

3. Expand the nodes as shown in Display 1 below.



**Display 1. Graph Template Item Stores**

You can select any of the graph names to see the underlying code, or you can run the following code to display the results in the log:

```
proc template;
    source Base.Freq.Graphics.<graph-template-name>;
run;
```

### **FREQPLOT Color**

When you create a graph of a multi-way table, you choose the colors of the bars using PROC TEMPLATE to define a style, and use the GraphData<n> style element to define the colors. The following code demonstrates how to assign colors and apply them to the output using the STYLE= option in the ODS statement:

```
ods graphics on;

proc template;
    define style colors;
        parent=styles.rtf;
```

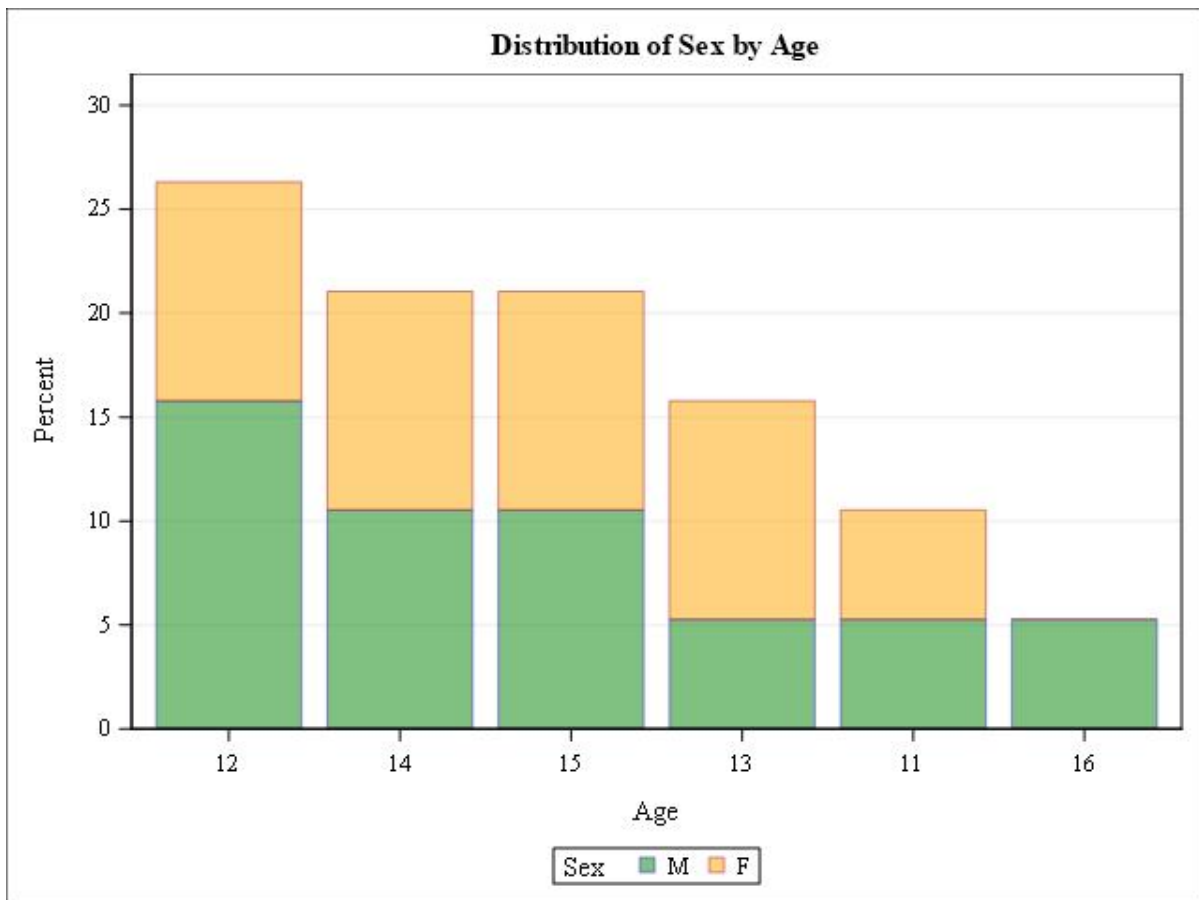
```

class GraphData1 /
  color=green;
class GraphData2 /
  color=orange;
end;
run;

ods rtf file="filename.rtf" style=colors;
proc freq data=sashelp.class order=freq;
  tables sex*age /plots= freqplot(twoway=stacked scale=percent);
run;
ods rtf close;

```

Output 12 is the resulting output.



### Output 12. Adding Color to FREQPLOT

#### USE GRAPH TEMPLATE LANGUAGE (GTL)

You can modify the predefined graph templates shown in Display 1 above using GTL. A graph template begins with a DEFINE STATGRAPH statement. Additional statements enable you to customize the graph to get the desired output.

#### Color Scheme

Suppose you are creating a series of graphs from a multi-way table, and you want to maintain the color scheme of the bars throughout all the graphs. By default, the color scheme changes when a certain category does not exist for one of the graphs. You can control this behavior with the



DISCRETEATTRMAP block and the DISCRETEATTRVAR statement in the PROC TEMPLATE step. The SEGMENTLABELTYPE=AUTO option in the BARCHARTPARM statement displays the values within the bars, and the SEGMENTLABELFORMAT controls the format of the value. The following sample code illustrates these modifications:

```
ods graphics on;

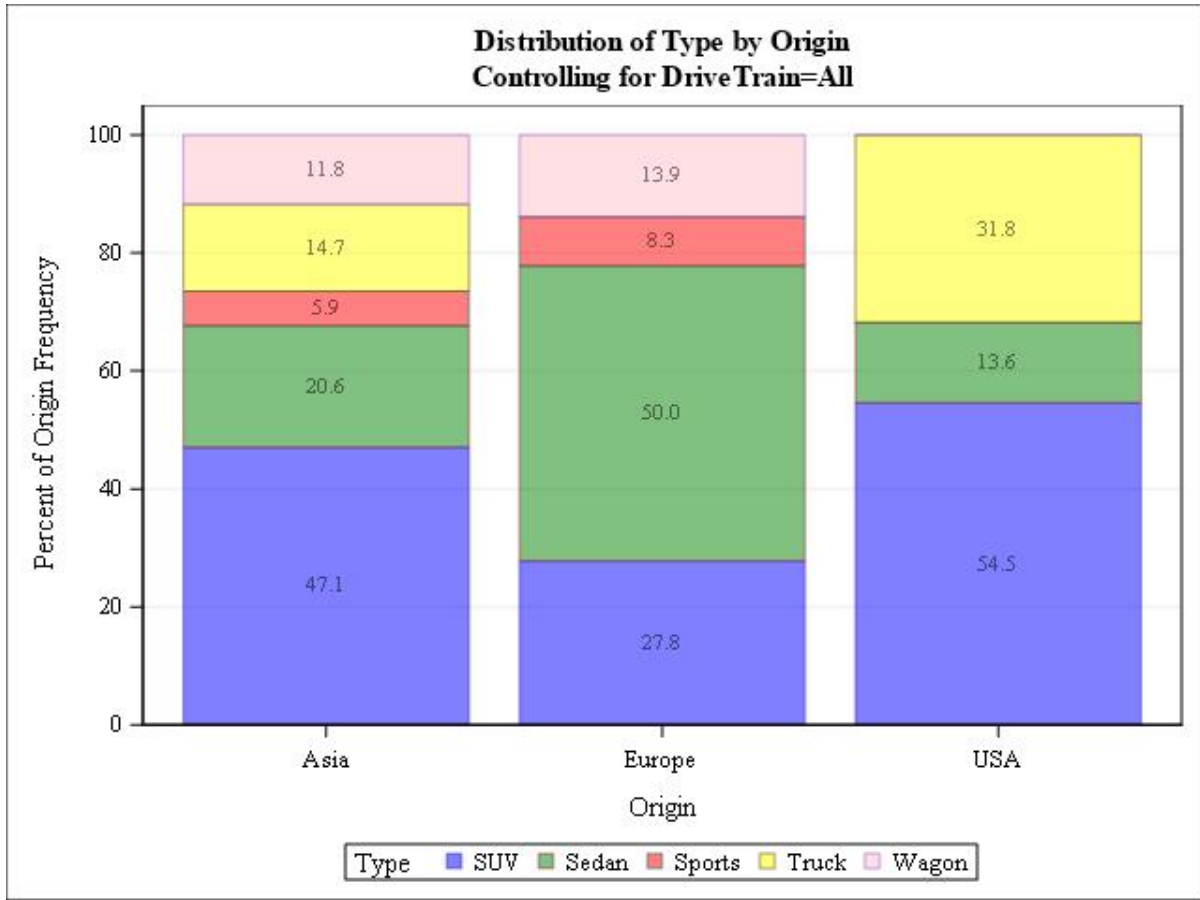
proc template;
  define statgraph Base.Freq.Graphics.StackedPercentChart;
    /* more code here */
    discreteattrmap name="colors" / ignorecase=true;
    value "Hybrid" / fillattrs=(color=orange );
    value "SUV" / fillattrs=(color=blue);
    value "Sedan" / fillattrs=(color=green );
    value "Sports" / fillattrs=(color=red);
    value "Truck" / fillattrs=(color=yellow);
    value "Wagon" / fillattrs=(color=pink);
  enddiscreteattrmap;
  discreteattrvar attrvar=_row var=_row attrmap="colors";
  /* more code here */
  barchartparm x=_COLVAR y=_PLOTVAR / segmentlabelformat=4.1
  segmentlabeltype=auto group=_row name="Rows"
  groupdisplay=_CLUSTER datatransparency=.5 rolename=(tip4=_FREQ)
  tip=(group x y tip4);

  /* more code here */
  end;
run;

proc freq data=sashelp.cars;
  tables drivetrain*type*origin /plots=freqplot(twoway=stacked
  scale=grouppct);
run;
```

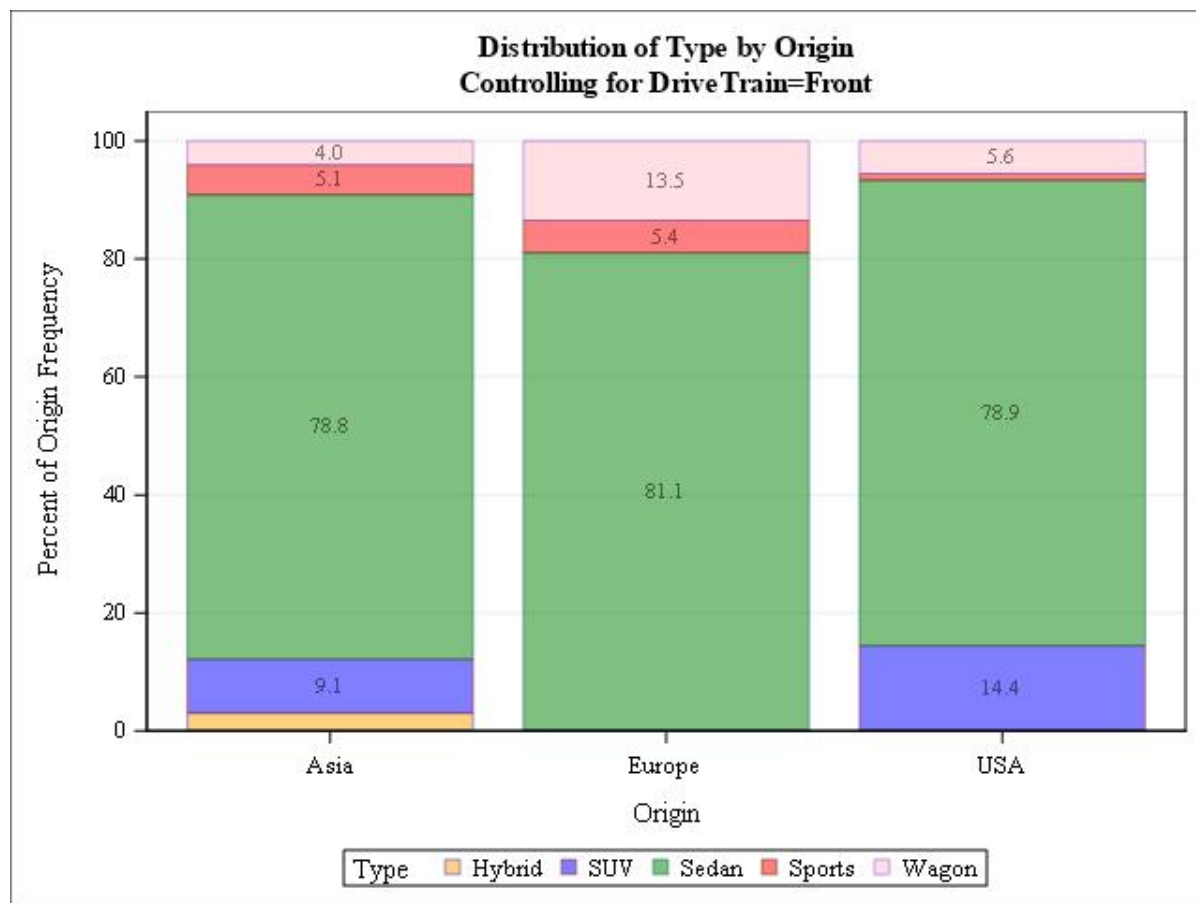
Note that the values are case sensitive.

Output 13 is the resulting output.



**Output 13. Using GTL to Show How Colors Are Maintained over Multiple Graphs**

Output 14 below shows how the color scheme is maintained even though different levels of the Type variable are represented.



**Output 14. Using GTL to Show How Colors Are Maintained over Multiple Graphs**

### Titles, Axis Labels, and Bar Labels

In the graph output, you might want to add a descriptive title and labels for each axis and add numbers to the bars to make them more readable. You can use GTL to do this, as shown below.

**Note:** Numbered circles included in the code correspond to a discussion after the code.

```
ods graphics on;

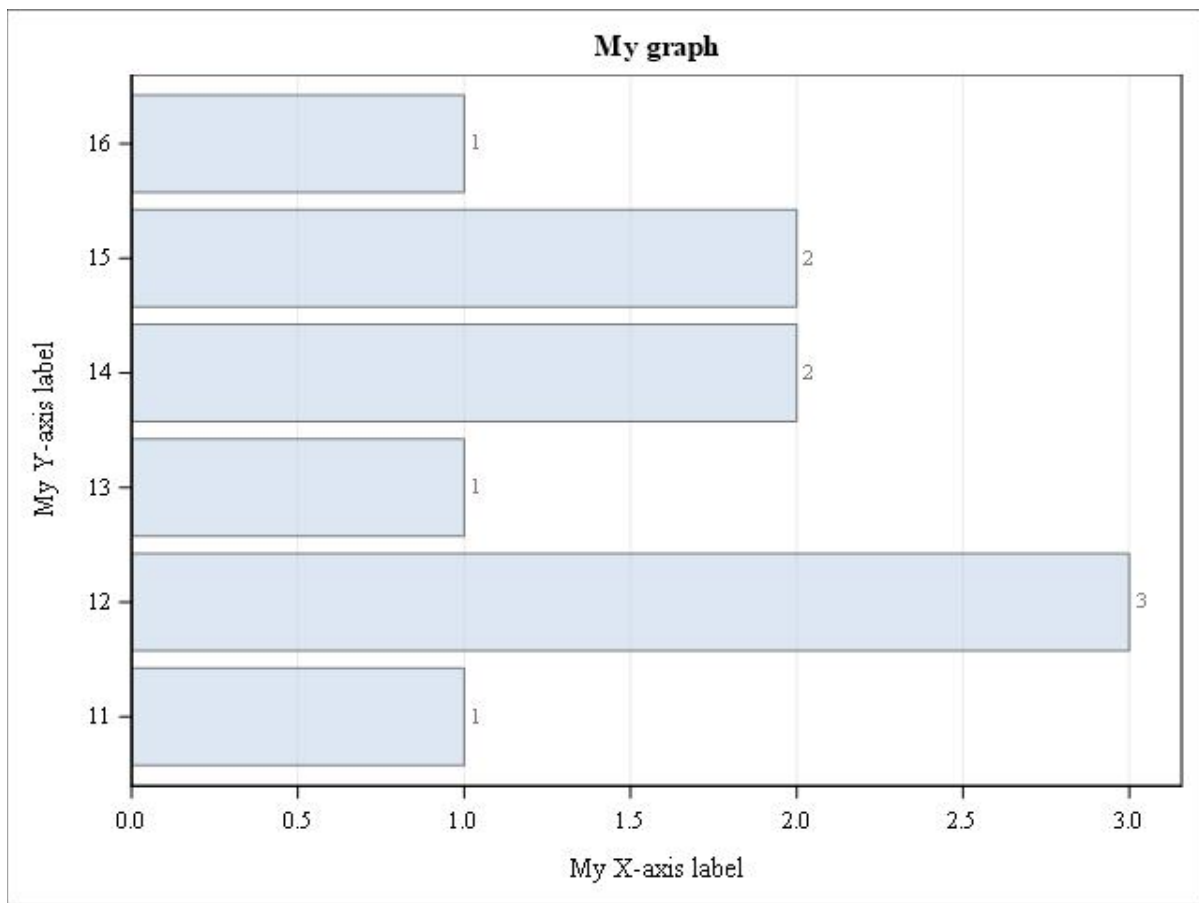
proc template;
  define statgraph Base.Freq.Graphics.OneWayFreqChart;
    /* more code here */
    entrytitle "My graph"; ❶
    if (_ORIENT)
      layout overlay / xaxisopts=(label="My X-axis label" shortlabel="My X-
axis label" ❷
      griddisplay=auto_on offsetmin=0 offsetmax=.05
      linearopts=(integer=0 viewmin=0 thresholdmax=_TMAX)) yaxisopts=
(label="My Y-axis label" shortlabel="My Y-axis label");
      barchartparm x=_LEVEL y=_FREQVAR / orient=horizontal
      datatransparency=.5 rolename=(tip3=_TIPVAR tip4=_TIPVAR2)
      tip=(x y tip3 tip4) datalabel=_freqvar; ❸ /** number at end of bar
  **/
  endlayout;
  /* more code here*/
end;
```

```
run;

proc freq data=sashelp.class;
  where sex="M";
  tables age / plots=freqplot;
run;
```

- ❶ The ENTRYTITLE statement in PROC TEMPLATE controls the title text.
- ❷ The LABEL option within the YAXISOPTS/XAXISOPTS options in the LAYOUT statement modifies the axis labels.
- ❸ The DATALABEL option in the BARCHARTPARM statement enables values to be added to the end of the bars.

Output 15 is the resulting output.



**Output 15. Changing Titles and Labels**

## CONCLUSION

PROC FREQ's ability to generate high-level statistics that other Base SAS® procedures cannot produce, paired with its ability to represent data both in tabular and graphical form, make this procedure a useful resource to all levels of SAS users. Through the use of PROC TEMPLATE and GTL, this paper provided you with the tools to modify PROC FREQ output to meet your needs. It included examples of customizing the traditional table output with labels, formats, and colors as well as introducing the graph capabilities in PROC FREQ and showing how these graphs can be enhanced.

## REFERENCES

SAS Institute Inc. 2016. "The FREQ Procedure." SAS/STAT® 14.1 User's Guide. Cary, NC: SAS Institute Inc. Available at [support.sas.com/documentation/cdl/en/statug/68162/HTML/default/viewer.htm#statug\\_freq\\_syntax08.htm](http://support.sas.com/documentation/cdl/en/statug/68162/HTML/default/viewer.htm#statug_freq_syntax08.htm)

McLawnhorn, Kathryn. 2016. "Seeing the FREQ procedure's one-way tables in a new light." *SAS Users Blog*. Available at [blogs.sas.com/content/sgf/2016/05/20/seeing-the-freq-procedures-one-way-tables-in-a-new-light/](http://blogs.sas.com/content/sgf/2016/05/20/seeing-the-freq-procedures-one-way-tables-in-a-new-light/)

SAS Institute Inc. 2009. "Customizing FREQ Procedure Output in SAS® 9.2." Cary, NC: SAS Institute Inc. Available at [support.sas.com/resources/papers/freq92.pdf](http://support.sas.com/resources/papers/freq92.pdf)

## ACKNOWLEDGMENTS

The author acknowledges Rayna Rowell, Bari Lawhorn, Jerry Leonard, and Lelia McConnell from SAS for their time and efforts to make this paper better for our customers.

## RECOMMENDED READING

- *Base SAS® Procedures Guide: Statistical Procedures*
- *Base SAS® Output Delivery System (ODS) Graphics Suite*

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Kathryn McLawnhorn  
SAS Institute, Inc.  
SAS Campus Drive  
Cary, NC 27513  
Email: [support@sas.com](mailto:support@sas.com)  
Web: [support.sas.com](http://support.sas.com)

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.