Data Quality Macros

Manitoba Centre for Health Policy

Development Data Analysis Environment

Version 2.0

Say Hong
11/5/2013

Table of Contents

MCHP Data Quality Macros

DQ_META Macro

Description: This macro generates a metadata dataset to be used in data quality and documentation processes. The output dataset will be saved as Metadata in the work directory

Syntax: %dq_meta(domain=, db=, fmt=)

Parameters: domain = libname of the

dataset db =

dataset name or prefix

fmt = location of a text file containing dataset name and variables name with their associated formats

Example: %***dq_meta***(domain=health, db=wrha_derca_,
fmt='T:\Sayh\test\testnewdqversion\diab_varfmt
.txt');

More Details about the Macro:

The following is an example of a **tab** delimited text file (fmt) containing datasets name, variables name and their associated format.

```
wrha_derca_tblpatient_1985apr     BandID           $DERCA_Band.
wrha_derca_tblpatient_1985apr     DerCodeID        $DerCA_Dercode.
wrha_derca_tblpatient_1985apr     GradDerReferID   $DERCA_DerRefer.
wrha_derca_tblpatient_1985apr     DiagCauseID      $DERCA_DiagCause.
wrha_derca_tblpatient_1985apr     DiagTypeID       $DERCA_DiagType.
wrha_derca_tblpatient_1985apr     DiagHNF1a        $DERCA_HNF1a.
wrha_derca_tblpatient_1985apr     InitMgmtID       $DERCA_InitMgmt.
wrha_derca_tblpatient_1985apr     DiagNephroID     $DERCA_Nephro.
wrha_derca_tblpatient_1985apr     RHAID            $DERCA_RHA.
wrha_derca_tblpatient_1985apr     ReferStatusID    $DERCA_ReferStatus.
wrha_derca_tblpatient_1985apr     RaceID           $DERCA_Race.
wrha_derca_tblpatient_1985apr     SchoolID         $DERCA_School.
wrha_derca_tblpatient_1985apr     VisitNoticeID    $DERCA_visitnotice.
wrha_derca_tblpatient_1985apr     StatFamilyID     $DERCA_FStat.
wrha_derca_tblpatient_1985apr     StatCareID       $DERCA_CStat.
wrha_derca_tblpatient_1985apr     ClinicStatID     $DERCA_ClinicStat.
wrha_derca_tblpatient_1985apr     GradCode         $DERCA_gradcode.
wrha_derca_tblvisit_  1985apr     InsulinRegimeID  $DERCA_InsulinRegime.
wrha_derca_tblvisit_  1985apr     JointContrID     $DERCA_JointContr.
wrha_derca_tblvisit_   1985apr    VisitLocID       $DERCA_VisitLoc.
wrha_derca_tblguardian_1985apr    RelationshipID   $DERCA_Relationship.
```

Please note that all formats mentioned in the above text file must be already loaded by SAS.

DQ_CONTENTS Macro

Description: This macro generates an overview table that contains dataset name, dataset label, number of records and number of fields

Syntax: %dq_contents(domain=, db=, memnum=)

Parameters: domain = libname of the dataset

db = dataset prefix or space separated list of dataset name or cluster dataset name

memnum = space separated list of cluster members, if blank then the macro will run for a specific dataset (non-cluster) or the whole cluster if the dataset is a cluster

Example: %***dq_contents***(domain=health, db=wrha_derca_)

Example:

```
%let DQ_Dir = T:\Sayh\test\testnewdqversion; /*specify
                                              location to
                                              save output */

%let DQ_name = contents;      /*specify excel output name*/

%dq_contents(domain=health, db= Ckd_2012_hsc_2004jan);

%dq_gen(dir=&dq_dir, wrkbook=&dq_name, save=Y)

/* Please refer to the automating the excel output section
      for more info regarding the DQ_GEN Macro */
```

DQ_VIMO Macro

Description: This macro is used to produce a VIMO table for a specific dataset or a specific cluster member or a combination of cluster members.

This macro can be used to perform a univariate data quality check if invalidchk=Y. Note that data quality check is based on the format defined, so that formats have to be loaded and DQ_META macro must be run before running the macro. Otherwise set invalidchk=N.

Note that the excel output produced by this macro is plain and unformatted, to automatically generate the VIMO table and GRAPH, please refer to the DQ_GEN macro.

Syntax:

```
%dq_vimo(ds=, invalidchk=N, memnum=, postals=,
        muncodes=, suppvar=, idvars=,
        nooutlier=)
```

Parameters:

ds = Name of input dataset, could be one or two level

invalidchk = specifies whether data quality check should be performed, default value is N (valid value = Y/N)

memnum = List of cluster members that are used to produce VIMO table, if blank then the macro will be run for a specific dataset (non-cluster) or the whole cluster if the dataset is a cluster

postals = Space separated list of postal code variables to check for invalid postal code, if blank then no invalid check will be performed.

muncodes = Space separated list of municipal code variables to check for invalid municipal code, if blank then no invalid check will be performed

suppvar = Space separated list of variables that are suppressed in SPDS, leave blank if none were suppressed

idvars = Space separated list of ID variables to be put in the ID category of the VIMO table. If blank, then only phin with format of z15 will be put in the ID category.

nooutlier = Fields to suppress outlier calculation. This parameter can be one of the following.

1. all (suppress outlier calculation for all numeric fields)

2. space separated list of numeric variables to suppress outlier calculation

3. location and name of the text file that contains variables to suppress outlier calculation.

Example:

```
%dq_vimo(ds=health.wrha_derca_tblpatient_1985a
         pr, invalidchk=Y, suppvar=bandid
         hlthother hsc mhsc);
%dq_vimo(ds = social.hcm_edi_2006jan,
          postals = postal);
```

Note that to run the DQ_VIMO macro with macro parameter invalidchk = Y, you must first do the following:

- define and load the format for at least one of the variable in the dataset
- run the DQ_META macro

Details regarding the range check

To perform a range check on the numeric variables, format with specific range need to be defined, for example to run a range check on age, birth date and program date where the range for age, birth date and program date fall within the following range:

1. Age must be between 0 to 110
2. Birth date must be between Jan. 1, 1900 to Nov. 01, 2012
3. Program date must be between Jan. 1, 2004 to Dec. 31, 2012

The following formats and a tab delimited text file containing datasets name, variables name and their associated format must be defined:

- `value yymmdddf  '01jan1900'd-'01Nov2012'd = 'valid';`
- `value yymmddd2f '01jan2004'd-'31Dec2012'd = 'valid';`
- `value agef  0-110 = 'valid';`
- `value $genderf      '1' = 'Male'`
`'2' = 'Female'`

Note that the name of the date format **MUST** start with yymmddd.

Creating a tab delimited text file containing datasets name, variables name and their associated format that will be used to generate the metadata for the DQ_VIMO macro.

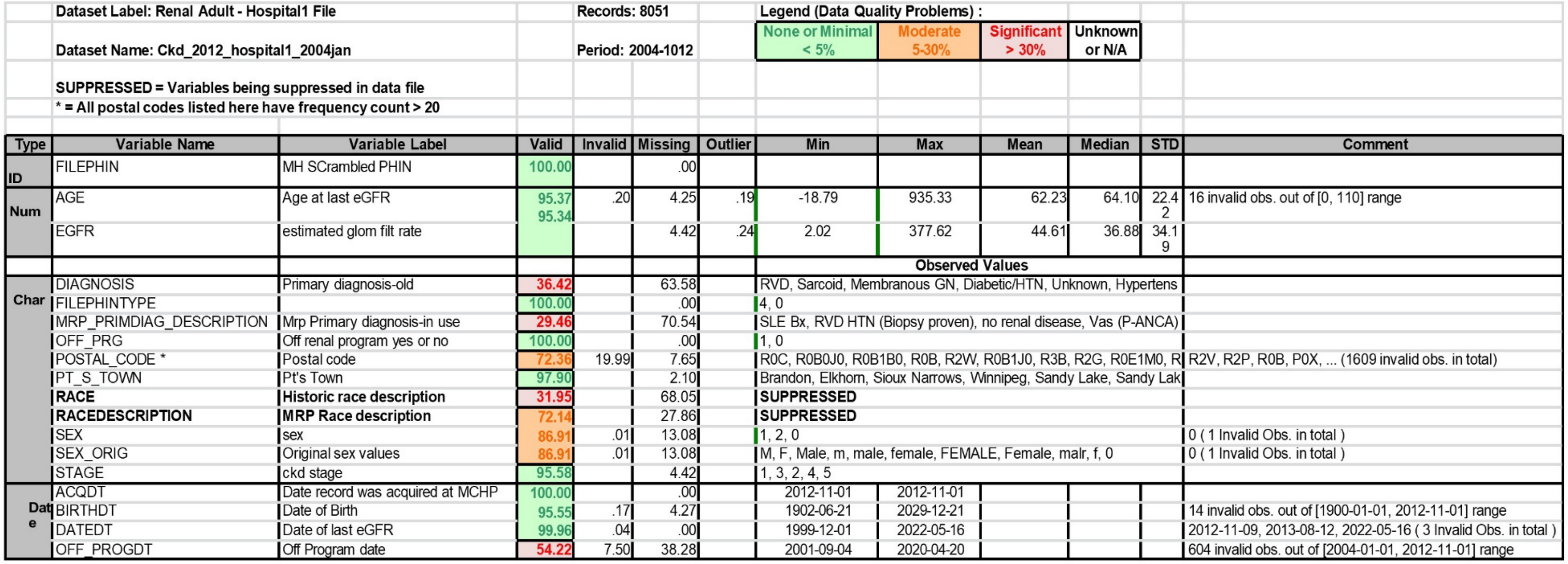
```
Ckd_2012_hsc_2004jan      birthdt             yymmdddf.
Ckd_2012_hsc_2004jan      DATEDT              yymmdddf.
Ckd_2012_hsc_2004jan      OFF_PROGDT          yymmddd2f.
Ckd_2012_hsc_2004jan      sex                 $genderf.
Ckd_2012_hsc_2004jan      age                 agef.
```

Example:

```
%include 'G:\dqmacro\*.sas';            /*Load DQ Macro*/

%let DQ_Dir = T:\Sayh\test\testnewdqversion; /*specify
                                         location to save
                                        output */
%let DQ_name = Ckd_HSC;            /*specify excel output
                                        name*/

proc format;
%include T:\Sayh\test\testnewdqversion\testvalid.txt';
run;

%dq_meta(domain=project, db=Ckd_2012_hsc_2004jan,
  fmt='T:\Sayh\test\testnewdqversion\hsc_varfmt.txt');

%dq_vimo(ds=project.Ckd_2012_hsc_2004jan,
  postals=postal_code, invalidchk=Y, suppvar=race);

%dq_gen(ds=project.Ckd_2012_hsc_2004jan, period=2004-2012,
  dir=&dq_dir, wrkbook=&dq_name, save=Y)

/* Please refer to the automating the excel output section
for more info regarding the DQ_GEN Macro */
```

Running the above codes will automatically generate the following VIMO table and chart. The DQ_GEN macro will automatically capture the dataset label (if the dataset had a label), total number of records, data set name, and label the time period of the dataset. The name of the data set will be used as the title of the VIMO chart.

Table 1



Dataset Label: Renal Adult - Hospital1 File | **Records: 8051**

Dataset Name: Ckd_2012_hospital1_2004jan | **Period: 2004-1012**

Legend (Data Quality Problems) :

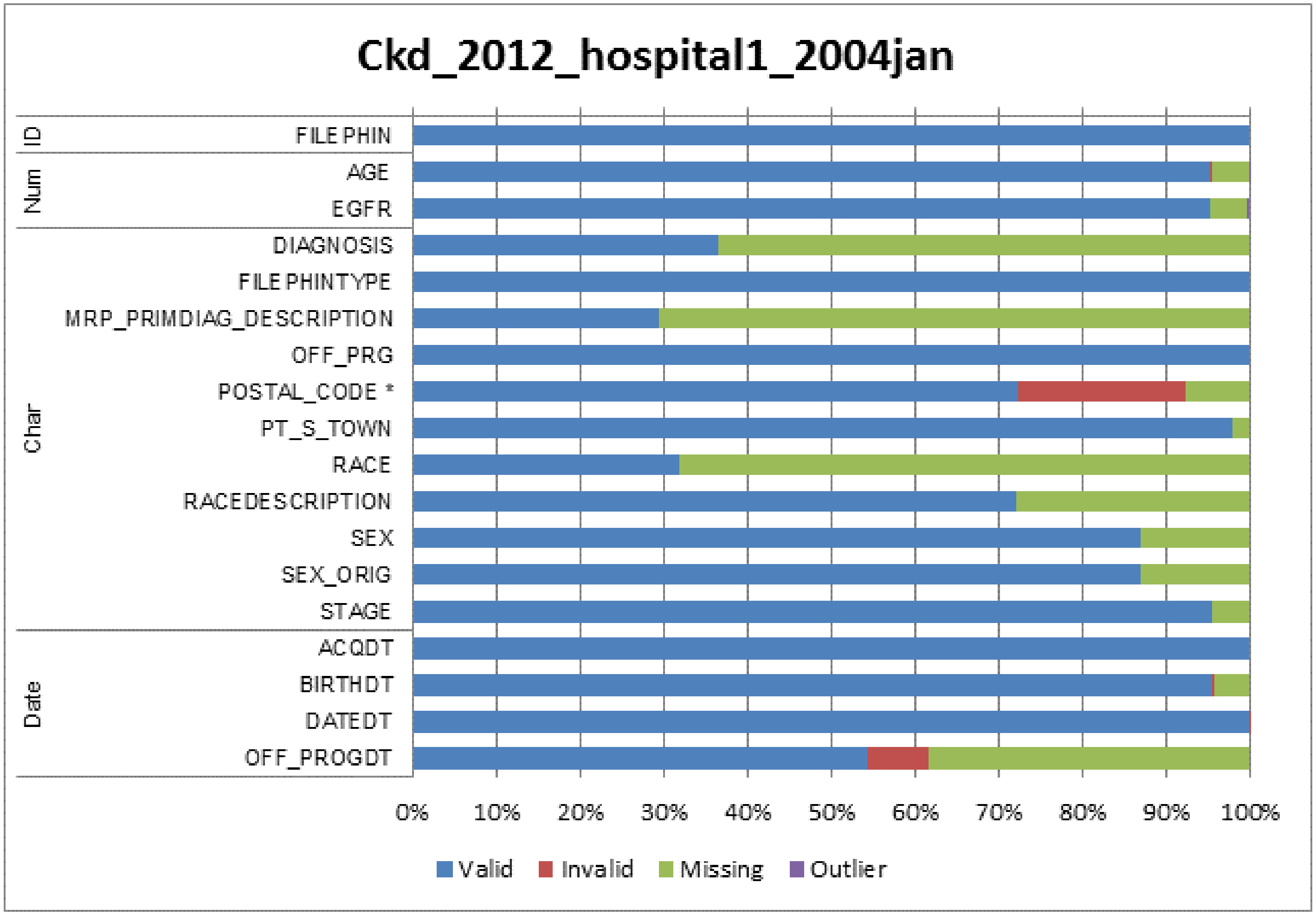
None or Minimal	Moderate	Significant	Unknown
< 5%	5-30%	> 30%	or N/A

SUPPRESSED = Variables being suppressed in data file

*** = All postal codes listed here have frequency count > 20**

Type	Variable Name	Variable Label	Valid	Invalid	Missing	Outlier	Min	Max	Mean	Median	STD	Comment
ID	FILEPHIN	MH SCrambled PHIN	100.00		.00							
Num	AGE	Age at last eGFR	95.37 95.34	.20	4.25	.19	-18.79	935.33	62.23	64.10	22.42	16 invalid obs. out of [0, 110] range
	EGFR	estimated glom filt rate			4.42	.24	2.02	377.62	44.61	36.88	34.19	
							Observed Values					
Char	DIAGNOSIS	Primary diagnosis-old	36.42		63.58		RVD, Sarcoid, Membranous GN, Diabetic/HTN, Unknown, Hypertens					
	FILEPHINTYPE		100.00		.00		4, 0					
	MRP_PRIMDIAG_DESCRIPTION	Mrp Primary diagnosis-in use	29.46		70.54		SLE Bx, RVD HTN (Biopsy proven), no renal disease, Vas (P-ANCA)					
	OFF_PRG	Off renal program yes or no	100.00		.00		1, 0					
	POSTAL_CODE *	Postal code	72.36	19.99	7.65		R0C, R0B0J0, R0B1B0, R0B, R2W, R0B1J0, R3B, R2G, R0E1M0, R					R2V, R2P, R0B, P0X, ... (1609 invalid obs. in total)
	PT_S_TOWN	Pt's Town	97.90		2.10		Brandon, Elkhorn, Sioux Narrows, Winnipeg, Sandy Lake, Sandy Lak					
	RACE	**Historic race description**	31.95		68.05		**SUPPRESSED**					
	RACEDESCRIPTION	**MRP Race description**	72.14		27.86		**SUPPRESSED**					
	SEX	sex	86.91	.01	13.08		1, 2, 0					0 (1 Invalid Obs. in total)
	SEX_ORIG	Original sex values	86.91	.01	13.08		M, F, Male, m, male, female, FEMALE, Female, malr, f, 0					0 (1 Invalid Obs. in total)
	STAGE	ckd stage	95.58		4.42		1, 3, 2, 4, 5					
Date	ACQDT	Date record was acquired at MCHP	100.00		.00		2012-11-01	2012-11-01				
	BIRTHDT	Date of Birth	95.55	.17	4.27		1902-06-21	2029-12-21				14 invalid obs. out of [1900-01-01, 2012-11-01] range
	DATEDT	Date of last eGFR	99.96	.04	.00		1999-12-01	2022-05-16				2012-11-09, 2013-08-12, 2022-05-16 (3 Invalid Obs. in total)
	OFF_PROGDT	Off Program date	54.22	7.50	38.28		2001-09-04	2020-04-20				604 invalid obs. out of [2004-01-01, 2012-11-01] range

Figure 1



Ckd_2012_hospital1_2004jan
ID
FILEPHIN
Num
AGE
EGFR
Char
DIAGNOSIS
FILEPHINTYPE
MRP_PRIMDIAG_DESCRIPTION
OFF_PRG
POSTAL_CODE *
PT_S_TOWN
RACE
RACEDESCRIPTION
SEX
SEX_ORIG
STAGE
Date
ACQDT
BIRTHDT
DATEDT
OFF_PROGDT
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
Valid
Invalid
Missing
Outlier

DQ_LINK Macro

Description: This macro creates a table that shows the number and percentage of linkable records of a specific dataset or a list of datasets. If the dataset is a cluster dataset, this macro can run for a list of cluster members.

Note that a record is considered linkable if the record's phin is coded as individual specific. i.e., phintype variable has the following values

0 = MH, verified against concurrent registries

1 = MH, redirected to this scrphin from filephin

2 = MCHP, modified sibling's scrphin

3 = MCHP, assigned scrphin from registry

6 = MCHP, MH PHIN not known at MCHP at acqdt

This macro also generates a distribution table for the phintype.

Syntax:

```
%dq_link(domain=, db=, phin=scrphin,
       type=scrphintype, memnum=)
```

Parameters: Domain = libname of the dataset

db   = dataset prefix or Space separated list of dataset name (or name of the cluster if memnum is non- empty)

phin = Name of phin variable (default value

is scrphin) type   = Name of phintype

variable (default value is scrphintype)

memnum = ALL or Space separated list of cluster members, if memnum=ALL then this macro will run for all cluster members of a specific cluster

Example:

```
%dq_link(domain=health,
        db=wrha_derca_tblpatient_1985apr,
        phin=filephin, type=filephintype)

%dq_link(domain=social,
        db=MFSL_CDCP_MHCW_2000JAN,
        phin=filephin, type=filephintype,
        memnum=all);

%dq_link(domain=social,
        db=MFSL_CDCP_MHCW_2000JAN,
        phin=filephin, type=filephintype,
        memnum=1 2);
```

DQ_LINKYR Macro

Description: This macro produces a percentage of linkable records over year for a specific dataset or a list of datasets

Syntax:

```
%dq_linkyr(domain=, db=, startyr=, endyr=,
         bydate=, phin=scrphin,
         type=scrphintype, ytype=F);
```

Parameters: domain = libname of the dataset

db   = dataset prefix or Space
         separated list of dataset name

startyr =

beginning year

endyr = ending

year

bydate = date variable (must be sas date)

phin   = name of phin variable (default value

is scrphin) type   = name of phintype (default

value is scrphintype)

ytype = Default value is F, if set to C then
        linkability will be run by calendar
        year, otherwise linkablity will be
        performed by fiscal year (valid value
        is F/C)

```
Example:     %dq_linkyr(domain=health,
                            db=MHCPL_CMORGANISM_19922010
                              MHCPL_CMRESULTS_19922010
                              MHCPL_CMSECTION_19922010
                              MHCPL_SPSEROTESTS_19922010
                              MHCPL_SPPARATESTS_19922010
                              MHCPL_SPSECTION_19922010,
                         startyr=1992,
                         endyr=2009,
                         bydate=RECEIVEDDT)
                         ;
```

Example:

```
    %include 'G:\dqmacro\*.sas';       /*Load DQ Macro*/

    %dq_linkyr(domain=health,
        db=MHCPL_CMORGANISM_19922010 MHCPL_CMRESULTS_19922010
        MHCPL_CMSECTION_19922010 MHCPL_SPSEROTESTS_19922010
        MHCPL_SPPARATESTS_19922010 MHCPL_SPSECTION_19922010,
        startyr=1992, endyr=2009, bydate=RECEIVEDDT)

    %dq_gen(Dir=&DQ_Dir, wrkbook=&DQ_name)
```

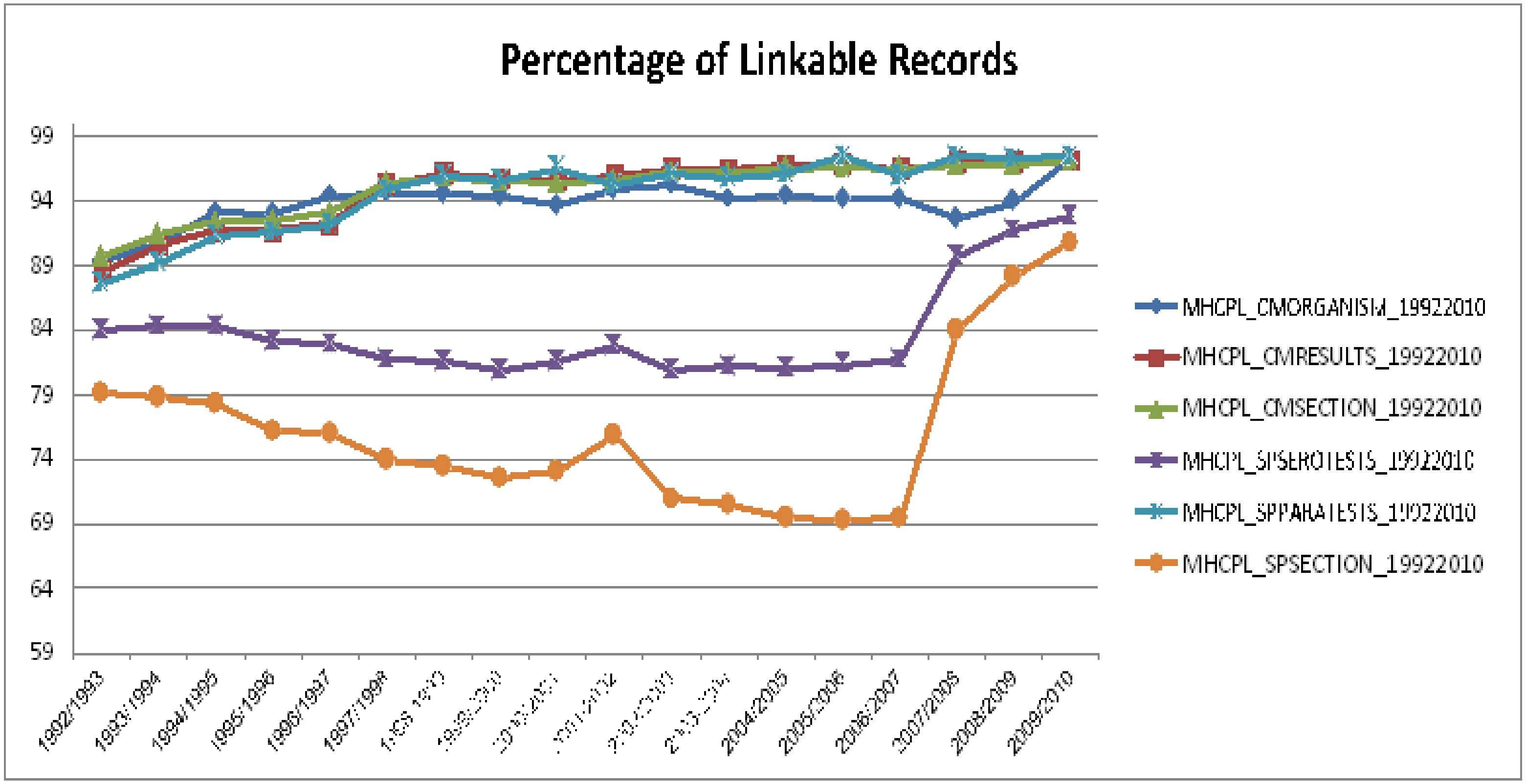
These codes automatically generate the following table and graph.

Table 2

Year	MHCPL_CMORGANSM_19922010	MHCPL_CMRESULTS_19922010	MHCPL_CMSECTION_19922010	MHCPL_SPSEROTESTS_19922010	MHCPL_SPPARAESTS_19922010	MHCPL_SPSECTION_19922010
1992/ 1993	89.3	88.5	89.6	84	87.6	79.1
1993/ 1994	90.7	90.6	91.4	84.3	89.2	78.7
1994/ 1995	93.1	91.7	92.5	84.3	91.3	78.3
1995/ 1996	93	91.7	92.5	83.1	91.6	76.2
1996/ 1997	94.3	92.2	93.1	82.9	92.1	76
1997/ 1998	94.6	95.2	95.3	81.7	94.8	73.9
1998/ 1999	94.5	96.1	95.8	81.5	95.9	73.4
1999/ 2000	94.3	95.7	95.5	80.8	95.6	72.5
2000/ 2001	93.6	95.6	95.4	81.5	96.5	73.1
2001/ 2002	94.9	95.9	95.6	82.8	95.2	75.9
2002/ 2003	95.2	96.5	96.3	80.9	96	70.9
2003/ 2004	94.2	96.4	96.3	81.2	95.7	70.5

Year	MHCPL_CMORGANSM_19922010	MHCPL_CMRESULTS_19922010	MHCPL_CMSECTION_19922010	MHCPL_SPSEROTESTS_19922010	MHCPL_SPPARAESTS_19922010	MHCPL_SPSECTION_19922010
2004/2005	94.4	96.7	96.5	81.1	96.1	69.5
2005/2006	94.1	96.8	96.6	81.3	97.5	69.3
2006/2007	94.1	96.5	96.5	81.7	95.9	69.4
2007/2008	92.6	97	96.8	89.6	97.5	83.9
2008/2009	93.9	97	96.8	91.7	97.3	88.1
2009/2010	97.2	97.1	97.1	92.8	97.5	90.8

Figure 2


Percentage of Linkable Records
99
94
89
84
79
74
69
64
59
1992/1993
1993/1994
1994/1995
1995/1996
1996/1997
1997/1998
1998/1999
1999/2000
2000/2001
2001/2002
2002/2003
2003/2004
2004/2005
2005/2006
2006/2007
2007/2008
2008/2009
2009/2010
MHCPL_CMORGANISM_19922010
MHCPL_CMRESULTS_19922010
MHCPL_CMSECTION_19922010
MHCPL_SPSEROTESTS_19922010
MHCPL_SPPARATESTS_19922010
MHCPL_SPSECTION_19922010

DQ_AGREEMENT Macro

Description: This macro checks the agreement between a dataset and the registry data (mhmrs_uniqphin_1970[regyr]) for the same individuals and produces Kappa Statistics for sex and date of birth
This macro can be run on a list of datasets or multiple cluster members within a specific cluster dataset.

Syntax: %dq_agreement(domain=, db=, regyr=2012, phin=scrphin, sex=sex, M=1, F=2, birthdt=birthdt, memnum=)

Parameters:

domain = libname of the dataset

db = Space separated list of dataset name or specific cluster dataset name

regyr = Latest year of available registry data (default value is 2012)

phin = Variable containing PHIN (default value is scrphin)

sex = Sex variable (default value is sex)
M = Representing value for males (default value is 1)
F = Representing value for females (default value is 2)
birthdt = Date of birth variable (default value is birthdt)

memnum = ALL or Space separated list of cluster members, if blank then the macro will run for a specific dataset (non-cluster) or the whole cluster if the dataset is a cluster

Example:

```
%dq_agreement(domain=health,
              db=wrha_derca_tblpatient_1985apr,
              phin=filephin, birthdt=birth_dt)
```

DQ_TREND Macro

Description: This macro performs a trend analysis over a specified time range, the output will be saved in graphic format (.PNG)

Syntax:

```
%dq_trend(ds=, startyr=, endyr=, bydate=,
         bydatetime=, byvar=_ALL_, byfmt=,
         bymonth=N, ytype=F,
         memnum=)
```

Parameters: ds = Name of

dataset startyr =

Beginning year

endyr = Ending

year

bydate = Date variable (must be SAS Date), leave blank if bydatetime is not blank

bydatetime = SAS Date time variable, leave blank if bydate is not blank

byvar = Optional, if omitted the trend analysis will be run for all records in the dataset

byfmt = Optional, format for byvar if any

bymonth = Default value is N, if set to Y then trend analysis will be run by month instead of year (valid value is Y/N)

ytype = Default value is F, if set to C then trend analysis will be run by calendar year, otherwise trend analysis will be performed by fiscal year (valid value is F/C)

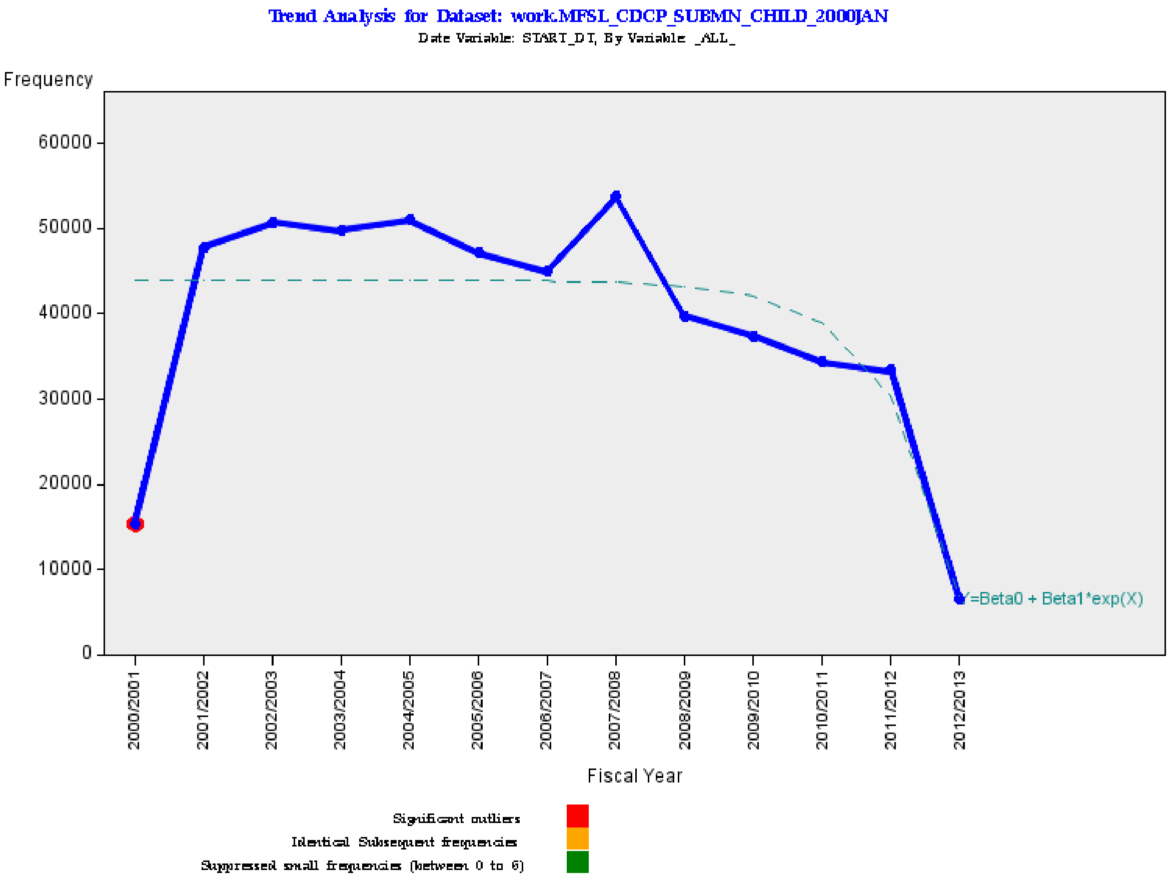
memnum = Space separated List of cluster members, if blank then the macro will run for a specific dataset (non-cluster) or the whole cluster if the dataset is a cluster

Example:

```
%dq_trend(ds=social.MFSL_CDCP_SUBMN_CHILD_2000J
         AN, startyr=2000, endyr=2012,
         bydate=START_DT);
```

Example:

```
%include 'G:\dqmacro\*.sas';             /*Load DQ Macro*/

%let dq_dir=T:\Sayh\test\testnewdqversion;  /*location to
                                              save graph*/

%dq_trend(ds=work.MFSL_CDCP_SUBMN_CHILD_2000JAN,
    startyr=2000, endyr=2012, bydate=START_DT);
```



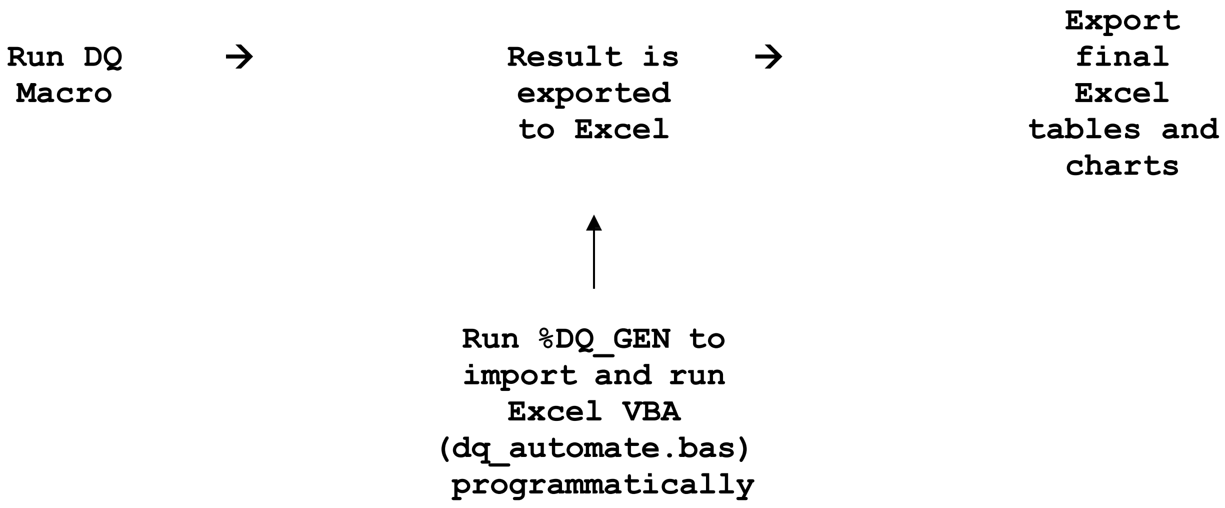
Automating the Excel Output

DQ_AUTOMATE.BAS

The dq_automate.bas contains the excel VBA code that has been developed to automatically generate the following tables and charts without manual intervention: VIMO table, VIMO graph, overview, linkability, phintype, agreement, linkability over years, and a line graph for linkability over years. This file will be run automatically by calling the DQ_GEN macro (see below).

DQ_GEN Macro

The excel output files created by the DQ macros (dq_vimo, dq_contents, dq_link, dq_linkyr and dq_agreement) are plain and unformatted. These files have to be copied and pasted into an excel template to produce VIMO table and graph as well as all other relevant DQ tables. In order to eliminate the tedious process of copying and pasting, an excel VBA code (dq_automate.bas) and DQ_GEN SAS macro have been developed to generate all the tables and graphs automatically. Since Excel VBA code can only run within its host application, Excel must be installed. The following Chart describes the flow of the process.


Run DQ Macro
Result is exported to Excel
Export final Excel tables and charts
Run %DQ_GEN to import and run Excel VBA (dq_automate.bas) programmatically

Description: This macro has been developed as an automated process for generating data quality tables and charts.

Running this macro will automatically open an unformatted excel output, and import Excel VBA code (dq_automate.bas) and run it to produce tables and chart without manual intervention.

Syntax:

```
%DQ_GEN(ds=, period=, Dir=, wrkbook=, memnum=,
       save=N, rnglen=);
```

Parameters:

ds   = same value as the VIMO macro

period = label the time period in the

output dir   = directory where Excel

is output

wrkbook = name of the Excel output

memnum = same value as the VIMO macro

save   = specify whether to save the output (valid value is Y/N). Default value is N

rnglen = this parameter allows the user to split the huge vimo table that contains hundreds of variables into mulitple smaller vimo tables, rnglen is the number of variables that each smaller vimo table contains. For example, if a vimo table contains 150 variables and
the user would like to split the table into
3 smaller tables each contains 50 variables, then set rnglen=50. Leave blank if no split is required.

Example: ***%dq_gen***(ds=health.wrha_derca_tblpatient_
1985apr, period=1985-2012,
dir=&dq_dir, wrkbook=&dq_name,
memnum=1)

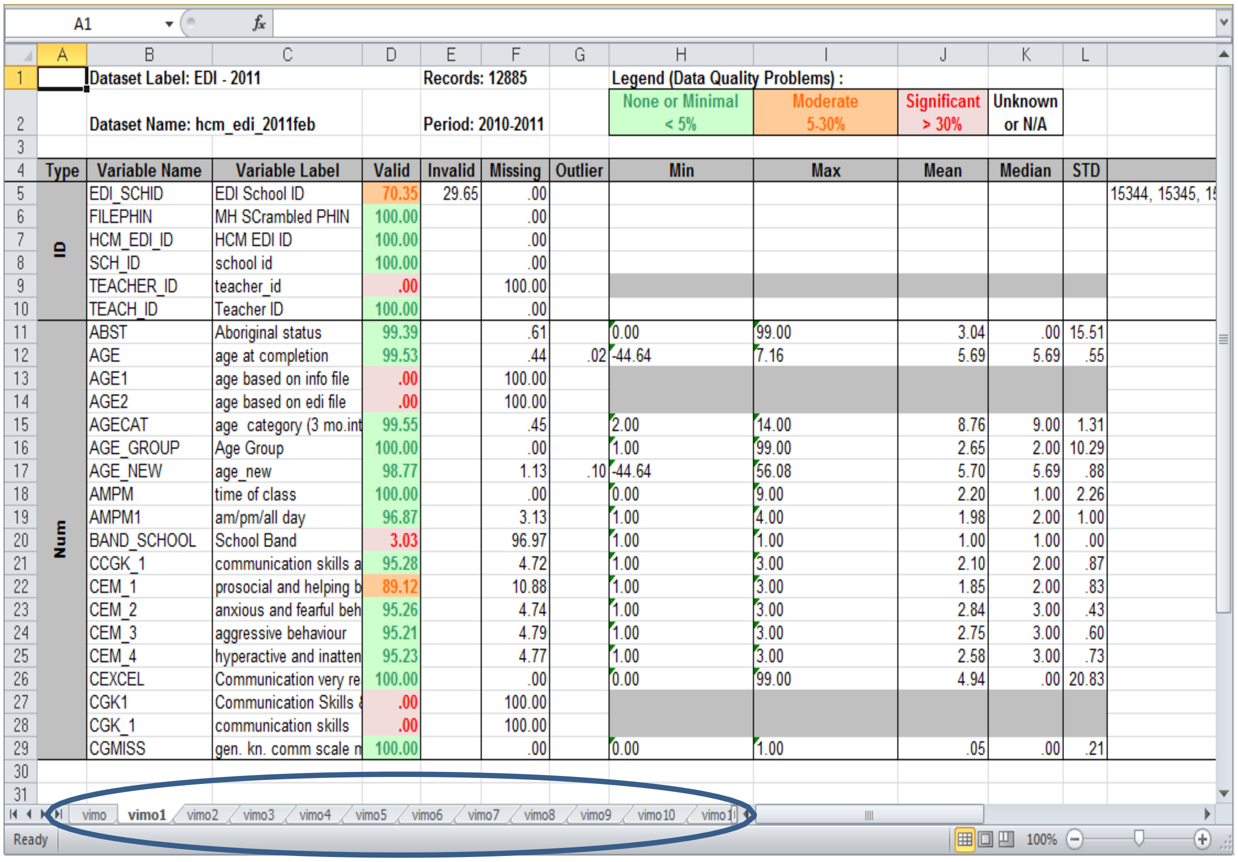
More details about DQ_GEN Macro

Before running the %DQ_GEN macro, the following **must be** done **once**.

1. Open Excel and click on Office button or File.
2. Click the Excel Options.
3. Click the Trust Center and Trust Center Settings…
4. Click the Macro Settings and select Trust Access to the VBA project object model. Example of splitting VIMO table into multiple smaller VIMO tables.

Example of splitting VIMO table into multiple smaller VIMO tables

```
%include 'G:\dqmacro\*.sas';

options mprint;

%let dq_dir = T:\Sayh\test\testnewdqversion;
%let dq_name = hcm_edi_2011feb;

%include 'G:\dmusers\sayh\hcmo\EDI\fmt.sas';

%dq_meta(domain=social, db=hcm_edi_2011feb,
fmt='G:\dmusers\sayh\hcmo\EDI\edi2011varfmt.txt');

%dq_vimo(ds=social.hcm_edi_2011feb, invalidchk=y,
     postals=CL_POSTALCODE NEW_P_CODE NEW_P_CODE2
     POSTAL_CODE POSTAL_CODE_HCM POSTAL_SCH POSTAL_USED
     P_CODE P_CODE_E,
     nooutlier='G:\dmusers\sayh\hcmo\EDI\DQ\nooutliervars2011s.
     txt', idvars=EDI_SCHID HCM_EDI_ID SCH_ID TEACHER_ID
     TEACH_ID);

  %dq_gen(ds=social.hcm_edi_2011feb, period=2010-2011,
    dir=&dq_dir, wrkbook=&dq_name, save=y, rnglen=25);
```



Dataset Label: EDI - 2011 Records: 12885

Dataset Name: hcm_edi_2011feb Period: 2010-2011

Legend (Data Quality Problems) :

None or Minimal < 5%	Moderate 5-30%	Significant > 30%	Unknown or N/A

Type	Variable Name	Variable Label	Valid	Invalid	Missing	Outlier	Min	Max	Mean	Median	STD	
ID	EDI_SCHID	EDI School ID	70.35	29.65	.00							15344, 15345, 15
	FILEPHIN	MH SCrambled PHIN	100.00		.00							
	HCM_EDI_ID	HCM EDI ID	100.00		.00							
	SCH_ID	school id	100.00		.00							
	TEACHER_ID	teacher_id	.00		100.00							
	TEACH_ID	Teacher ID	100.00		.00							
Num	ABST	Aboriginal status	99.39		.61		0.00	99.00	3.04	.00	15.51	
	AGE	age at completion	99.53		.44	.02	-44.64	7.16	5.69	5.69	.55	
	AGE1	age based on info file	.00		100.00							
	AGE2	age based on edi file	.00		100.00							
	AGECAT	age category (3 mo.int	99.55		.45		2.00	14.00	8.76	9.00	1.31	
	AGE_GROUP	Age Group	100.00		.00		1.00	99.00	2.65	2.00	10.29	
	AGE_NEW	age_new	98.77		1.13	.10	-44.64	56.08	5.70	5.69	.88	
	AMPM	time of class	100.00		.00		0.00	9.00	2.20	1.00	2.26	
	AMPM1	am/pm/all day	96.87		3.13		1.00	4.00	1.98	2.00	1.00	
	BAND_SCHOOL	School Band	3.03		96.97		1.00	1.00	1.00	1.00	.00	
	CCGK_1	communication skills a	95.28		4.72		1.00	3.00	2.10	2.00	.87	
	CEM_1	prosocial and helping b	89.12		10.88		1.00	3.00	1.85	2.00	.83	
	CEM_2	anxious and fearful beh	95.26		4.74		1.00	3.00	2.84	3.00	.43	
	CEM_3	aggressive behaviour	95.21		4.79		1.00	3.00	2.75	3.00	.60	
	CEM_4	hyperactive and inatten	95.23		4.77		1.00	3.00	2.58	3.00	.73	
	CEXCEL	Communication very re	100.00		.00		0.00	99.00	4.94	.00	20.83	
	CGK1	Communication Skills &	.00		100.00							
	CGK_1	communication skills	.00		100.00							
	CGMISS	gen. kn. comm scale m	100.00		.00		0.00	1.00	.05	.00	.21	

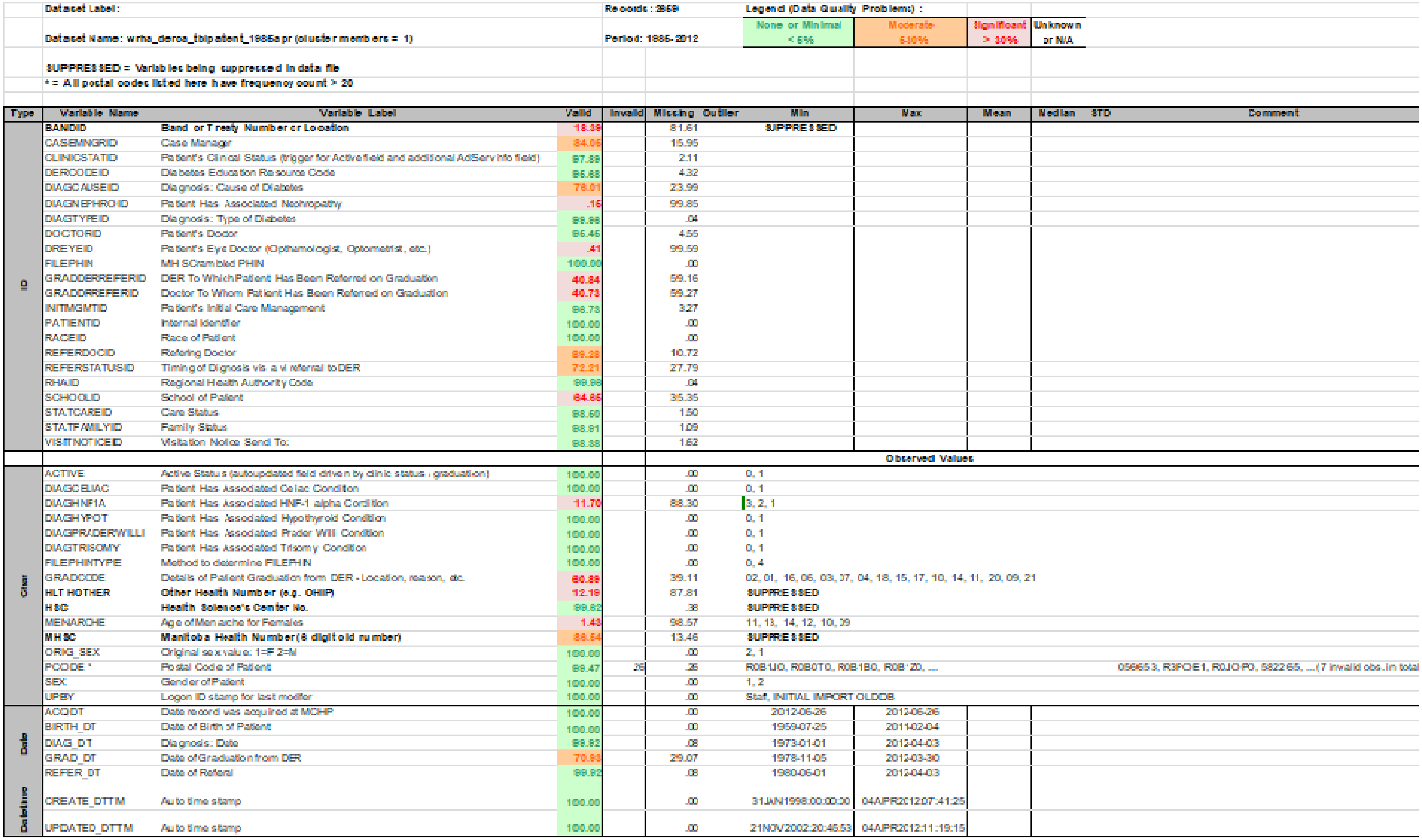
Vimo contains all variables, Vimo1 contains the first 25 variables, and Vimo2 contains the next 25 and so on…

Example

```
%include 'G:\dqmacro\*.sas';

%let dq_dir = T:\Sayh\test\testnewdqversion;  /*directory where
to save
                                          DQ output*/
%let dq_name = tblpatient;   /* Name of the excel DQ output */

proc format;                 /* load format */
  %include
'T:\Sayh\test\testnewdqversion\fmt_derca.txt';
run;

%dq_meta(domain=health, db=wrha_derca_,
       fmt='T:\Sayh\test\testnewdqversion\derca_varfmt.txt');


%dq_vimo(ds=health.wrha_derca_tblpatient_1985apr, invalidchk=Y,
       suppvar=bandid hlthother hsc mhsc, postals=pcode,
       memnum=1, idvars=bandid casemngrid clinicstatid
       dercodeid diagcauseid
             diagnephroid diagtypeid doctorid dreyeid
             gradderreferid graddrreferid initmgmtid patientid
             raceid referdocid referstatusid rhaid schoolid
             statcareid statfamilyid visitnoticeid);

%dq_contents(domain=health, db=wrha_derca_)

%dq_link(domain=health, db=wrha_derca_tblpatient_1985apr,
       phin=filephin, type=filephintype)

%dq_agreement(domain=health, db=wrha_derca_tblpatient_1985apr,
           phin=filephin, birthdt=birth_dt)

*  Note that the results of the above macros will be saved in the
   excel workbook with worksheets: vimo, linkability, phintype,
   overview and agreement;
*  Run %dq_gen to automatically generate DQ tables and chart;

%dq_gen(ds=health.wrha_derca_tblpatient_1985apr, period=1985-
      2012, Dir=&DQ_Dir, wrkbook=&DQ_name, memnum=1, save=Y);

Note: When running %dq_gen, an Open File - Security Warning
     window will pop up, click open and the program will run.
```

Outputs

VIMO Table



Dataset Label:

Records: 2859

Dataset Name: wrha_deros_tblpatient_1985apr (cluster members = 1)

Period: 1985-2012

Legend (Data Quality Problem:) :

None or Minimal < 5%	Moderate 5-30%	Significant > 30%	Unknown or N/A

SUPPRESSED = Variables being suppressed in data file

* = All postal codes listed here have frequency count > 20

Type	Variable Name	Variable Label	Valid	Invalid	Missing	Outlier	Min	Max	Mean	Median	STD	Comment
ID	**BANDID**	**Band or Treaty Number or Location**	18.39		81.61		SUPPRESSED					
	CASEMNGRID	Case Managr	84.05		15.95							
	CLINICSTATID	Patient's Clinical Status (trigger for Active field and additional AdServ Info field)	97.89		2.11							
	DERCODEID	Diabetes Education Resource Code	95.68		4.32							
	DIAGCAUSEID	Diagnosis: Cause of Diabetes	76.01		23.99							
	DIAGNEPHROID	Patient Has Associated Nephropathy	.15		99.85							
	DIAGTYPEID	Diagnosis: Type of Diabetes	99.96		.04							
	DOCTORID	Patient's Doctor	95.45		4.55							
	DREYEID	Patient's Eye Doctor (Opthamologist, Optometrist, etc.)	.41		99.59							
	FILEPHIN	MH SCrambled PHIN	100.00		.00							
	GRADDERREFERID	DER To Which Patient Has Been Referred on Graduation	40.84		59.16							
	GRADDRREFERID	Doctor To Whom Patient Has Been Referred on Graduation	40.73		59.27							
	INITMGMTID	Patient's Initial Care Management	96.73		3.27							
	PATIENTID	Internal Identifier	100.00		.00							
	RACEID	Race of Patient	100.00		.00							
	REFERDOCID	Refering Doctor	89.28		10.72							
	REFERSTATUSID	Timing of Dignosis vis a vi referral to DER	72.21		27.79							
	RHAID	Regional Health Authority Code	99.96		.04							
	SCHOOLID	School of Patient	64.65		35.35							
	STATCAREID	Care Status	98.50		1.50							
	STATFAMILYID	Family Status	98.91		1.09							
	VISITNOTICEID	Visitation Notice Send To:	98.38		1.62							
							Observed Values					
Char	ACTIVE	Active Status (autoupdated field driven by clinic status : graduation)	100.00		.00		0, 1					
	DIAGCELIAC	Patient Has Associated Celiac Condition	100.00		.00		0, 1					
	DIAGHNF1A	Patient Has Associated HNF-1 alpha Condition	11.70		88.30		3, 2, 1					
	DIAGHYPOT	Patient Has Associated Hypothyroid Condition	100.00		.00		0, 1					
	DIAGPRADERWILLI	Patient Has Associated Prader Willi Condition	100.00		.00		0, 1					
	DIAGTRISOMY	Patient Has Associated Trisomy Condition	100.00		.00		0, 1					
	FILEPHINTYPE	Method to determine FILEPHIN	100.00		.00		0, 4					
	GRADCODE	Details of Patient Graduation from DER - Location, reason, etc.	60.89		39.11		02, 01, 16, 06, 03, 07, 04, 18, 15, 17, 10, 14, 11, 20, 09, 21					
	HLTHOTHER	**Other Health Number (e.g. OHIP)**	12.19		87.81		**SUPPRESSED**					
	HSC	**Health Science's Center No.**	99.62		.38		**SUPPRESSED**					
	MENARCHE	Age of Menarche for Females	1.43		98.57		11, 13, 14, 12, 10, 09					
	MHSC	**Manitoba Health Number(6 digit old number)**	86.54		13.46		**SUPPRESSED**					
	ORIG_SEX	Original sex value: 1=F 2=M	100.00		.00		2, 1					
	PCODE *	Postal Code of Patient	99.47	.26	.26		R0B1J0, R0B0T0, R0B1B0, R0B1Z0, ...					056653, R3POE1, R0JOP0, 582265, ...(7 invalid obs. in total
	SEX	Gender of Patient	100.00		.00		1, 2					
	UPBY	Logon ID stamp for last modifer	100.00		.00		Staff, INITIAL IMPORT OLDDB					
Date	ACQDT	Date record was acquired at MCHP	100.00		.00		2012-06-26	2012-06-26				
	BIRTH_DT	Date of Birth of Patient	100.00		.00		1959-07-25	2011-02-04				
	DIAG_DT	Diagnosis: Date	99.92		.08		1973-01-01	2012-04-03				
	GRAD_DT	Date of Graduation from DER	70.93		29.07		1978-11-05	2012-03-30				
	REFER_DT	Date of Referal	99.92		.08		1980-06-01	2012-04-03				
Datetime	CREATE_DTTM	Auto time stamp	100.00		.00		31JAN1998:00:00:30	04APR2012:07:41:25				
	UPDATED_DTTM	Auto time stamp	100.00		.00		21NOV2002:20:46:53	04APR2012:11:19:15				

Overview Table

Domain	SPDS Table	Dataset Label	Number of Records	Number of Fields
HEALTH	WRHA_DERCA_TBLDOCTOR_1985APR		1370	7
HEALTH	WRHA_DERCA_TBLGUARDIAN_1985APR		6089	8
HEALTH	WRHA_DERCA_TBLLABTEST_1985APR		78234	12
HEALTH	WRHA_DERCA_TBLPATIENT_1985APR		2659	45
HEALTH	WRHA_DERCA_TBLSEENBY_1985APR	Pediatric Diabetes - Seen by	12578	3
HEALTH	WRHA_DERCA_TBLVISIT_1985APR		24224	34
HEALTH	WRHA_DERCA_TBLXXTEST_1985APR		27	7

Linkability Table

Dataset	Total Number of Records	Number of Linkable Records	% Linkable Re cords	Number of Linkable Individuals
WRHA_DERCA_TBLPATIENT_1985APR	2659	2201	82.78	2201

Phintype Table

FILEPHINTYPE	WRHA_DERCA_TBLPATIENT_1985APR
0 MH verified against concurrent registries	82.78
4 MCHP db specific ScrPHIN - No MH found	17.22

Agreement Table

Dataset Name	Degree of Agreement with Registry - Sex (Kappa Statistic)	Degree of Agreement with Registry – Date of Birth (Kappa Statistic)
wrha_derca_tblpatient_1985apr	0.9945	0.989

Referential Integrity Check Macro

Description: Referential integrity means that there is a matching key between two databases. One database (primary) contains a single record for each key variable - client information and variable labels (formats) are examples of primary tables. The other database (foreign) may contain any number of records for each key variable.

Primary Key should contain only unique values – no missing values are allowed. Foreign key may contain any number of values but all existing values must be in the primary table.

This macro check for the following criteria.

1. Primary Key is checked for any duplicate or missing values.
2. Values in the foreign table are matched to the primary table. Orphan values (those in the foreign table but not in the primary table) are identified.

Syntax

```
%dq_ref_int_check(primary= , foreign= ,
               key=, f_key= , debug=0,
               odsout=);
```

Parameters:

primary = Primary Dataset containing the primary key. This data set should contain only one record/key value. Primary key must not contain any missing values

foreign = Foreign dataset. This dataset may contain multiple values of the key variable. All key values in the foreign data must appear in the primary dataset.

key = Key variable or variables

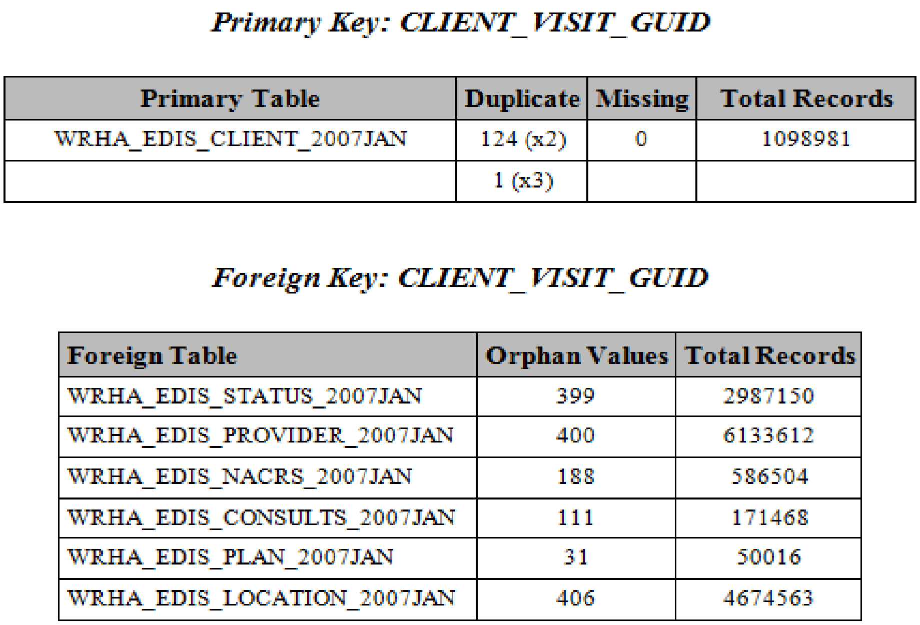
f_key = OPTIONAL foreign key variable(s). This option is not required if the key variables have the same name on both datasets. The order of the key variables must be the same in key and f_key.

Debug = OPTIONAL. If the word debug or debug=1 or debug=YES is passed then mprint and notes are turned on. Otherwise notes and mprint are turned off.

odsout = Location and name to save the output (output must end with an extension of .rtf). If blank, then output will be shown in sas output windows only.

Example:

```
* Example of running reference integrity check for a group of
  foreign tables with common foreign key variables;

%dq_ref_int_check(primary=health.Wrha_edis_client_2007jan ,
                 foreign=health.Wrha_edis_status_2007jan
                        health.Wrha_edis_provider_2007jan
                        health.Wrha_edis_nacrs_2007jan
                        health.Wrha_edis_consults_2007jan
                        health.Wrha_edis_plan_2007jan
                        health.Wrha_edis_location_2007jan,
                 key=CLIENT_VISIT_GUID, debug=1,
                 odsout='T:\Sayh\test\rtftest1.rtf') ;
```



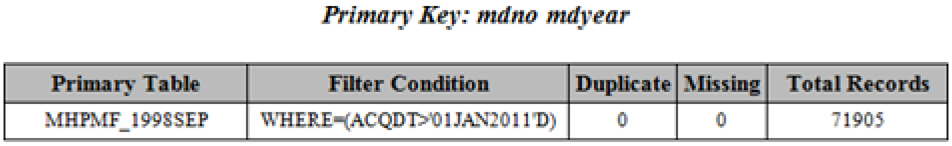
Primary Key: CLIENT_VISIT_GUID

Primary Table	Duplicate	Missing	Total Records
WRHA_EDIS_CLIENT_2007JAN	124 (x2)	0	1098981
	1 (x3)		

Foreign Key: CLIENT_VISIT_GUID

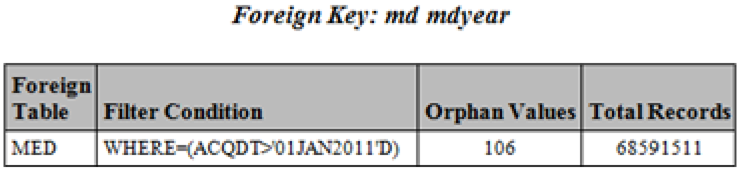
Foreign Table	Orphan Values	Total Records
WRHA_EDIS_STATUS_2007JAN	399	2987150
WRHA_EDIS_PROVIDER_2007JAN	400	6133612
WRHA_EDIS_NACRS_2007JAN	188	586504
WRHA_EDIS_CONSULTS_2007JAN	111	171468
WRHA_EDIS_PLAN_2007JAN	31	50016
WRHA_EDIS_LOCATION_2007JAN	406	4674563

```
* Example of filtering date variable from within

macro call; data med(index=(md mdyear)) ;
  set health.mhmed_1997apr(where=(acqdt>'01jan2010'd) keep=acqdt
                              md servdt acqdt);
  drop servdt ;
  if servdt< '30sep2010'd then
  mdyear='201009' ; else if
  servdt<'31dec2010'd then mdyear='201012'
  ; else if servdt<'31mar2011'd then
  mdyear='201103' ; else mdyear='201106' ;
run;

%dq_ref_int_check(primary="health.mhpmf_1998sep(where=(acqdt>'01ja
               n2011'd))", foreign =
               "med(where=(acqdt>'01jan2011'd))",
               key=mdno mdyear, f_key=md mdyear, debug=1,
               odsout='T:\Sayh\test\rtftest2.rtf')
```



Primary Key: mdno mdyear

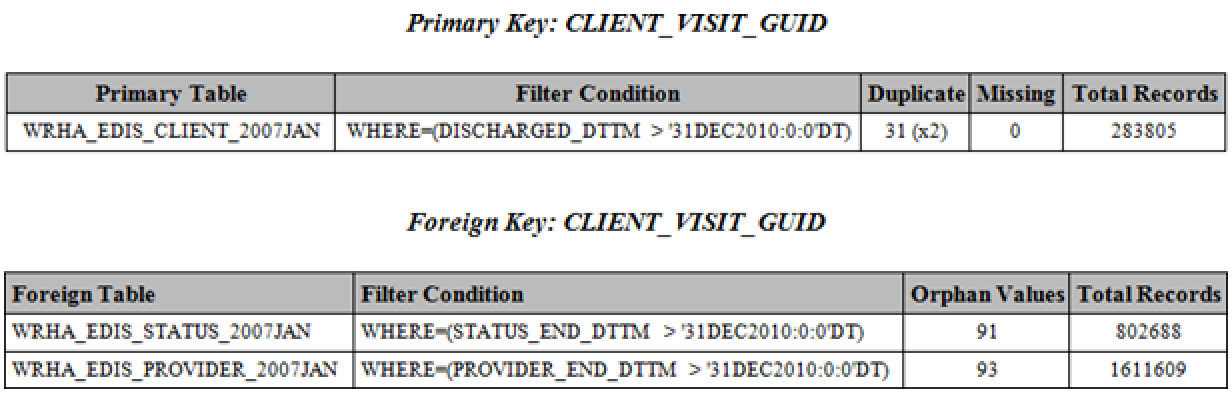
Primary Table	Filter Condition	Duplicate	Missing	Total Records
MHPMF_1998SEP	WHERE=(ACQDT>'01JAN2011'D)	0	0	71905



Foreign Key: md mdyear

Foreign Table	Filter Condition	Orphan Values	Total Records
MED	WHERE=(ACQDT>'01JAN2011'D)	106	68591511

```
* Example of filtering date time variable from within macro call;

%dq_ref_int_check(
primary=
"health.wrha_edis_client_2007jan(WHERE=(discharged_dttm >
'31dec2010:0:0'dt))", foreign=
"health.wrha_edis_status_2007jan(WHERE=(status_end_dttm >
'31dec2010:0:0'dt))"
"health.wrha_edis_provider_2007jan(where=(provider_end_dttm >
'31dec2010:0:0'dt))",
key=CLIENT_VISIT_GUID, debug=1, odsout="T:\Sayh\test\rtftest3.rtf")
```



Primary Key: CLIENT_VISIT_GUID

Primary Table	Filter Condition	Duplicate	Missing	Total Records
WRHA_EDIS_CLIENT_2007JAN	WHERE=(DISCHARGED_DTTM > '31DEC2010:0:0'DT)	31 (x2)	0	283805

Foreign Key: CLIENT_VISIT_GUID

Foreign Table	Filter Condition	Orphan Values	Total Records
WRHA_EDIS_STATUS_2007JAN	WHERE=(STATUS_END_DTTM > '31DEC2010:0:0'DT)	91	802688
WRHA_EDIS_PROVIDER_2007JAN	WHERE=(PROVIDER_END_DTTM > '31DEC2010:0:0'DT)	93	1611609

Note:

When filtering date and date time variables from within macro call, if a single quote is used to quote constant date or date time value, then a double quote must be used to quote the dataset, or vice versa.

Validation Macro

Description: This macro can be used to check for data inconsistency that involves two or more variables, examples of data inconsistency are pregnant man, hospital separation is occurring before admission, etc... This macro scans through the data and count the number of inconsistencies based on the validation rules specified by the user.

This macro can perform the following

1. Cross-field or within record check.
2. Cross-table check (Fields from one table can be checked for inconsistency with other fields in another table).
3.

Syntax:

```
%dq_validation(primary=, pkey=, secondary=, skey=,
               validaterule=, odsout=)
```

Parameters:

primary = name of the primary dataset

pkey = space-delimited list of key variable(s) from the primary dataset

secondary = name of the secondary dataset (leave blank if it is not a cross-table check)

skey = a space-delimited list of key variable(s) from secondary dataset (leave blank if key variable(s) from secondary dataset have the same name as primary dataset)

validaterule = location and name of the TAB delimited text file that contains the rules to check for inconsistencies among variables

odsout = location and name to save the output (output can be saved with an extension of .rtf, .pdf, or .html). If blank, then output will be shown in sas output windows only

Example:

```
%dq_validation(primary=random,
validaterule='T:\Sayh\test\testnewdqversion\validat
erule.txt',
odsout='T:\Sayh\test\testnewdqversion\validate.rtf'
)
```

Validation Rules

Validation rules are stored in a TAB delimited text file which will be used by the macro parameter validaterule=. This text file contains two columns; the first column is the error messages that specified by the user, the second column contains Boolean expression. For example, the first column contains the message: Pregnant man and the second column contains the expression: sex = ‘1’ and preg = ‘1’.

Example:

The following simulated data will be used as an example. Note that running DQ_VIMO on this dataset will not detect any data problem. However, there are 5 observations with data inconsistency. One observation contains a pregnant man, one contains an 80-year-old pregnant woman, and 3 observations with admission date occurred after separation date.

Obs	admitdt	sepdt	sex	preg	age
1	26APR2011	27APR2011	2	0	86
2	13DEC2011	16DEC2011	2	1	23
3	14AUG2010	19AUG2010	2	1	30
4	01DEC2011	02FEB2012	2	1	46
5	02MAY2012	10MAY2012	1	0	67
6	14NOV2010	24NOV2010	2	0	29
7	30APR2012	29APR2012	2	0	84
8	17SEP2011	01OCT2011	2	0	34
9	16JUN2011	01JUL2011	2	0	10
10	22SEP2010	20DEC2010	2	1	65
11	08APR2011	15MAY2011	1	1	30
12	07OCT2012	15NOV2012	1	0	29
13	11MAR2012	31MAR2012	2	0	42
14	01SEP2010	30NOV2010	1	0	68
15	20AUG2011	11SEP2011	1	0	49
16	20SEP2012	13OCT2012	1	0	72
17	02MAY2012	27MAY2012	1	0	29
18	11MAR2011	06APR2011	2	0	96
19	17JUN2011	14JUL2011	1	0	23
20	01AUG2012	02SEP2012	1	0	20
21	27DEC2012	29JAN2013	1	0	20

Obs	admitdt	sepdt	sex	preg	age
22	10OCT2010	15NOV2010	2	1	27
23	09AUG2012	16SEP2012	1	0	26
24	26JAN2011	25JAN2011	2	0	11
25	14JAN2012	23FEB2012	1	0	41
26	06APR2012	17MAY2012	1	0	29
27	20JAN2011	04MAR2011	1	0	77
28	24OCT2010	07DEC2010	2	1	46
29	30JUN2012	14AUG2012	1	0	20
30	10SEP2011	27OCT2011	2	0	42
31	23OCT2010	10DEC2010	21	1	80
32	11MAY2010	10MAY2010	1	0	20

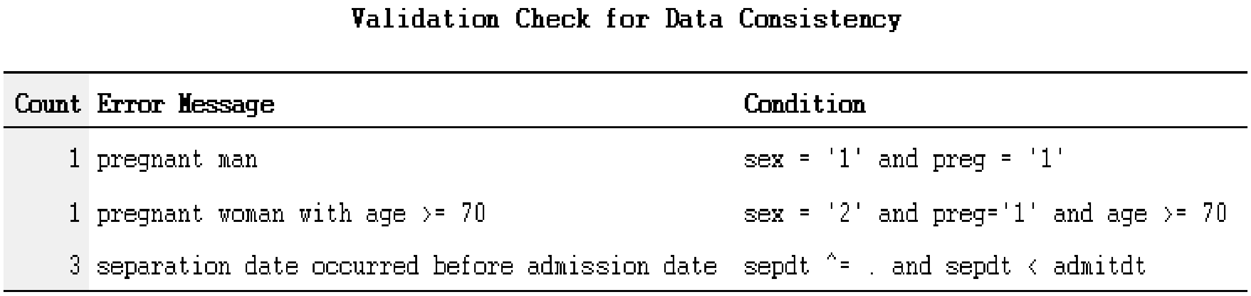
The following TAB delimited text file contains the validation rules that can be used to check for data inconsistency of the above simulated dataset. Note that, the first column contains the error messages and the second column contains the rules specified by the user.

Error message	Rule
Pregnant man	sex = ‘1’ and preg = ‘1’
Pregnant women with age >= 70	sex = ‘2’ and preg = ‘1’ and age >= 70
Separation date occurred before admission data	Sepdt ^= . and sepdt < admdt

Calling dq_validation macro:

```
%include 'G:\dqmacro\*.sas';

%dq_validation(primary=random,
             validaterule=T:\Sayh\test\testnewdqversion\validaterule.
             txt,
             odsout='T:\Sayh\test\testnewdqversion\validate.rtf')
```

Output produced by dq_validation macro:



Validation Check for Data Consistency

Count	Error Message	Condition
1	pregnant man	sex = '1' and preg = '1'
1	pregnant woman with age >= 70	sex = '2' and preg='1' and age >= 70
3	separation date occurred before admission date	sepdt ^= . and sepdt < admitdt

Note that if none of the data inconsistency was found, the following message will print in the SAS output windows:

******** **No data inconsistency was found based on the rules provided** ********

Caution: Note that this macro checks for data inconsistency based on the validation conditions (rules) that are specified by the user. It is important to make sure that all the rules are appropriately specified. If the output produces unexpected result, it is recommended to check the validation rules for errors.