

Graduate School Class Reminders

- ▶ Maintain six feet of distancing
- ▶ Please sit in the same chair each class time
- ▶ Observe entry/exit doors as marked
- ▶ Use hand sanitizer when you enter/exit the classroom
- ▶ Use a disinfectant wipe/spray to wipe down your learning space before and after class
- ▶ Media Services: 414 955-4357 option 2

Documentation on the web

- ▶ CRAN: <http://cran.r-project.org>
- ▶ R manuals: <https://cran.r-project.org/manuals.html>
- ▶ SAS: <http://support.sas.com/documentation>
- ▶ SAS 9.3: <https://support.sas.com/en/documentation/documentation-for-SAS-93-and-earlier.html>
- ▶ Step-by-Step Programming with Base SAS 9.4 (SbS):
<https://documentation.sas.com/api/docsets/basess/9.4/content/basess.pdf>
- ▶ SAS 9.4 Programmer's Guide: Essentials (PGE):
<https://documentation.sas.com/api/docsets/lepg/9.4/content/lepg.pdf>
- ▶ Wiki: <https://wiki.biostat.mcw.edu> (MCW/VPN)

Cheese cluster SAS program recovery system

- ▶ Since SAS programs are run, debugged, re-run, etc., it is very easy to make a programming error (whether syntactically or semantically) between runs or even accidentally to delete your program entirely (these dangers seem less pronounced for R programming although they do exist to some extent)
- ▶ On the server, all of our data files are backed up nightly
- ▶ However, if you create a new SAS program today and make an error, then it won't be backed up until later so you will not be able to recover it since a backup copy does not yet exist
- ▶ Also, even if it is backed up, it may take several hours to get the sysadmin to recover your SAS program which might be too late if you have an impending deadline
- ▶ Therefore, we have an automated system that **saves the last 9 versions of your SAS program, your .log, .lst, etc.**
- ▶ There is a hidden sub-directory `.USERNAME` where these are saved **triggered by each run of the SAS program**
- ▶ You can see how these are created by shell scripts:
`/usr/local/bin` directory with `sas` and `backup.ksh`

SAS Statistical Graphics: SGPROCs

- ▶ The legacy GPROCs remain for old code and some edge cases we will not be discussing these further
- ▶ The new SGPROCs largely cover the same territory and have many advantages in ease-of-use and superior capabilities
- ▶ There are two SGPROCs in particular
- ▶ SGPLOT will create many of the graphical displays that we are accustomed to: scatter plots, bar charts, histograms, box plots, heatmaps, time series and more
- ▶ SGPANEL has many of the same capabilities with the added ability to create panels for multiple displays on the same page
- ▶ Each of these have many opportunities for customization within the SGPROCs themselves
- ▶ In addition, there is an SGANNOTATE facility so that you can overlay text, points, lines, etc.
- ▶ We will need to use the latest documentation for SAS 9.4 since the SGPROCs are still developing (sometime buggy)
- ▶ SAS has recently made the 9.4 documentation easier to navigate (9.4 documentation is now on par with 9.3 or earlier)

SAS Statistical Graphics: SGPROCs

- ▶ In `~/autoexec.sas`, we have this line commented out
`ods graphics on / reset imagename("&fnroot"`
`imagefmt=pdf; footnote;`
- ▶ You place this line before your SGPROC to create Adobe Portable Document Format graphics stream files
- ▶ if your SAS program is `NAME.sas`, then the first page of your plot is `NAME.pdf`, the second page is `NAME1.pdf`, etc.
since the `&fnroot` macro variable resolves to `NAME`
(created by the `%_fn;` macro called in `~/autoexec.sas`)
and turn off the footnote it creates which is ugly for graphics
- ▶ For publication/presentations, it is helpful to have each page in a separate file. But, for programming and debugging, it is OFTEN easier to have them all in one file.
- ▶ So place the line `%_pdfjam;` after the SGPROC which will utilize the `pdfjam` utility (from TeX Live) to create a single PDF: `NAME.pdf` (see the man page: `man pdfjam`)
- ▶ Press F12 to view it
- ▶ Once you are happy, then comment it out: `%*_pdfjam;`

SAS Statistical Graphics: PROC SGPLOT

- ▶ You can get multiple plots on one page: RED are optional

```
proc sgplot data=NAME;
```

```
STATEMENT1 ...;
```

```
STATEMENT2 ...;
```

```
⋮
```

```
STATEMENTn ...;
```

```
run;
```

where STATEMENT1 ... STATEMENTn are *compatible*

- ▶ Similarly, there is a `group=VAR` where BLUE is optional

```
proc sgplot data=NAME;
```

```
STATEMENT1 ... / group=VAR1;
```

```
STATEMENT2 ... / group=VAR2;
```

```
⋮
```

```
STATEMENTn ... / group=VARn;
```

```
run;
```

- ▶ You can mix and match STATEMENTs with and without a `group` option

SAS Statistical Graphics: PROC SGPLOT

- ▶ Two STATEMENTS are compatible if they are the same
- ▶ The STATEMENTS that produce graphical output that we will consider are HISTOGRAM, DENSITY, VBOX/HBOX, HEATMAP, SERIES and SCATTER (N.B. there are others: see the docs)
- ▶ There are compatible statements that are NOT the same such as DROPLINE, INSET, KEYLEGEND/LEGENDITEM, REFLINE, STYLEATTRS, SYMBOLCHAR, XAXIS and YAXIS
- ▶ As well as the usual SAS statements like FOOTNOTE_n/TITLE_n, FORMAT, LABEL and WHERE
- ▶ And ODS statements like those about output data sets

```
ods trace on;
```

```
ods output NAME=NEW;
```

```
proc sgplot data=OLD;
```

```
STATEMENT ...;
```

```
run;
```

```
ods trace off;
```

VERY useful to understand what is being displayed

SAS Statistical Graphics: PROC SGPLOT HISTOGRAM

- RED and BLUE are optional

```
proc sgplot data=NAME;  
  histogram X1 ... / group=Y1;  
  histogram X2 ... / group=Y2;  
  ⋮  
  histogram Xn ... / group=Yn;  
run;
```

- The default is `scale=percent` which is usually what you want, but not always
- The alternative is `scale=count`
see `NTDB/sas/histogram.sas`

SAS Statistical Graphics: PROC SGPLOT DENSITY

- ▶ BEWARE: The default is `type=normal` for Normality
THIS IS EXTREMELY DANGEROUS! NEVER USE THE
DEFAULT EVEN IF YOU THINK NORMALITY MAKES
SENSE. LET THE DATA BE YOUR GUIDE.
USE THE OPTION `type=kernel`

- ▶ RED and BLUE are optional

```
proc sgplot data=NAME;  
density X1 / type=kernel ... GROUP=Y1;  
density X2 / type=kernel ... GROUP=Y2;  
:  
density Xn / type=kernel ... GROUP=Yn;  
run;
```
- ▶ see `NTDB/sas/density.sas`

SAS Statistical Graphics: PROC SGPLOT VBOX

- ▶ Box-plots are a standard way to summarize continuous outcomes often for two or more discrete categories
- ▶ You can overlay box-plots, but this is usually not visually appealing
- ▶ The VBOX statement creates vertical box-plots
- ▶ There is also HBOX for horizontal box-plots
- ▶ RED are optional

```
proc sgplot data=NAME;  
  vbox VAR1 / group=VAR2;  
run;
```
- ▶ Let's see the documentation to see how a box-plot is defined
- ▶ And NTDB/sas/vbox.sas

SAS Statistical Graphics: PROC SGPLOT HEATMAP

- ▶ A heat-map is a 2D plot of a 3D relationship with variables, X and Y, where the color plotted represents the height of the joint density in an $NXBINS \times NYBINS$ grid of cells
- ▶ Generally, the two variables are continuous
- ▶ HEATMAP supports categorical variables BADLY with options `discretex` for X and/or `discretey` for Y
the support is a little buggy at the moment
- ▶ Although you can overlay multiple heat-maps, typically that will NOT be a good idea

```
proc sgplot data=NAME;  
heatmap X=VAR1 Y=VAR2 / nxbins=N nybins=M;  
run;
```
- ▶ GO TO SGANNOTATE AND RETURN HERE
- ▶ See `NTDB/sas/heatmap.sas`

SAS Statistical Graphics: PROC SGPLOT SERIES

- RED and BLUE are optional

```
proc sgplot data=NAME;  
series X=VARX Y=VARY / ... GROUP=VAR1;  
series X=VARX Y=VARY / ... GROUP=VAR2;  
:  
series X=VARX Y=VARY / ... GROUP=VARn;  
run;
```

- see EHR/sas/series.sas

SAS Statistical Graphics: PROC SGPLOT SCATTER

- RED and BLUE are optional

```
proc sgplot data=NAME;  
scatter X=VARX Y=VARY / ... GROUP=VAR1;  
scatter X=VARX Y=VARY / ... GROUP=VAR2;  
:  
scatter X=VARX Y=VARY / ... GROUP=VARn;  
run;
```

- see EHR/sas/scatter.sas

SAS Statistical Graphics: SGANNOTATE %sganno macro

START HERE

- ▶ The ANNOTATE facility is to further customize graphics stream files with text, lines, etc.
- ▶ There are two versions
- ▶ The legacy GPROC ANNOTATE facility
- ▶ And the new SGANNOTATE facility
- ▶ In ~/autoexec.sas, we call two ANNOTATE macros:
%annomac; to load the legacy GPROC ANNOTATE facility which we will not discuss further
- ▶ And %sganno; to load the new SGANNOTATE facility which we will describe
- ▶ Each of these macros start with %SGname
- ▶ The most common arguments are
x1=VARX for the *x*-axis
y1=VARY for the *y*-axis
and drawspace="DATAVALUE" for drawing in the *data* space
- ▶ Typically, the most important arguments for a macro are at the top of the help and the rest have reasonable defaults

SAS Statistical Graphics: SGANNOTATE %sganno macro

- So in your .log you will see the following near the top

Following macros are available

%SGANNO_HELP

%SGARROW

%SGIMAGE

%SGLINE

%SGOVAL

%SGPOLYCONT

%SGPOLYGON

%SGPOLYLINE

%SGRECTANGLE

%SGTEXT

%SGTEXTCONT

Enter %SGANNO_HELP(macroname) for details

or %SGANNO_HELP(ALL) for details on all SGANNO macros.

See PTB/ecgleadi.sas

SAS Statistical Graphics: PROC SGPANEL

- ▶ PROC SGPANEL follows roughly the same syntax as PROC SGPLOT
- ▶ But, it allows you to create *panels* of your data
- ▶ Use the PANELBY statement typically with the ONEPANEL option to place them on one page and the COLUMNS=m option for the number of panel columns

```
proc sgpanel data=NAME;  
  PANELBY VAR1 ... VARn / onepanel columns=m;  
  STATEMENT1 ...;  
  STATEMENT2 ...;  
  ⋮  
  STATEMENTn ...;  
run;
```

See PTB/ecgpanel.sas

SAS Statistical Graphics: PROC SGPLOT STYLEATTRS

- ▶ Controls the colors of the lines/markers
styleattrs
datacontrastcolors=(black darkgray lightgray);
- ▶ Controls the colors of the *fill* pattern
styleattrs
datacolors=(black darkgray lightgray);
- ▶ You can see the SAS colors and their associated names by
running the following
(see /data/shared/04224/colors.sas)

```
proc registry list startat="COLORNAMES";  
run;
```