**Please read the following instructions carefully before beginning this lab.**

The SAS code provided as a part of this lab includes windows directory paths that will not exist on your computer. You will need to update the directory paths in order for the SAS code to work.

For this lab, please create a subfolder in the directory where your lab program is stored. You will write all SAS output to this subfolder using the commands taught during the lab.

For items so marked, include the SAS code in the complete SAS program that you will upload to Sakai to document completion of the lab. After you complete the entire SAS program, I recommend you save the program (save often during development) and exit your SAS session. Then, reopen SAS and run the entire program from start to finish, verify the SAS log has no unforeseen ERRORS or WARNINGS, and save the SAS log so that you can upload it to Sakai. The program and log should be named lab-01-PID.sas and lab-01-PID.log where PID is your student PID number.

The completed SAS program will be a long program with essentially unrelated sections of code. I recommend you comment the code for each section so you can find needed sections later (i.e., to study or as a reference for future assignments). Feel free to separate out the sections of SAS code into separate programs after lab. Constructing a “code library” of example SAS code is very useful, if well documented.

This lab is comprised of a series of enumerated sections. Many of them only require you to read and digest information. Others require you to write some SAS code and run it to examine what the code does. For sections that start with “**Include this code in your lab program**” make sure to include the referenced SAS code in the program that you submit at the end of lab. The presence of that code in your lab program is how we will determine if you completed the lab.

I recommend you organize the contents of this lab in a single folder with a subfolder for the SAS output. For example, if you are storing your SAS files in “C:\BIOS511\LAB-01” then create the folder “C:\BIOS511\LAB-01\ODSOUTPUT” for the SAS output.

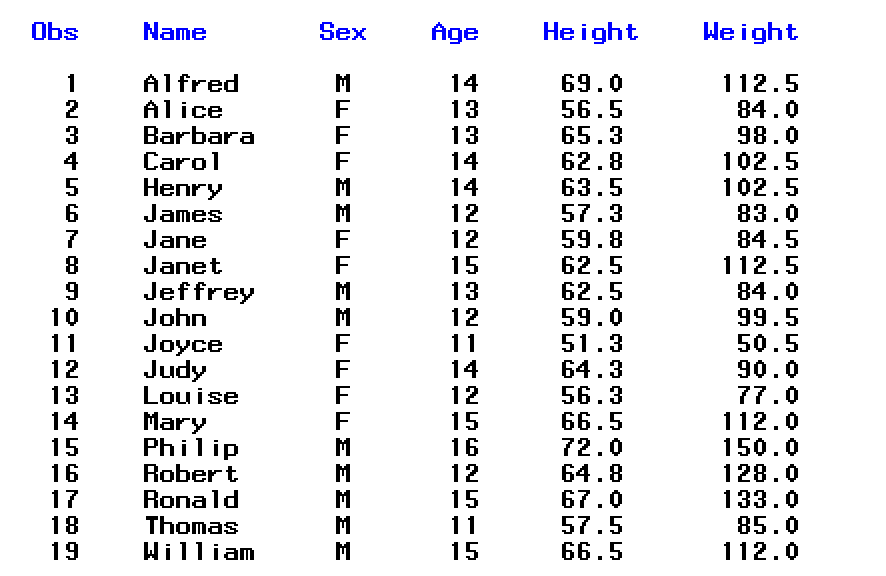
On SAS University Edition, you would instead use folder references like “/folders/myshortcuts/sasdata/ODSOUTPUT”. *Note that correct upper or lower case is required for folder specifications in SAS University Edition*.

### SAS’s Output Delivery System

The output delivery system (ODS) provides more visually appealing procedure output than traditional SAS listing output. In fact, SAS Studio (SS) and SAS University Edition (SUE) do not even produce traditional listing output (b/c it is so antiquated). Consider the following simple SAS program:

**proc** **print** data = sashelp.class; **run**;

When this program in SAS Display Manager (prior to SAS 9.4) the output would look as follows (by default):



Not very pretty! The modern ODS allows programs to direct output to one of several destinations: the listing destination (only supported by SAS Display Manager (SDM)), the PDF destination, the RTF destination, the HTML destination, and to a SAS data set. In addition, the SAS ODS allows programmers to restrict procedure output to just the portions they are really interested in.

**Learning Objectives: In this lab, you will learn how to use the ODS for two purposes:**

**(1) To produce SAS output in various formats: HTML, PDF, output data sets, etc.**

**(2) To control the amount of output that a SAS procedure produces.**

Note that SAS Studio (SS) users and SUE users have the ability to save output from the results tab as a PDF, HTML, or RTF file. The traditional SDM produces HTML output by default unless instructed otherwise. We will focus on using SAS ODS commands to create PDF files, HTML files, RTF files, and data sets programmatically rather than using the point-click interface provided SS and SUE. Using SAS ODS commands is critical when working in an environment where there are multiple programmers on the team or when SAS programs need to be executed frequently using different (perhaps updated) data sets as input.

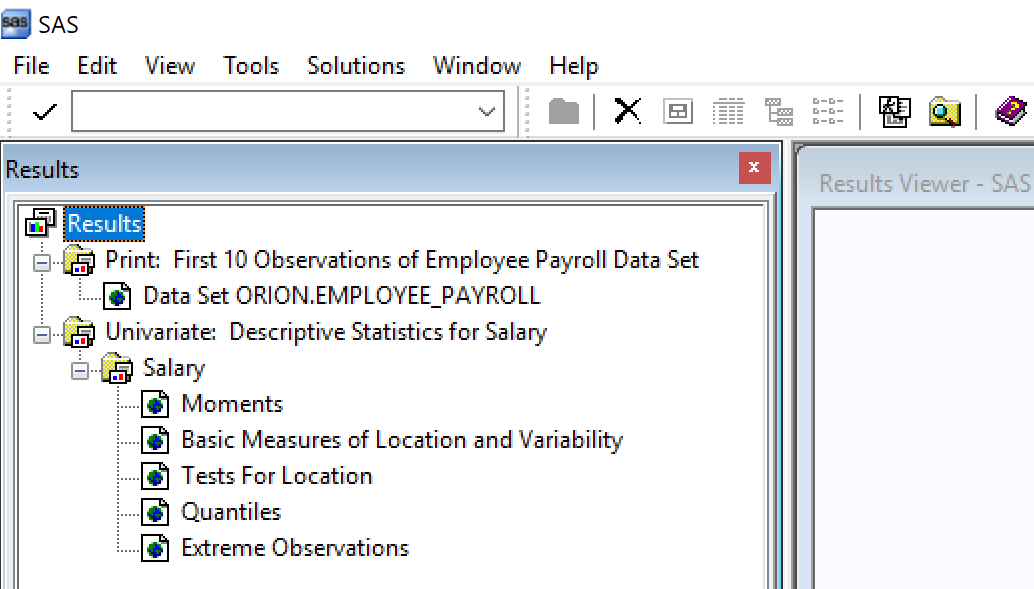
### Output Objects

1. First, you need to know that SAS procedures produce their results in units called *output objects*.
   1. In the SDM, these output objects are listed in the results window (on the left side of the SAS windowing environment) after a procedure step runs.
   2. Unfortunately SS/SUE does not produce such a list, but you can programmatically display the list of objects as described below.
2. **Include this code in your lab program.** Type in and run the code below (better practice than copy/paste). When PROC PRINT runs, it produces one output object. When this simple PROC UNIVARIATE step runs, it produces five output objects.

|  |
| --- |
| **proc** **print** data=orion.employee\_payroll(obs=**10**);  title 'First 10 Observations of Employee Payroll Data Set';  **run**;  **proc** **univariate** data=orion.employee\_payroll;  var salary;  title 'Descriptive Statistics for Salary';  **run**; |

1. *This item is only applicable if you are using the SDM*. With this SAS modality, you can explore what one see in the *Results Window*.

Figure 1: Image of the Results Window



Notice that there is one entry in the Results tree for each PROC step that was run. If you expand the tree as much as possible, the lowest-level items (one under Print and five under Univariate) are the output objects.

Double-click on one of these output object entries, such as **Tests for Location** under **Salary** under **Univariate: Descriptive Statistics for Item Salary**. Notice that the Results Viewer window scrolls to display the table you selected. The Results viewer is showing you HTML output as by default the Display Manager creates a temporary HTML file that is deleted when the SAS session closes.

Now go back to the Results window. Right-click on one of the output object entries, and select Properties at the bottom of the action list. The Properties window for an output object tells you its name (e.g., TestsForLocation) and some other potentially useful information.

1. **Include this code in your lab program.** One can programmatically use the ODS TRACE ON; and ODS TRACE OFF; statements around your PROC step code to have output object information printed out to the SAS log. This code works in both the SDM and SS/SUE. Try the code from above with ODS TRACE ON; and ODS TRACE OFF; now surrounding it:

|  |
| --- |
| ods trace on;  **roc** **print** data=orion.employee\_payroll(obs=**10**);  title 'First 10 Observations of Employee Payroll Data Set';  **run**;  **proc** **univariate** data=orion.employee\_payroll;  var salary;  title 'Descriptive Statistics for Salary';  **run**;  ods trace off; |

Notice that information on PROC PRINT’s one output object and PROC UNIVARIATE’s five output objects is written to the SAS Log. One generally uses ODS TRACE ON; and ODS TRACE OFF; to get information about ODS output objects but those statements are rarely included in a “final product” that would be turned in (after this lab).

The default ODS objects produced by PROC UNIVARIATE are named Moments, BasicMeasures, TestsforLocation, Quantiles, and ExtremeObs. Every ODS object has a unique name and objects can be selected for or excluded from the output using their unique name.

### ODS Destinations

1. ODS enables you to instruct SAS that you want to *deliver* particular output objects to particular *destinations*. For example, you might want to produce a two-way frequency table for presentation on the web (i.e., as HTML output), you might want to produce a table of descriptive statistics to include in a MS Word document (i.e., an RTF document), and/or you might want to mail either of these to a colleague.
2. To specify the form in which you want your output delivered by SAS, you use *ODS* *destinations*. Some of the destinations currently available, along with the ODS statement used to request each one, are listed in the following table:

|  |  |  |
| --- | --- | --- |
| Destination | Purpose | ODS Statement |
| HTML files | For display on the web | ODS HTML … ; |
| PDF files | For printing from the web | ODS PDF … ; |
| RTF files | To use in word processing | ODS RTF … ; |
| SAS Output window | SAS listing – not in SS/SUE | ODS LISTING … ; |
| Output data sets | Produce an output dataset | ODS OUTPUT … ; |

These are the destinations you will work with in today’s lab.

1. When you run a SAS procedure, it sends its output objects to whatever ODS destinations are currently open (any or all of them can be open).

**For the SDM**, by default, in SAS 9.4 the HTML destination is open for the SDM but the file created is temporary and will be deleted upon closing your SAS session. If you want to create a permanent HTML file, you must instruct SAS to do so. If you want some other destination to be used (i.e., PDF), you have to open it before submitting your PROC step(s). To see the output, you need to close the destination after the PROC step(s).

As noted above, the HTML destination is active when you start SAS. That means that all output objects will be delivered to the HTML destination unless you instruct SAS to change that behavior. Even if you open the PDF destination, SAS will still append objects to the HTML destination unless you instruct SAS to do otherwise.

If you do not want to create an HTML file (or want to stop appending to an HTML file), you need to submit the statement ODS HTML CLOSE; The HTML destination is closed from that point forward until you instruct SAS to open it back up. This line of code is commonly placed at the top of SAS programs written using the SDM.

**For SS/SUE**, by default procedure output is written to the Results tab which is also a temporary output storage facility. When you close SS/SUE, the Results tab output is deleted. In fact, any time you submit code in SS/SUE the Results tab is refreshed to only show newly produced output. The output in the Results tab is displayed as HTML but can be downloaded as HTML, PDF, or RTF. You cannot suppress output from being written to the Results tab even if you are separately directing SAS to create permanent files with ODS statements. One can think of the Results tab as always being there for quick reference.

After the SAS code from step 4, SDM users please add the ODS HTML CLOSE; statement to your SAS program so that there will be no active ODS destinations (from this point forward) in your SAS program. At this point, your SAS program (excluding the header section and preliminary options) should look as follows:

|  |
| --- |
| /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  SAS Code for Section # 2  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  .  .  .  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  SAS Code for Section # 4  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  .  .  .  ODS HTML CLOSE; |

**USE THIS SECTION COMMENTING CONVENTION FOR THE REMAINDER OF THE LAB.**

**For SDM users**, before going on (and after submitting the code ODS HTML CLOSE;), once again submit either the PROC PRINT step or the PROC UNIVARIATE step from items 2 or 4 (highlight just the step you want to run) and observe the message in your SAS log.

You should see the message: WARNING: No output destinations active. One will receive such a message any time they run a procedure that produces printed ODS output when there are no open destinations.

Since SS/SUE *always* writes to the Results Tab, this WARNING is not generally encountered with those SAS modalities.

1. Typically one uses ODS destination statements in pairs, as follows (NOTE: do not try to submit this code – it is incomplete):

|  |
| --- |
| ods pdf file='name of the PDF file you want to create';  /\*<procedure steps go here> \*/  ods pdf close; |

The first ODS PDF statement instructs SAS that you want your procedures to write their output into a PDF file. The second ODS PDF statement closes the PDF destination (i.e., instructs SAS that no more output will be written to this PDF so that the actual PDF file can be formally created).

Note that this code does not affect any other ODS destination that may be active. That is to say that one can create PDF, RTF, and HTML files all at the same time but only one file of each type may be created at a time.

1. **Include this code in your lab program.** Type in the example below. Do not copy the code. Type it yourself. Before you run it, make sure you have created the folder specified as a location for the PDF file.

|  |
| --- |
| ods pdf file = 'Path to Your “output” folder/freq9.pdf';  **proc** **freq** data=orion.customer;  tables country gender country\*gender;  title 'Frequency Distributions and Cross-tabulations';  **run**;  ods pdf close; |

After running the code, be sure to check the log for errors and correct any programs before moving on. Be sure to close the PDF file when you are finished viewing it. In general if a PDF file is open for viewing (or RTF/HTML file), and you wish to overwrite the contents (perhaps to add more output), SAS cannot overwrite a file unless it is closed.

If you using the SDM, the PDF file you created will probably pop open automatically in the results viewer window (note: sometimes you will have to close the results viewer before you are able to regenerate results as this effectively keeps the PDF file open).

If you are using SS/SUE, you will have to go into the folder on your computer where the file was created and open the file outside of SAS. The PDF file will not open automatically within SS/SUE. You must still remember to close the file if you wish to recreate it or modify it.

1. **Include this code in your lab program.** Now produce the same output in RTF (rich text format) form, ready for your favorite word processing software. All you need to do is use the same program, but replace “pdf” with “rtf” in the three places where it occurs. Note that instead of the “.rtf” extension, you can instead specify a “.doc” file extension, which corresponds to the extension for an older version of Microsoft Word.

|  |
| --- |
| ods rtf file = 'Path to Your “output” folder/freq10.rtf’;  **proc** **freq** data=orion.customer;  tables country gender country\*gender;  title 'Frequency Distributions and Cross-tabulations';  **run**;  ods rtf close; |

After running the code, be sure to check the log for errors and correct any programs before moving on. Microsoft Word will open and display your tables, ready for any text you might want to add or modifications you might want to make to the tables.

Note: To view correctly on a MAC machine you must open in word processing software that can correctly read files in RTF format. Some MAC word processing programs will incorrectly render the RTF code making titles and footnotes not appear correctly.

In your RTF file, did you notice that your title appeared in the header area rather than as part of the regular text? For more flexibility, you’ll often want titles and footnotes to be with the regular text rather than in the header and footer (especially if you want to copy the title, table, and footnotes to paste into another document). To accomplish this, use the BODYTITLE option in your opening ODS RTF statement, as follows:

ods rtf file=”…” bodytitle;

Repeat the code above with this option and name the file “freq10\_bodytitle.rtf”.

Does “The FREQ Procedure” appear near the top of the output you just produced? If so, you can turn it off in future runs of the program by submitting the global statement:

ODS NOPTITLE;

To return to the default behavior, you can submit the global statement:

ODS PTITLE;

***Incorporate this feature into your final program by adding the ODS NOPTITLE; statement just below the program header. You will seldom want the procedure title in the output.***

1. **Include this code in your lab program.** Creating an HTML file is as simple as making a PDF or RTF file.

|  |
| --- |
| ods html file = 'Path to Your “output” folder/freq11.html;  **proc** **freq** data=orion.customer;  tables country gender country\*gender;  title 'Frequency Distributions and Cross-tabulations';  **run**;  ods html close; |

Again, the only change from the earlier program is substituting “html” for “pdf” in three locations. If you are using the SDM, the produced HTML file will pop open in the results viewer. If using SS/SUE, you can open it outside of SS/SUE using any web browser.

### Limiting the Output Produced by a Procedure

1. Sometimes you are only interested in a few of the output tables produced by a SAS procedure. Some SAS procedures, such as PROC UNIVARIATE, yield pretty lengthy output! ODS gives you a way to cut down on undesired output.

Use the ODS SELECT statement to specify only the output objects you want.

Use the ODS EXCLUDE statement to exclude only the output objects you don’t want.

The syntax of the ODS SELECT/EXCLUDE statements are as follows:

ODS SELECT <output object names>;

ODS EXCLUDE <output object names>;

In order to find the name of the desired ODS output object, use one of the two methods discussed above (using ODS TRACE ON/OFF is my preferred approach); or simply look in the SAS documentation. For example, google “SAS 9.4 PROC FREQ ODS Table Names”. Once you know the names of the output objects you want to keep, you can use those names in the ODS SELECT or ODS EXCLUDE statement.

1. **Include this code in your lab** program. For the code below, compare use of the ODS SELECT and EXCLUDE statements.

|  |
| --- |
| ods pdf file = 'Path to Your “output” folder/select\_exclude13.pdf';  title "Moments and Extreme Observations";  ods select moments extremeobs;  **proc** **univariate** data=orion.employee\_payroll;  var salary;  **run**;  title "Basic Measures, Tests for Location, and Quantiles";  ods exclude moments extremeobs;  **proc** **univariate** data=orion.employee\_payroll;  var salary;  **run**;  title "All Univariate Procedure Output";  **proc** **univariate** data=orion.employee\_payroll;  var salary;  **run**;  ods pdf close; |

Note the procedure output that is included from the third PROC UNIVARIATE step. What does this imply about ODS SELECT and EXCLUDE statements?

One can also use ALL and NONE instead of specific object names. The default select list is ALL and the default exclude list is NONE. When ODS SELECT NONE; is submitted or ODS EXCLUDE ALL; no output is directed into the file being created. This is useful if all one wants to do is produce a data set of the output for further processing.

**Creating Output Data Sets Using ODS**

1. You can also use ODS to obtain a data set containing any procedure output. Many procedures provide options or statements for requesting output data sets, but these are limited to certain results. With ODS, you can get any procedure output in the form of a SAS data set.

To obtain an output data set, you use the ODS OUTPUT statement. That is to say, the OUTPUT destination is used to deliver ODS objects as SAS data sets for further processing. To use ODS OUTPUT, you need to reference the output objects by name.

The format of the SAS dataset produced does not always resemble the ODS object you see in printed output.

1. **Include this code in your lab program.** Let’s say that you want to deliver the “Quantiles” output object from an analysis of the variable “salary” to a temporary data set named “salary\_quant” (remember that one can get the name of the object from looking at the Results Window, by using ODS TRACE ON/OFF; or by googling “SAS 9.4 PROC UNIVARIATE ODS Tables”).

|  |
| --- |
| ods pdf file='Path to Your “output” folder/ods\_output\_15.pdf';  ods pdf select quantiles;  ods output quantiles = salary\_quant;  **proc** **univariate** data=orion.employee\_payroll;  var salary;  **run**;  title "Quantiles for Salary Variable";  **proc** **print** data = salary\_quant; **run**;  ods pdf close; |

Here there is no real reason to deliver the “Quantiles” ODS object to a SAS dataset. However, this general strategy will prove useful over the semester when you need to take results from one procedure and post-process them before display or to use as input into another procedure.

Find the WORK library in the explorer window (SDM) or the Libraries menu (SS/SUE University Edition), and double-click on the “salary\_quant” data set to see the temporary data set that was just created.

Note that many procedures (including the UNIVARIATE Procedure) have mechanisms to delivery specific analysis results to SAS datasets separately from the use of ODS OUTPUT statements. We will learn more about this soon. They key point here is that *any* procedure output can be delivered into a SAS data set using ODS OUTPUT.

**ODS Styles**

1. ODS styles control the colors, fonts, and font sizes used to create ODS output. SAS provides a default style for each ODS destination. For example, you might have noticed that the HTML files you’ve created with ODS have used a combination of blue and gray backgrounds. That is the default style for HTML files in when using the SDM.

If you want to see the additional styles that are available, make the following selections from the SAS menu bar (SDM):

Tools 🡪 Options 🡪 Preferences… 🡪 Results tab

You will see a pull-down list for selecting a default HTML style. Click on the pull-down list to see the available styles; scroll to see the entire list.

For SS/SUE, you can do the following:

Preferences… 🡪 Results tab

You will see a pull-down list for selecting a default style for HTML, PDF, and RTF files. The pull-down lists contain all the styles that are available for SS/SUE.

Both SDM and SS/SUE can programmatically produce a list of available styles using the syntax:

**proc** **template**;

list styles;

**run**;

No style previewer application is available but you will find them online if you google “SAS 9.4 ODS Styles Gallery”.

1. **Include this code in your lab program.** Try out at least two different styles but (in general) only submit assignments using styles that do not require a large amount of dark ink when printed (e.g., do not select a style that has a dark background). Substitute the name of each chosen style in the following code and include your favorite style in the submitted program:

|  |
| --- |
| ods pdf file = 'Path to Your “output” folder/fav\_style17.pdf' style=journal;  **proc** **freq** data=orion.customer;  tables country gender country\*gender;  title 'Frequency Distributions and Crosstabulations';  **run**;  ods pdf close; |

For RTF and PDF files, which are typically printed on non-color printers, styles such as Minimal, Printer, and sansPrinter work well. The Journal style is especially nice for a professional look.

Users can create their own style templates from scratch or by modifying an existing template using The TEMPLATE Procedure. Such topics are beyond the scope of this course.

**Page Composition with ODS: The STARTPAGE Option**

1. **Include this code in your lab program.** You can use combinations of the options described above, along with a new one, STARTPAGE, to construct nice pages of combined procedure output. The STARTPAGE option controls when SAS inserts a page break (and therefore goes to a new page).

By default, STARTPAGE=YES, and SAS inserts a new page at the beginning of each procedure’s output. In contrast, STARTPAGE=NO is set so that output from multiple procedures goes onto one page. The STARTPAGE=NO option instructs SAS to only go to a new page if the current page is filled or when you specify STARTPAGE=NOW. So if you have set STARTPAGE=NO and want to later force SAS to go to a new page, you can submit the following statement at the appropriate location:

ODS <Destination> STARTPAGE=NOW;

Note that STARTPAGE can be used with either the PDF or RTF destination.

Run the following code first without the STARTPAGE=NO option set and then with the STARTPAGE=NO option set. Lastly, add ODS PDF STARTPAGE=NOW; between the PROCs and compare the result to the first case.

|  |
| --- |
| ods pdf file = 'Path to Your “output” folder/start\_page18.pdf'  style=journal  startpage=no;  title1 'Descriptive Statistics for Price Variable';  **proc** **means** data=orion.employee\_payroll;  class employee\_gender;  var salary;  **run**;    ods select extremeobs;  **proc** **univariate** data=orion.employee\_payroll;  var salary;  **run**;  ods pdf close; |