%DIAGNOS: a Macro for a Quick Look into the Inside of Your Data Set
Yin You, Merck & Co., Inc., West Point, PA

Abstract
Data diagnostics is an essential step in statistical programming and analysis, as well as data edit checking. This paper will introduce one simple SAS® macro for statistical programmers and statisticians to have a quick look into the data of interest. This macro was written to provide lists of frequency counts for categorical variables and to summarize statistics for continuous variables for the entire or the selected variables of a data set. Also, the macro can identify numeric variables as categorical variables based on a given criteria. The macro output can display the variables according to their order in the input data set.

Introduction
The purpose of data diagnostics is to check the data and understand the data set better before performing any programming or statistical analysis. SAS software has two commonly used procedures to provide summary information for variables. Proc FREQ is widely used for listing frequency counts of categorical variables and Proc UNIVARIATE is used for summarizing statistics of continuous variables. Some macros have been developed (Young, A.C and Zhou, S.X., 1999) to generate and report the summary statistics of the treatment groups for either continuous or categorical variables or both.

This paper presents a SAS macro that uses Proc FREQ or Proc UNIVARIATE to produce an SAS output which handles categorical or continuous data correspondently to look at the entire or the selected variables of a data set. The output can display the variables according to their order in the input data set.

%Diagnos Macro
%DIAGNOS(
  iDS=,
  iVar=,
  iType=,
  iWhere=,
  iDrop=,
  iBy=,
  iOpt=,
  iCnt=10);

Eight keyword parameters are included in this macro. Only the parameter $iDS$ is required. The rest of the parameters are optional.

- **iDS:** Required, name the SAS data set diagnosed.
- **iVar:** Optional, input diagnosing variables separated by a space, if missing, then all variables in the data set are diagnosed and the order of output is based on the variable order in the data set.
- **iType:** Optional, specify the diagnosing variables’ type, N for numerical variables only; C for character variables only. If it is missing then both numerical and character variables will be diagnosed.
- **iWhere:** Optional, specify the conditions to subset the original data set for diagnosis. It is executed before the DROP statement.
- **iDrop:** Optional, variable names to omit from the original diagnosed data set. It is executed after the WHERE statement.
- **iBy:** Optional, variable names to define subgroups for the diagnosis.
- **iOpt:** Optional, options of PROC UNIVARIATE, such as PLOT, NORMAL.
- **iCnt:** Optional, counter to use frequency tables for the numerical variable if the possible value counter is not more than. The default is 10.
Function

%DIAGNOS can be used in the following cases:

- For categorical data, it provides frequency counts of observations in each category. Missing is treated as a category.
- For continuous variables, it prints out summary statistics, as well as, normality check and plots, including a horizontal bar chart, a box plot and a normal probability plot.
- If a continuous variable has only a few values, say, less than 10, the continuous variables will be treated as a categorical variable for frequency counts. The number can be reset through the iCNT parameter.
- DROP, WHERE and BY statement can be specified by user.

Example

1. This macro can easily diagnose all variables in the SAS data set. The data set name is the only required parameter for the macro. You may specifies a one-level name or a two-level name and with data-set-options. For example:
   ```sas
   %DIAGNOS(iDS=myds);
   %DIAGNOS(iDS=mylib.admin);
   %DIAGNOS(iDS=myds(keep=a: where=(age<50));
   ```

2. Diagnose all character variables in the SAS data set when you are interested in the character variables only. For example,
   ```sas
   %DIAGNOS(iDS=myds, iTYPE=c);
   ```
   Proc FREQ may produce a one-way frequency table for each variable in a data set. However, the one-way frequency table for numeric variables is too much to read if the variable had many outcomes. The macro can list the one-way frequency table for all character variables when you specify the type of variables you are interesting.

3. Diagnose all numeric variables in the SAS data set. For example,
   ```sas
   %DIAGNOS(iDS=myds, iTYPE=n);
   ```
   Proc UNIVARIATE may produce descriptive statistics for each numerical variable in a data set. In the clinical trial data sets, there are some classification or code variables often treated with the numerical type. In those cases, the counts and percentages of the variables are more useful than mean, standard deviation, or quantiles. A macro parameter is set up to get frequency tables for the numerical variable if the possible value counter is not more than 10. The macro parameter can be reset as need.

4. In order to diagnose all numeric variables, you can run the macro and get the detail on the distribution of each numerical variable, for example,
   ```sas
   %DIAGNOS(iDS=myds, iTYPE=n, iCNT=20);
   ```
   It will give the frequency tables for variables with distinct outcome values less than 20, and descriptive statistics on the rest numeric variables in the SAS data set.

5. Diagnose the specified variables, in the input SAS data set. For example,
   ```sas
   %DIAGNOS(iDS=myds, iVAR=age sex );
   %DIAGNOS(iDS=myds, iVAR=base:, iTYPE=n );
   ```

6. Excluded some variables in the input SAS data set for subset and diagnose the rest variables by subgroup. For example,
%DIAGNOS(iDS=myds,
   iDROP=alloc sex,
   iWHERE=%str(treated='Yes'),
   iBY=TRT);

7. Diagnose numerical variables in the input SAS data set and do normal check, box plot, bar chart and
   normal plot, and get the counts and percentage for the categorical variables, for example,
   %DIAGNOS(iDS=myds,
       iVAR=age siteno basebmi,
       iOPT=plot normal);

Conclusion
The demonstrated SAS macro presents an easy and quick way to diagnostics the data set for programmer
and statistician. The output of the macro provides the results from Proc FREQ or Proc UNIVARIATE.

Reference
Young, A.C and Zhou, S.X., 1999, “Using SAS Macro to Include Statistics Output in Clinical Trial
Summary Table”, Annual SAS Users Group International Conference, Paper 226.

Acknowledgements
SAS is a registered trademark of SAS Institute Inc..

The author would like to thank Susan Neidlinger and Guowei Wu for supporting and reviewing this paper.

Contact information
The author may be contacted at:

Yin You
Merck & Co., Inc.
HM-209
P.O Box 4
West Point, PA 19486-0004
Phone: 215-328-2081
Email: Yin_you@meck.com