Paper number: TT02

CHECK YOUR DATA MORE EFFICIENTLY

Jian Hua (Daniel) Huang, Forest Laboratories Inc, NJ

ABSTRACT:

%CHKDATA is a SAS macro program designed to check the data in an efficient and user-friendly way. First, the macro can check the data structure by generating three types of key information: the contents of dataset, its associated SAS format, and a collection of all variable names listed horizontally. Second, the macro can generate the distinct values and frequency counts for any specified variables. Third, the macro can define any potential data issues and generate the reports. In addition, %CHKDATA has one important special feature. The macro can work on multiple datasets at the same time. When processing multiple datasets, it combines data information from each input dataset and list them side-by-side in one report, therefore, people can easily review and even compare the information among all input datasets. This is especially helpful for people working on data integration or anything where multiple datasets are used and compared. In summary, %CHKDATA is a very useful tool for anyone who wants to review and understand data quickly and correctly.

BACKGROUND:

Whether the task is to create a report or to derive a new dataset, people start from some source data and, more importantly, they need to understand the source data correctly and comprehensively. Normally, people review the source data by simply opening the dataset and reading it directly. However, if the data is huge, which includes dozens of variables and thousands of records, and if there is not only one single dataset but multiple datasets are involved, and if these datasets are not from one study but from many different studies, reading all the data one-by-one could be slow and unreliable. It is better to have utility programs that help us review and understand the data more efficiently.

%CHKDATA is a SAS macro program designed to check the data in a more efficient and user-friendly way. The macro is originally called '%GETSTART' (sounds like 'get start'). Because, at the beginning, it was designed only to check the data structure and generate distinct values, so people could use such information and get works (i.e. table programming) started quickly. Later, more functions were added to the macro. In addition to collecting information on data structures, the macro also checks for the potential data issues and reports them properly and timely. The macro also becomes more powerful by its ability to process multiple datasets at the same time. Due to its increased functional scope, the macro was renamed to %CHKDATA (sounds like 'check data').

%CHKDATA MACRO:

The %CHKDATA macro has two major functions. First, the macro can check the data structures for input dataset and generate distinct values for any specified variables. Second, the macro can define and summarize potential data issues, and report them properly. In addition, %CHKDATA has one special feature in which the macro can process multiple datasets at the same time.

USING %CHKDATA TO CHECK THE DATA STRUCTURE AND GENERATE DISTINCT VALUES:

As mentioned before, %CHKDATA can generate three types of key information for data structures: the contents of dataset, the SAS format values, and a collection of all variable names which are listed horizontally on one page. Let us review these functions with below examples.

Example 1; display the contents of input dataset.

```
%chkdata(lib=xxxx21d, data=addm, content=yes, report=addm_content1);
```

Output:

```
Reported by SAS user: JHHUANG, through SAS program: TEST_CHKDATA.SAS, on date: 2008-09-29 at time: 14:53
List of the contents of each input dataset side-by-side

 Variable   Attrib    XXXX21D                          Variable   Attrib   XXXX21D
 Names      -utes     ADDM                             Names      -utes    ADDM
 ------------------------------------------------      ------------------------------------------------

 AGE        Type      1                                           Format   DATE
            Length    8
            Label     Age                              DOBN       Type     1
            Format                                               Length   8
                                                                 Label    Date of Birth (numeric)
 CENTRE     Type      2                                           Format
            Length    3
            Label     Centre Number                    INVNO      Type     2
            Format    $                                          Length   4
                                                                 Label    Investigator ID
 COUNTRY    Type      2                                           Format   $
            Length    14
            Label     Country                          ITT        Type     2
            Format                                               Length   7
                                                                 Label    ITT
 COUNTRYC   Type      2                                           Format   $YNS
```

Comments:

The contents of dataset: ADDM from one single study: XXXX21D is generated by calling %CHKDATA and the sample output is listed above. The output lists variable names and the values of four key attributes: ‘type’, ‘length’, ‘label’ and ‘format’. To save page space, if there is only one input dataset, each page contains two panels. If there are multiple input datasets, then the contents of each dataset will be listed side-by-side. We will discuss later about how %CHKDATA deal with multiple datasets.

Example 2; display the SAS formats of input study.

```
%chkdata(lib=xxxx21d, printfmt=yes, report=xxxx21d_format1);
```

Output:

Reported by SAS user: JHHUANG, through SAS program: TEST_CHKDATA.SAS, on date: 2008-09-26 at time: 15:21
List of SAS format of each input study side-by-side

Format Names	xxxx21d Start	End	Label
ACCEPTS	1	1	YES, ENTERELY ACCEPTABLE
	2	2	YES, SOMEWHAT ACCEPTABLE
	3	3	UNCERTAIN
	4	4	NO, SOMEWHAT UNACCEPTABLE
	5	5	NO, ENTERELY UNACCEPTABLE
	98	98	N.D.
	99	99	N.A.
ACTIO1S	1	1	None
	2	2	Dosage reduced
	3	3	Interrupted
	4	4	Discontinued (permanently)
	5	5	Dosage Increased
	6	6	Dose not changed
	7	7	Dose reduced
	8	8	Drug withdrawn
	9	9	Dose increased
	98	98	N.D.
	99	99	N.A.

Format Names	xxxx21d Start	End	Label
	6	6	NEW TREATMENT GIVEN
	7	7	OTHERS
	8	8	STOPPED (TEMPORAL)
	9	9	STOPPED (DEFINITIVE)
	10	10	DOSAGE REDUCED
	11	11	INTERRUPTED
	12	12	DISCONTINUED (PERMANENTLY)
	13	13	DOSAGE INCREASED
	14	14	DOSE NOT CHANGED
	15	15	DOSE REDUCED
	16	16	DRUG WITHDRAWN
	17	17	DOSE INCREASED
	18	18	DRUG WITHDRAWN PERMANENTLY
	98	98	N.D.
	99	99	N.A.
ACTIV1S	0	0	Screen
	1	1	Change Subject Status

Comments:

The SAS format of one single study: XXXX21D is generated. The output includes format name, its start and end values, and most importantly, the label. To save page space, if there is only one input study, each page contains two panels.

Example 3, list of all variable names on one page horizontally.

```
%chkdata(lib=issd, data=d_prof2, allvar=yes, report=prof2_allvar);
```

Output:

Reported by SAS user: JHHUANG, through SAS program: TEST_CHKDATA.SAS, on date: 2008-09-26 at time: 15:21
Check data from: ISSD.D_PROF2
List all variable names horizotally, so they are convenient to be reviewed or edited

Obs	var1	var2	var3	var4	var5	var6	var7	var8	var9	var10	var11	var12	var13	var14	var15
1	AGE	AGEGRP	AGEGRP_C	BIRTHDT	BIRTHDTC	BMI	BMIGRP	BMIGRP_C	CENTER	COMPLETE	COMPLE_C	COPDSEV	COPDSE_C	COPDSE_O	COUNTRY
Obs	var16	var17	var18	var19	var20	var21	var22	var23	var24	var25	var26	var27	var28	var29	var30
1	DEATH	DEATHDT	DEATHDTC	DEATH_C	DESIGN	DOSE	DOSE_C	DUR_STDY	DUR_TRT	FDSDT	FDSDT99	FDSDT99C	FDSDTC	HEIGHTCM	HEIGHTIN
Obs	var31	var32	var33	var34	var35	var36	var37	var38	var39	var40	var41	var42	var43	var44	var45	var46	var47
1	INITIALS	INVNO	ITT	ITT_C	LASTDT	LASTDTC	LDSDT	LDSDT99	LDSDT99C	LDSDTC	PATYRS	PERIOD	PERIOD_C	PID	PP	PP_C	RACE
Obs	var48	var49	var50	var51	var52	var53	var54	var55	var56	var57	var58	var59	var60	var61	var62	var63
1	RACEOTH	RACETYPE	RACETY_C	RACE_C	RACE_O	RAND	RAND_C	SAFETY	SAFETY_C	SCREENO	SCRNDT	SCRNDTC	SEX	SEX_C	SMOKER	SMOKER_C
Obs	var64	var65	var66	var67	var68	var69	var70	var71	var72	var73	var74	var75
1	SMOKER_O	STUDYID	TERMSPEC	TREASON	TREASO_C	TREASO_O	TREATC	TREATGP	TREATGPC	TREATSQ	WEIGHTKG	WEIGHTLB

Comments:

There are totally 75 variables (var1-var75) existed in input dataset: D_PROF2. All variable names are collected and listed on one page horizontally, therefore, it is easy to review the variable names and copy and paste them into other SAS programs.

In addition to displaying data structure, %CHKDATA can also check the distinct values and their frequency counts for any specified variables. The following two examples demonstrate how %CHKDATA generates the distinct values for any specified variables from the input datasets.

Example 4; display the distinct values of any specified variables (the output is not sorted by any ID variables).

```
%chkdata(lib=xxxx22d, data=medicati, report=cmed1_chkdata1,
         univar=startyyy, idvar=);
```

Output:

Check data from:xxxx22D.MEDICATI
List unique values and frequency count of: STARTYYY
List First 30 Observation Only

Obs	STARTYYY	COUNT
1	04	1
2	1974	1
3	1976	1
4	1978	1
5	1980	2
6	1982	1
7	1984	5
8	1985	2
9	1986	2
10	NA	4
11	UK	15

Comments:

The distinct values and their frequency counts of variable: STARTYYY are generated and listed as above. Some strange values of STARTYYY, such as: ‘04’, ‘NA’, ‘UK’, have been detected by %CHKDATA. The information is very useful as it reminds people to pay attention to those strange values when working on the dataset. Btw, if multiple variables are listed under the macro option ‘UNIVAR=’, the macro will generate distinct values and frequency counts for each input variable and list them on separated pages. Again, it is also able to apply this function on multiple datasets at the same time. We will discuss this special feature later.

Example 5; display the distinct values of any specified variables (the output is sorted by ID variables first).

```
%chkdata(lib=xxxxpk09d, data=advs,  report=vital2_chkdata1,
            univar=timing, idvar=visitno visitid period);
```

Output:

Check data from: XXXXPK09D.ADVS
List unique values and frequency count of: TIMING, sorted by ID variables: VISITNO VISITID PERIOD

Obs	VISITNO	VISITID	PERIOD	TIMING	COUNT
1	-2.00	SCREENING	999	-99.00	198
2	-1.00	TREATMENT PERIOD 1 DAY -1	999	-99.00	63
3	1.00	TREATMENT PERIOD 1 DAY 1	1	0.00	63
4	1.00	TREATMENT PERIOD 1 DAY 1	1	0.08	63
5	1.00	TREATMENT PERIOD 1 DAY 1	1	0.50	63
6	1.00	TREATMENT PERIOD 1 DAY 1	1	2.00	63
7	1.00	TREATMENT PERIOD 1 DAY 1	1	6.00	63
8	1.00	TREATMENT PERIOD 1 DAY 1	1	12.00	63
9	2.00	TREATMENT PERIOD 1 DAY 2	1	24.00	63
10	3.00	TREATMENT PERIOD 1 DAY 3	1	0.00	63
11	3.00	TREATMENT PERIOD 1 DAY 3	1	0.08	63
12	3.00	TREATMENT PERIOD 1 DAY 3	1	0.50	63
13	3.00	TREATMENT PERIOD 1 DAY 3	1	2.00	63
14	3.00	TREATMENT PERIOD 1 DAY 3	1	6.00	63
15	3.00	TREATMENT PERIOD 1 DAY 3	1	12.00	63
16	4.00	TREATMENT PERIOD 1 DAY 4	1	24.00	63
17	5.00	TREATMENT PERIOD 1 DAY 5	1	48.00	63
18	6.00	TREATMENT PERIOD 2 DAY -1	1	-99.00	57
19	7.00	TREATMENT PERIOD 2 DAY 1	2	0.00	57
20	7.00	TREATMENT PERIOD 2 DAY 1	2	0.08	57
21	7.00	TREATMENT PERIOD 2 DAY 1	2	0.50	57
22	7.00	TREATMENT PERIOD 2 DAY 1	2	2.00	57
23	7.00	TREATMENT PERIOD 2 DAY 1	2	6.00	57
24	7.00	TREATMENT PERIOD 2 DAY 1	2	12.00	57
25	8.00	TREATMENT PERIOD 2 DAY 2	2	-99.00	12
26	8.00	TREATMENT PERIOD 2 DAY 2	2	24.00	57
27	9.00	TREATMENT PERIOD 2 DAY 3	2	0.00	57
28	9.00	TREATMENT PERIOD 2 DAY 3	2	0.08	57
29	9.00	TREATMENT PERIOD 2 DAY 3	2	0.50	57
30	9.00	TREATMENT PERIOD 2 DAY 3	2	2.00	57
31	9.00	TREATMENT PERIOD 2 DAY 3	2	6.00	57
32	9.00	TREATMENT PERIOD 2 DAY 3	2	12.00	57
33	10.00	TREATMENT PERIOD 2 DAY 4	2	24.00	57
34	11.00	TREATMENT PERIOD 2 DAY 5	2	48.00	57

Comments:

In this case, the variable TIMING is first sorted by three ID variables: VISITNO, VISITID, and PERIOD. Then its distinct values and frequency counts, within sorted variables, are generated and listed as above.

USING %CHKDATA TO DEFINE AND REPORT POTENTIAL DATA ISSUES:

We just discussed in detail how %CHKDATA can check the data structures and data values. In the next paragraph, we will discuss how %CHKDATA can check for any potential data issues. We use the following examples to demonstrate this function.

Example 6, check data and report potential data issue.

```
%chkdata (lib=xxxx22d, data=medicati, report=cmed1_issue1, listobs=15,
          idvar=MEDIC_TR dose dose1, fmtvar=MEDIC_TR,
          issue=%str(which variable to use as the correct CMED dosage: dose or
                     dose1?),
         where=(dose ^=dose1));
```

Output:

Report data issue: which variable to use as the correct CMED dosage: dose or dose1?
Reported by SAS user: JHHUANG, through SAS program: TEST_CHKDATA.SAS, on date: 2008-09-26 at time: 15:22
Check data from: XXXX22D.MEDICATI
List first 30 observation only

Obs	MEDIC_TR	DOSE	dose1
1	ENAP	10.00000	200
2	BECLAZONE	250.00000	200
3	FLIXOTIDE	250.00000	100
4	MONOPRIL	10.00000	100
5	VERAPAMIL	80.00000	100
6	ZINNAT	500.00000	100
7	FLIXOTIDE	250.00000	.
8	PRESTARIUM	4.00000	.
9	OMEZ	20.00000	50
10	BECLAZONE	250.00000	50
11	URSOSAN	500.00000	18
12	ENALAPRIL	10.00000	25
13	BECOTIDE	300.00000	25
14	VERAPAMIL	80.00000	400
15	CAPOTEN	12.50000	400

Comments:

In this case, two variables: DOSE, DOSE1 are found from the same input dataset: MEDICATI. These two variables have the similar name and label, it is hard to tell which one represents the real dosage of concomitant medication (MEDIC_TR). The potential data issue is defined as: 'which variable to use as the correct CMED dosage: dose or dose1?'. The macro defines this potential data issue in the where statement: 'where= (dose1 ^=dose)' and then generates corresponding output as above. This report is saved as a permanent list file by the macro option 'report=cmed1_issue1'. Later, the report can be sent to the corresponding group, i.e. data management group, for further review or data cleaning. In addition, by assigning macro option 'fmtvar=MEDIC_TR', it removes its associated format of variable 'MEDIC_TR', so the value of MEDIC_TR can be printed and fit on one page.

SPECIAL FEATURE, %CHKDATA CHECKS MULTIPLE DATASETS AT THE SAME TIME:

%CHKDATA has one special feature; it can deal with multiple input datasets, from different studies, at the same time. Below are examples of this important feature.

Example 7; display the contents of multiple datasets from different studies.

```
%chkdata(lib=xxxx22d xxxx30d xxxx31d, data=adcm medicati, content=yes,
         report=cmed1_content);
```

Output:

Reported by SAS user: JHHUANG, through SAS program: TEST_CHKDATA.SAS, on date: 2008-09-26 at time: 15:21
List of the contents of each input dataset side-by-side

Variable Names	Attrib -utes	XXXX22D MEDICATI	XXXX30D ADCM	XXXX31D ADCM
ATC_TEXT	Type	2	2	2
	Length	200	200	200
	Label	ATC Text	ATC Text	ATC Text
	Format	$	$	$
BATCHNO	Type	1		
	Length	8		
	Label	Batch number		
	Format			
CAS	Type		2	2
	Length		10	10
	Label		CAS number	CAS number
	Format		$	$

Comments:

The contents of two datasets: ADDM and MEDICATI, from three different studies: XXXX22D XXXX30D XXXX31D, are generated and listed on one page, side-by-side. The output allows people to review and compare the contents of all input datasets in an easy and quick way. This special feature is very helpful for people who work on the data integration or anything dealing with multiple datasets at the same time. It is worth to mention here that every input dataset, i.e. ADDM and MEDICATI, is not necessary to be existed in each input study. %CHKDATA can automatically detect which dataset exists, and then list the contents for those datasets that exist. For example, in this case, MEDICATI exists only in XXXX22D, and ADCM exists in the other two studies: XXXX30D, XXXX31D.

Example 8; display SAS format for multiple studies (list 3 studies per page).

```
%chkdata (lib=xxxx21d xxxx24d xxxx25d xxxxpk09d xxxx30d xxxx31d, printfmt=yes,
          ncolpage=3, report=prof2_format2);
```

Output:

Page 1

Reported by SAS user: JHHUANG, through SAS program: TEST_CHKDATA.SAS, on date: 2008-09-26 at time: 15:21
List of SAS format of each input study side-by-side

Format	xxxx21d			xxxx24d			xxxx25d		
Names	Start	End	Label	Start	End	Label	Start	End	Label
ACCEPTS	1	1	YES, ENTERELY ACCEPTABLE	1	1	YES, ENTERELY ACCEPTABLE	1	1	YES, ENTERELY ACCEPTABLE
	2	2	YES, SOMEWHAT ACCEPTABLE	2	2	YES, SOMEWHAT ACCEPTABLE	2	2	YES, SOMEWHAT ACCEPTABLE
	3	3	UNCERTAIN	3	3	UNCERTAIN	3	3	UNCERTAIN
	4	4	NO, SOMEWHAT UNACCEPTABLE	4	4	NO, SOMEWHAT UNACCEPTABLE	4	4	NO, SOMEWHAT UNACCEPTABLE
	5	5	NO, ENTERELY UNACCEPTABLE	5	5	NO, ENTERELY UNACCEPTABLE	5	5	NO, ENTERELY UNACCEPTABLE
	98	98	N.D.	98	98	N.D.	98	98	N.D.
	99	99	N.A.	99	99	N.A.	99	99	N.A.
ACTIO1S	1	1	None	1	1	None	1	1	None
	2	2	Dosage reduced	2	2	Dosage reduced	2	2	Dosage reduced
	3	3	Interrupted	3	3	Interrupted	3	3	Interrupted
	4	4	Discontinued (permanently)	4	4	Discontinued (permanently)	4	4	Discontinued (permanently)
	5	5	Dosage Increased	5	5	Dosage Increased	5	5	Dosage Increased
	6	6	Dose not changed	6	6	Dose not changed	6	6	Dose not changed
	7	7	Dose reduced	7	7	Dose reduced	7	7	Dose reduced
	8	8	Drug withdrawn	8	8	Drug withdrawn	8	8	Drug withdrawn
	9	9	Dose increased	9	9	Dose increased	9	9	Dose increased
	98	98	N.D.	98	98	N.D.	98	98	N.D.
	99	99	N.A.	99	99	N.A.	99	99	N.A.

Page 2

Reported by SAS user: JHHUANG, through SAS program: TEST_CHKDATA.SAS, on date: 2008-09-26 at time: 15:21
List of SAS format of each input study side-by-side

Format	xxxxpk09d			xxxx30d			xxxx31		
Names	Start	End	Label	Start	End	Label	Start	End	Label
ACCEPTS	1	1	YES, ENTIRELY ACCEPTABLE	1	1	YES, ENTERELY ACCEPTABLE	1	1	YES, ENTERELY ACCEPTABLE
	2	2	YES, SOMEWHAT ACCEPTABLE	2	2	YES, SOMEWHAT ACCEPTABLE	2	2	YES, SOMEWHAT ACCEPTABLE
	3	3	UNCERTAIN	3	3	UNCERTAIN	3	3	UNCERTAIN
	4	4	NO, SOMEWHAT UNACCEPTABLE	4	4	NO, SOMEWHAT UNACCEPTABLE	4	4	NO, SOMEWHAT UNACCEPTABLE
	5	5	NO, ENTIRELY UNACCEPTABLE	5	5	NO, ENTERELY UNACCEPTABLE	5	5	NO, ENTERELY UNACCEPTABLE
	98	98	N.D.	98	98	N.D.	98	98	N.D.
	99	99	N.A.	99	99	N.A.	99	99	N.A.

Comments:

%CHKDATA generates SAS formats for six studies at one time and lists them side-by-side. The macro option of ‘ncolpage=3’ (stands for ‘number of columns listed per page’) requests that each page lists SAS formats for three studies.

SUMMARY:

In summary, %CHKDATA is a very useful tool to check data efficiently. It can check both the data structures and distinct data values; therefore, people can get start quickly on their work. In addition, the macro can also define potential data issues detected from input datasets, and report these issues properly to the data management team. It helps staff to continually keep the data clean and correct. Finally, the macro can check multiple datasets, even from different studies, at the same time, and summarize the information in one report. This special feature is especially helpful for people who work on data integration or anything dealing with multiple datasets at the same time.

%CHKDATA has its limitation as well. Once the macro defines a potential data issue, it generates the report and saves it into a separated file. It will be better if the macro can automatically concatenate all correlated data issues together, and save them into one big final report.

CONCLUSION:

%CHKDATA macro is simple and practical. It will be a very useful tool for people who want to review and understand data quickly and correctly.

CONTACT INFORMATION:

Your comments and questions are valued and encouraged. Please contact the author at:

Daniel Huang, MSc, MPH,
Forest Research Institute
Harborside Financial Center,
Plaza V, Jersey City, NJ 07311
Tel: 1-201-427-8291
Email: Jian.huang@frx.com