***Frequency weighting***

SPSS macros by Kirill Orlov

kior@akado.ru, ttnphns@gmail.com

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

All rights reserved

*Частотное взвешивание.* Достижение нужных долевых размеров групп респондентов одномерным или многомерным (rim) взвешиванием. Можно выбрать общее N, наложить ограничение на взвешивание отдельных ячеек или наблюдений, взвесить несколько подвыборок параллельно, учесть начальные веса.

*Прочтите «*[*О SPSS макросах*](https://www.spsstools.net/ru/KO-aboutmacros)*» что они такое и как их запускать.*

# МАКРОС !KO\_WEIGR: ВЗВЕШИВАНИЕ ГРУПП

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

!KO\_weigr vars= *v1 v2* /\*Одна или несколько группирующих переменных, в которых надо изменить

/\*соотношение численностей групп, поименно

/wname= /\*Имя для весовой переменной (по умолчанию weight\_$)

/restr= *6* /\*Опционально: ограничение изменения веса - не вводить (NONE, тж п/у);

/\*стараться не допускать частоты в ячеке ниже порога (число);

/\*запрет на ячеи отрицательных остатков (NEGRES); сдерживание переменной

/\*(имя\_переменной)

/wvar= /\*Опционально: переменная с начальными весами

/splvar= /\*Опционально: делать взвешивание раздельно для подвыборок,

/\*задаваемых этой переменной (массив выйдет сортированным по ней)

/total= /\*Нужная сумма весов выборки (симулированное N): число;

/\*SELECTED (тж п/у, начальная сумма весов вошедших во взвеш-е наблюдений);

/\*или имя\_переменной из VARS (начальная сумма весов отобранных перед взвешиванием

/\*в этой переменной)

/report= /\*Отчитываться о взвешивании: YES (тж п/у) или NO

/hist= /\*Опционально, при REPORT=YES: гистограмма полученных весов (YES или NO, тж п/у) /mode= PAIR /\*Стиль задания PROPS: PAIR (тж п/у) или LIST

/method= /\*Если VARS несколько: взвешивание ободочное (RIM, тж п/у) или декартово (CART)

/iter= *5* /\*Число итераций; действует только при ободочном взвешивании

/props= *'1 .6' '2 .4' / '1 .35' '2 .15' '3 .50'* /\*Для каждой переменной VARS: либо ALLEQ/ALLVEQ

/\*(все группы взвесить поровну), либо список групп (кодов) и требуемых для них размеров;

/\*размеры либо долями, либо частотами – везде);

/\*можно ключевые слова MISSING, SYSMIS, ELSE/ELSEV (вместо кода),

/\*HOLD, ADJUST (вместо доли).

Минимум надо задать VARS, PROPS. PROPS должна идти последней.

Макрос исполняет частотное взвешивание групп. Изменением весов наблюдений в рабочем массиве макрос приводит указанные пользователем группы к заданным численностям (долям от общего размера выборки N). Размер выборки можно использовать старый или указать любой другой. Группирующих переменных может быть одна или несколько. Группы, участвующие в взвешивании, необходимо указать; неуказанные группы будут выключены из выборки (их вес будет *sysmis*). Опционально вы можете сдерживать изменение веса тех или иных наблюдений или целых подгрупп. Также можно проделать взвешивание раздельно по подвыборкам и учесть фоновые, начальные веса наблюдений.

Результат работы макроса – весовая переменная *WEIGHT\_$* (вы можете задать и другое имя), взвешивающая массив. В этой переменной наблюдения, не прошедшие процедуру взвешивания, получают sysmis-веса; остальные наблюдения получат в весовой переменной валидные значения и составят взвешенную выборку. Какие наблюдения войдут и какие не войдут во взвешивание – вы определите заданием подкоманды PROPS (а также фильтрацией, если массив данных в состоянии фильтрованности).

В окне результатов – таблица с получившимися после взвешивания краевыми частотами. Макрос вычисляет также *Weighting Efficiency* proportion. Эта величина *WE* есть:

где *ui* – исходный вес наблюдения *i*; *wi* – получившийся вес этого наблюдения; всего весов, как и наблюдений в выборке, *n*. Если исходных весов не было, т.е. все *ui* = 1, то формула упрощается до:

*Weighting Efficiency* proportion – это коэффициент сбалансированности весов (и вообще неотрицательных значений). Когда получившиеся в итоге взвешивания веса *w* различаются между наблюдениями массива мало, коэффициент высок, он близится к 1. Если же веса *w* различаются между наблюдениями сильно – относительно их общего уровня, – коэффициент смещен в сторону 0. Если взвешивание имело цель имитировать в выборке структуру популяции, *WE* ниже 0.7-0.8 может считаться признаком того, что выборка до проделанного взвешивания недостаточно хорошо гармонировала с популяцией; или другими словами, что взвешивание ради упомянутой имитации оказалось искажением такой силы, какую не следует игнорировать: это «рискованное» взвешиваине. Индекс *WE* не нуждается в том, чтобы сумма *w* равнялась *n* или равнялась сумме *u*; он характеризует только значения *w*, только их профиль. Подробнее о коэффициенте *WE*[[1]](#footnote-1) – см. «Приложение».

В ходе работы макрос создает временные переменные с именами, в которых есть 5 подряд идущих символов ‘$’, например, v$$$$$2. Тождественных имен должно не быть среди переменных вашего массива.

Черты этого макроса:

Одномерное «ячейковое» взвешивание (по одной переменной или по комбинации нескольких) или многомерное «ободочное» (rim aka raking) взвешивание. См. п/к METHOD.

Вы можете выбрать манеру задания целевых размеров – парно (‘код доля’ ‘код доля’) или списочно (‘код код’ доля доля).

Вы можете предпочесть задать целевые частоты вместо целевых долей, хотя это менее универсальный способ.

Вы можете использовать скорописные ключевые слова ALLEQ/ALLVEQ для задания равных долей всем группам.

Вы можете использовать ключевые слова SYSMIS, MISSING, ELSE/ELSEV на месте кода группы и ключевые слова HOLD и ADJUST на месте целевой доли.

Группы, которые вы умолчите, не зададите им целевые размеры – исключаются из процедуры взвешивания. Их веса будут sysmis. Также sysmis веса получат выфильтрованные наблюдения, если входящий массив в статусе фильтрованности. Те и другие наблюдения – не участвуют во взвешивании.

Если вы хотите, чтобы неучаствующие во взвешивании наблюдения сохранили свое исходное значение в весовой переменной, задайте п/к WVAR и WNAME как одну и ту же переменную.

Вы можете задать целевой полный объем выборки несколькими способами (п/к TOTAL).

Вы можете указать переменную с базовыми, исходными весами.

Исходные веса 0 остаются в итоге таковыми; целевые размеры 0 – допускаются. Наблюдения с весом 0 на выходе из макроса включены во взвешенную выборку, в отличие от наблюдений с sysmis-весом.

Вы можете проделать взвешивание в подвыборках параллельно, указав макросу переменную для расщепления массива.

Вы можете осуществить разного рода дифференцированное сдерживание изменения весов.

ПРИМЕР 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'.

В первой команде численности двух составляющих переменную *SEX* групп, мужчины (код 1) и женщины (код 2) уравниваются: им заданы доли по 0.5 (50%).

Вторая команда делает двумерное взвешивание: по полу численности уравниваются, а по возрасту (три группы) требуемое соотношение 50%–30%–20%.

Третья команда эквивалентна второй, если в переменной *SEX* нет пропущенных значений, т.е. существуют только две валидные категории, 1 и 2. Если же в ней есть пропущенные данные, использование ALLVEQ вместо ALLEQ сделает третью команду эквивалентной второй.

Четвертая команда это одномерное взвешивание по комбинации категорий двух переменных, т.е. по переменной с 2×3=6 группами и целевыми долями, определенными заданными краевыми долями.

Следующая развилка ориентирует, какой метод частотного взвешивания когда использовать.

*Есть одна группирующая переменная*. Одномерное взвешивание. **Укажите ее** в VARS и задайте целевые (т.е. нужные, которые требуется достичь, - обычно это те, что в генеральной совокупности) доли взвешиваемых групп в PROPS.

*Есть несколько группирующих переменных.*

*Целевые доли для ячей перекрестной классификации (кросс-таблицы), образуемой ими, известны.* Это тоже одномерное взвешивание. **Создайте переменную** с перекрестными группами и действуйте, как выше. Например, если есть переменная ПОЛ (2 группы) и ВОЗРАСТ (3 группы), создайте группирующую переменную с 2×3=6 группами.

*Целевые доли для ячей перекрестной классификации, образуемой ими, не даны, но являются для вас функцией целевых краевых долей.* (Имеется в виду, что первые можно получить перемножением вторых. Это подразумевает отсутствие корреляции между группирующими переменными в популяции.) Это тоже одномерное взвешивание, как выше. Но чтобы не создавать физически переменную с перекрестными группами, вы можете **воспользоваться опцией** METHOD=CART (декартово взвешивание). Тогда надо **указать все** ваши группирующие переменные в VARS и целевые краевые доли в PROPS, а макрос сам посчитает целевые ячейковые доли за вас.

*Целевые доли для ячей перекрестной классификации, образуемой ими, неизвестны и не считаются функцией краевых долей, либо они просто неважны вам. Известны (или имеют значение) только целевые краевые доли этой таблицы.* Это есть многомерное, ободочное взвешивание (rim weighting = raking weighting = iterative proportional fitting): METHOD=RIM. Вы должны **указать все** ваши группирующие переменные в VARS и целевые краевые доли в PROPS. Этот метод используется по умолчанию п/к METHOD, когда переменных VARS более одной.

См. ПРИМЕРЫ 5 и 6 для уяснения различия между декартовым взвешиванием и ободочным взвешиванием.

Следующая развилка касается только многомерного, ободочного взвешивания. Она – о том, использовать ли опцию сдерживания изменения весов (п/к RESTR).

*Вы полагаете (или готовы согласиться), что диспропорционирующие силы действовали в каждой из переменных независимо*. «Диспропорционирующие силы» это нечто, что привело к перекосу наблюдаемых в выборке частот по сравнению с целевыми, «подлинными», популяционными. Действовали независимо – значит: перекос частот в переменной ПОЛ (берем здешний пример) произошел безотносительно к групповой принадлежности в остальных указанных в VARS переменных, в данном случае – в переменной ВОЗРАСТ. Другими словами, перепредставленность в выборке, скажем, женщин, произошла, как полагает исследователь, с одинаковой «силой» во всех возрастных категориях. И аналогично со стороны ВОЗРАСТА: перекос частот между возрастами произошел безотносительно к групповой принадлежности в переменной ПОЛ. При описанных, независимых со стороны группирующих переменных, перекосах перекос в какой-л. одной из них *сопровождается* перекосом в другой лишь в меру и силу существующей природной сцепленности между этими признаками (т.е. неравенства популяционного хи-квадрат нулю). Тогда делайте rim-взвешивание обычное, без сдерживания. Умолчите RESTR или укажите RESTR=NONE.

*Вы полагаете, что диспропорционирующая сила (или силы) действовала внутри перекрестной классификации, и определенным образом.* Т.е. перекашивались ячейковые частоты, а перекос краевых стал просто внешним выражением и следствием этих внутритабличных перекосов.

*Диспропорционирующая сила поразила фактор, ответственный за сцепленность признаков (т.е. группирующих переменных).* «Ответственный за сцепленность» фактор надо понимать как в факторном анализе – некое латентное свойство, общее для группирующих переменных, из-за которого они в популяции имеют ненулевой хи-квадрат. Если группирующие переменные или их часть сцеплены (в популяции) фактором(ами), то бережливо будет помыслить, что частотные перекосы в выборке случились именно «вдоль» фактора, т.е. произошли между ячейками, наполненными его милостью (ячейки с положительным хи-квадратовым остатком), нежели между ячейками, наполненными возмущениями против него (ячейки с отрицательным остатком). Раз так, то при восстановлении популяционных, целевых частот нужен перевзвес только среди ячеек с неотрицательными residuals[[2]](#footnote-2). **Задайте** RESTR=NEGRES.

*Диспропорционирующая сила выбирала ячейки иначе, и у вас есть гипотеза или сведения о том, между какими ячейками случился перекос.* «Гипотеза или сведения» должно выражаться введением сдерживающей перевзвешивание переменной: **задайте** RESTR=*переменная*. Значения (от 0 до 1) сдерживающей переменной у респондентов, принадлежащих одной перекрестно-классификационной ячейке, могут быть одинаковы или различаться. Подробнее – см. п/к RESTR.

Заметим, что сдерживающую взвешивание переменную вы можете применить вообще всегда (а не только при rim-взвешивании), если ее значения как-то варьируют. Постольку, поскольку эта переменная в общем выражает собой то, в какой степени разрешено изменять частотный вес у респондента.

### Подкоманды

# VARS

Одна или несколько группирующих числовых переменных, поименный список. Если переменная указана одна, будет одномерное взвешивание. Если переменных несколько, по умолчанию будет многомерное, rim-взвешивание; но вы можете заказать одномерное декартово взвешивание – см. п/к METHOD.

Каждое уникальное валидное или пользовательски-пропущенное (user-missing) значение переменной задает свою группу в ней; системное пропущенное (system-missing) значение также составляет группу в ней. Таким образом, всякое наблюдение массива принадлежит какой-л. группе. Те из групп данной переменной, которые вы собираетесь ввести в взвешивание, вы укажете в подкоманде PROPS. Переменные VARS должны не иметь следующих отрицательных значений: *-999* и *-9999*.

**WNAME**

В этой необязательной подкоманде вы можете указать имя для создаваемой весовой переменной. По умолчанию/незаданию имя переменной будет *WEIGHT\_$*. Когда создаваемая переменная уже существует в массиве, она заменяется (обновляется).

# PROPS

Эта подкоманда должна идти последней. В ней перечислите, для каждой переменной VARS, коды групп вместе с целевыми (желаемыми) для них размерами, выраженными в виде долей (можно и в виде абсолютных частот ­– см. об этом ниже).

*Скорописные слова* *ALLEQ/ALLVEQ.*

Если нужно по данной переменной взвесить все группы поровну, вы можете – вместо перечня кодов и размеров – использовать кл. слово ALLEQ или ALLVEQ (слово можно окавычить). Все группы данной переменной получат равные целевые размеры. При этом ALLEQ каждое уникальное значение переменной (даже если оно user-missing), считает группой, и sysmis-значение также считает группой: все группы входят во взвешивание. ALLVEQ только валидные значения в переменной считает группами и берет во взвешивание, а все пропуски получат в весовой переменной sysmis-веса. (Техническое замечание: если за ключевым словом ALLEQ или ALLVEQ стоят квадратные скобки [ ], то они игнорируются.)

*Задание групп и размеров.*

Существует два эквивалентных способа написать подкоманду PROPS – «парный» и «списочный». Вы можете по своему удобству пользоваться тем или другим, одним для всех переменных VARS. Сейчас будет описан «парный» способ. Отличие «списочного» способа от него дано в п/к MODE.

Когда вы задаете список, каждая пара код–размер должна быть в апострофах или кавычках, а задания для разных переменных должны отделяться слешем (/) и идти в том порядке, в каком переменные написаны в VARS. Например: **‘1 *0.6*’ ‘2 *0.4*’ / ‘1 *0.15*’ ‘2 *0.25*’ ‘3 *0.6*’**. Коды групп не обязаны быть положительными целыми. Код группы не обязан быть валидным значением. Задание нулевого размера для группы допускается, наблюдения этой группы получат 0 в весовой переменной.

Следующие ключевые слова можно использовать, один раз для переменной, на месте кода группы:

SYSMIS - означает «группа, образуемая системным пропущенным (system-missing) значением».

MISSING - означает «все пропущенные (user-missing, system-missing) значения как единая группа». Это слово *неупотребимо* для одной и той же переменной наряду со словом SYSMIS. MISSING надо с осторожностью применять в сочетании с кодами, которые являются user-missing, т.к. MISSING означает *все еще неупомянутые* пропущенные значения. Например, если код 2 имеет user-missing статус, то **‘1 *0.15*’ ‘MISSING *0.6*’ ‘2  *0.25*’** приведет к неожиданному результату: поскольку код 2 уже подразумевается MISSING, идущей раньше, то группы с кодом 2 как самостоятельной группы (отдельной от прочих пропущенных значений) уже не существует после этого для макроса. А вот **‘1 *0.15*’ ‘2  *0.25*’ ‘MISSING *0.6*’** вполне допустимо: MISSING будет подразумевать под собой совокупность пропущенных значений кроме значения 2.

ELSE - означает «каждая из неупомянутых групп». Может стоять только в *последней* паре 'код размер' у переменной.

ELSEV - означает «каждая из неупомянутых валидных групп». Может стоять только в *последней* паре 'код размер' у переменной. В отличие от ELSE, ELSEV считает все неупомянутые пропущенные значения в переменной единой группой, и она не допускается во взвешивание. Использованное после MISSING, ELSEV эквивалентно ELSE, поскольку MISSING уже подобрало собой все пропуски.

Группы, которые вы не укажете, не будут взяты макросом в процедуру взвешивания и весовая переменная /выйдет пуста – system missing – для этих наблюдений. Окончательно *совокупность взятых в процедуру взвешивания* наблюдений – взвешенная выборка – определяется списочным (listwise) исключением: если в переменной группа не взята, то эти наблюдения исключаются и из остальных переменных VARS.

Следующие ключевые слова можно использовать на месте целевых долей (но не абсолютных частот):

HOLD - означает «сохранить за группой ее наблюдаемую долю»: именно ту долю, которая была у нее *среди взятых* во взвешивание групп данной переменной *до* момента списочного исключения наблюдений. HOLD это то же что написать упомянутую наблюдаемую долю числом. HOLD надо применять с оглядкой: **‘1  *0.15*’ ‘2 *0.25*’ ‘ELSE *HOLD*’** означает удержать за всеми прочими группами, кроме 1 и 2, их наличные доли, однако вы должны быть уверены, что в сумме эти последние составляют 0.6, чтобы сумма всех целевых долей сложила 1 (см. *Сумма долей*, ниже).

ADJUST - означает «сделать такую долю, чтобы сумма долей всех взятых в данной переменной групп составила 1». Все группы, которые ADJUST в данной переменной, изменят свою наблюдаемую (*до* момента списочного исключения наблюдений) долю пропорционально - в одно число раз. Например, **‘1  *0.15*’ ‘2 *0.25*’ ‘ELSE *ADJUST*’** назначает всем прочим кроме 1 и 2 группам целевые доли, которые пропорционально отличаются от наблюдаемых их долей, а в сумме дают 0.6.

*Сумма долей.* *Задание частотами*. В норме сумма целевых долей по каждой переменной должна равняться 1 (поэтому указывайте доли с достаточным числом десятичных цифр). Если она не равна 1, итоговая общая сумма весов будет отличаться от заданной в подкоманде TOTAL во столько раз, во сколько сумма целевых долей отличалась от 1. Зная это правило, целевые размеры в принципе можно указывать и не в долях, а прямо в сырых *частотах* (см. ПРИМЕР 4, последний абзац). Но это менее гибкий способ задания численностей, нежели долями; и потребуется п/к TOTAL в форме TOTAL=*число*. Вы можете задать все целевые размеры или долями, или частотами – не смешанно. Важно: не употребляете ключевые слова ALLEQ/ALLVEQ, HOLD, ADJUST, если вы задаете целевые размеры сырыми частотами.

Внимание. Будьте аккуратны при составлении подкоманды PROPS. Следите, чтобы число кодов и отвечающих им размеров было одинаково, и чтобы не было лишних или упущенных кавычек/апострофов.

**MODE**

По умолчанию и при MODE=PAIR макрос ожидает, что коды и размеры в PROPS написаны для всякой переменной VARS «парным» способом: ‘код размер’ ‘код размер’… MODE=LIST разрешает (требует) написать PROPS для всякой переменной в списочной манере: ‘код код…’ размер размер… В списочном способе сначала пишутся коды, причем их список – в кавычках или апострофах, а затем идут соответствующие им размеры. Разумеется, список кодов и размеров должен быть одной длины. Все ключевые слова и правила, приведенные выше в п/к PROPS, действительны и при MODE=LIST. Вот некоторые примеры:

|  |  |
| --- | --- |
| Парный способ: | Эквивалентная запись списочным способом: |
| /‘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 |

Если окавыченный список кодов длинный, можете просто продолжать его с новой строки. Либо использовать принятый в SPSS знак конкатенации + (см. ПРИМЕР 3).

ПРИМЕР 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.

Эти две команды эквивалентны.

ПРИМЕР 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.

Эти три команды эквивалентны. Во 2-й команде список кодов просто перенесен на новую строку, что разрешено. В 3-й команде перенос делается с конкатенацией текстов знаком +.

# TOTAL

Здесь закажите нужную сумму весов выходящей весовой переменной, т.е. симулированный ею объем выборки. Возможно три способа:

SELECTED - (тж. по умолчанию или незаданию подкоманды) это сумма весов, равная числу (или исходной сумме весов) наблюдений, взятых в процедуру взвешивания. Что такое наблюдения, взятые в процедуру взвешивания, см. п/к PROPS.

*Число* - укажите нужную сумму весов числом.

#### Имя переменной (из VARS) - сумма весов, равная числу (или исходной сумме весов) наблюдений всех групп, взятых в этой переменной в процедуру взвешивания; это число (сумма весов) наблюдений, отмечавшееся в переменной до списочного исключения наблюдений (см. п/к PROPS).

Если группирующая переменная единственна или во всех группирующих переменных во взвешивание берутся все существующие в них группы, то TOTAL=SELECTED и TOTAL= *переменная* дадут одну и ту же сумму весов.

ПРИМЕР 4. Разные TOTAL. Пусть надо взвесить по двум переменным *VAR1* и *VAR2*. Образуемая ими полная (с учетом всех пропущенных значений) частотная таблица:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | VAR2 | | | | Всего |
| VAR1 |  | Группа A (код=1) | Группа B (код=2) | Группа C (код=3) | Группа D (sysmis) |  |
| Группа 1 (код=1) | 10 | 12 | 18 | 7 | 47 |
| Группа 2 (код=2) | 14 | 7 | 8 | 2 | 31 |
| Всего |  | 24 | 19 | 26 | 9 | 78 наблюдений в массиве |

!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'.

* Во всех трех командах отбираемые во взвешивание группы и их целевые доли одни и те же. В массиве все 78 респондентов распадаются по переменной *VAR1* на 2 группы, и обе они взяты во взвешивание, т.к. упомянуты в PROPS. Эти группы назначено уравнять по численности (целевые доли по 0.5). По переменной *VAR2* 78 респондентов разбиваются на 4 группы: три закодированных плюс system-missing пропуск. Согласно PROPS, эта последняя группа не взята во взвешивание, т.к. она не упомянута ни одним из ключевых слов SYSMIS, MISSING или ELSE. Только три закодированных группы взяты в *VAR2* (их суммарная численность 24+19+26=69). Для группы A целевая доля – 0.333; для группы B – наблюдаемая сейчас, т.е. 19/69 (не 78) =0.275; для группы C – оставшееся, т.е. 1-(0.333+0.275)=0.392.
* Итак, от *VAR1* взято во взвешивание 78 респондентов, но от *VAR2* – 69. «Списочное исключение» – исключение из обеих переменных всех наблюдений, невзятых хотя бы в одной из них – допускает во взвешивание 69 респондентов: всех тех, что взяты в *VAR2*. В первом из трех пусков макроса TOTAL=SELECTED (то же что умолчание), т.е. на выходе сумма весов весовой переменной, N выборки, будет 69 – сколько окончательно респондентов вошло во взвешивание.
* Во втором пуске TOTAL=*VAR1*, поэтому N выборки на выходе будет 78. Столько было отобрано в взвешивание респондентов согласно *VAR1*, до списочного исключения. В данном случае это столько наблюдений, сколько всего их есть в массиве.
* В третьем пуске макроса пользователь задал TOTAL произвольным числом 780, таким и будет симулированное им N выборки.

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

* Этот случай кажется дающим результат как в первом пуске выше, потому что группа D (sysmis), хоть и добавлена теперь во взвешивание, имеет целевую долю 0. В действительности результат будет другим. Т.к. группа D взята, от *VAR2* во взвешивание вошли 78 (а не 69, как раньше) респондентов. Поэтому целевая доля “hold” для группы B уже иная: 19/78=0.244. Соответственно, другой стала и целевая “adjust” доля для группы C: 1-(0.333+0.244+0)=0.423. Окончательно во взвешивание взято 78 респондентов, поэтому TOTAL=SELECTED даст на выходе N выборки 78.

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

* В этом примере целевые численности групп заданы не долями, а частотами. (В этом случае TOTAL=число обязательно.) Сумма целевых частот тут = 100. Поскольку TOTAL задано как 1, N выборки на выходе будет 1\*(100/1)=100. Вы не можете употреблять кл. слова ALLEQ/ALLVEQ, ADJUST и HOLD, если задаете целевые размеры не долями. Вы не можете задавать для одних переменных частоты, для других доли: для всех переменных надо только либо так, либо этак. Если вы задаете размеры сырыми частотами и суммы частот оказались неодинаковы для разных переменных, наибольшая сумма станет итоговой N взвешенной выборки.

**METHOD**

Эта подкоманда действует, если переменных VARS более одной. Когда переменная единственная, макрос делает одномерное (ячейковое) взвешивание по этой переменной. Когда переменных несколько, подкоманда METHOD дает возможность выбрать между многомерным ободочным (rim) взвешиванием и одномерным (ячейковым) взвешиванием по комбинации переменных:

RIM - (тж. по умолчанию/незаданию подкоманды) ободочное (rim aka raking) взвешивание. Это многомерное взвешивание в том смысле, что взвешивание делается последовательно и независимо по каждой переменной. Процесс итеративен и вам понадобится п/к ITER.

CART - декартово взвешивание, это взвешивание по комбинации переменных. Это одномерное взвешивание в том смысле, что оно делается так, словно имеет место взвешивание по единственной переменной, которая является полной комбинацией (декартовым произведением) групп между несколькими переменными. Эта опция существует в макросе, чтобы вам не тратить усилий по ручному созданию упомянутой комбинаторной переменной.

При METHOD=CART целевые доли находятся в элементарных ячейках перекрестной таблицы, задаваемой переменными VARS, и они вычисляются макросом простым перемножением заданных вами целевых краевых долей PROPS. Это означает, что между переменными VARS мы полагаем иметь нулевую связь (т.е. отсутствие взаимодействия) в популяции. Это также означает, что мы полагаем в популяции число групп равным величине декартового произведения между множествами групп (категорий) переменных VARS. Если вы готовы на эти два допущения, вы можете использовать METHOD=CART вместо того, чтобы вручную создавать комбинаторную переменную, вычислять ячейковые целевые доли и взвешивать по ней.

При METHOD=RIM целевые доли – это сами краевые доли PROPS, которые вы задали. Подгонка ведется непосредственно к ним – без постулирования целевых долей в ячейках перекрестной таблицы. Концептуальное допущение в этом методе то, что мы согласны рассматривать «искажение» частот в данных до взвешивания – искажение, которое мы хотим «отыграть назад» взвешиванием – как случившееся независимо по VARS-переменным. Этот метод часто применяется, когда мы не знаем целевых кросс-табуляционных ячейковых долей, но не желаем налагать нулевую связь (декартовым взвешиванием), либо мы не заботимся строго о ячейковых долях.

ПРИМЕР 5. Одномерное взвешивание по пересечению двух переменных. *X* есть 5-группная переменная с кодами 1 2 3 4 5. *Y* есть 3-группная переменная с кодами 1 2 3. Положим, что нам не дали 15 целевых долей, но дали целевые краевые доли, пояснив, что в нашем случае связь между *X* и *Y* должна стать нулевая. Это есть случай для применения 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.

* Теперь покажем, что же в сущности сделал этот пуск. Создадим переменную *XY* с 5×3=15 категориями, вычислим «вручную» 15 целевых ячейковых долей для нее, и взвесим по этой переменной:

compute xy= x\*10+y. /\*X и Y это целочисленные коды между 1 и 9, проще всего создать

execute. /\*декартово произведение так

matrix. /\*Посчитаем ячейковые целевые доли

compute x\_props= {.10, .25, .15, .30, .20}. /\*Целевые доли по X

compute y\_props= {.50, .35, .15}. /\*Целевые доли по Y

compute xy\_props= t(x\_props)\*y\_props. /\*Перемножение

print xy\_props /format= f8.4. /\*дающее 5x3=15 целевых долей для переменной XY

print msum(xy\_props). /\*Проверим, сумма 1

end matrix. /\*Полученные доли вставлены в следующий вызов макроса:

!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.

* Весовые переменные *WEI\_1* и *WEI\_2* идентичны. Мы получили выборку с долями внутри кросстабуляции *X* на *Y*, равными тем 15-ти долям, являющимся функцией краевых целевых долей .10, .25, .15, .30, .20 для X и .50, .35, .15 для *Y*. Результаты идентичны, т.к. во втором случае мы задали в PROPS все 15 целевых долей – именно это подразумевало декартово взвешивание.
* Взвешивание достигнет заказанных целевых долей, если в данных представлены все 5×3=15 подгрупп.
* Взвешивание одномерное, оно неитеративное.
* Поскольку в ситуации нулевой ассоциации между целевыми краевыми долями профиль у *X* один и тот же на всяком уровне *Y* (как равным образом и наоборот), то мы могли бы получить тот же результат еще одним способом: двухшаговым одномерным взвешиванием в условиях п/к SPLVAR – см. ПРИМЕР 11.

ПРИМЕР 6. Те же переменные *X*, *Y* и те же целевые краевые, что выше, но взвешивание будет 2-мерное ободочное.

!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.

* Результат (веса наблюдений), полученный в этом взвешивании, отличается от такового полученного в ПРИМЕРЕ 5. Мы добились тех же целевых краевых долей, как там, но ячейковые доли в двумерной кросс-таблице не равны тем 15-ти, что были целевыми в одномерном взвешивании.
* Ободочное взвешивание нацелено на корректировку только краевых распределений, поскольку идея позади этого метода есть допущение, что диспропорции среди ячейковых частот внутри перекрестной таблицы были результатом наложения независимых краевых диспропорционирующих процессов. Rim-взвешивание “проигрывает” это обратно, как бы разматывая. Искажающий процесс – вот что характеризуется словом «независимый» со стороны *X* и стороны *Y* (так что и процесс исправления тоже независимый с этих сторон).
* В ПРИМЕРЕ же 5 мы имели ту частную ситуацию, где ряды и столбцы перекрестной таблицы целевых долей демонстрирует *состояние* независимости (поэтому мы смогли применить краевые доли напрямую) – это не подразумевает независимый «процесс».

# ITER

Эта подкоманда действует, только если группирующих переменных несколько и заказано ободочное взвешивание, и в этом случае она требуется. Укажите в ней число итераций для подгонки краевых распределений. Чем больше итераций, тем точнее достигаются целевые доли групп. Большее число итераций требуется, когда группирующих переменных и/или групп в них много, а также при наложении RESTR.

# RESTR

С помощью этой подкоманды (она – придумка автора макроса) можно сдержать изменение веса у отдельных наблюдений массива или целых ячей в перекрестной таблице, образуемой группирующими переменными VARS. Сдерживание одних наблюдений или целых ячей от изменения веса перекладывает с них задачу изменения веса на другие наблюдения/ячеи, тем самым заставляя их изменить вес «за себя и того парня». Макрос не слушается сдерживания в группах, которым задана целевая доля 0. Выберите:

NONE - (тж. по умолчанию/незаданию подкоманды) Не налагается ограничений, все взятые в процедуру взвешивания наблюдения массива могут в полной мере изменять свой вес.

*Число* - (возможно только при многомерном, rim взвешивании) Стараться не допускать падения частоты в непустой элементарной ячейке ниже этого значения. Употребите, если без сдерживания частота в ячейке вышла недопустимо низкой для вас. Укажите положительное число. Разумеется, вы не можете указать это число несоразмерно большим, если ячей много, а выборка невелика.

NEGRES - (возможно только при многомерном, rim взвешивании) Ячейкам с отрицательными наблюдаемыми частотными остатками (frequency residuals) запретить изменять свою частоту – эти респонденты не изменят свой входящий вес. Изменение веса будет только среди респондентов, принадлежащих ячейкам с неотрицательными остатками.

*Имя переменной* - Заданное пользователем сдерживание. Значения переменной могут варьировать между 0 и 1. Сдерживающее значение имеет прямой смысл: это доля исходного веса наблюдения, которой разрешено подвергнуться изменению (насколько же велико будет это изменение и будет ли оно увеличение или уменьшением, зависит от требований со стороны целевых долей и от величины сдерживающих значений у других наблюдений; однако в общем, конечно, тенденция такова что чем ближе к 0 сдерживающее значение, тем меньше будет и изменение веса данного наблюдения).

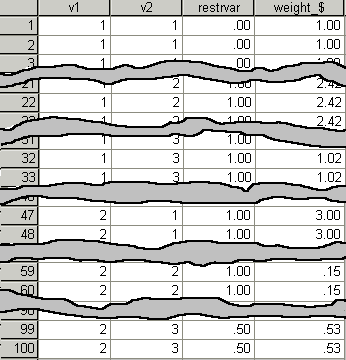
*О сдерживающей переменной:*

* Сумма значений внутри *группы* не может равняться 0.
* При одномерном взвешивании (включая METHOD=CART) равенство или неравенство сумм в разных группах не влияет на результат. Введение сдерживающей переменной в одномерное взвешивание преследует цель сдерживать отдельные наблюдения от изменения их веса.
* При многомерном (rim) взвешивании сумма внутри *ячейки* перекрестной таблицы может равняться 0 (что значит, что ячейка полностью защищена от перевзвеса). Средняя арифметическая значений внутри ячейки будет той долей исходной частоты или суммы весов ячейки, которой позволено измениться. В многомерном взвешивании введение сдерживающей переменной имеет целью сдерживать ячейки и/или отдельные наблюдения от изменения их веса.
* Если значение у наблюдения – пропуск, вес у этого наблюдения будет пропуск: такое наблюдение не берется во взвешивание (так же, как выфильтрованное наблюдение).

RESTR=NONE эквивалентно сдерживающей переменной со всеми значениями =1; RESTR=NEGRES эквивалентно двоичной сдерживающей переменной (1 если респондент принадлежит ячейке с неотрицательным остатком, 0 если с отрицательным остатком).

ПРИМЕР 7. Многомерное взвешивание со сдерживающей переменной.

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



* *V1* состоит в массиве из 2-х групп (коды 1 и 2), *V2* из 3-х (коды 1, 2, 3). По PROPS, все группы в обеих переменных взяты во взвешивание, т.е. какие-то веса будут у всех наблюдений массива.
* В *V2* определенная целевая доля задана для группы 1, а для двух других заказано пропорционально изменить их наблюдаемые доли, в сумме они должны дать 0.45.
* Согласно сдерживающей переменной RESTRVAR, запрещено изменять частоту ячейки (*V1*=1,*V2*=1), т.к. все сдерживающие значения для этих наблюдений = 0. Поэтому в весовой переменной *WEIGHT\_$* вес там не изменился, остался = 1. Частоту ячейки (*V1*=2,*V2*=3) было разрешено изменять не в полной мере (сдерживающие значения = 0.5). Если бы там не было сдерживания (значение было бы 1), вес ее наблюдений уменьшился бы сильнее.

# WVAR

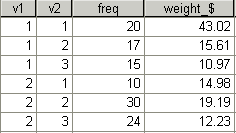
Если массив данных имеет весовую переменную, которую надо учитывать, т.е. использовать как исходные веса, укажите ее здесь. Массив не обязан быть фактически взвешен ею. Любые значения в этой переменной, кроме валидных положительных, трактуются макросом на входе как валидный вес 0, и в новой весовой переменной выйдут как 0 (если наблюдение взято во взвешивание).

Частный случай употребления WVAR – это когда массив представляет собой не данные «респонденты X переменные», а данные, агрегированные в группы. Переменная, содержащая частоты этой таблицы, должна быть тогда указана в WVAR (ПРИМЕР 8).

Если WVAR совпадает именем с переменной, заданной в WNAME, то переменная WVAR, она же WNAME, будет переписана только для наблюдений, которые взяты во взвешивание; а у наблюдений, не участвующих во взвешивании, на выходе останутся те значения весов, которые были у них изначально. Этим можно пользоваться, чтобы изменять одну и ту же весовую переменную постепенно частями (ПРИМЕР 9).

ПРИМЕР 8. Перевзвес частотной таблицы[[3]](#footnote-3).

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



* Входящие данные это агрегированные данные, частотная таблица, вытянутая в два столбца. Частоты – переменная *FREQ*. Макрос пересчитал частоты на заказанные в PROPS краевые распределения. Новые частоты – переменная *WEIGHT\_$*.

ПРИМЕР 9. Изменение весовой переменной только у некоторых наблюдений.

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.

* В обоих пусках макроса заданы обе подкоманды WVAR и WNAME, и весовая переменная в них одна и та же, *WEI*. В этих условиях макрос сохранит необработанным наблюдениям их исходный вес. Первоначально *WEI* посчитана как 1 для всех наблюдений.
* В 1-м пуске макроса участвуют наблюдения *SEX*=1 (фильтр). Они изменят свой вес. Прочие наблюдения сохранят в *WEI* исходное значение (т.е. 1).
* Во 2-м пуске макроса участвуют наблюдения *SEX*=2 (фильтр). Они изменят свой вес. Прочие наблюдения сохранят в *WEI* исходное значение (для наблюдений *SEX*=1 это будут значения, полученные в предшествующем пуске). Если в *V2* есть пропуски, то эти наблюдения не войдут во взвешивание – поскольку PROPS=ALLVEQ, эти наблюдения сохранят в *WEI* свой исходный вес (равный 1, т.к. они пока не меняли его).

# SPLVAR

Вы можете указать категориальную переменную (или даже несколько), задающую непересекающиеся подвыборки, для которых макрос проделает одно и то же взвешивание раздельно, параллельно. Выходящий массив окажется сортирован и расщеплен по значениям этой переменной (сочетаниям значений, если переменных несколько). Если TOTAL=*число*, то это число относится к каждой подвыборке; таким образом, на выходе суммы весов всех подвыборок будут равны, чем можно пользоваться, если стоит задача не только привести подвыборки к одинаковой структуре групп в них, но и сравнять численность подвыборок.

Следите за тем, чтобы группы, которым вы задали целевые размеры в п/к PROPS, присутствовали в данных *все* в каждой отдельной подвыборке. В подвыборках, где отсутствуют некоторые из заявленных в PROPS групп, взвешивание пройдет некорректно.

ПРИМЕР 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.

* В опросе учащихся школ представлено 30 школ (коды school 1 до 30) , и ученики разбиты по возрасту на три уровни, level: 1 (младшие), 2 (средние) и 3 (старшие). Опрос был проведен два раза – в два разных года (коды year 1 и 2). Школы и уровни одни и те же в оба года, но индивиды-респонденты разные. Размер всей выборки – 8401, несбалансированный между годами и между школами.
* Исследователь хочет уравнять вклады всех школ в каждом году, и уравнять N в обоих годах. Далее, он желает сохранить распределение возрастов (уровней), какое оно наблюдается в выборке сейчас: 37%:37%:26%.
* Одно решение (выше) – сделать взвешивание в двух годах параллельно (SPLVAR=*YEAR*), запросив равное итоговое 4200.5 для каждого года. Само взвешивание 2-переменное ободочное: по *SCHOOL* (все категории равны) и по *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.

* Другой и эквивалентный подход в этих обстоятельствах – обработать две переменные, являющие комбинации: *SCHOOL*\**YEAR* (30\*2=60 категорий) and *LEVEL*\**YEAR* (3\*2=6 категорий) – вместо *SCHOOL* и *LEVEL* под SPLVAR год. П/к TOTAL=8401 необязательна, поскольку по умолчанию база из всех 8401 наблюдений, взятых во взвешивание, сохраняется как сумма весов.

ПРИМЕР 11. Показывает синтаксис, эквивалентный по результату тому, что было в ПРИМЕРЕ 5 (см. там).

!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.

* В первой команде мы взвешиваем по *X*, но делаем это параллельно в подвыборках, задаваемых *Y*. После этого, во второй команде мы взвешиваем по *Y* всю выборку, используя веса, полученные в первой команде, в качестве начальных весов.
* С тем же эффектом мы могли бы поменять порядок двухшагового взвешивания на обратный: взвесить сначала по *Y* в подвыборках *X*, и затем взвесить по *X*, используя начальные веса, только что полученные.
* Результаты не будут совпадать с полученными в ПРИМЕРЕ 5, если в данных существуют не все 5×3=15 подгрупп.

**REPORT**

По умолчанию и REPORT=YES выводится сводка результов: таблица/ы с получившимися после взвешивания краевыми частотами (долями групп), а также некоторая статистика весов и Weighting Efficiency proportion. Вы можете запретить вывод, указав REPORT=NO.

**HIST**

Опциональная HIST=YES строит гистограмму полученных весов. HIST не работает, если REPORT=NO.

### Особые режимы

Макрос слушается фильтрованности массива (команда FILTER, USE): выфильтрованные наблюдения получат вес sysmis: они не участвуют во взвешивании. Не используйте расщепление массива – для этого в макросе есть подкоманда SPLVAR. Макрос не слушается взвешенности массива автоматически: для учета исходных весов вы должны указать переменную в WVAR. Макрос снимает расщепленность с массива, если SPLVAR не использована. Макрос игнорирует временные (под TEMPORARY) преобразования.

### Некоторые вопросы

*Нужно ли удалять созданную макросом весовую переменную (weight\_$) перед новым пуском макроса?* Нет. Это не играет роли. Если она осталась, макрос ее пересчитает. Другое дело, если вы собираетесь ее использовать как начальные веса, - тогда укажите ее в WVAR.

*При многомерном взвешивании целевые доли не достигнуты, несмотря на большое число итераций.* Rim-взвешивание не всегда полностью успешно, если в частотной таблице, образуемой входящими группирующими переменными, относительно много пустых (с частотой 0) ячеек или если они специфически расположены. Кроме того, риск неудовлетворительного результата увеличивается при наложении сдерживания (подкоманда RESTR). Если увеличение ITER не помогает, приходится довольствоваться тем, что получилось. Изменением порядка переменных в списке VARS можно выбрать, на какую из них переложить большую и меньшую недостигнутость.

*Получились отрицательные веса у некоторых респондентов.* Это может случиться при употреблении сдерживания переменной или NEGRES. Если одним наблюдениям или целым ячейкам запрещается сбрасывать вес, другие должны сделать это за себя и за них. Когда эта задача нереалистична, получаются отрицательные веса. При RESTR=переменная старайтесь во избежание этого не допускать, чтобы преобладали низкие сдерживающие значения над небольшим числом высоких (близких к 1) сдерживающих значений.

*Получились system-missing веса у некоторых взятых в процедуру взвешивания респондентов*. А взяты ли они? – проверьте свое задание подкоманды PROPS. Если взяты, то пропусков в весовой переменной на выходе не может быть, кроме случая, когда массив фильтрован (командами FILTER или USE).

*В распечатываемых таблицах размер выборки больше должного*. Проверьте PROPS (в частности, равна ли сумма целевых долей 1). Если правильно – посмотрите, не получились ли отрицательные веса (см. выше).

***Приложение***

Математический смысл Weighting Efficiency proportion, или коэффициента сбалансированности для неотрицательных данных. Верно следующее равенство:

где есть квадратное евклидово расстояние (различие) между всеми взятыми попарно весами. Становится понятно, что *WE* есть просто-напросто расстояние *D*2, переведенное в коэффициент сходства, колеблющийся между 0 и 1, методом обращения. Перевод некоторого различия *d* в сходство *s* путем обращения имеет формулу , или, в более общем виде, , где const – положительная константа. В нашем случае, мы видим, .

Вообще, если const велико сравнительно с *d*, то *s* близится к 1, а если const мало сравнительно с *d*, то *s* близится к 0. Отсюда понятен смысл *WE*: если *D*2, квадратичная неодинаковость значений весов, велика по сравнению с валовым уровнем их значений (именно, квадратом их суммы), то коэффициент сбалансированности *WE* будет низкий, а если *D*2 мало по сравнению с валовым уровнем их значений, то коэффициент сбалансированности *WE* будет высокий.

Что касается более общей формулы *WE* с исходными неодинаковыми весами *u*, то она не меняет ничего принципиально. Имеем то же обращение расстояния в сходство, , но здесь оба участника – расстояние и константа, вычислены по взвешенным формулам: , и . В случае с *u* мы имеем то привнесение, что элементарная квадратная разница в паре значений,, ценится больше там, в тех парах значений, где произведение велико, т.е. где оба первоначальных веса, и , велики. Т.е. введены «весовые коэффициенты» *u*, усиливающие или ослабляющие отдельные *w*-разницы сравнительно друг с другом.

Итак, коэффициент сбалансированности, действующий для набора неотрицательных данных – это мера одинаковости их значений на фоне (или «относительно») их величины (приподнятости над 0). Коэффицент наиболее близок к 1, когда значения различаются мало и все они высоки. И наоборот, он наиболее близок к 0, когда значения различаются сильно, и низкие из них близки к 0. Коэффициент снижается, отступает от 1 с увеличением неодинаковости значений, т.е. с обострением профиля. Это первостепенный эффект для него. Если все значения одинаковы, коэффициент равен 1 при любой их величине (но >0).

Коэффициент сбалансированности полностью коррелирует, отрицательно, с известным коэффициентом вариации (ст. отклонение, деленное на среднюю). С ростом *n* эта криволинейная функциональная связь (форма которой зависит от распределения, рождающего значения) спрямляется в линейную.

1. Величина *WE∙n* известна как Effective Base (эффективная база), она используется для осмотрительной, консервативной оценки интервальных мер (дов. интервал и т.п.) при помощи так наз. Effective base weighting. Эта опция присутствует в SPSS Custom Tables с 24-й версии. [↑](#footnote-ref-1)
2. Надо сознавать, что сильный перекос частот между ячейками может менять знак остатков в некоторых ячейках. Наша опция RESTR=NEGRES полагает, что такого не произошло и ячейки, бывшие положительными по остаткам в нормальных данных, таковыми и остались в наблюдаемых данных. То есть опция рассчитывает, что перекос между ячейками был *несильным*. (Впрочем, rim-взвешивание вообще, да и любое частотное перевзвешивание выборки – считается методологически допустимым только при «не больших» диспропорциях.) Если применение опции NEGRES изменило в ваших данных знак остатков в ячейках, ее применение в данном конкретном случае следует посчитать сомнительным, не оправдавшим концептуального допущения. [↑](#footnote-ref-2)
3. Тот же результат можно получить в SPSS общим логлинейном анализом (GENLOG):

   genlog v1 v2 /cstruct= freq /design= v1 v2 /save= pred(weight\_$). Итерации можно регулировать подкомандой CRITERIA. Массив должен быть взвешен переменной, которая содержит *ожидаемые* целевые ячейковые частоты – это есть перемноженные целевые доли групп, умножить на размер выборки. [↑](#footnote-ref-3)