## Introduction to SAS

All programs and data sets used for these notes are available from my course website. These files are gpa.sas, gpa.txt, gpa.csv, gpa\_names.csv, cereal.sas, and cereal.csv.

## Background

SAS is a very widely used statistical software package. The software was originally developed by students and professors at North Carolina State University in the 1960s and 1970s. These individuals eventually left the university and formed the SAS Institute based in the Research Triangle area of North Carolina. The SAS Institute is now the largest privately owned software company in the world. Originally, SAS stood for "Statistical Analysis System" but it is now just simply "SAS".

When I was going to graduate school, there were no real competitors. SAS by far was the most powerful statistical software package with respect to what it could do. All statistics graduate students learned how to use SAS. Software like SPSS was not used by statisticians but by some individuals in other fields, like the social sciences, that did not necessarily need the best software. Part of SAS's appeal was that SAS was cheap for those in academia. Once statistics graduate students received their degree, these students would ask their companies if they could use SAS. The SAS Institute would then charge thousands of dollars in yearly subscription fees per user license to these companies!

The statistical software environment began to change on February 29, 2000 when R version 1.0.0 was released. Unlike SAS, R is completely free. The growth of R has been huge to a point now that R is the predominant statistical software in Departments of Statistics. Still, SAS is used a lot in academia and in industry. Currently, I think students need to have a background in both

	Int	ro.3
-		
SAS		83
File Edit View Tools Run Sol	uuoons Window Heip ▼ □ 😅 🖬 🚑 D.   X 🖦 🖏 🕫 🏟 வ   ★ X 🛈 🏈	
✓		
	Log-Christian NDTE: Copyright (c) 2002-2012 by SAG Institute Inc., Cary, NC, USA. NDTE: SAG (r) Proprietary Software 3.4 (TSIN2) Licensed to UNIVERITY OF NEDMARKA'S SFA TAR, Site 70080298. NDTE: This session is executing on the X64_7PRD platform.	*
Favorite Computer Folders	NUTE: Updated analytical products: SAG/STAT 13.2 SAG/STAT 13.2 SAG/TR 13.2 SAG/TR 13.2 SAG/TR 13.2 SAG/TR 13.2 SAG/TR 13.2 NUTE: Additional host information: X64_7PR0 HIN 6.1.7601 Service Pack 1 Horkstation NUTE: SAS initialization used: real time 5.89 seconds cpu time 1.31 seconds	
Results Q Explorer	Output - (Untitled)  Log - (Untitled)  CAUSer/Chris Ln 1, Col 1	

There are a number of windows within SAS:

- Enhanced editor: Statisticians primarily write computer programs to perform statistical analyses. Programs for SAS are written here. This editor has syntax highlighting so that different colors are used for different types of code. Also, the editor enables code folding so that multiple lines of code can be "folded" into one line.
- Log: During a running of a program, information regarding the execution of code (e.g., syntax errors) will be printed here.
- Results Explorer: The output resulting from running a program (e.g., information about the fit of a regression model) will

software packages. It will be interesting to see if this is still true in the future :).

## Installation

It is not necessarily easy for an individual to purchase SAS outside of academia! For example, go to the SAS website (http://www. sas.com) and you will notice that there is no "purchase here" link.

For all of UNL, the Department of Statistics is the distributor of SAS. Inidviduals can obtain SAS for a small fee from the department. The SAS installation on a computer is very large (> 10GB), and there can be some issues that occur during installation. Note that SAS is also available on the computers in the two small computer rooms within our department as well as a few other computer rooms on campus.

### Basics

Below is what SAS looks like when you open it.

Intro.4

appear here. By default, all output created is HTML-based and opened within this window.

- Output: This is where the output from procedures used to go by default (prior to SAS version 9.3). It still can be useful to use this window at times.
- Explorer/Results: Data sets are listed in the Libraries location of the Explorer tab. Links to specific output are in the Results tab.

These windows are all re-sizable and can be moved around. Important buttons on the main toolbar include:

- Submit \*: Select this button to run the whole or a highlighted portion of a program.
- Break **O**: Select to stop a program while it is running.
- Save **\Box**: Select to save the program.

All SAS programs have a .sas file extension. These files are simply plain text so they can be opened in programs such as Notepad. SAS is not case sensitive with its programs.

### Simple program

My program gpa.sas is a very basic SAS program. It reads in a data set located in the files GPA.txt (plain text file using space delimiters, NOT tab delimiters) and gpa.csv (plain text file using comma delimiters). Both of these files contain the same 20 observations involving high school and college GPA of students. Our eventual purpose will be to use high school GPA to estimate college GPA with a regression model.

To open the SAS program, simply drag and drop it into the Enhanced Editor window or select FILE > OPEN. Below is a screen capture of the program in the Enhanced Editor.

gpa.sas	
<pre>dm "log; clear; odsresults; clear;";</pre>	
options linesize=64 nonumber nodate;	
*****	*******
* AUTHOR: Chris Bilder	*:
* DATE: 4-22-16	*
* FURPOSE: Data analysis example using the gpa data set	*;
***************************************	******;
*Will appear as the first line of every page of output;	
title1 "Chris Bilder, STAT 850";	
<pre>infile "C:\data\gpa.txt" firstobs=2; input HS College; run;</pre>	
*Read in the data set from a comma delimitted text file;	
data set1;	
<pre>infile "C:\data\gpa.csv" firstobs=2 delimiter=","; input HS College ;</pre>	
run;	
title2 "The HS and College GPA data set";	
<pre>proc print data=set1;</pre>	
run;	

Important aspects in the beginning of the program:

- Line 1: This clears any information that may be in the log window. If you also want to clear information in the Results Viewer (I usually do), include odsresults; clear; within the quotes. The dm code means "display manager".
- Line 2: Various options can be set here. I specify these options to help with producing my lecture notes. For example, the linesize = 64 limits the number of characters on each line

Intro.7

• Indenting of the code between data and run is standard practice to make code easier to read.

To run the code so far, I highlight from the beginning of the program to the end of the datastep and then select submit. This code produces no output itself. Below is what the Log window looks like after running this segment of code.

```
1 dm "log;clear;odsresults;clear;";
2 options ps=50 ls=75 pageno=1;
3
5 * NAME: Chris Bilder
                                                 *;
6 * DATE: 4-22-16
                                                 *;
7 * PURPOSE: Example with gpa data set
                                                 *:
9
    *Will appear as the first line of every page of output;
10
    title1 'Chris Bilder, STAT 850';
11
12
13
14 *Read in the data set from a space delimitted text file;
15
    data set1;
     infile 'C:\data\gpa.txt' firstobs=2;
16
17
      input HS College;
18
    run;
NOTE: The infile 'C:\data\gpa.txt' is:
     Filename=C: \data \gpa.txt,
     RECFM=V,LRECL=32767,File Size (bytes)=213,
     Last Modified=22Apr2016:22:38:43,
     Create Time=22Feb2012:10:27:54
NOTE: 20 records were read from the infile 'C:\data\gpa.txt'.
     The minimum record length was 5.
     The maximum record length was 9.
NOTE: The data set WORK.SET1 has 20 observations and 2 variables.
NOTE: DATA statement used (Total process time):
                       0.03 seconds
     real time
     cpu time
                       0.01 seconds
```

The notes provided with the datastep indicate the code ran properly. In the Explorer window, the data set is shown in the of output in the Results Viewer to be no larger than 64.

- Comments begin with a asterisk \* and are color coded in green. Note that only one asterisk is actually needed at the beginning of the line. I use more than one at times because this is a standard way in SAS programs to make code more readable.
- The other code is colorized according to its purpose as well. This can be controlled by selecting TOOLS > OPTIONS > ENHANCED EDITOR > APPEARANCE.
- The title1 statement gives information that will be printed on line 1 of every page in the Results Viewer.
- Semicolons end every complete statement in SAS!

Next in the program is a *datastep*. These are used to change information in a data set. In this case, it is used to read in the data from a file. Below is an explanation of the code:

- Line 1: All dataseteps begin with data. The word after it gives the name of a data set to be created. This name needs to start with a letter and can include numbers and underscores.
- Line 2: The infile statement indicates where the data file is located. The firstobs option indicates to SAS to start reading information from the file beginning at line 2. This is used here here because the first line in the file contains variable names. If the file was comma delimited, the option delimiter="," could be added to this line of code.
- Line 3: The input statement declares the variable names. Again, these names need to start with a letter and can include numbers and underscores.
- Line 4: The **run** commands tells SAS to execute the code.
- Code folding is available by selecting the minus symbol  $\exists$  next to the data line. This will put a plus symbol it is place. Selecting the plus will unfold the code.

Intro.8

*Work* library (select LIBRARIES > WORK). This is the default location for SAS to put data sets. Selecting **set1** in the library opens the data set into a window.

	X	Y	
1	3.04	3.1	
2	2.35	2.3	
3	2.7	3	
4	2.05	1.9	
5	2.83	2.5	
6	4.32	3.7	
7	3.39	3.4	
8	2.32	2.6	
9	2.69	2.8	
10	0.83	1.6	
11	2.39	2	
12	3.65	2.9	
13	1.85	2.3	
14	3.83	3.2	
15	1.22	1.8	
16	1.48	1.4	
17	2.28	2	
18	4	3.8	
19	2.28	2.2	
20	1.88	1.6	-

Comments:

- Data sets in the work library are actual files located on your computer, and they are stored in temporary folders that are deleted when SAS is closed (C:\Users\Chris\AppData\Local\Temp\SAS Temporary Files on my computer). We will see later how to create a permanent SAS data set in a user created library later.
- The data set needs to be closed before it can be modified again via code.

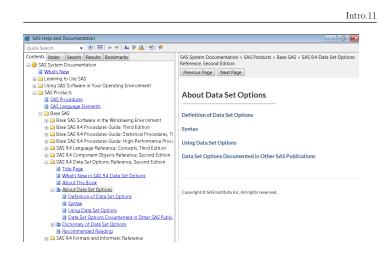
As mentioned earlier, SAS output is displayed in the Results Viewer. A simple way to put information into this window is

by printing the data set with proc print: title2 'The HS and College GPA data set'; proc print data=set1; run;

Chr The HS a			STAT 8	
	Obs	HS	College	
	1	3.04	3.1	
	2	2.35	2.3	
	3	2.70	3.0	
	4	2.05	1.9	
		2.83	2.5	
		4.32	3.7	
		3.39	3.4	
-		2.32	2.6	
-		2.69	2.8	
		0.83	1.6 2.0	
-		3.65	2.0	
-		1.85	2.3	
		3.83	3.2	
-		1.22	1.8	
		1.48	1.4	
	17	2.28	2.0	
-	18	4.00	3.8	
	19	2.28	2.2	
	20	1.88	1.6	

Below is an explanation of the code:

• A standard way to organize code and output is to use the second line of every page in the Results Viewer to indicate what



Intro.12

- 6 ×

is being displayed. I did this through a title2 statement. Note that this statement can be located on other lines in the Enhanced Editor, but needs to be before the **run** statement.

- The proc line defines what procedure to run. For most procedures, there is a **data** option which allows one to specify the data set to be used. When a procedure needs a data set, one can get away with not specifying any with **data** because SAS assumes the last active data set is being used. I recommend against doing this to prevent programming errors!
- The **run** line simply runs the procedure.

### Help

SAS Help and Documenta

SAS consists of many different *products* which can be purchased separately. These products include: Base, STAT, GRAPH, ETS, IML, OR, QC, ... The datastep and **proc print** are part of Base. This product organization is helpful to know when finding "help" because SAS organizes its help system in this format.

One way to find help is to select the help icon @ on the toolbar. After selecting this icon, I found the help for a data step and proc print in SAS:

#### 👻 🕑 🖽 😓 🔿 🗛 📎 📇 🛃 🕫 Contents Index Search Results Bookmarks SAS System Documentation > SAS Products > Base SAS > Base SAS 9.4 Procedures Guide, Third Edition > Procedures > PRINT Procedure SAS System Documentation What's New Learning to Use SAS Previous Page Next Page Using SAS Software in Your Operating Environment B SAS Products PRINT Procedure SAS Procedures SAS Language Elements Syntax - Overview Concepts Using - Examples - Base SAS Base SAS Software in the Windowing Environment Base SAS 9.4 Procedures Guide, Third Edition Supports the Output Delivery System. For details, see <u>Output</u> Delivery System: Basic Concepts in SAS Output Delivery System User's Guide. Tips: 🗎 Title Page What's New in Base SAS 9.4 Procedures Anoul The Book Concept Concept Concept Concept Another Decodure Concept Another Decodure Concept About This Book You can use the ATTRIB, FORMAT, LABEL, and WHERE statements. See <u>SAS Statements: Reference</u>. For more information, see <u>Statements with the Same Function in Multiple Procedures</u>. The SAS Tutorial <u>Printing a Simple Listing</u> on support.sas.com shows basic PRINT procedure execution. Table of Procedure Tasks and Examples Syntax E DS2 Procedure PROC PRINT < option(s) > EXPORT Procedure BY <DESCENDING> variable-1 <<DESCENDING> variable-2 > <NOTS FCMP Procedure PAGEBY BY-variable; FCMP Special Functions and Call Routine # Im FCmp Function Editor SUMBY BY-variable E 🍙 FEDSQL Procedure ID variable(s) </ STYLE <(location(s))>=<style-override>>; FONTREG Procedure E TORMAT Procedure Bornel in Code une FOLST Procedure GROVY Procedure SUM variable(s) </ STYLE <(location(s))>=<style-override>>; I IN FSLIST Procedure VAR variable(s) </ STYLE <(location(s))>=<style-override>>; Table of Procedure Tasks and Examples Statement Task Example PROC PRINT Print observations in a data set Ex. 1, Ex. 2, Ex. 3, Ex. 5, Ex. 8 BY Produce a separate section of the report for each BY group Ex. 3, Ex. 4, Ex. 5, Ex. 6, Ex. 8 Identify observations by the formatted values of the variables that you list instead of by observation numbers ID Ex. 7 PAGEBY Statemen SUM Statement PAGEBY Control page ejects that occur before a page is full Ex.3 SUMBY Statement SUMBY Limit the number of sums that appear in the report Ex. 4, Ex. 5, Ex. 6, Ex. Overview: PRINT Procedure SUM Total values of numeric variables Ex. 6 VAR Select variables that appear in the report and determine their order Ex. 1, Ex. 2, Ex. 8 Using ODS Styles with Base SAS Report Writ Error Processing in the PRINT Procedure Outs The Examples: PRINT Procedure The PRINT Procedure

Below is a zoomed in version of part of the proc print help: Syntax

PROC PRINT <option(s)>;

BY <DESCENDING> variable-1 <<DESCENDING> variable-2 ...> <NOTSORTED>; PAGEBY BY-variable; SUMBY BY-variable; ID variable(s) </STYLE <(location(s))>=<style-override>>; SUM variable(s) </STYLE <(location(s))>=<style-override>>; VAR variable(s) </STYLE <(location(s))>=<style-override>>;

### Table of Procedure Tasks and Examples

Statement	Task	Example
PROC PRINT	Print observations in a data set	<u>Ex. 1, Ex. 2, Ex. 3, Ex. 5, Ex. 8</u>
<u>BY</u>	Produce a separate section of the report for each BY group	<u>Ex. 3, Ex. 4, Ex. 5, Ex. 6, Ex. 8</u>
ID	Identify observations by the formatted values of the variables that you list instead of by observation numbers	<u>Ex. 7</u>
PAGEBY	Control page ejects that occur before a page is full	<u>Ex. 3</u>
<u>SUMBY</u>	Limit the number of sums that appear in the report	<u>Ex. 4, Ex. 5, Ex. 6, Ex. 8</u>
<u>SUM</u>	Total values of numeric variables	<u>Ex. 6</u>
VAR	Select variables that appear in the report and determine their order	<u>Ex. 1, Ex. 2, Ex. 8</u>

The syntax provides a way to understand what each portion of the code does along with examples. For example, the **var** statement allows one to select particular variables to print. Thus,

proc print data=set1;

Intro.15

some examples.

data set1;

• Include data in the program: Rather than reading a small data set from an external file, it can be more simple to just include the data as part of the code.

input HS College; datalines; 3.04 3.1 2.35 2.3 2.7 3.0 2.55 2.45 2.83 2.5 4.32 3.7 3.39 3.4 2.32 2.6 2.69 2.8 2.83 3.6 2.39 2.0 3.65 2.9 2.85 3.3 3.83 3.2 2.22 2.8 1.98 2.4 2.88 2.6 4.0 3.8

2.28 2.2

; run;

Notes:

- There are no semicolons after each line of data.
- A semicolon cannot be put at the end of the last line of data! Instead, it needs to be on a new line after the data.
- Some older SAS users may still use a cards statement rather than a datalines statement before entering the data.

run;

prints only the high school GPAs.

Other ways to find help are through web searches. Many web searches direct you to SAS's online documentation. Below is a screen capture for the datastep:

💊 About Data Set Optic 🗙	
← → C 🗋 support.sas.co	m/documentation/cdl/en/ledsoptsref/68025/HTML/default/viewer.htm#p1tkszfr
🗰 Apps 🌘 yihui/lyx · GitHub	🕅 Chris Bilder's STA
SSASS THE POWER TO KNOW Providing software solutions since 1 support.sas.com Knowled; Go to Documentation Home	ge Base Support Training & Books Store Support Communities
SAS(R) 9.4 Data Set Op	tions: Reference, Third Edition
PDF Contents About	2
Currents About What's New in SAS 0.4 Data Set Options About This Book About This Book About This Book About This Book Options - Optimions of Data Set Options - Syntax - Data Set Options Data Set Options Documented in Other SAS Publications El Dictionary of Data Set Options Recommended Reading	Previous Page   Next Page About Data Set Options Definition of Data Set Options Syntax Using Data Set Options Data Set Options Documented in Other SAS Publications
	Previous Page   Next Page   Top of Page

### Import data into SAS

d

There are a number of file formats for data outside of SAS so there are a number of ways to get data into a SAS data set. Below are

Intro.16

• Mutiple observations can be read from the same line by using the double ampersand symbol @@

data set2	;					
input H	IS College	@@;				
datalin	les;					
3.04 3.	1 2.35	2.3	2.7 3		2.55	2.45
2.83 2.	5 4.32	3.7	3.39	3.4	2.32	2.6
2.69 2.	8 2.83	3.6	2.39	2	3.65	2.9
2.85 3.	3 3.83	3.2	2.22	2.8	1.98	2.4
2.88 2.	6 4 3.8	3	2.28	2.2	2.88	2.6
;						
run;						

• Non-numeric (character) variables: When there are variables that include non-numerical values, one needs to include a dollar sign \$ after the variable name.

```
data set3;
    infile "C:\data\gpa_names.csv" firstobs=2 delimiter=",";
    input HS College first_name $ last_name$;
run.
```

There does not need to be a space between the variable name and \$.

• Specific placement of data: The file format may dictate a very fine specification of which character positions contain values for particular variables. These positions can be specified with a single ampersand symbol @.

```
data set4;
  infile "C:\data\gpa_names.csv" firstobs=2 delimiter=",";
  input @1 HS @6 College;
run;
```

One can also specify a range such as HS 1-4 for the high school variable. This range format can be helpful when there are embedded blank spaces within a variable value.

• proc import: There are procedures available to import data

as well. Below is how I imported the comma delimited data set

```
proc import out=set6 datafile="C:\data\gpa.csv" DBMS=CSV
  replace;
  getnames=yes;
  datarow=2;
```

run;

Note that the data set cannot be open in Excel when trying to import it.

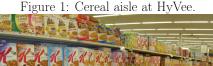
The data set can be imported with **proc** import via pointand-click methods too by selecting FILE > IMPORT DATA to bring up the Import Wizard. In the first step, select the comma separated values format and then click on NEXT >. The second step allows you to browse to the file. The third step allows you to specify the library to put the data set (work library is o.k. for now). The last step allows you to specify a new SAS program to put the **proc** import code into in case you want to perform the importation with code in the future.

SAS is not perfect when importing data! Always remember to examine the data AFTER you read it in to make sure it is correct. Also, examine the log window for important statements about the results of the datastep. I provide two examples in my program that show when SAS does not import data correctly. Overall, I have found using a comma delimited format, rather than a space delimited format, helpful to prevent errors.

The data set can be made more descriptive through the use of labels. Below is a simple example.

```
data set3label;
  set set3;
  label HS = "High School GPA"
      College = "College GPA"
      first_Name = "First Name"
      last_Name = "Last Name";
run;
```

Intro.19





Again, one should check the log window and view the actual data file itself to make sure the data was correctly exported.

### Cereal data

This examples reinforces some of the concepts learned earlier in this section and also shows new ways to use a datastep and a procedure.

A few years ago, I collected information on the nutritional content of dry cereals at a grocery store. This was done by first noting that one side of one aisle in many grocery stores usually contains all the cereals within a store. For example, Figure 1 shows what the cereal aisle used to look like in the HyVee at 5020 N 27th St. before its recent renovation. My research hypothesis was that there were different mean nutritional contents by shelf. For example, lower shelves may have more sugar content cereals than higher shelves.

The data used for this example was collected from a store a few years ago (not the HyVee in the picture). Note that there were only four shelves at this store and my sample size was 10 from Notice how I used a previously created data set in the datastep with a **set** statement. This instructs SAS to create a new data set based on this previous one. We will use this type of syntax a lot!!! To see the inclusion of labels, open the data set in the work library.

Can Excel files be imported? Yes, but I recommend using a more simple comma delimitted format most of the time. With 32bit and 64-bit computers/software, working directly with Excel files has become more difficult than it should be! If you still want to work directly with Excel files, I recommend using proc import to import the data into SAS.

# Export data out of SAS

There are a number of ways to get data from a SAS data set into an external data file. Below are some examples.

• Datastep: The outfile and put statements are used with it. data \_null\_;

```
set set1;
file 'C:\data\export1.csv' delimiter=",";
put HS College;
run;
```

The \_null\_ name for the data set is a way to not actually create a new data set for the operation. One could actually use something like data set\_new, but this is not necessary.

- proc export: This procedure works similar to proc import.
  - proc export data=set1 outfile="C:\data\export2.csv" DBMS=CSV
    replace;
    putnames=yes;

run;

The data can also be exported via FILE > EXPORT DATA and following similar methods as with importing the data.

```
Intro.20
```

each shelf. Below is how I read the data into SAS and print the first five observations.

```
title1 "Chris Bilder, STAT 850";
```

			Cereal data				
Obs	ID	Shelf	Cereal	size_g	sugar_g	fat_g	sodium_mg
1	1	1	Kellog's Razzle Dazzle Rice Crispies	28	10	0	170
2	2	1	Post Toasties Corn Flakes	28	2	0	270
3	3	1	Kellogg's Corn Flakes	28	2	0	300
4	4	1	Food Club Toasted Oats	32	2	2	280
5	5	1	Frosted Cheerios	30	13	1	210

There are many options that one can use with a data set by specifying them within parantheses right after the data set name. The previous code shows how only the first five observations of the cereal data set are used by **proc print**. Other options include **drop** and **keep** to specify which variables to use (see program for an example). While all of the data is not displayed here, the shelves are numbered from lowest (1) to highest (4).

We need to adjust the nutritional content variables (sugar\_g, fat\_g, and sodium\_g) for the serving size because cereal boxes tend to have different serving sizes. Below is how I make the adjustment by a datastep.

```
data set1;
  set cereal;
  sugar = sugar_g/size_g; *sugar content per cereal divided by
```

```
serving size;
  fat = fat_g/size_g;
  sodium = sodium_mg/size_g;
  *remove the old variables below from the data set;
  keep ID Shelf Cereal sugar fat sodium;
run:
title2 "Cereal data adjusted for serving size";
proc print data=set1(obs = 5);
run:
                       Chris Bilder, STAT 850
                 Cereal data adjusted for serving size
     Obs ID Shelf Cereal
                                                    fat sodium
                                          sugar
              1 Kellog's Razzle Dazzle Rice Crispies 0.35714 0.000000 6.0714
       1 1
       2 2
              1 Post Toasties Corn Flakes
                                         0.07143 0.000000 9.6429
                                         0.07143 0.000000 10.7143
       3 3
              1 Kellogg's Corn Flakes
                                         0.06250 0.062500 8.7500
       4 4
              1 Food Club Toasted Oats
              1 Frosted Cheerios
                                         0.43333 0.033333 7.0000
       5 5
```

There are a number of statements in **proc print** which can make printing of output nicer. Below is how I use the **where** and **var** statements for illustrative purposes. I also remove the observations numbers with the **noobs** option.

```
proc print data=set1 noobs;
  where shelf=1;
  var shelf cereal sugar;
run;
```

Chris Bilder, STAT 850 Shelf #1 of the cereal data set Cereal sugar

1         Kellog's Razzle Dazzle Rice Crispies         0.35714           1         Post Toasties Corn Flakes         0.07143           1         Kellogg's Corn Flakes         0.07143           1         Food Club Toasted Oats         0.06250           1         Frosted Cheerios         0.43333           1         Food Club Frosted Flakes         0.35484	Shelf	ereal sug	ar
Kellogg's Corn Flakes         0.07143           Food Club Toasted Oats         0.06250           Frosted Cheerios         0.43333	1	ellog's Razzle Dazzle Rice Crispies 0.357	14
1         Food Club Toasted Oats         0.06250           1         Frosted Cheerios         0.43333	1	ost Toasties Corn Flakes 0.071	43
1 Frosted Cheerios 0.43333	1	ellogg's Corn Flakes 0.071	43
	1	ood Club Toasted Oats 0.062	50
1 Food Club Frosted Flakes 0.35484	1	osted Cheerios 0.433	33
	1	ood Club Frosted Flakes 0.354	84
1 Capn Crunch 0.4444	1	apn Crunch 0.444	44
1 Capn Crunch's Peanut Butter Crunch 0.33333	1	apn Crunch's Peanut Butter Crunch 0.333	33
1 Post Honeycomb 0.37931	1	ost Honeycomb 0.379	31
1 Food Club Crispy Rice 0.06061	1	ood Club Crispy Rice 0.060	61

The where statement can be used with most procedures and has additional flexibility. For example, the use of where shelf = 1 and sugar < 0.1 with proc print will print just low sugar cereals on the first shelf. Also, a similar use of this option can be done within the proc print line with data=set1(where=(shelf = 1 and sugar < 0.1)).

Sorting a data set can be useful to see aspects of the data which otherwise may be more difficult to detect without it. Also, some procedures may require that data be sorted prior to their use. Below is how **proc sort** can be used for this purpose.

```
proc sort data=set1;
  by shelf sugar;
```

run;

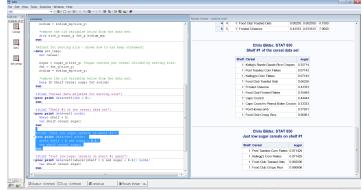
No output will be generated. Rather the data set will be rearrnaged in the work library.

Intro.23

VIEW	TABLE: Work.S	iet1				
	ID	Shelf	Cereal	sugar	fat	sodium
1	10	1	Food Club Crispy Rice	0.0606060606	0	10
2	4	1	Food Club Toasted Oats	0.0625	0.0625	8.75
3	2	1	Post Toasties Corn Flakes	0.0714285714	0	9.6428571429
4	3	1	Kellogg's Com Flakes	0.0714285714	0	10.714285714
5	8	1	Capn Crunch's Peanut Butter Crunch	0.33333333333	0.0925925926	7.4074074074
6	6	1	Food Club Frosted Flakes	0.3548387097	0	5.8064516129
7	1	1	Kellog's Razzle Dazzle Rice Crispies	0.3571428571	0	6.0714285714
8	9	1	Post Honeycomb	0.3793103448	0.0172413793	7.5862068966
9	5	1	Frosted Cheerios	0.43333333333	0.0333333333	7
10	7	1	Capn Crunch	0.4444444444	0.0555555556	7.4074074074
11	11	2	Rice Crispies Treats	0.3	0.05	6.33333333333
12	18	2	Food Club Toasted Oats	0.303030303	0.0454545455	4.5454545455

### My SAS set-up

How do I work with SAS on my computer? If I am only using a single laptop monitor, this is what a screen capture of my working environment looks like:



I try to have two main windows of SAS available at all times. The log window is not shown because its is exactly the same size as the Results Viewer and behind it.

Intro.24

When I am in my office, I take advantage of my portrait oriented monitors and use the following work environment:

• • • • •	3 Po 🛍 🛚	이 🗵 🏢 🖬 🤧 🐯 🚉	0		
cerealsas					
sodium = sodium_mg/s.	ize_g;				
*remove the old varia	ables belo	. from the data act.			
drop size_g sugar_g :	fat_g sodi	am_mg;			
run:					
*Adjust for serving si:	ze - shows	how to use keep sta	atemer	it.	
<pre># data set_temp;     set_cereal;</pre>					
set cereal)					
<pre>sugar = sugar_g/size fat = fat_g/size_g;</pre>	g; *sugar	content per cereal	divid	led by serving size;	
sodium = sodium_mg/s:	ize_g;				
*remove the old varia	ables below	e from the data pat-			
keep ID Shelf Cereal					
run;					
title2 "Cereal data ad	justed for	serving size";			
<pre>Here print data=set1(of run;</pre>	og = 5);				
title2 "Shelf #1 of the					
Sproc print data=set1 n	e dereai di pobs;	sta set";			
<pre>where shelf = 1; var shelf cereal suge</pre>					
run:	a.,				
title2 *Just low sugar	cereals of	n shelf €1°:			
Bproc print data-set1 no where shelf = 1 and s	cobs;				
<pre>where shelf = 1 and :   var shelf cereal sug</pre>	sugar < 0.1	L) -			
run;					
title2 *Just low sugar	cereals or	shelf #1 again":			
Sproc print data=set1(wd	here=(shel:	f = 1 and sugar < 0.	1)) :	ioobs;	
<pre>var shelf cereal sug run;</pre>	ac;				
1					
*Sort the data by shele	d and suga	0			
	d and suga				
*Sort the data by shele Results Viewer - sashtmilhtm		) Toasted Oals	0.062	250 0.062500 8.7500	
Results Viewer - sashtmLhtm		) Toasted Oats		250 0.082500 8.7500 333 0.033333 7.0000	
Results Viewer - sashtmLhtm	1 Food Club	) Toasted Oats			
Results Viewer - sashtmLhtm	1 Food Club	) Toasled Oals heerios	0.433		
Results Viewer - sashtmLhtm	1 Food Club 1 Frosted C	) Toasted Oats heerios Chris Bilder, STAT 8	0.433 350	333 0.033333 7.0000	
<     Results Viewer - sashtmLhtm     4     4	1 Food Club 1 Frosted C	) Toasled Oals heerios	0.433 350	333 0.033333 7.0000	
<     Results Viewer - sashtmLhtm     4     4	1 Food Club 1 Frosted C	o Toasled Oels heerics Chris Bilder, STAT 8 elf #1 of the cereal da	0.433	333 0.033333 7.0000	
<     Results Viewer - sashtmLhtm     4     4	1 Food Chit 1 Frosted C Shelf Cere	o Toasled Oels heerics Chris Bilder, STAT 8 elf #1 of the cereal da	0.433 350 ata se	sugar	
<     Results Viewer - sashtmLhtm     4     4	1 Frosted C 1 Frosted C Shelf Cere 1 Kelo	o Toesled Oets heerios Chris Bilder, STAT & If #1 of the cereal di Ial	0.433 350 ata se	sugar	
<     Results Viewer - sashtmLhtm     4     4	1 Frosted C 1 Frosted C Shelf Cere 1 Kelo 1 Post	o Toasted Oats heerios Chris Bilder, STAT & If #1 of the cereal di al g's Razzle Dazzle Roce Cri	0.433 350 ata se	333 0.033333 7.0000 tt 9.35714	
<     Results Viewer - sashtmLhtm     4     4	1 Food Chill 1 Frosted C Shelf Cere 1 Kello 1 Post 1 Kello 1 Kello	o Toasted Oets heerios Chris Bilder, STAT & If #1 of the cereal di al gis Razzle Dazzle Rice Cri Toasties Com Fiekes	0.433 350 ata se	sugar 0.033714 0.07143	
<     Results Viewer - sashtmLhtm     4     4	1 Food Ckill 1 Frosted C Sheff Cerre 1 Kello 1 Post 1 Kello 1 Food 1 Food 1 Kello 1 Food 1 Food 1 Kello 1 Food 1 Food 1 Kello 1 Food 1	D Toasted Cets heerios Chris Bilder, STAT & Elf #1 of the cereal di el g's Razzle Dazzle Rice Cri Toasties Con Fiskos g's Corn Fiskos	0.433 350 ata se	sugar 0.03714 0.07143 0.07143	
<     Results Viewer - sashtmLhtm     4     4	1 Food Chill 1 Frosted C Shelf Cerc 1 Kello 1 Post 1 Kello 1 Food 1 Foo	D Toasted Oets heerios Chris Bilder, STAT & elf #1 of the cereal di val gis Razzle Dazzle Rice Cri Toastec Com Fiakes gg's Com Fiakes (chia Toasted Oets	0.433 350 ata se	sugar 0.33774 0.07143 0.06250	
<     Results Viewer - sashtmLhtm     4     4	1 Food Club 1 Frosted C Shelf Cere 1 Kello 1 Post 1 Kello 1 Fros 1 Fros	o Toasted Oats hearins Chris Bilder, STAT & If #1 of the cereal di al gis Razze Dazzle Roe Cro Toastes Corn Fiskes gis Corn Fiskes Cub Toasto Oats	0.433 350 ata se	sugar 0.033333 7.0000 st 0.033334 7.0000 0.03334 0.07143 0.07143 0.07143 0.02520 0.043333	
<     Results Viewer - sashtmLhtm     4     4	1 Food Chill 1 Frosted C Sheff Cerre 1 Kello 1 Post 1 Kello 1 Fros 1 Fros 1 Fros 1 Food 1 Frost 1 Cerre 1 Cerre 1 Kello 1 Food 1	D Toasted Oats heerins Chris Bilder, STAT E If #1 of the cereal di al gis Razzk Dazzk Roe Cri Toastes Com Fiskes Cub Toasted Oats ted Chaerios Cub Toasted Oats	0.433 350 ata se	t sugar 0.053714 0.07143 0.07550 0.06550 0.45333 0.35484 0.34444	
<     Results Viewer - sashtmLhtm     4     4	1 Food Ckill 1 Frosted C Sheff Cerr 1 Kello 1 Poos 1 Kello 1 Fros 1 Fros 1 Fros 1 Fros 1 Fros 1 Cerr 1 Cer	Charles Didder, STAT F hearies Chris Bilder, STAT F elf #1 of the cereal di el #1 gls Razzle Diazle Ree Cri Tossies Crin Fielkes gls Can Fielkes Lobar Tossido dia Lobar Tossido Halos Clab Frossio Flakes Clab Frossio Flakes	0.433 350 ata se ispies	t U0743 0.05714 0.05714 0.05714 0.05250 0.04353 0.04533 0.04544 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05564 0.05576 0.05576 0.05576 0.05576 0.05764 0.05764 0.05764 0.05576 0.05576 0.05764 0.05576 0.05564 0.05576 0.05576 0.05564 0.055664 0.055664 0.055664 0.055664 0.055664 0.055664 0.056	
<     Results Viewer - sashtmLhtm     4     4	1 Food Ckill 1 Frosted C Shelf Cerr 1 Kello 1 Food 1 Food 1 Food 1 Cepp 1 Cepp 1 Post 1 Rest 1 Food 1 Food 1 Post 1 Food 1 Post 1 Rest 1 Post 1 Rest 1 Rest	Toasted Oats heartins Chris Bilder, STAT ( Eff #1 of the cereal di al gis Razzo Dazzis Ree Ori Toastes Com Fiskes gis Com Fiskes (Chu Toasted Oats I Chu Frosked Oats I Chu Frosked Fiskes I Chu Frosked Fiskes I Church Peanut Butter C	0.433 350 ata se ispies	st 0.033333 7.0000 st 0.35774 0.07743 0.07743 0.0743 0.07433 0.043333 0.35464 0.44444 0.43333	
Results Viewer - sashtmLhtm	1 Food Ckill 1 Frosted C Shelf Cerr 1 Kello 1 Food 1 Food 1 Food 1 Cepp 1 Cepp 1 Post 1 Rest 1 Food 1 Food 1 Post 1 Food 1 Post 1 Rest 1 Post 1 Rest 1 Rest	Tossied Oals hearins Chris Bilder, STAT E If #1 of the cereal di al gis Rozzb Rozzb Roc Cri Tossies Com Flakes Ogis Rozm Flakes Lick Tossied Oals Ind Chab Tossied Oals Ind Chab Tossied Oals Ind Chab Tossied Flakes In Crunch Peand Butter C Howycomb	0.433 350 ata se ispies	t U0351 0.033333 7 0.000 t 0.05714 0.05714 0.05250 0.04353 0.04553 0.04554 0.045444 0.035644 0.035641 0.	
Results Viewer - sashtmLhtm	1 Food Ckill 1 Frosted C Shelf Cerr 1 Kello 1 Food 1 Food 1 Food 1 Cepp 1 Cepp 1 Post 1 Rest 1 Food 1 Food 1 Post 1 Food 1 Post 1 Rest 1 Post 1 Rest 1 Rest	Transied Outs hearins Chris Bilder, STAT 6 If #1 of the cereal di al go Razzle Dazzle Ree Cri Toasles Com Fieles go Com Fieles Gal Charlos I Chu Toasle Charlos I Chu Toasle Charlos I Chu Toasle Charlos I Chu Toasle Charlos I Chu Charlos I Church's Peand Butter C Horwycomb I Chu Chisty Ree	0.433 350 ata se	t U0351 0.033333 7 0.000 t 0.05714 0.05714 0.05250 0.04353 0.04553 0.04554 0.045444 0.035644 0.035641 0.	
Results Viewer - sashtmLhtm	1 Food Citil 1 Frosted C Sheft Cerc 1 Kello 1 Food 1 Food 1 Food 1 Cape 1 Cape 1 Cape 1 Food 1 Food	Trastied Guids hereiros Chris Bilder, STAT E elf #1 of the cereal di el el di de Charlos gis Carr Fleikes (Cab Trostel Guids Cab Trostel Guids (Cab Trostel Guids) (Cab Trostel Guids) (Cab Trostel Guids) (Cab Charlos (Cab Charlos Fleikes (Cab Charlos Fleikes) (Cab	0.433 350 ata see spies	t ugar 0.35714 0.05714 0.07143 0.07143 0.0720 0.04250 0.04250 0.04250 0.04250 0.04250 0.04504 0.43434 0.04604 0.44444 0.0460404 0.0460404	
<     Results Viewer - sashtmLhtm     4     4	1 Food Citil 1 Frosted C Sheft Cerc 1 Kello 1 Food 1 Food 1 Food 1 Cape 1 Cape 1 Cape 1 Food 1 Food	Transied Outs hearins Chris Bilder, STAT 6 If #1 of the cereal di al go Razzle Dazzle Ree Cri Toasles Com Fieles go Com Fieles Gal Charlos I Chu Toasle Charlos I Chu Toasle Charlos I Chu Toasle Charlos I Chu Toasle Charlos I Chu Charlos I Church's Peand Butter C Horwycomb I Chu Chisty Ree	0.433 350 ata see spies	t ugar 0.35714 0.05714 0.07143 0.07143 0.0720 0.04250 0.04250 0.04250 0.04250 0.04250 0.04504 0.43434 0.04604 0.44444 0.0460404 0.0460404	
<     Results Viewer - sashtmLhtm     4     4	1 Food Child 1 Frosted C Shelf Care 1 Kelo 1 Kelo 1 Fooc 1 Fooc 1 Fooc 1 Cape 1 Cape 1 Cape 1 Cape 1 Fooc 1 Fooc 1 Fooc 1 Fooc	Trastied Guids hereiros Chris Bilder, STAT E elf #1 of the cereal di el el di de Charlos gis Carr Fleikes (Cab Trostel Guids Cab Trostel Guids (Cab Trostel Guids) (Cab Trostel Guids) (Cab Trostel Guids) (Cab Charlos (Cab Charlos Fleikes (Cab Charlos Fleikes) (Cab	0.433 350 ata se ispies Crunch 350 shelf	t ugar 0.35714 0.05714 0.07143 0.07143 0.0720 0.04250 0.04250 0.04250 0.04250 0.04250 0.04504 0.43434 0.04604 0.44444 0.0460404 0.0460404	
<     Results Viewer - sashtmLhtm     4     4	1 Food Chill 1 Frested C Shelf Cere 1 Kele 1 Kele 1 Kele 1 Food 1 Food	Trastied Guis     Trastied Guis     There of the cereal di     def of the cereal di     def     de	0.433 350 ata se ispies Crunch 350 shelf	t sugar 0.03533 7.0000 t sugar 0.03574 0.03544 0.03544 0.03544 0.03544 0.03544 0.03544 0.03544 0.04645 0.00000000000000000000000000000000000	
<     Results Viewer - sashtmLhtm     4     4	1 Food Chill 1 Frosted C Shelf Cere 1 Kello 1 Post 1 Kello 1 Foos 1 Foo	Trastied Guis     Trastied Guis     therris     Chris Bilder, STAT (         if #1 of the cereal di         if         if of the cereal di         if         if and the cereal di         if         if         if and         if	0.433 350 ata se ispies Crunch 350 shelf	t US7r4 0.07743 0.07745 0.07745 0.07745 0.07745 0.07745 0.07745 0.07745 0.	
<     Results Viewer - sashtmLhtm     4     4	1 Food Child 1 Frosted C Sheff Cerr 1 Kello 1 Food 1 Kello 1 Food 1 Foo	Trasted Cuis hearins Chris Bilder, STAT F Eff of the cereal di eff of the cereal di eff of the cereal di eff of the cereal di eff of the cereal of the cereal of the cereal of the cereal of the cereal of the cereal of the cereal of the cereal of the cerea	0.433 350 ata se Spies Crunch 350 shelf su su	t sugar 0.03333 0.03333 7.0000 t sugar 0.03574 0.0743 0.0743 0.0744 0.0743 0.0744 0.0744 0.0744 0.0744 0.0744 0.0745 0.0744 0.0755 0.0744 0.0753 0.03533 0.03533 0.03533 0.03533 0.03533 0.03533 0.035464 0.04545 0.03533 0.035464 0.04545 0.03533 0.03554 0.03554 0.03555 0.03554 0.03555 0.03554 0.03555 0.03554 0.03555 0.03554 0.03555 0.03554 0.03555 0.03554 0.03555 0.03555 0.03555 0.03555 0.03555 0.03555 0.000 0.03555 0.0055 0.	
<     Results Viewer - sashtmLhtm     4     4	1 Food Club 1 Frosted C Shelf Cere 1 Kelo 1 Post 1 Foo 1 Fo	Trasted Cuis hearins Chris Bilder, STAT F Eff of the cereal di eff of the cereal di eff of the cereal di eff of the cereal di eff of the cereal of the cereal of the cereal of the cereal of the cereal of the cereal of the cereal of the cereal of the cerea	0.433 350 spies Crunch 350 shelf su 0.433 350 stata se spies 350 shelf 0.071 0.071	133         0.033333         7.0000           t         0.05714         0.05714           0.07743         0.07743         0.07743           0.07743         0.02743         0.02744           0.04544         0.44544         0.33333           0.04644         0.33733         0.37601           0.00001         0.07601         0.07601           ##1         420         500	