***Frequency weighting***

SPSS macros by Kirill Orlov

kior@akado.ru, ttnphns@gmail.com

<https://www.spsstools.net/en/KO-spssmacros>

All rights reserved

*Frequency weighting.* Achieving wanted proportion sizes of respondent groups by univariate or multivariate (rim) weighting. You can select total N, impose restriction upon weighting individual cells or cases, weight several subsamples in parallel, take account of initial weights.

*Read “*[*About SPSS macros*](https://www.spsstools.net/en/KO-aboutmacros)*” what are they and how to run them.*

# MACRO !KO\_WEIGR: WEIGHTING GROUPS

Version 5, Jun 2023 (Version 1, Aug 2001). Tested on SPSS Statistics 22, 26, 28.

!KO\_weigr vars= *v1 v2* /\*One or more grouping variables in which the ratios of group sizes must be

/\*changed, name-by-name

/wname= /\*Name for the weight variable (by default, weight\_$)

/restr= *6* /\*Optional: restriction to weight change: don’t impose (NONE, default);

/\*or try to not allow cell frequences lower than this (number); or

/\*ban change the cells with negative residuals (NEGRES); or inhibit by variable

/\*(varname)

/wvar= /\*Optional: variable with initial weights

/splvar= /\*Optional: do weighting separately for subsamples defined by this

/\*variable(s) (dataset will come out sorted by it)

/total= /\*Needed sample sum of weights (simulated N): number;

/\*or SELECTED (default, initial sum of weights of cases taken into the weighting);

/\*or varname one of VARS (initial sum of weights of cases taken into the weighting in

/\*this variable)

/report= /\*Report of the done weighting: YES (default) or NO

/hist= /\*Optional, under REPORT=YES: histogram of weights obtained (YES or NO, default)

/mode= PAIR /\*Style of specifying PROPS: PAIR (default) or LIST

/method= /\*If multiple VARS: rim weighting (RIM, default) or Cartesian (CART)

/iter= *5* /\*Number of iterations; active only with rim weighting

/props= *'1 .6' '2 .4' / '1 .35' '2 .15' '3 .50'* /\*For each of VARS variables: either

/\*ALLEQ/ALLVEQ (all groups weight equally),

/\*or list of groups (codes) and required sizes for

/\*them; sizes either as proportions or as counts - everywhere;

/\*may use keywords MISSING, SYSMIS, ELSE/ELSEV (in place of a code),

/\*HOLD, ADJUST (in place of a size).

Minimal specification VARS, PROPS. PROPS must go the last.

The macro performs frequency weighting of groups. By altering case weights in the working dataset, the macro takes the groups indicated by the user to the specified sizes (proportions of the total sample size N). The sample size can be initial or any user-specified. There can be one or several grouping variables. The groups that participate in weighting should be indicated; not mentioned groups will be excluded from the sample (their weight will be *sysmis*). Optionally, you can restrict weight modification of some cases or of whole subgroup cells. Also, you can do weighting separately by subsamples or make use of initial, background weights.

The macro’s result is the emergent variable *WEIGHT\_$* (you can specify other name) weighting the dataset. In this variable, cases not undergone the weighting procedure receive sysmis weights; other cases receive valid values in the weight variable and will constitute the weighted sample. Which cases will and which will not enter the weighting will be defined by you specifying the PROPS subcommand (as well as by filtering, if the dataset is in filtered state).

Output window contains table(s) with the marginal frequencies achieved with the weighting. The macro computes as well *Weighting Efficiency* proportion. This quantity *WE* is:

where *ui* is the initial weight of case *i*; *wi* is the resultant weight of the case; *n* weight in toto, as there are cases in the sample. If there were no initial weights, that is, all *ui* = 1, the formula simplifies to:

*Weighting Efficiency* proportion is a coefficient of balancedness of weights (and nonnegative values, in general). When weights *w*, occurred as the result of weighting, differ among dataset cases little, the coefficient is high, it approaches 1. But if weights *w* differ among dataset cases much – relative their overall level, – the coefficient is shifted toward 0. If the weighting aimed to imitate in the sample the structure of the population, *WE* below 0.7-0.8 can be considered the sign of that the sample before weighting matched the population insufficiently well; or, in other words, that the weighting pursuing the said imitation showed to be a distortion of such a strength that should not be ignored: it was a “risky” weighting. Index *WE* is needs not that sum of *w* be equal to *n* or be equal to the sum of *u*; it decribes only values *w*, only theirs profile. More detail about *WE*[[1]](#footnote-1) coefficient – see “Appendix”.

During the work, the macro creates temporary variables with names having 5 symbols ‘$’ in succession, for example, v$$$$$2. There should be no names identical to them among the variables of your dataset.

Features of the macro:

Univariate “cell” weighting (by one variable or by combination of several) or multivariate “rim” (aka “raking”) weighting. See s/c METHOD.

You can choose the manner target sizes are specified – paired (‘code proportion’ ‘code proportion’) or listed (‘code code’ proportion proportion).

You can prefer to specify target counts rather than target proportions, though this is less universal way.

You can use shorthand keywords ALLEQ/ALLVEQ to set equal proportions to all groups.

You can use keywords SYSMIS, MISSING, ELSE/ELSEV in place of group code and keywords HOLD and ADJUST in place of target proportion.

Groups which you withhold, which you don’t set target sizes to – are excluded from weighting procedure. Their weights will be sysmis. Sysmis weights will also get filtered out cases, if the input dataset is in filtered state. Both these and those cases – they do not participate in the weighting.

If you want cases not participating in the weighting to preserve their initial value in the weight variable, specify s/c WVAR and WNAME as one and the same variable.

You can set target total sample size several ways (s/c TOTAL).

You can indicate variable with baseline, initial case weights.

Initial case weights 0 remain so in the end; target sizes 0 are allowed. Cases with weight 0 at output from the macro are included in the weighted sample, in contrast to cases with sysmis weight.

You can perform weighting in subsamples, in parallel, by specifying macro a variable to split the dataset.

You can carry out various kind of differential restriction of weights change.

EXAMPLE 1.

!KO\_weigr vars= sex /props= '1 .5' '2 .5'.

!KO\_weigr vars= sex age /iter= 5 /props= '1 .5' '2 .5' /'1 .5' '2 .3' '3 .2'.

!KO\_weigr vars= sex age /iter= 5 /props= ALLEQ /'1 .5' '2 .3' '3 .2'.

!KO\_weigr vars= sex age /method= CART /props= ALLEQ /'1 .5' '2 .3' '3 .2'.

In the first command, counts of two groups comprising variable *SEX*, males (code 1) and females (code 2), are being equalized: they were specified equal proportions 0.5 (50%).

The second command does two-variate weighting: by gender the counts are being equalized, and by age (three groups) the requested ratio is 50%–30%–20%.

The third command is equivalent to the second, if variable *SEX* lacks missing values, that is, only two categories, 1 and 2, exist there. While if it has missing data, use of ALLVEQ instead of ALLEQ will make the third command equivalent to the second.

The forth command is a univariate weighting by combination of categories of two variables, i.e., by the variable with 2×3=6 groups and target proportions defined by the specified marginal proportions.

The following fork directs a user, which method of frequency weighting to use when.

*There is single grouping variable*. Univariate weighting. **Indicate it** in VARS and specify target (i.e. the needed, those to arrive at, - usually these are what are observed in the population) proportions of the being weighted groups in PROPS.

*There are multiple grouping variables.*

*Target proportions for cells of the cross-classification (the cross-table) defined by them are known.* This is also univariate weighting. **Create variable** with the cross-groups and act as above. For example, if there are variables SEX (2 groups) and AGE (3 groups), create grouping variable with 2×3=6 groups.

*Target proportions for cells of the cross-classification defined by them are not given, yet they are the functions of the target marginal proportions for you.* (Meaning, that the former can be obtained by cross-multiplying of the latter. And this assumes the absence of correlation between the grouping variables in the population.) This is also univariate weighting, as before. But in order not to create physically the variable with the cross-groups, you may **use the option** METHOD=CART (Cartesian weighting). So, you have to **indicate all** your grouping variables in VARS and target marginal proportions in PROPS, and the macro will, instead of you, compute the target cell proportions.

*Target proportions in cells of the cross-classification defined by them are not known and are not considered to be functions of marginal proportions, or they are just unimportant to you. Known are (or have significance) only target marginal proportions of this table.* This is multivariate, rim weighting (= raking weighting = iterative proportional fitting): METHOD=RIM. You must **indicate all** your grouping variables in VARS and target marginal proportions in PROPS. This method is applied by default of the METHOD s/c when VARS variables are more than one.

See EXAMPLES 5 and 6 to understand the difference between Cartesian weighting and rim weighting.

The following fork is for multivariate, rim weighting only. It is about whether to use the option to restrict weights change (s/c RESTR).

*You suppose (or are ready to accept) that disproportioning forces acted in each of the variables independently*. “Disproportioning forces” is something what have led to the warp of the frequencies observed in the sample in comparison with the target, “authentic”, populational ones. “Acted independently” means: the frequencies distortion in variable SEX (taking the current example) took place irrelatively to group membership in other variables indicated in VARS, in this case – the variable AGE. Differently put, overrepresentation in the sample, say, of females, happened, as the researcher things, with the same “intensity” in all age groups. And analogously from the side of AGE: the distortion of frequencies between age groups was irrelative to the group membership in variable SEX. Under the described, independent from the sides of grouping variables, distortions a warp in any one of them *is accompanied* by warp in other only in the degree of existing natural coherence between these attributes (i.e., the inequality of the population chi-square association to zero). Then do rim weighting usual, without restriction. Omit RESTR or specify RESTR=NONE.

*You suppose that disproportioning force(s) acted inside the cross-classification, and in a certain way.* I.e., it distorted cell frequencies, while distortion in marginal frequencies appeared an outer expression and consequence of these within-table warps.

*The disproportioning force affected the factor responsible for the attributes’ (the grouping variables) cohesion.* “Responsible for cohesion” factor should be understood as in factor analysis – some latent trait common for the grouping variables, due to which they have a nonzero chi-square in the population. If the grouping variables or some of them are linked (in population) by factor(s) it would be parsimonious to think that frequency distortions in the sample happened right “along” the factor(s), it will be parsimonious to think that the frequency warps in the sample occurred just “along” the factor, i.e. they took place between the cells filled owing to its charity (the cells with positive chi-square residuals) rather than between the cells filled by disturbance against it (cells with negative residuals). If so, then at the restoration of the population, target frequencies, there needed is re-weight only among cells with non-negative residuals[[2]](#footnote-2). **Specify** RESTR=NEGRES.

*The disproportioning force was selecting cells differently, and you have a hypothesis or evidence about that between which cells the warp had occurred.* “Hypothesis or evidence” must be expressed by introducing of a reweight-restraining variable: **specify** RESTR=*variable*. Values (0 through 1) of the restraining variable can be equal or different for respondents belonging to the same cell of the cross-classification. See details in s/c RESTR.

Let’s notice that you may use reweight-restraining variable generally always, not only in rim weighting, if its values somehow vary. Because this variable expresses by itself, generally, to what degree it is allowed to change the respondent’s weight.

### Subcommands

# VARS

One or more numeric grouping variables, name-by-name list. If single variable is specified, there will be univariate weighting. If multiple variables, then multivariate rim weighting will be by default; but you can request univariate Cartesian weighting – see METHOD s/c.

Each unique valid or user-missing value in a variable defines own group in it; system-missing value also constitutes one group in it. Thus, every case in the dataset belongs to some group. Those groups in a given variable that you are going to enter into the weighting – you will indicate them in subcommand PROPS. Variables VARS must not have the following negative values: *-999* and *-9999*.

**WNAME**

In this optional subcommand you may indicate a name for the weight variable being created. By default/unspecification, the variable name will be *WEIGHT\_$*. When a variable with the same name as the being created variable already exists in the dataset, it is replaced (updated).

# PROPS

This subcommand must go the last. Enumerate in it, for each variable VARS, the group codes together with the target (the wanted) sizes for them expressed in proportions (may also in absolute frequencies, counts – see about it below).

*Shorthand words ALLEQ/ALLVEQ.*

If need to weight all groups equally in a given variable, you may use keyword ALLEQ or ALLVEQ (may quote the word) – instead of typing codes and sizes. All groups of that variable will receive equal target sizes. ALLEQ considers each unique value in the variable (even a user-missing one) as a group, and system-missing also a group: and all the groups enter the weighting. ALLVEQ considers only valid values in the variable to be groups and to enter the weighting, while all missing values will receive sysmis weights. (Technical remark: if after keyword ALLEQ or ALLVEQ square brackets [ ] are found, they are ignored.)

*Specifying groups and sizes.*

There are two equivalent modes to write the PROPS subcommand – “paired” and “listed”. You may use as you like this or that way, one for all variables VARS. The “paired” mode will be described now. Difference of the “listed” mode from it is given in s/c MODE.

When you specify a list, each pair code–size must be in apostrophes or quotes, and specifications for different variables must be separated with slash (/) and must go in the order in what the variables are listed in VARS. For instance: **‘1 *0.6*’ ‘2 *0.4*’ / ‘1 *0.15*’ ‘2 *0.25*’ ‘3 *0.6*’**. Group codes need not be positive integers. Group code does not have to be a valid value. Giving zero size for a group is allowed, cases of that group will receive 0 in the weight variable.

The following keywords can be used, once for a variable, in place of a group code:

SYSMIS - means “group formed by system-missing value”.

MISSING - means “all missing (user-missing, system-missing) values as a single group”. This word is *not* usable for the same variable along with keyword SYSMIS. MISSING should be carefully applied in combination with codes that are user-missing, because MISSING means *all not yet mentioned* missing values. For example, if code 2 is user-missing status then **‘1 *0.15*’ ‘MISSING *0.6*’ ‘2  *0.25*’** will lead to unexpected result: since code 2 is already implied by MISSING going earlier, group with code 2 as an independent group (separate from other missing values) does not exist after that for the macro. But **‘1 *0.15*’ ‘2  *0.25*’ ‘MISSING *0.6*’** is quite all right: MISSING will imply here the totality of missing values except value 2.

ELSE - means “each one of the not mentioned groups”. It may stand only in the *last* pair 'code size' of a variable.

ELSEV - means “each one of the not mentioned valid groups”. It may stand only in the *last* pair 'code size' of a variable. Unlike ELSE, ELSEV considers all not mentioned missing values in the variable as a single group, and the group is not admitted to weighting. When used after MISSING, ELSEV and ELSE are equivalent since MISSING has already picked all missing values.

Groups that you will not specify, will not be taken by the macro into the weighting procedure, and the weight variable will come out blank – system missing – for those cases. The final *totality of the cases taken into weighting* procedure, the weighted sample, is defined by listwise deletion: if in a variable a group is not taken, these cases are excluded from other variables VARS as well.

The following keywords can be used in place of target proportions (but not of absolute frequencies):

HOLD - means “preserve the group its observed proportion”: namely, that proportion what the group had *among the groups taken* in that variable into the weighting – and had it *prior* the moment of listwise deletion of cases. HOLD is the same as to write the mentioned observed proportion explicitly by number. HOLD should be applied with care: **‘1  *0.15*’ ‘2 *0.25*’ ‘ELSE *HOLD*’** means to preserve for all groups other than 1 and 2 their present proportions, but you must be sure that the sum of those proportions constitutes 0.6, so that the sum of all target proportions amounts to 1 (see *Sum of proportions*, below).

ADJUST - means “make such proportion so that the sum of all proportions from groups taken in a given variable make up 1”. All groups that are ADJUST in the given variable will alter their observed (*prior* the moment of listwise case deletion) share proportionally, by the same no. of times. For example, **‘1  *0.15*’ ‘2 *0.25*’ ‘ELSE *ADJUST*’** sets, to all groups other than 1 and 2, target proportions that are different from their observed proportions by the same factor, while in sum they give 0.6.

*Sum of proportions. Specification via counts.* Normally, the sum of target proportions for each variable should equal 1 (therefore specify proportions with enough number of decimal digits). If it is not equal to 1, the final total sum of weights will differ from the one specified in subcommand TOTAL by the number of times the sum of target proportions differed from 1. Knowing that rule you might specify target proportions, in principle, right in *frequencies* (counts) instead of proportions (see EXAMPLE 4, last paragraph). This however is less flexible mode of specifying the sizes than that by proportions; and you will need s/c TOTAL of the form TOTAL=*number*. You may specify all the target sizes either as proportions or as counts, not mixed. Important: don’t use keywords ALLEQ/ALLVEQ, HOLD, ADJUST if you are specifying target sizes by raw frequencies, counts.

Attention. Be accurate while composing PROPS subcommand. Watch the number of codes and their corresponding sizes to be equal, and there are no extraneous or missing quotes/apostrophes.

**MODE**

By default and with MODE=PAIR, the macro expects that codes and sizes are written in PROPS by “paired” mode for every variable: ‘code size’ ‘code size’… MODE=LIST allows (demands) to write PROPS by “listed” mode for every variable: ‘code code…’ size size… In listed mode, codes are written first, and their list is in quotes or apostrophes; and then their corresponding sizes go. Of course, lists of codes and sizes must be of the same length. All keywords and rules expressed earlier in s/c PROPS are true for MODE=LIST too. Here’s some examples:

|  |  |
| --- | --- |
| Paired mode: | Equivalent record by listed mode: |
| /‘1 0.6’ ‘2 0.4’ /‘1 0.15’ ‘2 0.25’ ‘3 0.6’ | /‘1 2’ 0.6 0.4 /‘1 2 3’ 0.15 0.25 0.6 |
| /‘1 0.15’ ‘3 0.25’ ‘MISSING ADJUST’ | /‘1 3 MISSING’ 0.15 0.25 ADJUST |
| /‘1 0.15’ ‘-2 0.25’ ‘ELSE HOLD’ /ALLEQ | /‘1 -2 ELSE’ 0.15 0.25 HOLD /ALLEQ |

If the quoted list of codes is long, you may simply break/continue it on next line. Or use the accepted in SPSS concatenation sign + (see EXAMPLE 3).

EXAMPLE 2.

!KO\_weigr vars= sex age /iter= 5 /props= ALLEQ /'1 .5' '2 .3' '3 .2'.

!KO\_weigr vars= sex age /iter= 5 /mode= LIST /props= ALLEQ /'1 2 3' .5 .3 .2.

These two commands are equivalent.

EXAMPLE 3.

!KO\_weigr vars= age /mode= LIST /'1 2 3 6' .3 .1 .2 .4.

!KO\_weigr vars= age /mode= LIST /'1 2

3 6' .3 .1 .2 .4.

!KO\_weigr vars= age /mode= LIST /'1 2 ' +

'3 6' .3 .1 .2 .4.

These three commands are equivalent. In the 2nd command the list of codes is simply carried over to a new line, what the macro allows. In the 3rd command carry over is done by string concatenation with + symbol.

# TOTAL

Order here the needed sum of weights of the output weight variable, i.e., the sample size simulated by it. There are three ways:

SELECTED - (also default/unspecification of the s/c) it is the sum of weights equal to the number (or the initial sum of weights) of cases taken into the weighting procedure. What are the cases being taken into weighting procedure – see s/c PROPS.

*Number* - specify the needed sum of weights as a number.

#### Varname (of VARS) - sum of weights equal to the number (or the initial sum of weights) of cases of all the groups taken in this variable into the weighting procedure; it is the number (sum of weights) of cases observed in the variable prior listwise deletion of cases (see s/c PROPS).

If grouping variable is single or if in all grouping variables all existing in them groups are being taken in the weighting, then TOTAL=SELECTED and TOTAL= *variable* will give the same sum of weights.

EXAMPLE 4. Different TOTAL. Suppose we need to weight by two variables, *VAR1* and *VAR2*. The complete (with missing values accounted) frequency cross-table defined by them:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | VAR2 | | | | Total |
| VAR1 |  | Group A (code=1) | Group B (code=2) | Group C (code=3) | Group D (sysmis) |  |
| Group 1 (code=1) | 10 | 12 | 18 | 7 | 47 |
| Group 2 (code=2) | 14 | 7 | 8 | 2 | 31 |
| Total |  | 24 | 19 | 26 | 9 | 78 cases in all in the dataset |

!KO\_weigr vars= var1 var2 /iter= 5 /total= SELECTED /props= '1 .5' '2 .5' /'1 .333' '2 HOLD' '3 ADJUST'.

!KO\_weigr vars= var1 var2 /iter= 5 /total= var1 /props= '1 .5' '2 .5' /'1 .333' '2 HOLD' '3 ADJUST'.

!KO\_weigr vars= var1 var2 /iter= 5 /total= 780 /props= '1 .5' '2 .5' /'1 .333' '2 HOLD' '3 ADJUST'.

* In all three commands the groups being selected for weighting and their target proportions are the same. All the 78 respondents of the dataset partition in 2 groups by variable *VAR1*, and both are taken into weighting because are mentioned in PROPS. These groups are appointed to equalize in size (target proportions are 0.5). By variable *VAR2*, 78 respondents partition in 4 groups: three coded plus system-missing group. According to PROPS, that last group is not taken into the weighting because it isn’t mentioned by any of keywords SYSMIS, MISSING, or ELSE. Only the three coded groups are taken in *VAR2* (their summative frequency is 24+19+26=69). For group A target proportion is 0.333; for group B – the one observed currently, i.e. 19/69 (not 78) =0.275; for group C – the rest, i.e. 1-(0.333+0.275)=0.392.
* So, 78 respondents are taken into the weighting in *VAR1*, but 69 – in *VAR2*. “Listwise deletion” – exclusion from both variables cases not taken at least in one of them – admits 69 respondents in the weighting: all those taken in *VAR2*. Now, in the first of the three macro runs, TOTAL=SELECTED (also default), i.e., at output the sum of weights of the weight variable, N of the sample, will be 69 – because that is how many respondents entered the weighting finally.
* In the second run, TOTAL=*VAR1*, so the sample N at output will be 78. This is how many respondents were selected into the weighting according to *VAR1*, prior listwise deletion. In this instance, that is the number of cases existing in in all the dataset.
* In the third macro run, the user specified TOTAL by an arbitrary number 780, so that will be the simulated by him N of the sample.

!KO\_weigr vars= var1 var2 /iter= 5 /total= SELECTED /props= '1 .5' '2 .5' /'1 .333' '2 HOLD' '3 ADJUST' 'SYSMIS 0'.

* This case seems to should be giving result like in the first run above, because group D (sysmis), despite it is now added into the weighting, has target proportion 0. But actually the result will be different. Since group D is taken, there 78 respondents (and not 69 as before) from *VAR2* entered. Therefore the “hold” target proportion for group B is now different: 19/78=0.244. Correspondingly, there became different the “adjust” target proportion for group C: 1-(0.333+0.244+0)=0.423. Finally, 78 respondents entered the weighting, so TOTAL=SELECTED will give N of sample 78 at output.

!KO\_weigr vars= var1 var2 /iter= 5 /total= 1 /props= '1 50' '2 50' /'1 20' '2 45' '3 35'.

* In this example target sizes are specified as raw frequencies (counts), not proportions. (In such a case TOTAL=number is necessary.) The sum of target frequencies = 100 here. Because TOTAL is specified as 1, N of sample at output will be 1\*(100/1)=100. You must not use keywords ALLEQ/ALLVEQ, ADJUST, and HOLD when you specify target sizes not by proportions. You may not specify frequencies for some variables but proportions for other: must do either this or that way for all the variables. If you specify the sizes as counts and their sums appear different for different variables, the largest sum will become the resultant N of the weighted sample.

**METHOD**

This subcommand is in effect if VARS variables are more than one. When there is a single variable, the macro carries out univariate (cell) weighting by that variable. When variables are multiple, subcommand METHOD gives the opportunity to choose between the multivariate rim weighting and the univariate (cell) weighting by the combination of variables:

RIM - (also default/unspecification of the s/c) rim aka raking weighting. It is a multivariate weighting in the sense that the weighting is done successively and independently by each variable. This process is iterative and you will need ITER s/c.

CART - Cartesian weighting, it is weighting by the combination of variables. The weighting is univariate in the sense that it is done so as if there takes place weighting by the single variable which is the complete combination (Cartesian product) of groups between several variables. The option exists in the macro in order for you not to waste effort on manual creation of the said combinatorial variable.

With METHOD=CART, target proportions are located in the elementary cells of the cross-table defined by VARS variables, and are calculated by the macro by simple cross-multiplication of the target marginal proportions PROPS you specify. This means that we are presuming to have zero association (i.e., no interaction) between VARS variables in the population. This also means that we are presuming the number of groups in the population to be equal to the size of the Cartesian product between the sets of groups (categories) of VARS variables. If you consent to both suppositions, you may use METHOD=CART instead of manual creation of the combinatorial variable, calculation the cell target proportions, and weighting by the variable.

With METHOD=RIM, target proportions are the marginal proportions PROPS that you specified, themselves. The fitting goes right to them, without positing target proportions in the cells of the cross-tabulation. The conceptual assumption with this method is that we agree to consider the “distortion” of frequencies in the data prior the weighting – the distortion we want to “play back” by the weighting – as having occurred independently by VARS variables. This method is frequently applied when we don’t know the target cross-tabulation cell proportions but don’t want to impose zero association (i.e., Cartesian weighting), or we don’t bother strictly about the cell proportions.

EXAMPLE 5. Univariate weighting by the crossing of two variables. *X* is a 5-group variable with codes 1 2 3 4 5. *Y* is a 3-group variable with codes 1 2 3. Suppose we were not given the 15 target proportions but were given the target marginal proportions, and were instructed that in our case the association between *X* and *Y* must become zero. This is the case to use METHOD=CART.

!KO\_weigr vars= x y /wname= wei\_1 /mode= LIST /method= CART

/props= '1 2 3 4 5' .10 .25 .15 .30 .20 /'1 2 3' .50 .35 .15.

* Now let’s show what that run essentially has done. Let’s create variable *XY* with 5×3=15 categories, calculate “manually” the 15 target cell proportions for it, and weight by the variable:

compute xy= x\*10+y. /\*X and Y are integer codes between 1 and 9, the simplest way to create

execute. /\*the Cartesian product is like this

matrix. /\*Calculate target proportions in cells

compute x\_props= {.10, .25, .15, .30, .20}. /\*Target proportions by X

compute y\_props= {.50, .35, .15}. /\*Target proportions by Y

compute xy\_props= t(x\_props)\*y\_props. /\*Cross-multiplication

print xy\_props /format= f8.4. /\*yielding 5x3=15 target proportions for the XY variable

print msum(xy\_props). /\*Check the sum to equal 1

end matrix. /\*The obtained propotions inserted in the following call of the macro:

!KO\_weigr vars= xy /wname= wei\_2 /mode= LIST

/props= '11 12 13 21 22 23 31 32 33 41 42 43 51 52 53'

.05 .035 .015 .125 .0875 .0375 .075 .0525 .0225 .15 .105 .045 .1 .07 .03.

* Weight variables *WEI\_1* and *WEI\_2* are identical. We’ve got our sample with the proportions inside the *X* by *Y* crosstabulation equal to those 15 proportions which are the function of the marginal target proportions .10, .25, .15, .30, .20 for X and .50, .35, .15 for *Y*. The results are identical because in the second case we specified in PROPS all 15 target proportions – precisely what the Cartesian weighting implied.
* The weighting will reach the target proportions specified if all 5×3=15 subgroups are represented in the data.
* The weighting is univariate, it is non-iterative.
* Because in the situation of zero association between target marginal proportions the profile of *X* is one and the same of every level of *Y* (and likewise vice versa), we might obtain the same result but yet one more way: two-step univariate weighting under s/c SPLVAR – see EXAMPLE 11.

EXAMPLE 6. Same *X*, *Y* variables and same target marginals, as above, but weighting will be 2-variate rim.

!KO\_weigr vars= x y /wname= wei\_4 /iter= 6 /mode= LIST

/props= '1 2 3 4 5' .10 .25 .15 .30 .20

/'1 2 3' .50 .35 .15.

* The result (the case weights) obtained in this weighting is different from that obtained in EXAMPLE 5. We attained the same target marginal proportions as there, but the two-way cross-tab cell proportions differ from those 15 ones having been target in the univariate weighting.
* Rim weighting aims to correct marginal distributions only, because the notion behind this method assumes that disproportions among cell frequencies inside the cross-table had been the result of superposition of independent marginal disproportioning processes. Rim weighting “stages” to unwind it back. It was the distortion *process* characterized as independent for the *X* and the *Y* sides (so the rectification process is likewise independent for the sides).
* But in EXAMPLE 5, we had a particular situation where the rows and columns of the cross-table of target proportions show the *state* of independence (therefore we could utilize marginal proportions directly) – this does not imply independent “process”.

# ITER

This subcommand acts only when there are multiple variables in VARS and rim weighting is requested, and it is then required. Specify in it the number of iterations to fit the marginal distributions. The more iterations the more precisely group target proportions will be achieved. You will need more iterations when there are many grouping variables and/or groups in them, and also when you put on RESTR.

# RESTR

With the help of this subcommand (it is the macro author’s gimmick) one can restrict altering of weight for separate cases in the dataset or for entire selected cells in the cross-table formed by the grouping variables VARS. Restraining some cases or entire cells from their weight change shifts the task of weight change onto other cases/cells, thereby making the latter to change weight “for themselves and instead that guy”. The macro does not obey requests of restraining in groups which target proportion is specified as 0. Select:

NONE - (also default/unspecification of the s/c) Don’t put restriction, all dataset cases taken into the weighting procedure can in full change their weight.

*Number* - (possible only in multivariate, rim weighting) Try not to allow the frequency in a nonempty elementary cell to fall below this value. Use this if you see that without restriction a frequency in some cell turned out intolerably low for you. Specify positive number. Of course, you cannot set this number to be disparately big if there are many cells while the case sample is not big.

NEGRES - (possible only in multivariate, rim weighting) Forbid cells with negative observed frequency residual to change their frequency – these respondents (cases) will not change their input weight. Weight change will occur only among respondents belonging to cells with nonnegative residuals.

*Varname* - User-specified restraining. Values of that variable may vary between 0 and 1. The restraining value bears straightforward meaning: it is the share (proportion) of the initial case weight that is allowed to get altered (however, how big will be that alteration and whether it will be increase or decrease depend on demands on the side of target proportions and on the size of restraining values of other cases; still, in general, the tendency will be of course that the closer to 0 is the restraining value for a given case the smaller will be weight change for that case).

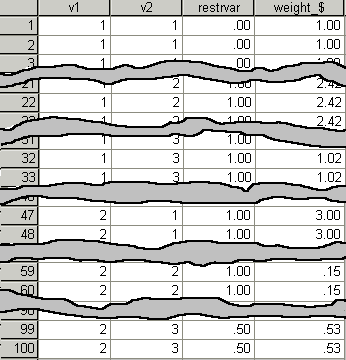
*About restraining variable:*

* Sum of values inside a *group* cannot be equal to 0.
* With univariate weighting (including METHOD=CART), equality or nonequality of sums in different groups does not affect the result. Introducing of restraining variable in univariate weighting targets the aim to restrain separate cases from changing their weight.
* With multivariate (rim) weighting, the sum within a *cell* of the cross-table may equal 0 (which means the cell is completely protected from re-weighting). Arithmetic mean of values inside a cell will be that share of the initial frequency or sum-of-weights that is allowed to get changed in the cell. In multivariate weighting, introducing of a restraining variable has the aim to restrain cells and/or individual cases from changing their weight.
* If a case value is missing, the case’s weight will be missing: such a case is not taken into the weighting (same as a case filtered out).

RESTR=NONE is equivalent to the restraining variable with all values =1; RESTR=NEGRES is equivalent to binary restraining variable (1 if respondent belongs to a cell with nonnegative residual, 0 if to a cell with negative residual).

EXAMPLE 7. Multivariate weighting with restraining variable.

!KO\_weigr vars= v1 v2 /restr= restrvar /iter= 7 /props= '1 .6' '2 .4' / '1 .55' 'ELSE ADJUST'.



* *V1* in the dataset consists of 2 groups (codes 1 and 2), *V2* of 3 groups (codes 1, 2, 3). According to PROPS, all the groups in both variables are taken into the weighting, so some weights will be possessed by all cases of the dataset.
* In *V2*, a specific target proportion is set for group 1, while for two other it is requested to alter their observed proportions proportionally, in sum they must yield 0.45.
* According to restraining variable RESTRVAR, it is prohibited to change the frequency in cell (*V1*=1,*V2*=1) because all restraining values for these cases = 0. Therefore, in the weight variable *WEIGHT\_$* weight did not change there, remained = 1. Frequency in cell (*V1*=2,*V2*=3) was allowed to change not in full scale (restraining values = 0.5). If there had been no restraining (value had been 1) the weight of the cell’s cases would have decreased stronger.

# WVAR

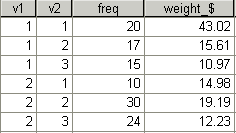
If the dataset has weight variable that should be taken into account, i.e., used as initial weights, indicate it here. The dataset does not have to be actually weighed by it already. Any values of that variable except positive valid ones are treated by the macro as valid weight 0 and will come out as 0 in the output weight variable (provided the case was taken into the weighting).

A specific case of using WVAR is when the dataset presents itself not “respondents X variables” data but data aggregated in groups. The variable containing frequencies of this table must be then specified in WVAR (EXAMPLE 8).

If WVAR coincides by name with the variable specified in WNAME, the WVAR variable, aka WNAME, will be rewritten only for cases taken into the weighting; but cases not participating into the weighting will retain at output their weight values they had initially. It may be used in order to modify one and the same weight variable gradually by parts (EXAMPLE 9).

EXAMPLE 8. Re-weighting of a frequency table[[3]](#footnote-3).

!KO\_weigr vars= v1 v2 /wvar= freq /iter= 5 /props= '1 .6' '2 .4' /'1 .5' '2 .3' '3 .2'.



* Input data are aggregated data, frequency cross-table stretched out in two columns. The frequencies are variable *FREQ*. The macro recalculated the frequencies onto the requested in PROPS marginal distributions. New frequencies are variable *WEIGHT\_$*.

EXAMPLE 9. Modifying weight variable only for some cases.

compute wei= 1.

compute filt= sex=1.

filter by filt.

!KO\_weigr vars= v1 /wvar= wei /wname= wei /props= ALLEQ.

compute filt= sex=2.

!KO\_weigr vars= v2 /wvar= wei /wname= wei /props= ALLVEQ.

* In both runs of the macro, both subcommands WVAR and WNAME are specified, and the weight variable is one and the same in them, *WEI*. In these conditions, the macro will leave the unprocessed cases their initial weight. Primordially, *WEI* is computed as 1 for all cases.
* In the 1st run of the macro, cases *SEX*=1 (filter) participate. They will change their weight. Other cases will keep in *WEI* the initial value (i.e. 1).
* In the 2nd run of the macro, cases *SEX*=2 (filter) participate. They will change their weight. Other cases will keep in *WEI* the initial value (for cases *SEX*=1 that will be values obtained in the preceding run). If there are missings in *V2*, these cases won’t enter the weighting – because PROPS=ALLVEQ, these cases will keep in *WEI* their initial weight (equal to 1, because they haven’t changed it so far).

# SPLVAR

You may indicate a categorical variable (or several such) defining disjoint subsamples for each of which the macro will do all the same weighting separately, in parallel. The output dataset will occur to be sorted and split by values of that variable (combination of values, if several variables). If TOTAL=*number* then this number pertains to each subsample; thus, sum of weights in all subsamples will be equal at output, which is your use if your task is not only to bring the subsamples to the same group structure in them but also to equalize the sizes of the subsamples.

Watch after so the groups which you specified target sizes to in PROPS s/c be present *all* in the data in every individual subsample. In subsamples where some of the groups, declared in PROPS, are absent, the weighting will pass incorrectly.

EXAMPLE 10.

!KO\_weigr vars= school level /splvar= year /iter= 7 /total= 4200.5 /mode= LIST /props=

'ALLEQ' /'1 2 3' .37 .37 .26.

split file off.

* In the survey of school pupils, there represented are 30 schools (coded 1 through 30) , and the pupils are divided by age into levels 1 (junior), 2 (medium), and 3 (senior). The survey was conducted two times – in two different years (coded 1 and 2). The schools and the levels are the same in both years, but the individual respondents are different. The whole sample size is 8401, unbalanced between the years and between the schools.
* The researcher wants to equalize contributions of all the schools in each year, and to equalize N in both years. Second, he wants to keep the distribution of ages (levels) as it is, observed in the sample currently: 37%:37%:26%.
* One solution (above) would be to do weighting in two years in parallel (SPLVAR=*YEAR*), requesting equal total 4200.5 for each year. The weighting itself is 2-variate rim: by *SCHOOL* (all categories equal) and by *LEVEL* (37%:37%:26%).

compute school#year= school.

if year=2 school#year= school#year\*100. /\*There are 60 different codes thereby

compute level#year= level.

if year=2 level#year= level#year\*100. /\*There are 6 different codes thereby

!KO\_weigr vars= school#year level#year /iter= 7 /mode= LIST /props=

'ALLEQ' /'1 2 3 100 200 300' .185 .185 .13 .185 .185 .13.

* Another and equivalent approach in this setting would be to process two combination variables: *SCHOOL*\**YEAR* (30\*2=60 categories) and *LEVEL*\**YEAR* (3\*2=6 categories) – instead of *SCHOOL* and *LEVEL* under SPLVAR by year. S/c TOTAL=8401 is unnecessary, because all the 8401 case base taken into the weighting is preserved as the sum-of-weights by default.

EXAMPLE 11. Shows syntax equivalent, by result, to what was in EXAMPLE 5 (see there).

!KO\_weigr vars= x /wname= wei\_3 /splvar= y /mode= LIST

/props= '1 2 3 4 5' .10 .25 .15 .30 .20.

!KO\_weigr vars= y /wname= wei\_3 /wvar= wei\_3 /mode= LIST

/props= '1 2 3' .50 .35 .15.

split file off.

* In the first command, we weight by *X*, but do it in parallel in subsamples defined by *Y*. After that, in the second command we weight by *Y* the whole sample, using the weights obtained in the first command as the initial weights.
* With the same effect, we could permute the order of the two-step weighting: weight first by *Y* in subsamples of *X*, and then weight by *X*, using the initial weights just obtained.
* The results will coincide with the ones obtained in EXAMPLE 5, if all the 5×3=15 subgroups exist in the data.

**REPORT**

By default and REPORT=YES, summary of results is reported: table(s) with the obtained marginal frequencies (groups’ proportions) after the weighting, and also some weight statistics and Weighting Efficiency proportion. You can suppress the report by specifying REPORT=NO.

**HIST**

Optional HIST=YES plots histogram of the resultant weights. HIST does not work if REPORT=NO.

### Special regimes

The macro obeys dataset filteredness (commands FILTER, USE): filtered out cases will receive weight sysmis and won’t participate in the weighting. Do not use dataset splitting – there’s the special subcommand SPLVAR for that in the macro. The macro does not recognize the weighted status of the dataset automatically: to take into account initial weights you must indicate that weight variable in WVAR. The macro takes off dataset splitness if SPLVAR was not used. The macro ignores temporary (under TEMPORARY command) transformations.

### Some questions

*Should I delete the weight variable created by the macro (weight\_$) before new run of the macro?* No. That does not play any role. If the variable remained, the macro will recalculate it. Another thing is when you want to use it as starting weights, - then indicate it in WVAR.

*At multivariate weighting, the target proportions were not reached despite a big number of iterations.* Rim weighting is not always completely successful if in the frequency table formed by the input grouping variables there are relatively many empty (with frequency 0) cells or if they are peculiarly located. Additionally, the risk of unsatisfactory result increases when restriction (subcommand RESTR) is put on. If increasing of ITER doesn’t help one has to be content with what came out. By changing the order of variables in VARS list it is possible to select, on which variable to place greater and lesser non-achievement.

*There came out negative weights in some respondents.* It can happen with using a restraining variable or NEGRES. If some cases or whole cells are forbidden to put off their weight, others have to do it for themselves and instead of them. When this task is unrealistic, negative weights appear. With RESTR=variable try, to avoid that, not to allow a situation where there dominate low restraining values over a small amount of high (close to 1) restraining values.

*There came out system-missing weights in some respondents taken into the weighting procedure*. Were they really taken? – check your specification of PROPS subcommand. If they are taken there cannot be missings in the weight variable at output, except in a situation when the dataset is filtered (by commands FILTER or USE).

*In the print-out tables, the sample size is larger than it should be*. Check PROPS (in particular, does the sum of target proportions really equal 1). If correct – then check if negative weights have appeared (see above).

***Appendix***

Mathematical meaning of Weighting Efficiency proportion aka the coefficient of balancedness for nonnegative data. The following equality holds:

where is the squared euclidean distance (difference) between all case weights taken in pairs. It comes clear that *WE* is simply the distance *D*2 transformed into a similarity coefficient ranging between 0 and 1 by the inversion method. Transforming of some dissimilarity *d* into similarity *s* by inverting it has the formula , or, in more general aspect, , where const is some positive constant. In our case, we see, .

Generally, if const is big comparatively *d*, then *s* approaches 1, and if const is small comparatively *d*, then *s* approaches 0. The meaning of *WE* comes clear from it: if *D*2, the quadratic disparity among weights' values, is big comparatively the values’ overall level (namely, the square of their sum), the balancedness coefficient *WE* will be low, and if *D*2 is small comparatively the values’ overall level, the balancedness coefficient *WE* will be high.

In regard to the more general formula of *WE* with the unequal initial weights *u*, it doesn’t change anything principally. We have the same conversion of a distance into a similarity, , only here both participants – the distance and the constant, are computed by weighted formulae: , and . In the case with *u*, we have this bringing that an elementary quadratic difference in a pair of values,, is valued stronger in those pairs of values where the product is big, i.e., where both initial weights, and , are big. That is, “weight coefficients” *u*, strengthening or weakening individual *w*-differences comparatively each other, have been introduced.

So, the coefficient of balancedness, effective for an array of nonnegative data – is a measure of nonequality of their values against the background of (or “relative’) their magnitude (lift above 0). The coefficient is most close to 1 when the values differ little and they are all high. And conversely, it is most close to 0 when the values differ strong and the low ones of them are close to 0. The coefficient gets lower, retreats from 1, as dissimilarity of values grows, i.e., as the profile gets sharper. It is the overriding effect for it. If all values are identical, the coefficient equals 1 whatever their magnitude (but >0).

The coefficient of balancedness is completely correlated, negatively, with the well-known coefficient of variation (st. deviation divided by the mean). As *n* grows, this curvilinear functional relation (which shape depends on the distribution generating the values) straightens into the linear.

1. Quantity *WE∙n* is known as Effective Base, it is used for cautious, conservative estimation of interval measures (confidence interval, etc.) with the help of the so called Effective base weighting. This option is present in SPSS Custom Tables from 24th version. [↑](#footnote-ref-1)
2. One is to be aware that strong warp of frequencies between cells can change sign of residuals in some cells. Our option RESTR=NEGRES supposes that it did not occur and that cells having been residual-positive in “normal” data remained so in the being observed data. So the option expects that the warp between the cells was *not strong*. (However, rim weighting and any frequency reweighting of sample in general is methodologically allowable only if disproportions are “not big”.) If using NEGRES option changed sign of cell residuals in your data its usage in this particular case should be considered dubious, not having justifed the conceptual assumption. [↑](#footnote-ref-2)
3. The same result could be obtained in SPSS general loglinear analysis (GENLOG):

   genlog v1 v2 /cstruct= freq /design= v1 v2 /save= pred(weight\_$). Iterations can be controlled by subcommand CRITERIA. The dataset must be weighted by a variable which contains *expected* target cell frequencies – these are the target proportions for groups multiplied between them, and multiplied by the sample size. [↑](#footnote-ref-3)