

## Paper RIV-11

**How to Replicate Excel Stacked Area Graphs in SAS®**

Sharon Avrunin-Becker, Westat, Rockville, MD

**ABSTRACT**

Have you ever been given a graph and asked to reproduce the same thing using SAS? As you stare at the graph you realize you have no idea what to call this type of graph to even begin to try to figure out how to program it. Your helpful SAS programming coworkers have no idea but suggest you try looking for some kind of stacked bar chart. As you start desperately looking through all types of SAS documentation and conference papers you finally realize the solution may reside within PROC Gplot. This paper walks you through the steps to eliminate the original panic of determining the graph type and then take you through the steps of creating a very close replica of an Excel Stacked Area Plot.

**INTRODUCTION**

The local ice cream store wants to figure out how many ice cream cones of their 3 top flavors they are selling a week. They have been tracking this themselves, but now they are ready for you to help them out. They hand you an Excel file with the numbers they have gathered so far and a printout of the graph they want you to replicate. The graph looks like this and you have no idea what to even begin to Google for to be able to recreate it in SAS.

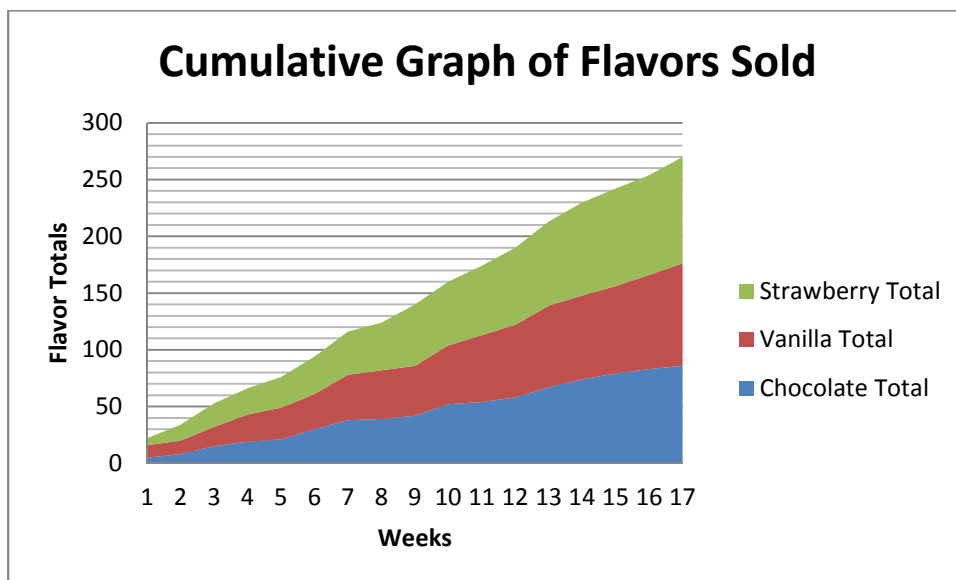


Figure 1. Example of an Excel Stacked Area Plot

**IDENTIFY GRAPH**

The first step before you can begin to program this in SAS is to understand the type of graph you need to reproduce. Most non SAS programmer types usually turn to Excel when they are trying to do it themselves and you know this is where the data resides. Therefore, you need to open Excel and look through their list of graphs and charts and determine which one most closely resembles the one you are trying to recreate. Once you have done this you realize that the graph in Figure 1 looks like a Stacked Area Plot. Now you can turn to SAS and resolve this issue programmatically.

## THE STEPS IN EXCEL...

Open Excel and click on the Insert tab. Next click on Other Charts in the Charts section. Then click on All Chart Types at the bottom.

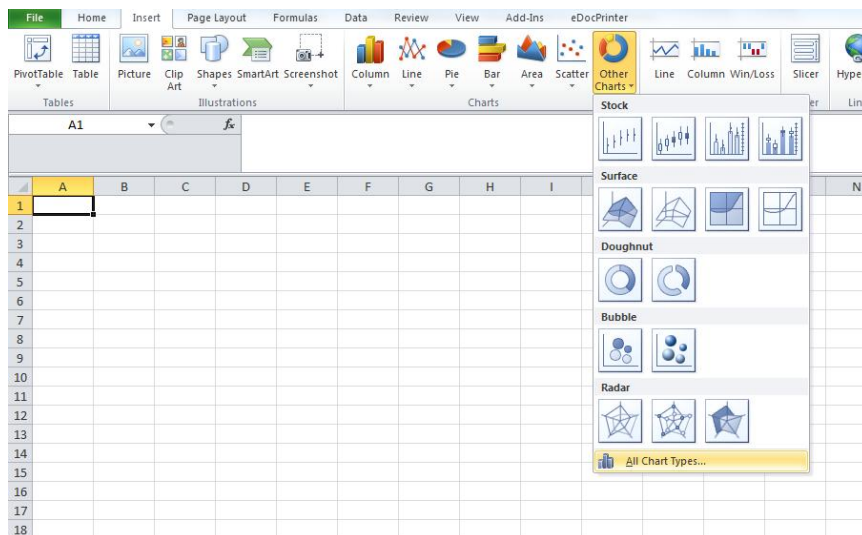


Figure 2a. Steps 1-4 in Excel to identify chart

Once the insert chart table is open, scroll through until you find the graph that most closely resembles what you are looking for.

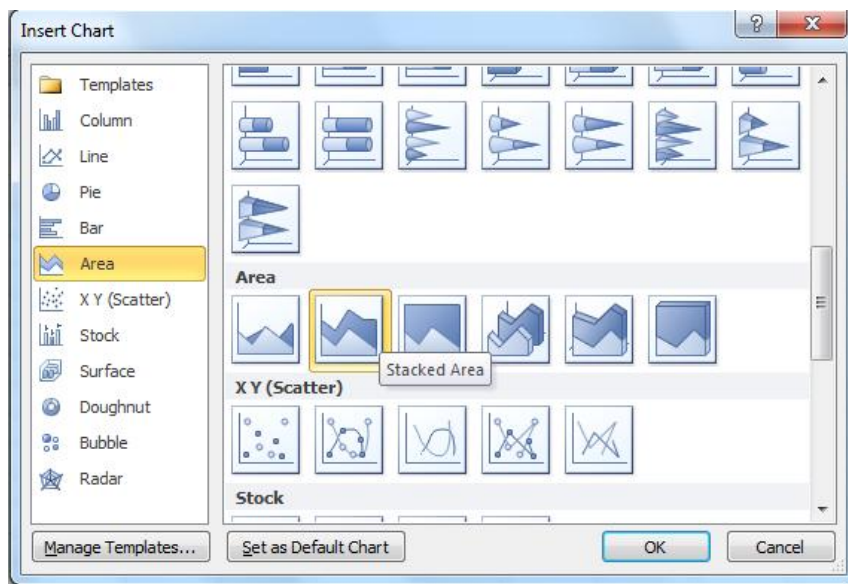


Figure 2b. Step 5 in Excel to identify chart

## IMPORT DATA

The next step is to get your data into SAS. Save yourself time and import the Excel data into SAS. The Import Wizard can help you out or just use a quick Proc Import as shown below.

```
PROC IMPORT OUT= WORK.X
            DATAFILE= "H:\MY DOCUMENTS\SESUG 2013\Sample.xlsx"
            DBMS=EXCEL REPLACE;
            RANGE="Sheet1$";
            GETNAMES=YES;
            MIXED=NO;
```

```

SCANTEXT=YES;
USEDATE=YES;
SCANTIME=YES;
RUN;

```

### USE PROC GPLOT

Using PROC GPLOT with the Overlay option and specifying 3 areas gives you a graph that is almost identical to the original Excel graph. You also need to set your symbol statement to INTERPOL=JOIN.

#### THE PROC GPLOT WITH OVERLAY CODE...

```

SYMBOL1 INTERPOL=JOIN;

PROC GPLOT DATA=X;
PLOT CHOCOLATE_TOTAL*WEEK VANILLA_TOTAL*WEEK STRAWBERRY_TOTAL*WEEK /
OVERLAY AREAS=3 AUTOVREF CVREF=GRAY VAXIS=AXIS1 HAXIS=AXIS2;
RUN;

QUIT;

```

#### HERE IS THE OUTPUT...

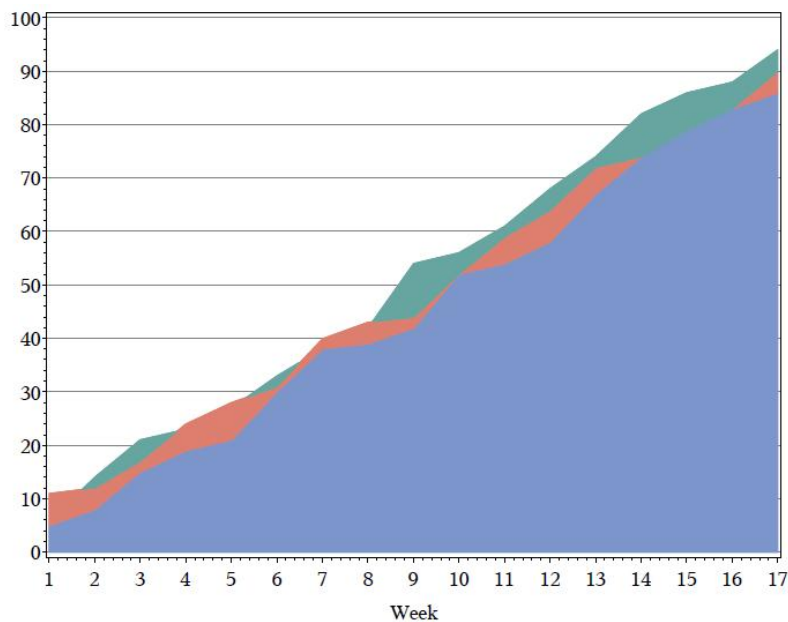


Figure 3. Output from PROC GPLOT

## SPECIFYING COLORS

If you would like to switch your colors you can do this by adding in a pattern statement. The v in the code below is the value statement specifying you want a solid pattern.

### THE COLORS ARE SPECIFIED...

```
PATTERN1 V=S COLOR=PURPLE;
PATTERN2 V=S COLOR=PINK;
PATTERN3 V=S COLOR=YELLOW;
```

### HERE IS THE OUTPUT...

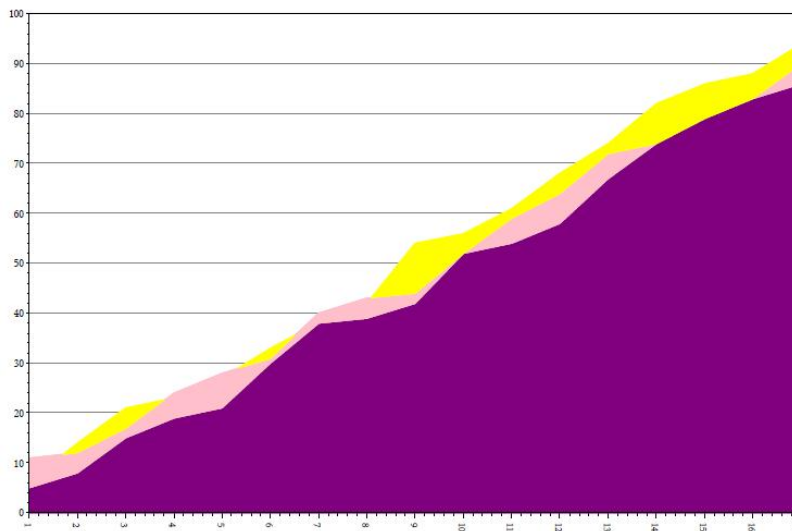


Figure 4. Output from PROC GPLOT with color specified

## ADDING IN A LEGEND

As nice as this table looks, you have no idea what these colors and mountains represent. Now is the time to add in a legend. The legend has lots of options associated with it. You can assign where on the graph you want to display your legend, you can specify the labels on the values, and you can also have a frame around your legend. You can even add in macro variables to display totals. Don't forget that the legend statement needs to be added to your PROC GPLOT code for it to be displayed.

### THE SIMPLE LEGEND SPECIFICS...

```
LEGEND1 LABEL=(HEIGHT=2 POSITION=TOP JUSTIFY=CENTER "THE MAIN FLAVORS")
          VALUE=(HEIGHT=1.5 "CHOCOLATE" "VANILLA" "STRAWBERRY")
          ACROSS=1 DOWN=3 POSITION=(TOP LEFT INSIDE) FRAME
MODE=PROTECT;
```

```
PROC GPLOT DATA=X;
PLOT CHOCOLATE_TOTAL*WEEK VANILLA_TOTAL*WEEK STRAWBERRY_TOTAL*WEEK/
OVERLAY AREAS=3 AUTOVREF CVREF=GRAY VAXIS=AXIS1 HAXIS=AXIS2
LEGEND=LEGEND1;
RUN;
```

HERE IS THE OUTPUT...

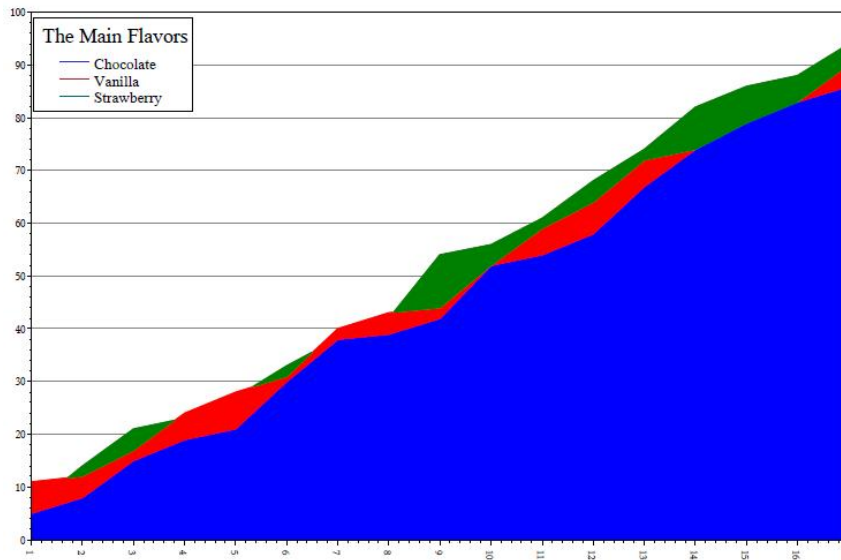


Figure 5. Output from PROC GPLOT with legend

#### ADDING IN MACRO VARIABLES...

```
DATA X(KEEP=WEEK CHOCOLATE_TOTAL VANILLA_TOTAL STRAWBERRY_TOTAL NPC);
  SET X;
  NPC = CHOCOLATE_TOTAL + VANILLA_TOTAL + STRAWBERRY_TOTAL;
  CALL SYMPUTX('GTOTC',CHOCOLATE_TOTAL);
  CALL SYMPUTX('GTOTV',VANILLA_TOTAL);
  CALL SYMPUTX('GTOTS',STRAWBERRY_TOTAL);
  CALL SYMPUTX('GTOTALL',NPC);
RUN;
```

```
LEGEND1 LABEL=(HEIGHT=2 POSITION=TOP JUSTIFY=CENTER "THE MAIN FLAVORS
(N=&GTOTALL.)")
  VALUE=(HEIGHT=1.5 "CHOCOLATE (N=&GTOTC.)" "VANILLA
(N=&GTOTV.)" "STRAWBERRY (N=&GTOTS.)")
  ACROSS=1 DOWN=3 POSITION = (TOP LEFT INSIDE) FRAME
  MODE=PROTECT;
```

HERE IS THE OUTPUT...

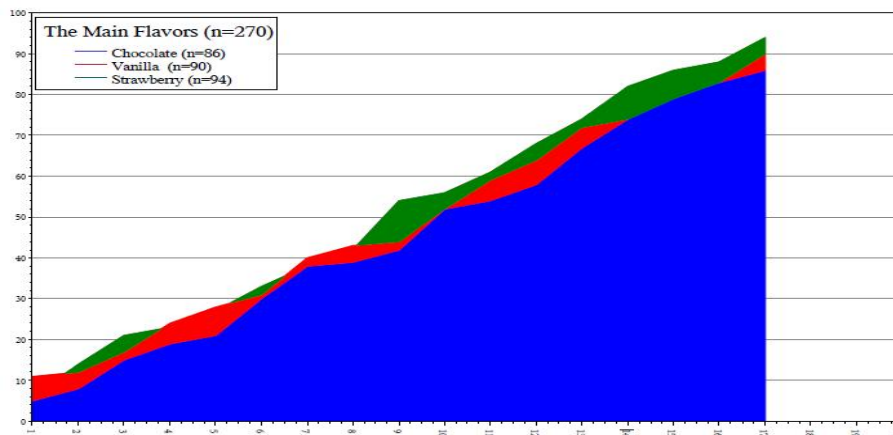


Figure 6. Output from PROC Gplot with totals specified

### ADDING IN LABELS AND TITLES

The axis statements let you add in labels for your X and Y axis. You can specify the start and end numbers, you can rotate your labels, and you can specify how often to place a tic mark. A title is added as you would any SAS title. Don't forget to reference the axis in your PROC Gplot statement.

#### THE AXIS STATEMENTS...

```

AXIS1 LABEL=(A=90 'Flavor Totals') OFFSET=(0,0) ORDER=(0 TO 100 BY 10)
        VALUE=(H=1 );
AXIS2 LABEL=( 'Weeks' ) OFFSET=(0,0) ORDER=(1 TO 20 BY DAY) VALUE=(H=1
        A=270 );

```

```

PROC Gplot DATA=X;
PLOT CHOCOLATE_TOTAL*WEEK VANILLA_TOTAL*WEEK STRAWBERRY_TOTAL*WEEK/
OVERLAY AREAS=3 AUTOVREF CVREF=GRAY VAXIS=AXIS1 HAXIS=AXIS2
LEGEND=LEGEND1;
RUN;

```

HERE IS THE OUTPUT...

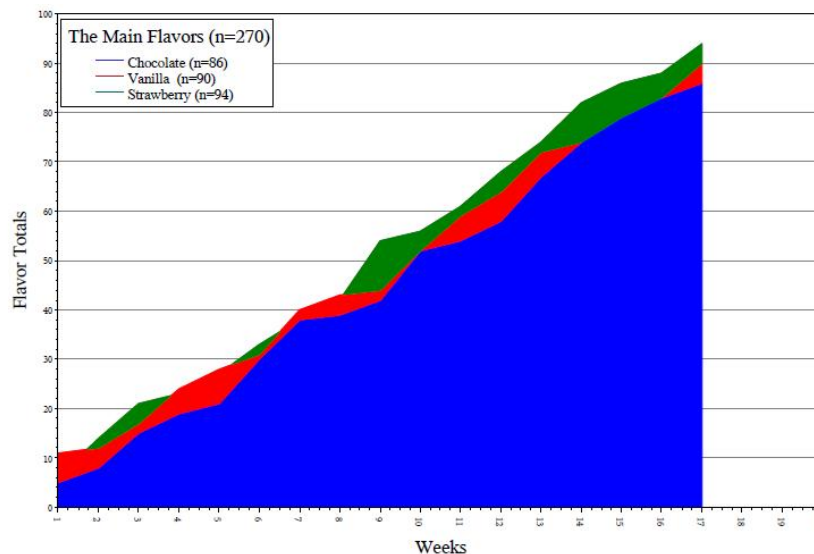


Figure 7. Output from PROC Gplot with axis defined

THE TITLE STATEMENT...

`TITLE1 H=3 J=C FONT=TIMES BOLD "CUMULATIVE GRAPH OF FLAVORS SOLD";`

HERE IS THE OUTPUT...

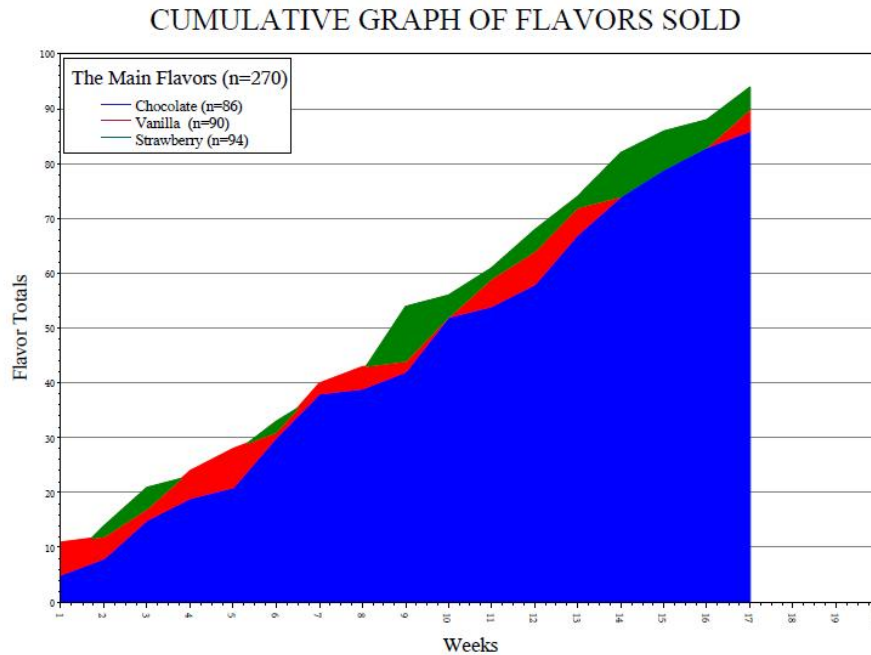


Figure 8. Output from PROC GPLOT with title added in.

## CONCLUSION

This paper touched on some simple features of the GPLOT procedure. GPLOT has too many options and uses for just one paper to be able to cover everything. The most important thing to remember is if it can be done in Excel, SAS can do it too.

## ACKNOWLEDGMENTS

Many thanks to Mike Rhoads, Michael Raithel, Rick Mitchell, and Sarah Woodruff

## CONTACT INFORMATION

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

Sharon Avrunin-Becker  
Westat  
1650 Research Blvd. – WB432  
Rockville, MD 20850  
Work Phone: (301) 738-8384  
Email: [SharonAvruninBecker@westat.com](mailto:SharonAvruninBecker@westat.com)

## DISCLAIMER

The contents of this paper are the work of the author and do not necessarily represent the opinions, recommendations, or practices of Westat.

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.