

/\* Transform qualitative variables in binary variables  
DECEMBER 2007

Programmers: Adrien FRANCAIS  
Statistical Enginners

Albert Bonniot Institute  
Team 11-Outcome of cancer and critical illnesses  
UJF-INSERM U823  
38706 LA TRONCHE CEDEX  
FRANCE  
04 76 54 94 00

## MACRO VARIABLES

TABLE : work table which includes origin variable.

VARIABLE : variable to transform in several binary variables

MISS : if you want, you can precise a value for missing data  
For example, you can put miss=999 to avoid missing values in the file.

VARCIBLE : outcome variable (binary variable)

VARSORTIE : table which contains obtained binary variables

Before using this macro, you have to :

- Prepare your database with correct variables
- Compile the macro.
- Execute the macro :

Example to compute on your computer to directly and easily apply the macro:

```
data try;
input dc14jours      sexe transfert dnr sympt @20;
cards;
0      1      0      0      601
0      1      1      0      602
1      1      1      0      602
1      1      1      0      602
0      1      1      0      605
0      0      1      0      606
0      1      0      0      607
1      0      1      1      608
1      0      1      0      624
1      1      1      0      610
0      0      0      0      602
0      1      1      0      601
0      1      1      0      609
```

```

0    1    0    0    609
1    0    1    1    610
0    1    0    0    611
0    0    1    0    612
1    1    0    1    613
0    1    0    1    614
1    1    1    1    615
1    0    1    0    611
1    0    1    0    .
0    0    0    0    .
0    1    1    0    .
0    0    0    0    614
0    0    .    0    613
1    0    1    1    612
1    1    1    1    601
1    0    1    0    601
1    0    1    0    602
0    1    0    0    608
0    1    0    0    608
0    0    0    0    .
1    0    1    0    .
1    0    0    1    .
0    1    0    0    .
0    1    1    1    609
1    1    1    1    .
0    1    1    0    617
0    0    1    0    605

```

```

;
run;

```

```

%creationvarbinaires(table=try,variable=sympt,miss=999,varcible=dc14jours,varsortie=tablei
ntermediaire);
%creationvarbinaires(table=tableintermediaire,variable=sexe,miss=999,varcible=dc14jours,va
rsortie=tablefin);

```

You can see LOG screen for information of algorithm.

If you use this macro for work to be published please use the following citation:

Francais Adrien

Transform qualitative variables in binary variables

\*/

```
%macro creationvarbinaires(table=,variable=,miss=,varcible=,varsortie=);
```

```
%let error=0;
```

```
%let errormodal=0;
```

```
/*some errors messages*/
```

```
%if &table = %then %do;
```

```

        %put "ERROR: You didn't give the work table : Please enter the name of work
table";%let error=1;
%end;
%if &variable = %then %do;
        %put "ERROR: You didn't give the variable to test : Please enter a qualitative
variable";%let error=1;
%end;
%if &varcible = %then %do;
        %put "ERROR: You didn't give outcome variable : Please enter a binary
variable";%let error=1;
%end;
%if &varsortie = %then %do;
        %put "ERROR: You didn't give the name of final table : Please enter the name of the
final table";%let error=1;
%end;

/*to test if the outcome variable is binary*/
%put "We test if the outcome variable is binary";
%if &table ne %then %do;
        ods output onewayfreqs=aaa(keep=&varcible);
        title "Test of variable &varcible";
        proc freq data=&table;tables &varcible;run;
        data aaaa;set aaa;id=1;run;

        data aaaaa;set aaaa;by id;    retain errormodal 0;
        if first.id and &varcible ne 0 then errormodal=1;
        if last.id and &varcible ne 1 then errormodal=1;
        call symput ('errormodal',errormodal);
        run;
%end;
%if &errormodal = 1 %then %do;
        %put "ERROR: You didn't give a correct outcome variable : Please enter a binary
variable";%let error=1;
%end;

%if &error = 1 %then %do;
        %put "PLEASE CORRECT ERROR(S) DETAILED ABOVE";
%end;

/*if there is no error then we start the macro*/
%if &error ne 1 %then %do;

        proc sort data=&table;by descending &variable ;run;
        /*length of variables must be inferior to 8!!!*/
        data &table;set &table; /*if &variable=. then &variable=697;*/
        %if &miss ne %then %do;
                if &variable=. then &variable=&miss;
        %end;
        rename &variable=%substr(&variable,1,8);run;

```

```
%let varreduit=%substr(&variable,1,8);

/*Proc glmmod produces automatically binary variables*/
proc glmmod data=&table noprint
    outdesign=zzz (drop=&varreduit.0)
        prefix=&varreduit
        zerobased;

    class &varreduit;
    model &varcible = &varreduit;
run;

/*we add new binary variables*/
data &varsortie;merge &table zzz;run;
%put %upcase("New binary variables created are available in the table '&varsortie'");
%end;
%mend;
```