

Using a Single SAS® Radar Graphic to Describe Multiple Binary Variables within a Population

Kevin Sundquist, Faith Parsons, Jacob Julian, Sean Mota;

Center for Cardiovascular Behavioral Health, Columbia University Medical Center, NY

ABSTRACT

In health care and epidemiological research, there is a growing need for basic graphical output that is clear, easy to interpret, and easy to create. SAS® 9.3 has a very clear and customizable graphic called a Radar Graph, yet it can only display the unique responses of one variable and would not be useful for multiple binary variables. In this paper we describe a way to display multiple binary variables for a single population on a single radar graph. Then we convert our method into a macro with as few parameters as possible to make this procedure available for everyday users.

INTRODUCTION

In this paper we demonstrate the usefulness of the SAS Radar Graphic and why there is a need for the expansion of its capabilities. The current SAS Proc for the Radar Graphic is simple and easy to use if you have a variable that fits the specific criteria. The current GRADAR procedure creates a graphical representation of the counts or percentages of unique observations for a single variable; it is akin to the FREQ procedure except displayed graphically across a circular spoke network with N spokes, where each spoke represents unique variable outcomes. You can also do a cross-frequency by adding a variable that will split the plot into different categories, which creates a graph with N spokes and M different line or plot overlays.

This type of graphic, although very useful, has limitations when it comes to real world applications. What if you want to compare three or more positive response rates? This is the challenge we encountered at our Center: we have data collected from different study sites that contain multiple binary variables. We wanted a single graphical display that illustrates the proportion of positive (yes) response rates for each binary variable and categorize the responses by the study site. Using several SAS procedures and data management techniques, we developed a method to derive a single radar-ready variable from multiple variables. We then we created a macro that allows other users create a single radar graph from multiple binary variables.

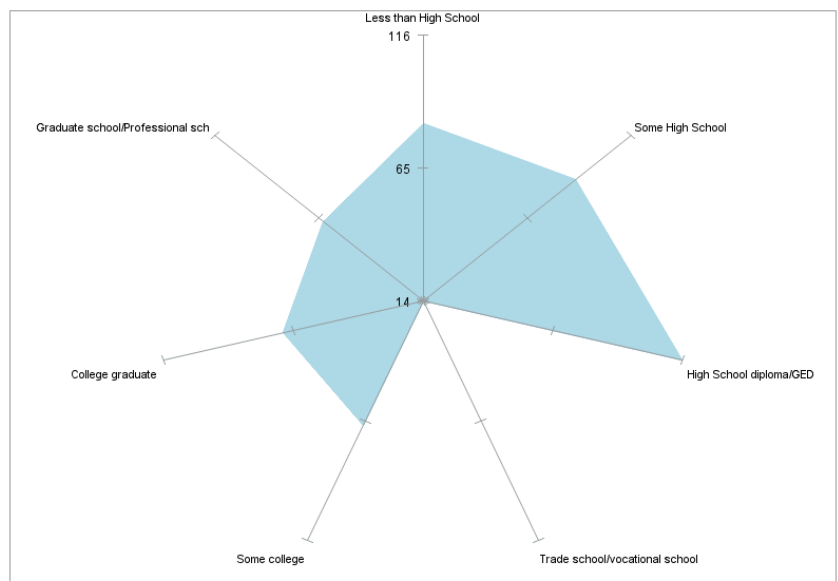
CREATING A “RADAR READY” DATASET

In this paper, we discuss steps on how to create a radar-ready variable from multiple variables, convert the process into a macro, and ways you can change the look of your output.

Sample Figure Code:

```
goptions reset=all border;
title "Sample Radar Graphic";
proc GRADAR data=Dta01;
  chart educ /
  starinradius=0
  cstars=(lightblue)
  starfill=(solid)
;run;
quit;
```

Figure 1. Sample radar graph



The graphic on Figure 1, is a visual representation of the education variable from one of the research studies at the Center for Behavioral Cardiovascular Health (CBCH) at Columbia University Medical Center. The figure shows a typical radar graph which displays the distribution of responses to the different education categories. The radar graph works well for one variable with multiple categories, however, we have several binary variables with Yes/No responses that we would like to visually represent in one graph. So this brings us to the first part of this paper: how to graphically compare more than two binary variables on a single radar graph.

HOW TO GRAPH MULTIPLE BINARY VARIABLES ON A SINGLE RADAR GRAPHIC

The first step to this process is to transform your data:

In order to graphically display your variables in SAS using GRADAR you will need to create a dataset with the following characteristics: each row corresponds to the name of the binary variable name with (yes) or (no) responses (see Figure 2).

Figure 2. Comparison of starting dataset and transformed dataset

Starting dataset							Transformed dataset	
ID	Var_1	Var_2	Var_N			RadarReady_Name	PercentResponse
1	1	0	-	1			Var_1	15
2	0	0	-	1			Var_2	20
...	-
N	1	1	-	0			Var_N	50

Below are the steps to create a radar-ready variable from your dataset.

1. The first step is to identify your variables of interest and an Identification variable that uniquely identifies the respondents or observations, for example (ID, ParticipantID, UniqueID). From this you sort by the unique ID you just identified, this is very important for the next step.

- Sort by ID:

```
proc SORT data = Example1; by ID; run;
```

2. Next, we use a TRANSPOSE procedure, which will create a data set similar to our transformed dataset (Figure 2) but it will be spread out as shown below (Table 1):

- Transpose your dataset

Table 1. Transposed and formatted dataset

ID	_Name_	_label_	Response_1
1	Var_1	Lable1	1
1	Var_2	Label2	1
1	Var_3	Lable3	0
2	Var_1	Lable1	0
2	Var_2	Label2	0
2	Var_3	Lable3	1
.....
N	Var_3	Label3	1

```
Proc TRANSPOSE data=Example1
  Prefix = Response_
  out=Transposed_data;
  var Var_1 Var_2 Var_3 ;
  by ID;
  run;
```

- Now we manipulate the “Transposed_Data” dataset and apply some basic data management techniques. We will drop out all responses that are not our responses of interest (yes or no) and only keep the variables ID, and _Name_. Remembering how the data is formatted, if we only keep the values of interest these steps will give us the variable names repeated N times for N number of responses. You might also want to rename _Name_ to something more logical like RadarVar.

- Keep only ID and _Name_, rename _Name_ and drop out non-responding rows.

```
DATA RadarData;
set Transposed_Data;
where Response_1=1;
rename _NAME_ = RadarVar;
keep _Name_ ID;
run;
```

- Next run a frequency on this data to find the count of desired response observations.

Table 3. Frequency Output

RadarVar	COUNT	PERCENT
Var_1	# Obs	%
Var_2	# Obs	%
Var_3	# Obs	%

```
proc FREQ RadarData;
Table
RadarVar/out=RadarVarCount;
run;
```

- From Table 3, we will only use RadarVar and COUNT. The percentage shown from the frequency table is not what we want to represent in our radar graph, because it is not representative of percentage of positive responses over the TOTAL number of responses from our initial dataset. To get the actual percentages we need to create a macro variable that will be the count of our initial dataset or if you already know the N you can also use a hard coded method.

```
data _NULL_;
set Example1 nobs=n;
call symput('N',trim(left(put(n,20.))));
stop;
run;
```

- The statement above will create a macro variable referenced as &N that will be the count of observations in the starting dataset; in our example this would be Example1.
- This is the last step for creating a “Radar Ready” dataset. In steps 3 and 4, we created a dataset with each variables count of positive responses and created MACRO variable that is the total N of the original dataset. In this step, we will compute the percentages by dividing the counts by the total N. The data step below illustrates one way to compute these rates.

RadarVar	COUNT	N	Pct
Var_1	# Obs	N	# Obs / N
Var_2	# Obs	N	# Obs / N
Var_3	# Obs	N	# Obs / N

Table 4. Final Dataset (*Only RadarVar and Pct*)

```
data Final_Radar;
set RadarVarCount;
pct=(count/&N)*100;
drop percent count;
run;
```

6. Now we will move on to the graphing portion of this paper, we will go over how to: add/format an axis, change the color or appearance of the graphic, change the inner radius, and how to overlay another radar graphic or multiple.

```
goptions reset=all border; A.
axis2 order=(0 to 100 by 20) value=none; B.
axis3 order=(0 to 100 by 20) value=( tick=1 ""); C.
title " Radar For Var1 - Var3 ";
proc GRADAR data= Final_Radar;
chart radarvar/ freq = pct D.
staraxis=(Axis3 Axis2 Axis2) E.
starinradius=0 F.
/*overlayvar=&split*/
cstars=(lightblue) G.
starfill=(solid) H.
wstar =(1)
lstar =(1) J.
;
run;
quit;
```

We will describe each part of the GRADAR step and the customizable features.

- Before the GRADAR we need to reset all borders previously defined.
- Axis2 and axis3 will be our axes for the graphic (you could use whichever name you like), for this style we advise a “simpler is better” approach. We first Name the Axis = “Axis2” then we use the order statement to create the minimum, maximum and interval lengths of the axis “Order=(Min=0 to Max=100 by interval=20). This will give you an axis that goes from 0 to 100 by ticks of 20, perfect for a percentage scale that goes from 0 to 100. Then value=none will make the axis be formatted such that it does not show the value of each tick such as “0 20, 40 ...” This makes the graphic look messy so we will drop this feature in all except for the first axis, this brings us to axis3. Now we repeat the above process except we name the axis “axis3” and when we get to the value statement we use the tick=1 option after value= to create a small number representing each tick on the axis (0,20,40,60,80,100). We will come back to this in step E.
- Titles are important**, Radar graphics are not common place yet and a descriptive title can make all of the difference!
- After you specify the data with proc GRADAR you will choose your graphing variable. From before we know that the data is set up such that the variable of interest is a combination of two variables, the name of the variable and its positive response frequency. So we will CHART our **RadarVar** then specify the FREQ= option and input **PCT**, **it should look like this: “Chart RadarVar / FREQ=PCT”**. This tells SAS that the frequencies have already been calculated for each of the row RadarVar values and that they are contained in the PCT variable. THERE SHOULD BE NO SEMI-COLON AFTER THIS STEP.
- Now as we think back to step B we will use the STARAXIS option to specify which axis’s will have which format, from our specifications (axis2 and axis3). For visual clarity we suggest only using Axis3 once and then axis2 for the rest of the axis’s in your graphic. The option is created so that

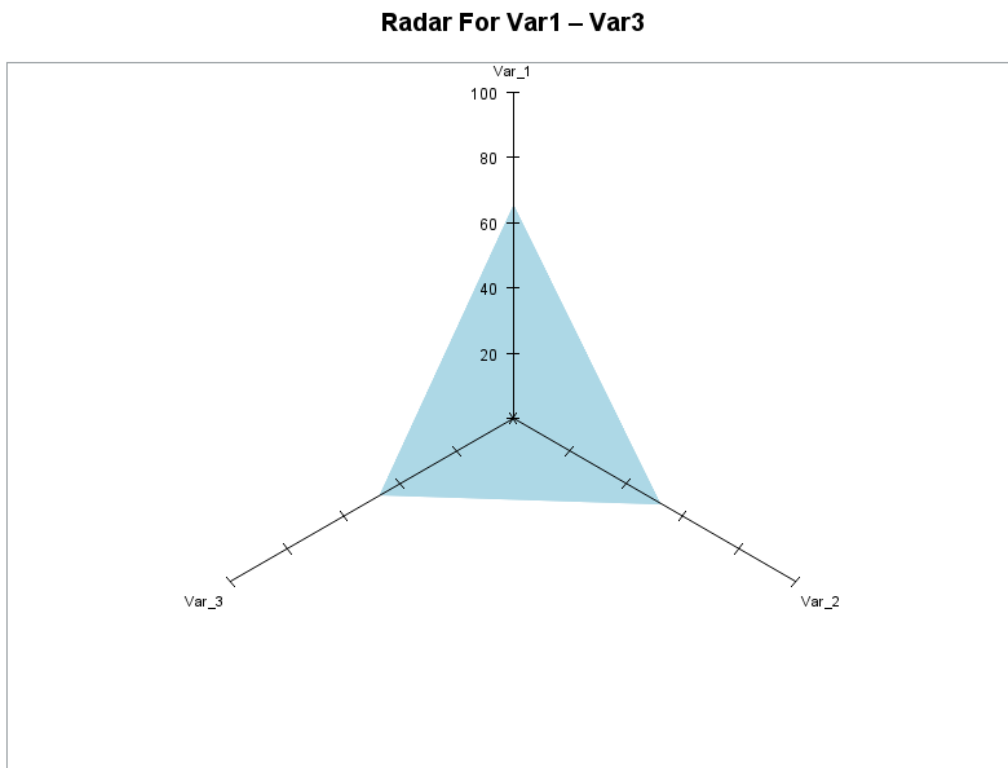
for each variable you must specify its axis format or else SAS creates its own, that WILL not match the form of the specified axis's, thus for this statement we must do as follows. If you are using 3 variables the statement will look like this: "Staraxis=(Axis3 Axis2 Axis2)" or if you had 5 variables the statement would be: "Staraxis=(Axis3 Axis2 Axis2 Axis2 Axis2)" again with no semi-colon.

- f) The Starinradius statement tells SAS the radius of a circle inside of 0, this can be useful in other situations but for our purposes we need the statement to read starinradius=0 no semi-colon
- g) The Cstars statement means "Color of stars" so you can specify the color of your plot
- h) The StarFill specifies the fill of your graphic so you can choose Solid or Empty
- i) Wstar states the weight of the line that is used to draw the outline of your chart
- j) Lstar states the type of line that is used, 1 is a solid line

Note: More information on customization of the radar graphic can be found by following this link:

<http://support.sas.com/documentation/cdl/en/graphref/67881/HTML/default/viewer.htm#p16s3bvpmpzqiron19ucimqqoze1b.htm>

7. Output 2. Final Product (with example Data in Sup. Files):



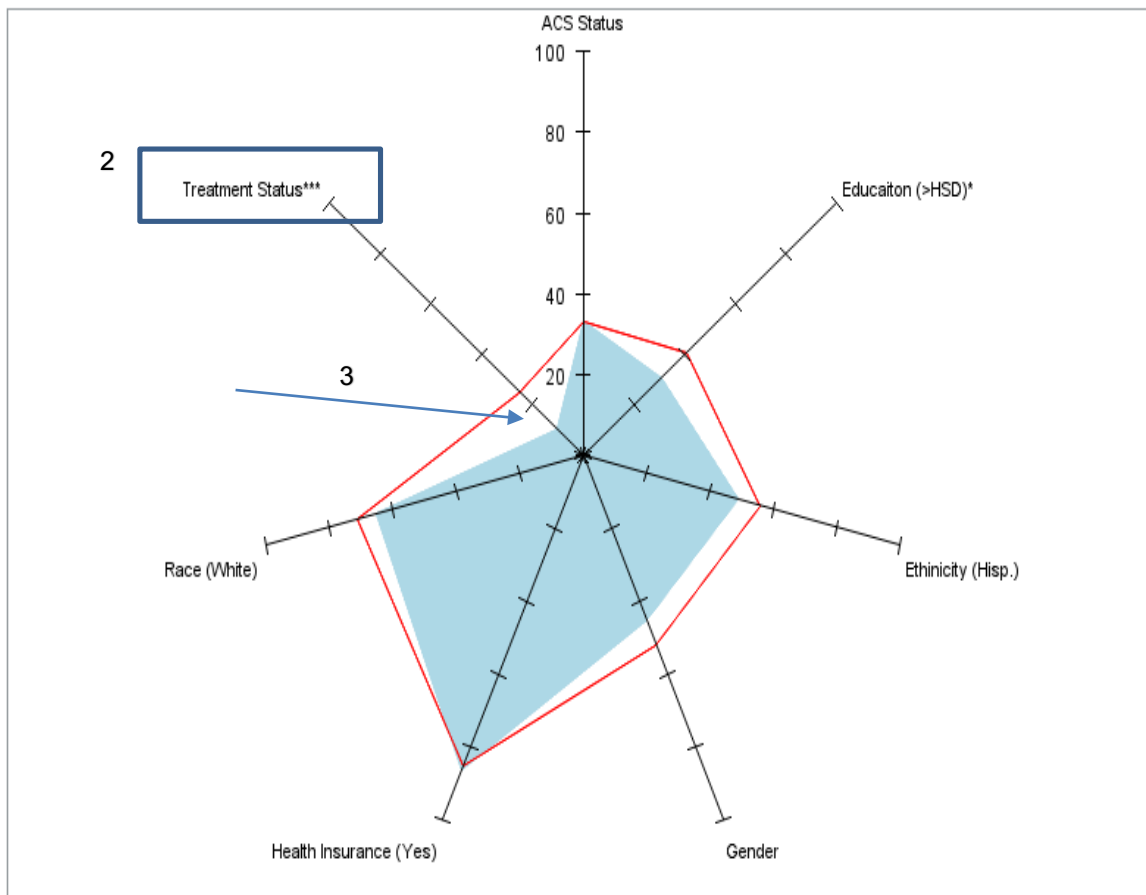
THE “RADARREADY_SPLIT” MACRO

WHAT IS THE “RADARREADY_SPLIT” MACRO?

Output 3. Example of graphical output from a current study on treatment rates in Psychopathology patients.

Demographics Compared for those with and without Psychopathology

* $p < 0.05$ ** $p < 0.01$ *** $p < 0.001$



From the graphic above let us highlight the key applications of the macro's output

1. In the graphic above we can assess the positive response rates of demographic and study questions conditional on if patients screened positive or not for a psychopathology (in this case depression or PTSD). (Yes is the red line and No is the blue shaded area).
2. You will notice that the Treatment Status label has three *** next to its name, the macro will add stars ranging from (*, **, ***) to the labels of variables that have significantly different response rates when split by your “Split” binary variable using Chi-squared and Fishers exact tests.
3. Other than the stars signifying the differences on the variables label names you can visually see which variables have similar and not so similar positive response rates, look at the relatively large space between the blue and red line on the Treatment status spoke. Now look at the Health Insurance spoke, you can see obviously that the response rates of these two groups are almost equal.

HOW TO SET UP YOUR DATA FOR THE “RADARREADY_SPLIT” MACRO.

- The data should be set up such that each row corresponds to one observation.
- The data should only contain the ID variable, the Split variable, and the binary variables for analysis.
- All of the data should be labeled such that the label is what the user wants to display as the axis heading.
- Only Binary variables, formatted as 1 or 0; where 1 is considered a “Positive” or the response of interest.
- Must have more than 2 variables not including the ID and Split variables.

Example of a properly set up dataset:

ID	Split	Var_1	Var_2	Var_N
1	1	1	0	-	1
2	0	0	0	-	1
...	-	...
N	0	1	1	-	0

The macro must be first run, to do this copy and paste it from the supplementary files (#2). Then you must create your dataset for graphing, remembering the conditions above. Then fill out the macro statements below and hit run.

```
%radarready_split(data,id,split,title) ;
```

- **Data:** The dataset you created according to the conditions stated above.
- **ID:** The Unique ID that identifies each observation.
- **Split:** The Variable you would like to split the graph by.
- **Title:** The Descriptive title you would like your graph to have

The final output will be a proc CONTENTS of your dataset, your graphic, and a table of each variables Chi-square and Fisher's Exact Test P-values, and a description of what the P-values mean.

CONCLUSIONS

Radar graphics are an exciting and powerful way to display data, and we wanted to find a way to use this type of graphic in SAS with our binary variables. Using the processes described in the first section other SAS users are now able to create easily interpretable, simple, and practical radar graphics with multiple binary variables. We also introduced our “RadarReady_Split” macro that has many features and automated processes that creates a truly meaningful and wonderful graphical display of binary data.

REFERENCES

1. Graphics. (n.d.). Retrieved March 16, 2016, from <https://support.sas.com/rnd/datavisualization/yourGraphs/businessQuick/radar/>
2. SAS/GRAPH(R) 9.4: Reference, Fourth Edition. (n.d.). Retrieved March 16, 2016, from http://support.sas.com/documentation/cdl/en/graphref/67881/HTML/default/viewer.htm#p16s3bvp_mzqiron19ucimqqoze1b.htm

CONTACT INFORMATION

Kevin Sundquist

KS3204@cumc.columbia.edu

Phone: (212) 342-4484

Center for Behavioral Cardiovascular Health
Columbia University Medical Center
622 West 168th St PH9 309
New York, NY 10032