%CHECKDATA: An Enhanced Data Diagnostic Macro

John Stanmeyer, Trade Resources Company, Washington, D.C.

ABSTRACT

Often when receiving a large dataset from a client it is useful to perform a review of the summary statistics of each numeric variable, as well as to do a frequency/rank tabulation on each character variable. PROC MEANS and PROC FREQ only go so far to this end (especially when dealing with date fields). Therefore, Trade Resources Company has developed a macro called %CHECKDATA to easily run a comprehensive diagnostic of any data set.

THE PROBLEM:

When receiving datasets with dozens (or hundreds) of variables and thousands (or hundreds of thousands) of observations, taking a high-level view can really help in the identification of any upstream problems that may have crept into the data.

To this end, one might use the following two procedures:

PROC MEANS DATA=*dataset* MIN MEAN MAX SUM N NMISS; RUN;

and

PROC FREQ DATA=*dataset* ORDER=FREQ; RUN;

However, these simple diagnostics do not help identify certain potential data problems. For example, for most purposes the PROC MEANS statistics in their native format are not useful with date variables. Quite often when working with time-related databases we would like to know, at a glance, the MIN and MAX of various date fields in date format, not as SAS integers such as 15,864.

Furthermore, while it is nice to know the total number of observations (N), and the total number of missing values (NMISS) in a dataset, we might also want to know the number of negative, zero and positive values for each variable. Knowing such information helps recognize any problems (or just the protocols) related to currency fields such as credits, debits, and adjustments. We might also want to know the total number of non-missing values.

Finally, the minimum (MIN) and MEAN values are not always the most helpful statistics which can be generated. Suppose, for example, a sales dataset has a rebate variable. Usually, this variable is zero, but it will have a positive value if a particular sale had a rebate. Suppose we want to know, at a glance, the range of rebates and the average rebate for those sales having rebates. The MIN function will return a zero. What we really want is the minimum positive value—a “MINPOS” as it were. Similarly, the MEAN function will return an average that is skewed downward by the many records which have zeroes in the rebate field. What we really want is a mean of the positive values—a “MEANPOS.”

Regarding PROC FREQ reports on the character variables, sometimes a certain field (such as an invoice number) might have so many possible values that a PROC FREQ would go on for dozens of pages. What we’d really like to see is the top-ranking handful of values for each character variable, along with the number of missing values, and perhaps the length of the longest value.

THE SOLUTION:

In order to achieve the above improvements, Trade Resources Company has developed a macro called %CHECKDATA that will automatically process any dataset and produce the "enhanced" PROC MEANS report on the numeric fields, along with a slightly enhanced PROC FREQ report on the character fields. This tool, in conjunction with PROC COMPARE, aids in the identification of changes to, or problems in, data received from upstream.

THE OUTPUT OF THE MACRO:

First, the macro produces a table with the following columns:

- Field name
- Minimum value
- Minimum positive value
- Mean value
- Mean of positive values
- Maximum value
- Sum of all values
- Number of observations with missing values
- Number of observations with negative values
- Number of observations with zero values
- Number of observations with positive values
- Total number of observations populated (i.e., non-missing)

In this report, date fields will be formatted as dates instead of integers.

For the PROC FREQ section, each character variable's frequency tabulation is printed on a new page. Up to 44 unique values are tabulated and sorted by rank. If there are more than 44 values, the remainder are collapsed into a single line item called "XX other values" (XX will show the quantity of other unique values). If there are any missing values, they will become the very first line item in the report, shown as "***Missing***", in order to draw attention to the fact that missing values are present.

HOW TO USE THE MACRO:

First, you must %INCLUDE the macro code. Save the CHECKDATA.SAS file (available in the CODE directory of the CD-ROM accompanying this paper) to your active SAS directory. Then submit the following coding:

```
%INCLUDE 'checkdata.sas';
```

There are two ways to call the macro. The first way has no parameters and runs the macro against the most recent data set (_LAST_):

```
%CHECKDATA;
```

The second way takes a dataset name as a parameter and runs the macro on that dataset:

```
%CHECKDATA(libname.dataset);
```

Under either method, the macro will know automatically if there are no numeric variables, or if there are no character variables. In the former case, the numeric statistics portion of the macro will not execute. In the latter case, the frequency tabulation portion of the macro will not execute.

HOW THE MACRO WORKS:

- If no dataset name was passed as a macro parameter, the macro uses the _LAST_ dataset by default.
- A PROC CONTENTS is dumped into a temporary dataset and the

fields FORMAT, INFORMAT, and TYPE are used to identify numeric and date fields. A running list of date fields is generated and finally dumped into a macro variable for later use.

- Several PROC MEANS are run: one to get the summary statistics normally available to PROC MEANS, and for each additional statistic (e.g., MINPOS), a separate data step and PROC MEANS are run.
- The PROC MEANS outputs are dumped into temporary datasets, concatenated into a single dataset, and transposed.
- The numeric data is now actually contained in character fields, which allows the mixing of date and decimal formats within the columns.
- In printing the numeric stats, all numeric fields are formatted using the BEST format to allow full precision while at the same time eliminating trailing zeroes. All date fields are formatted MMDDYY10.
- In printing the character variable frequency tables, first the maximum length is calculated and stored in a macro variable for display in the title.
- Percent and cumulative percent are manually recalculated so that the "other values" bucket may be used if more than 44 unique values are found. Note that 44 is an arbitrary number and can be replaced in the code with any value N for the number of top unique values desired. (Also, a nearby 43 must be replaced with N-1).
- Finally the macro "cleans up" after itself. It deletes the temporary datasets that it created and resets the _LAST_ dataset name to be the same as it was before the macro was called. It also restores certain system options (NOTES, SYMBOLGEN, MPRINT, and MLOGIC) which were deactivated for the purpose of keeping the log short and clean.

CONCLUSION:

The %CHECKDATA macro is a powerful tool for generating comprehensive at-a-glance diagnostics. It can be used to quickly and easily produce multiple reports based on any dataset, large or small, to give the analyst a quick method of spotting problems or trends in the data.

AUTHOR CONTACT:

Your comments and questions are valued and welcome. Contact the author at:

John W. Stanmeyer
Trade Resources Company
1634 Eye Street, N.W.
Washington D.C., 20006
Phone: (202) 659-0920
Fax: (202) 785-0160
E-mail: stanmeyer@traderes.com

APPENDIX A: THE CODE

```
%MACRO CHECKDATA(INPUTDSN);

OPTIONS NONOTES NOSYMBOLGEN NOMPRINT NOMLOGIC;

DATA _NULL_; FILE LOG;
PUT "----------------------------------------";
PUT "RUNNING DATA CHECK ON DATASET &INPUTDSN.";
PUT "----------------------------------------";
RUN;

%LET DATEVARS=;

%IF "&INPUTDSN." = "" %THEN %LET INPUTDSN=_LAST_;

proc contents data=&INPUTDSN. noprint out=datevars ;

proc sort data=datevars; by varnum; run;

data datevars; set datevars;
     if _N_=1 then call
symput("INPUTDSN",TRIM(LEFT(TRIM(LEFT(LIBNAME))!!"."!!TRIM(LEFT(MEMNAME)))));

data datevars
     charvars (keep=name length)
     numvars (keep=name);
      set datevars;
     if type=2 then output charvars; else
     if (index(upcase(informat),"DATE") or index(upcase(format),"DATE") or
         index(upcase(format),"DAY") or index(upcase(format),"DOW") or
        (index(upcase(format),"Y") and index(upcase(format),"M") and
index(upcase(format),"D")))
          then output datevars; else
      if type=1 then output numvars;
     RUN;

data dum; dum=1; run;

data dum; set dum numvars; run;

data _null_; set dum nobs=nobs;
     call symput('nobs',nobs); stop; run;

%if %eval(&nobs.) > 1 %then %do;

data datevars; set datevars end=last;
     LENGTH DATEVARS OLDDATEVARS $200;
     RETAIN OLDDATEVARS;
     if _N_=1 then datevars=TRIM(LEFT(NAME));
     ELSE DATEVARS=TRIM(LEFT(OLDDATEVARS))!!" "!!TRIM(LEFT(NAME));
     OLDDATEVARS=DATEVARS;
     IF LAST THEN CALL SYMPUT("DATEVARS",DATEVARS);
     RUN;

proc delete data=datevars; run;

data _NULL_; set &INPUTDSN. NOBS=NOBS; CALL SYMPUT("NOBS",TRIM(LEFT(NOBS))); STOP; RUN;

%IF "&DATEVARS." NE "" %THEN %DO;

DATA CHECKDSN; SET &INPUTDSN.;
     FORMAT &DATEVARS. MMDDYY10.; RUN;

%END;

%ELSE %DO;
  DATA CHECKDSN; SET &INPUTDSN.;
%END;
```

```
proc means data=CHECKDSN min max mean n nmiss sum noprint; *maxdec=2;
var _numeric_;
output out=M5a (drop=_type_ _freq_ rename=(_stat_=STAT));

proc means data=CHECKDSN noprint maxdec=2;
var _numeric_;
output out=M5b (drop=_type_ _freq_) sum=;
data m5b; set m5b; stat="SUM";

data m5c; set CHECKDSN (keep=_NUMERIC_);
     array vars _Numeric_;
     do over vars; if ((-1)*vars) > 0 then vars=1; else vars=0; end;
proc means data=m5c noprint maxdec=2;
var _numeric_;
output out=M5c (drop=_type_ _freq_) sum=;
data m5c; set m5c; stat="NUMNEG";

data m5d; set CHECKDSN (keep=_NUMERIC_);
     array vars _Numeric_;
     do over vars; if vars>0 then vars=1; else vars=0; end;
proc means data=m5d noprint maxdec=2;
var _numeric_;
output out=M5d (drop=_type_ _freq_) sum=;
data m5d; set m5d; stat="NUMPOS";

data m5e; set CHECKDSN (keep=_NUMERIC_);
     array vars _Numeric_;
     do over vars; if vars=0 then vars=1; else vars=0; end;
proc means data=m5e noprint maxdec=2;
var _numeric_;
output out=M5e (drop=_type_ _freq_) sum=;
data m5e; set m5e; stat="NUMZERO";

data m5f; set CHECKDSN (keep=_NUMERIC_);
     array vars _Numeric_;
     do over vars; if vars=. then vars=1; else vars=0; end;
proc means data=m5f noprint maxdec=2;
var _numeric_;
output out=M5f (drop=_type_ _freq_) sum=;
data m5f; set m5f; stat="NUMMISS";

data dum; set CHECKDSN (keep=_NUMERIC_);
     array vars _Numeric_;
     do over vars; if vars LE 0 THEN vars=.; end;

proc means data=dum noprint maxdec=2;
var _numeric_;
output out=M5g (drop=_type_ _freq_) min=;
data m5g; set m5g; stat="MINPOS";

proc means data=dum noprint maxdec=2;
var _numeric_;
output out=M5h (drop=_type_ _freq_) mean=;
data m5g; set m5g; stat="MEANPOS";

%IF "&DATEVARS." NE "" %THEN %DO;

data M5a; Set M5a M5b M5c M5d M5e M5f M5g M5h; IF STAT="STD" THEN DELETE;
     format _numeric_ best.;
     FORMAT &DATEVARS. MMDDYY10.; RUN;

%END;

%ELSE %DO;

data M5a; Set M5a M5b M5c M5d M5e M5f M5g M5h; IF STAT="STD" THEN DELETE;
     format _numeric_ best.;
     RUN;
%END;

proc transpose data=M5a out=m5a; var _all_;
```

```
data m5a; format COL5 $16.; set m5a; _label_=""; drop _label_; run;

data m5a; set _last_ (firstobs=2 rename=(_NAME_=FIELD
                                COL1=N COL2=MIN COL3=MAX COL4=MEAN COL5=SUM
                                       COL6=NUMNEG COL7=NUMPOS COL8=NUMZERO COL9=NUMMISS
COL10=MINPOS COL11=MEANPOS));

data m5a (drop=charn cnummiss cnumneg cnumpos cnumzero);
            set m5a (rename=(N=CHARN
                             NUMNEG=CNUMNEG
                             NUMPOS=CNUMPOS
                             NUMZERO=CNUMZERO
                             NUMMISS=CNUMMISS
                            ));

     IF INDEX(UPCASE("&DATEVARS."),UPCASE(TRIM(LEFT(FIELD)))) THEN DO;
        SUM="----------------";
        NON_MISS=INPUT(TRIM(LEFT(CHARN)),MMDDYY10.);
        NUMNEG=INPUT(TRIM(LEFT(CNUMNEG)),MMDDYY10.);
        NUMMISS=INPUT(TRIM(LEFT(CNUMMISS)),MMDDYY10.);
        NUMZERO=INPUT(TRIM(LEFT(CNUMZERO)),MMDDYY10.);
        NUMPOS=INPUT(TRIM(LEFT(CNUMPOS)),MMDDYY10.);
     END;

     ELSE DO;
             NON_MISS=INPUT(TRIM(LEFT(CHARN)),BEST.);
             NUMNEG=INPUT(TRIM(LEFT(CNUMNEG)),BEST.);
             NUMMISS=INPUT(TRIM(LEFT(CNUMMISS)),BEST.);
             NUMZERO=INPUT(TRIM(LEFT(CNUMZERO)),BEST.);
             NUMPOS=INPUT(TRIM(LEFT(CNUMPOS)),BEST.);
             ARRAY VARS MIN MINPOS MAX MEAN;
             DO OVER VARS;
                 VARS=PUT(INPUT(TRIM(LEFT(VARS)),BEST.),BEST10.4);
             end;
             SUM=PUT(INPUT(TRIM(LEFT(SUM)),BEST.),COMMA16.2);
          END;

ARRAY VARS2 NON_MISS NUMNEG  NUMMISS  NUMZERO NUMPOS;

DO OVER VARS2; IF VARS2="0" THEN VARS2="."; END;

label FIELD="Field   *Name    *"
      Min  ="   Minimum*     Value"
      MinPos  ="   Min Pos*     Value"
      Max  ="   Maximum*     Value"
      Mean ="      Mean*     Value"
      MeanPos ="    Mean of*    Pos Values"
      Sum  ="      Numeric*        Sum"
    NumMiss="Number*Missing"
    NumNeg ="Number*Negative"
    NumZero="Number*of Zeroes"
    NumPos ="Number*Positive"
   Non_Miss="Number*Populated";
RUN;

proc print data=m5a label split="*" noobs uniform;
 VAR  FIELD  MIN MINPOS MEAN MEANPOS MAX SUM  NUMMISS NUMNEG NUMZERO NUMPOS NON_MISS;
 title3 "ENHANCED SUMMARY OF NUMERIC VARIABLES IN &INPUTDSN. (&NOBS. OBS)";
 run; TITLE3;

PROC DELETE DATA=WORK.m5a; RUN;
PROC DELETE DATA=WORK.m5b; RUN;
PROC DELETE DATA=WORK.m5c; RUN;
PROC DELETE DATA=WORK.m5d; RUN;
PROC DELETE DATA=WORK.m5e; RUN;
PROC DELETE DATA=WORK.m5f; RUN;
PROC DELETE DATA=WORK.m5g; RUN;
PROC DELETE DATA=WORK.m5h; RUN;
proc delete data=work.dum; run;
proc delete data=work.numvars; run;
```

```
proc delete data=work.checkdsn; run;

%end;

data dum; dum=1; run;

data dum; set dum charvars; run;

data _null_; set dum nobs=nobs;
     call symput('nobs',nobs-1); stop; run;

%if %eval(&nobs.) > 0 %then %do;

DATA CHECKDSN; SET &INPUTDSN. (keep=_char_);

%do loop=1 %to &nobs.;

data dum; dum=&loop.; set charvars point=dum;
     call symput('charvar',trim(left(name)));
     call symput('charlen',trim(left(put(max(length,20),best3.))));
     stop; run;

proc freq data=checkdsn order=freq noprint; tables &charvar./missing out=TEMPCHAR; run;

data _null_; set TEMPCHAR end=last;
     retain maxlen 0;
     if &charvar. ne ' ' then thislen=length(trim(&charvar.)); else thislen=0;
     if thislen > maxlen then maxlen = thislen;
     if last then call symput('MAXLEN',trim(left(put(maxlen,best3.))));
     run;

data TEMPCHAR (drop=NumOthers ocount opercent);
     length &charvar. $&charlen.; format &charvar. $char&charlen..;
     set TEMPCHAR (rename=(count=ocount percent=opercent)) end=last;
     if &charvar.='' then &charvar.='***Missing***';
     retain ncount npercent;
     if _n_<44 then do; COUNT=ocount; PERCENT=opercent; end;
        else do; COUNT+ocount; PERCENT+opercent;
        NumOthers+1;
        If NumOthers>1 Then &CharVar.=trim(left(Put(NumOthers,Best9.)))!!" other values";
        end;
     if _n_<43 or last then do;
                                CCOUNT+COUNT;
                                CPERCENT+PERCENT;
                                output;
                           end;
     run;

proc print data=TEMPCHAR noobs uniform label split="*";
     var &charvar. COUNT CCOUNT PERCENT CPERCENT;
     title3 "FREQUENCY ANALYSIS OF FIELD &charvar. (Max. length: &MaxLen.) IN DATASET
&INPUTDSN."; sum count percent; run; TITLE3;

%end;

proc delete data=work.dum; run;
proc delete data=work.charvars; run;
proc delete data=work.TEMPCHAR; run;
proc delete data=work.checkdsn; run;

%end;

OPTIONS _LAST_=&INPUTDSN.;

%LET INPUTDSN=; %LET NOBS=;

OPTIONS NOTES SYMBOLGEN MPRINT MLOGIC;

%MEND CHECKDATA;
```

APPENDIX B: Sample Output (Numeric Variables Only)

ENHANCED SUMMARY OF NUMERIC VARIABLES

Field Name	Minimum Value	Min Pos Value	Mean Value	Mean of Pos Values	Maximum Value	Numeric Sum	Number Missing	Number Negative	Number of Zeroes	Number Positive	Number Populated
CONNUMT	1001.07005	1001.07005	4424.52131	4424.521305	7159.6455	46,514,992.48	.	.	.	10513	10513
GRADET	0.90015837	0.90015837	2.08730752	2.087307517	3.29930345	21,943.86	.	.	.	10513	10513
CUSCODT	288146.665	288146.665	443271.442	443271.4419	1094016.05	4,660,112,668.70	.	.	.	10513	10513
CUSCATT	0.9000366	0.9000366	1.60811348	1.608113477	6.59720541	16,906.10	.	.	.	10513	10513
SALINDT	1/17/1998	1/17/1998	11/18/2002	11/18/2002	12/7/2007	----------	.	.	.	10513	10513
SALEDT	1/12/1998	1/12/1998	12/7/2002	12/7/2002	12/12/2007	----------	.	.	.	10513	10513
SHIPDAT	1/22/1998	1/22/1998	11/28/2002	11/28/2002	11/27/2007	----------	.	.	.	10513	10513
PAYDT	2/7/1998	2/7/1998	12/31/2002	12/31/2002	1/25/2008	----------	10	.	.	10503	10503
QTYT	0.90045009	0.90045009	111.549868	111.5498682	7939.31692	1,172,723.76	.	.	.	10513	10513
GRSUPRT	1.82264221	1.82264221	19.9957716	19.99577164	29.7984313	210,215.55	.	.	.	10513	10513
EARLPYT	0	0.20722547	0.25672132	0.498598055	0.67633827	2,698.91	.	.	5100	5413	10513
REBATE1T	0	0.31086241	0.25906556	0.527616481	0.68810702	2,723.56	.	.	5351	5162	10513
DINLFRTT	2.51372007	2.51372007	3.27820625	3.278206248	3.83337879	34,463.78	.	.	.	10513	10513
DWAREHT	0.55154036	0.55154036	0.61254599	0.612545993	0.67419335	6,439.70	.	.	.	10513	10513
DBROKT	-2.57343	1.03972487	1.26908005	1.269080049	1.45505753	13,341.84	.	13	.	10500	10513
CONTAINT	0.48668927	0.48668927	0.63426913	0.634269129	0.75121501	6,668.07	.	.	.	10513	10513
INTNFRT	1.84014598	1.84014598	2.24688053	2.246880528	2.58209601	23,621.45	.	.	.	10513	10513
INLFPWT	0	0.1533369	0.18732675	0.187344571	0.21461879	1,969.37	.	.	1	10512	10513
INLFWWT	0	0.04293499	0.05237227	0.052377254	0.0600915	550.59	.	.	1	10512	10513
WHSET	0	0.61336587	0.74823042	0.748301602	0.85819149	7,866.15	.	.	1	10512	10513
INLFWCT	0	0.07593368	1.31407215	1.516281477	23.1124358	13,814.84	.	.	1402	9111	10513
TCBROKHT	0.01226916	0.01226916	0.01497747	0.014977467	0.01724067	157.46	.	.	.	10513	10513
COMM1T	0	0.25393493	0.50988542	0.62952735	0.83178086	5,360.43	.	.	1998	8515	10513
COMM2T	0	0.15306924	0.00208448	0.284599096	0.53759627	21.91	.	.	10436	77	10513
BROKER1T	180718.98	180718.98	457695.745	457695.7454	1093462.07	4,811,755,371.50	.	.	.	10513	10513
BROKER2T	0	180772.226	2997.58087	409267.113	597163.071	31,513,567.70	.	.	10436	77	10513
CREDITT	0	0.00309987	0.07826908	0.078328688	0.85033071	822.84	.	.	8	10505	10513
ADVERTT	0	.	0	.	0	0	.	.	10513	.	10513
WARRT	0	0.11725839	0.34686006	0.353415371	0.48543795	3,646.54	.	.	195	10318	10513
ROYALT	0.02318329	0.02318329	0.1148622	0.114862199	0.16058372	1,207.55	.	.	.	10513	10513
DINDIRST	0.10863067	0.10863067	0.17355654	0.173556542	0.26157326	1,824.60	.	.	.	10513	10513
INDIRST	0.22904487	0.22904487	1.68449389	1.684493886	8.43056928	17,709.08	.	.	.	10513	10513
DINVCART	3.45352147	3.45352147	4.41468628	4.414686281	5.79497246	46,411.60	.	.	.	10513	10513
INVCART	0.11453274	0.11453274	0.16792131	0.167921308	0.21462451	1,765.36	.	.	.	10513	10513
PACKT	1.42678358	1.42678358	1.78614624	1.78614624	2.06543752	18,777.76	.	.	.	10513	10513
VCOMT	217.124118	217.124118	286.996235	286.9962353	391.596511	3,017,191.42	.	.	.	10513	10513