

Self-validating data

Christof Binder, BIOP AG, Basel, Switzerland
Dante diTommaso, F. Hoffmann-La Roche Ltd, Basel, Switzerland

ABSTRACT

SAS® programmers routinely work with data from external sources. Standard operating procedures, driven by resource constraints, often instruct them to assume that data are correct and to expend no resources on data review. Inevitably, data violate such optimistic procedures, leading to undesirable results. This paper presents a set of three macros for testing data plausibility up front, thus avoiding long debugging sessions of programs that run under a faulty assumption of data cleanliness. These techniques include: Comparing actual and expected variable distributions; Looking at the distribution of combinations of variables; and Plausibility testing using Benford's Law.

INTRODUCTION

The data from clinical trials is the basis of the work for programmers.

However the quality of the data can be suboptimal for several reasons. One possibility why the data on your desk is not in a perfect shape is that there were errors in processing it. A lab-device can output several instead of only one observation, a CRF gets copied somehow and entered multiple times, a problem with the extraction of the data may lead to multiple observations, data conversion might not work properly.

As a programmer you have to be reasonably sure that the data you are working with are in fact plausible.

The validation techniques presented in this paper do not check single datum for agreement within a larger database. Validation checks, queries, and manual review by the data manager / monitor and are not the subject of this paper.

Instead, these techniques focus on whether some corruption could have disrupted data collection and transfer processes. Are the data reasonably consistent with the fundamental assumption of clinical trials that trial subjects are a valid, representative sample of the broader target population?

NORMALITY / POISSON DISTRIBUTION

The distribution of variables is one of the properties that can give away some flaws in the data. Although a random sample from a population does not need to follow a normal or poisson distribution for continuous and categorical variables respectively, a quick look at the distribution can be revealing. What would need to follow a normal distribution are the residuals of an analysis, but testing this goes beyond the scope of this paper.

The distribution is simple for variables that are not dependent of the treatment. It usually is an aim of a study to show some difference between treatments. So we cannot expect these variables (efficacy variables) to follow a simple distribution for the complete population. If the treatment group is known the distribution can be plotted for each treatment group separately.

```

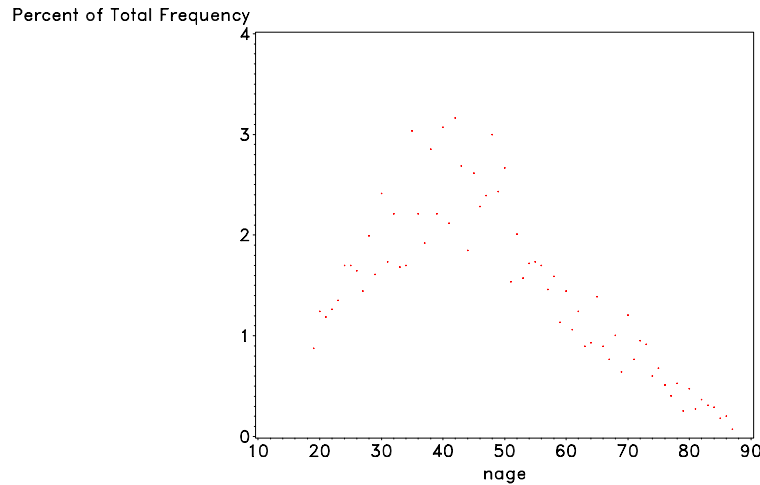
/*****
* macro test_dist
* purpose: test if a variable follows normal or poisson distribution
* parameters: ds          : dataset where the variable is to be found
*              test_var   : name of the variable to test
*              norm_poisson : Normal distribution (N) or poisson (P)
*                          default N
*              trt_group   : name of the treatment group variable
*                          default NULL
* input:
* sample usage:
*   %test_dist( ds = DM, test_var = AGE); *does DM.AGE follow a normal;
*                                     *distribution;
*   %test_dist( ds = DM, test_var = SEX, norm_poisson="P");
*                                     *does DM.SEX follow a poisson distribution;
*****/

```

PhUSE 2006

```
* %test_dist( ds = VS_PULSE, test_var = PULSE, trt_group = GROUP);
*               *does the variable PULSE follow a normal distribution ;
*               *stratified by the variable GROUP;
*****/
```

A sample output may look like this:



COMBINATION OF VARIABLES

Several – basically unrelated – variables of one patient can be concatenated together and the frequency of these combinations can be examined. The expectation would be that the frequency distribution is more or less uniform or at least there should not be one (or a few combination(s) that pop up much more than the rest. If we see a large frequency of a special combination, without a clinical explanation, then some included variable may have unintended repeats. To be clear: this can be perfectly OK, given study design, but may warrant closer inspection. The number of possible combinations can grow rather quickly with the number of variables concatenated. It is therefore best to concentrate on a few variables that are unrelated and expected to vary across the patients in a study. If the inclusion/exclusion criteria restricts the possible values of a variable, it is even expected that the combinations are not uniformly distributed.

```
/******
* macro test_comb
* purpose: test if a combination of variables is evenly distributed
* parameters: ds          : dataset where the variable is to be found
*              char_vars: names of the character variables to combine and test
*              num_vars  : names of the numeric variables to combine and test
*              plot      : should a plot of the data be produced.
*                      default Y
* input:
* sample usage:
* %test_comb(ds = AE, char_var = AETERM AESER, num_var= AESTDY);
*****/
```

BENFORD'S LAW

Benford (1938) observed that the distribution of the initial digits of seemingly unrelated data is not uniform but the digit "1" is expected much more frequently than the digit "2" and more frequently than the digit "3" and so forth. The probability of a number starting with the digit D is $P_D = \log_{10} (1 + 1/D)$. The frequencies are shown in Table 1.

D	P_D	D	P_D
1	0.301	6	0.067
2	0.176	7	0.058
3	0.125	8	0.051
4	0.097	9	0.046
5	0.079		

Table 1: Distribution of the frequencies for the digits 1 to 9

The underlying reason for this distribution are far from trivial, but it allows us to have a simple indication if something

in the data might need a closer look.

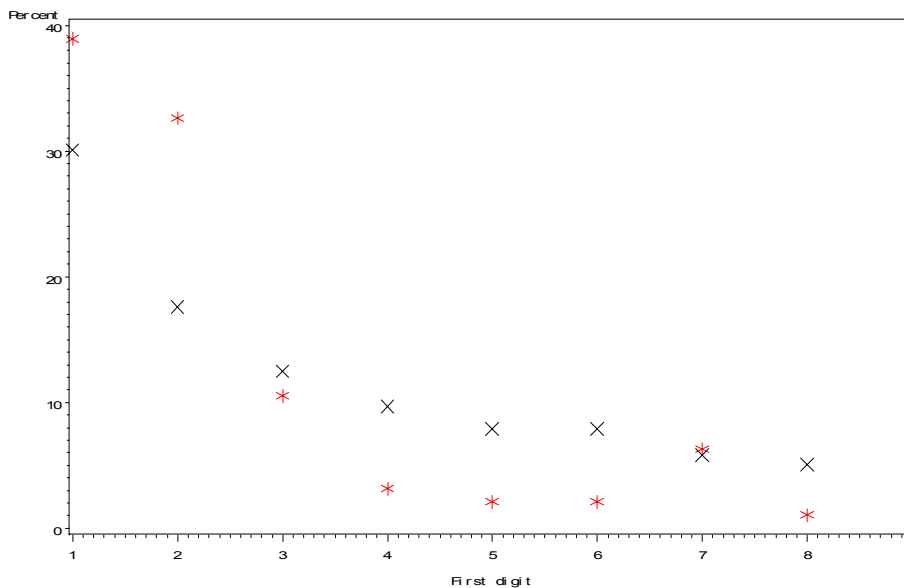
It is however important to note that this distribution is only valid for data that is not dimensionless. Therefore variable that contain dimensionless data need to be excluded from the investigation. Dimensionless variables include things like subject identifiers (e.g subject id, center id) record numbers or parameter numbers. Variables that are dimensioned are measurements (e.g. weight, height or labor measurements).

```

/*****
* macro test_benford
* purpose: test if the numeric variables follow Benford's law
* parameters: ds      : dataset where the variable are to be found
*              excl    : list of variables to exclude (dimensionless variables)
*              plot     : should a plot of the data be produced.
*                      default Y
* input:
* sample usage:
* %test_benford(ds = LB, excl = USUBJID LBSEQ LBGRPID);
*****/

```

A sample output might look like this:



CONCLUSION

The techniques presented here are simple to apply to clinical trial data. They can raise red flags, efficiently drawing attention to potential problems in the data. These tests make no conclusions about the quality of your data, but only help visualize and understand your clinical database. We cannot offer criteria for 'failing' any of these tests, as this judgment can only be made by a study team in the context of a specific study. Suspicious results could, however, could draw attention to unusual properties that are well worth looking into.

Running these tests on your data can help you to gain confidence in the quality of the data. They might also help you to spot interesting properties of the data that could have been missed otherwise.

REFERENCES

Frank Benford: The law of anomalous numbers, Proceedings of the American Philosophical Society, Vol. 78, No. 4, (March 1938), pp. 551--572.

Cited at http://en.wikipedia.org/wiki/Benford's_law

CONTACT INFORMATION

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

Christof Binder
Biometrical Practice, BIOP
Centralbahnstrasse 9
CH-4051 Basel, Switzerland

Dante diTommaso
F. Hoffmann-La Roche Ltd
CH-4002 Basel, Switzerland

PhUSE 2006

Work Phone: ++41 (0)61 227 94 42
Fax: ++41 (0)61 227 94 49
Email: christof.binder@biop.ch
Web: www.biop.ch

Work Phone: ++41 (0)61 687-4314
Email: dante.di_tommaso@roche.com
Web: http://www.roche.com/home/divisions/div_phar.htm

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

APPENDIX

Source code for the macros presented above. This is still work in progress and the authors welcome comments and suggestions for improvement and additions.

```
/* *****  
* macro test_dist  
* purpose: test if a variable follows normal or poisson distribution  
* parameters: ds          : dataset where the variable is to be found  
*              test_var   : name of the variable to test  
*              norm_pois  : Normal distribution (N) or poisson (P)  
*                          default N  
*              trt_group  : name of the treatment group variable  
*                          default NULL  
* input:  
* sample usage:  
*   %test_dist( ds = DM, test = AGE); *does DM.AGE follow a normal distribution;  
*   %test_dist( ds = DM, test = SEX, dist="P"); *does DM.SEX follow a ;  
*                                           *poisson distribution;  
*   %test_dist( ds = VS_PULSE, test = PULSE, trt_group = GROUP);  
*           *does the variable PULSE follow a normal distribution ;  
*           *stratified by the variable GROUP;  
* *****/  
  
%macro test_dist ( ds =, test_var =, norm_pois = "N", trt_group = );  
  
    data __temp;  
        set &ds;  
    run;  
  
    %if &norm_pois = "N" %then %do;  
        ** stratified by treatment group;  
        %if %length( &trt_group ) %then %do;  
            proc freq data = __temp;  
                table &test_var*&trt_group / out = __gtemp;  
            run;  
  
            proc sort data=__gtemp;  
                by &trt_group;  
            run;  
  
            proc plot data = __gtemp;  
                plot percent*&test_var = '*' ;  
                by &trt_group;  
            quit;  
        %end;  
  
        ** no stratification;  
    %else %do;  
        proc freq data = __temp;  
            table &test_var / out = __gtemp;
```

PhUSE 2006

```

run;

proc plot data = __gtemp;
    plot percent*&test_var = '*';
quit;
%end;
%end;

%if &norm_pois = "P" %then %do;
    %* similar for poisson distribution;
    %* snip;
%end;

%** clean up;
proc sql;
    drop table __temp, __gtemp;
quit;
%mend;

/*****
* macro test_comb
* purpose: test if a combination of variables is evenly distributed
* parameters: ds          : dataset where the variable is to be found
*              char_vars: names of the character variables to combine and test
*              num_vars  : names of the numeric variables to combine and test
*              plot       : should a plot of the data be produced.
*                          default Y
* input:
* sample usage:
*   %test_comb(ds = AE, char_var = AETERM AESER, num_var= AESTDY);
*****/
%macro test_comb (ds =, char_var =, num_var =, PLOT = Y);
    %let plot = %upcase(%substr(&plot,1,1));

    data _temp;
        set &ds;
        keep &char_var &num_var;
    run;

    data _temp;
        set _temp;
        length comb $500;
        %let n=1;
        %let var = %scan(&char_var, &n);
        %do %while ( %length( &var ) );
            comb = trim(comb) !! "-" !! trim(left(&var));
            %let n = %eval (&n + 1);
            %let var = %scan(&char_var, &n);
        %end;

        %let n=1;
        %let var = %scan(&num_var, &n);
        %do %while ( %length( &var ) );
            comb = trim(comb) !! "-" !! (put(&var, best. -1));
            %let n = %eval (&n + 1);
            %let var = %scan(&num_var, &n);
        %end;
    run;

    proc freq data =_ temp;

```

PhUSE 2006

```

table comb / out = __temp;
run;

%if &plot eq Y %then %do;
    proc plot data = __temp;
        plot comb * percent = '*';
    quit;
%end;

%** clean up;
proc sql;
    drop table __temp, __temp;
quit;

%mend;

/*****
* macro test_benford
* purpose: test if the numeric variables follow Benford's law
* parameters: ds      : dataset where the variable are to be found
*              excl   : list of variables to exclude (dimensionless variables)
*              plot    : should a plot of the data be produced.
*              default Y
* input:
* sample usage:
*   %test_benford(ds = LB, excl = USUBJID LBSEQ LBGRPID);
*****/
%macro test_benford ( ds, excl, plot=YES );
    %let plot = %upcase(%substr(&plot,1,1));

    data __temp;
        set &ds;
        %if %length(&excl) %then drop &excl; ;
    run;

    data __temp;
        set __temp;
        keep digit;
        array _num {*} _numeric_;
        array _chr {*} _character_;

        do i = 1 to dim( _num );
            dig = substr( put(_num{i}, ??best12. -L), 1, 1 );
            digit = input( dig, ??1. );
            if digit then output;
        end;

        do i = 1 to dim( _chr );
            dig = substr( left(_chr{i}), 1, 1 );
            if index('123456789', dig) then do;
                digit = input(dig, 1.);
                if digit then output;
            end;
        end;
    run;

    proc freq data = __temp;
        table digit /out = __gtemp;
    run;

```

PhUSE 2006

```
%if &plot eq Y %then %do;
  data __gtemp;
    set __gtemp;
    lable percent = 'Percent'
        digit    = 'First digit'
        ;
    array benf[9] _temporary_ (30.1 17.6 12.5 9.7 7.9 6.7 5.8 5.1 4.6);
    expected = benf[digit];
  run;

  proc plot data = __gtemp;
    plot percent*digit = '*'
        expected*digit = 'x' / overlay;
  quit;
%end;

proc sql;
  drop table __gtemp, __temp;
quit;

%mend;
```