***Multiple Response tools***

SPSS macros by Kirill Orlov

kior@akado.ru, ttnphns@gmail.com

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

All rights reserved

*Инструменты для наборов множественного ответа.* Собрание разнообразных макросов для работы с переменными наборов множественного ответа – категориальных наборов (MRC) и дихотомических наборов (MRD), за исключением перекодировки их друг в друга (это см. в коллекции “Categorical – Binary recodings”).

* Для ремонта категориального набора множественного ответа (придание правильного строения, разупорядочение набивки, устранение повтора ответов, устранение конфликта несовместимых ответов, сброс пустых переменных, слитие наборов) используйте [!KO\_REGMRC](#_МАКРОС_!KO_REGMRC:_ОТЛАДКА).
* Для создания или обновления переменной «нет ответа на вопрос» в дихотомических наборах множественного ответа используйте [!KO\_MRDNA](#_МАКРОС_!KO_MRDNA:_СНАБЖЕНИЕ).
* Если надо почистить данные (добавить, отъять ответы) категориальной переменной или категориального набора множественного ответа данными других категориальных переменных, у которых тот же список вариантов ответов – используйте [!KO\_DONRECI](#_МАКРОС_!KO_DONRECI:_ПОВЕРКА).
* [!KO\_STRMRC](#_МАКРОС_!KO_STRMRC:_РАЗВЕРТКА) и [!KO\_STRMRD](#_МАКРОС_!KO_STRMRD:_РАЗВЕРТКА) создают готовый к анализу набор множественного ответа из текстовой переменной, содержащей набивку ответов. [!KO\_ERRDISP](#_МАКРОС_!KO_ERRDISP:_ПРОВЕРКА) – подсобный макрос.

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

*Ошибка “Protected directory”.* Некоторые из макросов, описанных в текущем документе, пишут временные файлы на жесткий диск. Если вы не обладаете полными правами Администратора вашего компьютера, это может вызвать ошибку, сообщающую среди прочего: *“SPSS Statistics cannot access a file... specifies a protected directory...”* и значающую, что дефолтная директория, какую макрос хочет использовать, защищена на вашем ПК. Чтобы решить эту проблему, в окне синтаксиса скомандуйте: CD 'myfolder'., где 'myfolder' есть путь/имя некоторой папки, куда вам разрешено сохранять файлы.

**Категориальный набор множественного ответа** (Categorical multiple response set, MRC) это несколько переменных, объединенных тем, что они вместе выступают единым хранилищем данных. Данные (значения) это дискретные категории, варианты ответа на вопрос с множественным (неальтернативным) ответом; так что у одного респондента бывает набито несколько разных ответов в строке. Каждая переменная MRC-набора, таким образом, имеет смысл «еще одно отвечание респондента». Ячейки, оставшиеся незаполненными ответами, можно оставить пустыми или заполнять каким-нибудь кодом-наполнителем, которому надо придать во всех переменных набора статус пропущенного значения. Обычно стараются придать MRC-набору *правильное строение*: это такое наполнение кодами ответов, при котором каждая следующая переменная – все менее заполнена ими; другими словами, пустая/невалидная ячейка не может находиться левее валидного кода. Первая переменная, таким образом, первична. Если респондент ответов не дал, то в 1-ю переменную ему ставят код «нет ответа».

**Двоичный, или дихотомический набор множественного ответа** (Dichotomous multiple response set, MRD) это альтернативный по отношению к MRC способ хранить те же данные. В MRD-наборе переменные соответствуют категориям (вариантам ответа или признакам). Переменных столько, сколько их, а значения в переменных двоичны: 1 («да», «есть», «выбрано») и 0 («нет», «не выбрано»). Это напоминает набор фиктивных (dummy) переменных, но в отличие от фиктивных переменных MRD-набор может иметь более одной единицы в строке, поскольку MRD соответствует вопросу на множественный ответ (тогда как dummy-набор соответствует вопросу на единичный (альтернативный) ответ. Аналогично тому как в категориальных переменных обычно предусматривают код для «нет ответа на вопрос», в MRD предусматривают переменную «нет ответа на вопрос», которая равна 1, если все остальные переменные набора равны 0, и равна 0, если это не так.

# МАКРОС !KO\_REGMRC: ОТЛАДКА КАТЕГОРИАЛЬНОГО НАБОРА МНОЖЕСТВЕННОГО ОТВЕТА

Version 3, May 2012 (Version 1, Feb 2000). Tested on SPSS Statistics 22, 26, 28.

!KO\_regmrc vars= *v1 to v8* /\*Переменные, составляющие MRC набор, писать надо ч-з to

/nvars= /\*Опционально: можно указать число переменных (ради скорости)

/check= YES /\*Опционально: отсеивать повторение одного и того же кода - YES или NO (тж п/у);

/\*после YES можно перечислить отдельные коды для такой правки;

/\*вставка слова EXCEPT между YES и кодами - править кроме этих кодов

/sole= LEAVE *10* /\*Опционально: удалить из компании (DELETE) или сохранить в одиночестве (LEAVE)

/\*код(ы), несовместимые с прочими; после кл слова перечислите эти код(ы)

/random= /\*Опционально: рандомизовать порядок кодов - YES или NO (тж п/у)

/crop= YES /\*Опционально: удалить лишние (пустые, без валидных ответов)

/\*переменные - NO (тж п/у) или YES (несовместимо с SELFEXE=NO)

/filler= *0* /\*Опционально: код-наполнитель, принятый в наборе (число)

/selfexe= /\*Макросу исполнять себя самому (YES, тж п/у) или не исполнять (NO).

Минимум надо задать VARS.

Макрос придает категориальному набору множественного ответа (MRC) *правильное* строение, т.е. такое, когда каждая следующая в наборе переменная потенциально менее заполнена кодами ответов, чем предыдущая (см. описание выше). Неправильное строение обычно возникает в результате чисток или в ходе набивки, и оно возможно, если у респондента меньше ответов, чем всего переменных в наборе. Исправление макрос делает перемещением в наборе значений справа налево на свободные места, если таковые окажутся. Свободные места это пропущенные (user- или system-missing) значения. Это предполагает, что если во вводимом MRC используется код-наполнитель, то он имеет там статус пропущенного значения.

Помимо придания правильного строения, макрос обладает вдобавок 4-мя опциями. (1) Вы можете затребовать устранить из данных дублирование кодов – всех или определенных. (2) Вы можете проконтролировать, чтобы код, несовместимый с остальными кодами, был единственным ответом респондента или был удален. (3) Вы можете рандомизовать последовательность набивки кодов у респондента – лишить ее закономерности. (4) Вы можете избавиться от лишних, не содержащих валидных кодов, переменных.

Макрос не затрагивает наблюдения, не имеющие валидных данных во входящих переменных VARS.

Макрос не создает новых переменных, но изменяет сами входящие переменные. Вы можете сделать эти изменения временными, на одну ближайшую процедуру: для этого, в условиях п/к SELFEXE=NO, поставьте пуск макроса под TEMPORARY.

Макрос создает временные переменные с именами, содержащими пять символов *$* подряд, например, *v$$$$$.\_2*. Поэтому желательно, чтобы таких имен не было в вашем массиве. Если вы пускаете макрос в режиме SELFEXE=NO и создаете перед этим скретч-переменные, их именам также лучше не иметь пять символов *$* подряд.

***Подкоманды***

**VARS**

Укажите категориальные числовые переменные, образующие MRC – все или нужные. Это должны быть смежные в массиве данных переменные и указать их надо диапазоном, через “to”: *var* to *var.* Вы можете указать и несколько MRC-наборов, словно это один набор, что позволяет слить наборы (см. ПРИМЕР 5).

**NVARS**

Эта необязательная подкоманда может ускорить процесс отсеяния кодов-повторов – т.е. когда вы задаете CHECK=YES. Вы можете захотеть ускорения, если переменных много. Укажите в NVARS число входящих переменных VARS.

**CHECK**

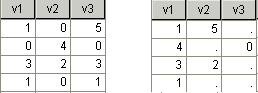
По умолчанию/незаданию и при CHECK=NO макрос не отсеивает повторения одних и тех же валидных кодов у респондента, если они имеют место. CHECK=YES отсеивает повторения, так что на выходе каждый валидный код будет встречаться не более раза. Если надо, чтобы эта проверка и устранение дублирования была осуществлена не для всех валидных кодов, а только определенных, перечислите эти коды после слова YES, например: CHECK=YES 1 3 (только неотрицательные числа могут быть в списке). Если, напротив, надо запретить отсеивание повторов у определенных кодов, перечислите их после вставки слова EXCEPT, например: CHECK=YES EXCEPT 1 3.

Упоминание кода более раза не ведет к ошибке: CHECK=YES 1 3 3 это то же, что CHECK=YES 1 3.

На месте кодов позволено писать *имена* содержащих их *переменных*, это открывает возможность задавать разным респондентам разные коды; коды в этих переменных не обязаны быть неотрицательными.

ПРИМЕР 1.

!KO\_regmrc vars= v1 to v3 /check= YES.



* MRC имел «неправильное» строение: код-заполнитель (здесь = 0) встречался левее валидных значений. Макрос исправил это, переместив значения влево. Освободившиеся места остались пустыми. Чтобы заполнить их, добавьте подкоманду /FILLER=0.
* Заказано было также устранить дублирование значений. Значение теперь встречается у наблюдения не больше раза.

ПРИМЕР 2.

!KO\_regmrc vars= v1 to v9 /check= YES 1 3 4 x.

* Дублирование только кодов 1, 3, 4 и X будет устранено. X это имя числовой переменной, и значения в ней могут быть разными для разных респондентов.

**SOLE**

Нередко бывает, что один или несколько вариантов ответа по смыслу несовместимы со всеми другими вариантами. К примеру, «Затрудняюсь ответить» и «Ничего из списка» несовместимы с другими ответами, а также друг с другом. Любой из этих двух вариантов, если выбран, может быть только единственным вариантом ответа у респондента. SOLE избавляет MRC-набор от нарушений этого правила. Вы можете предпочесть: или удалить код, несовместимый с другими обнаруженными у респондента кодами, или удалить эти другие, оставив респонденту только тот код. Испольуйте соответственно ключевое слово DELETE или LEAVE с перечислением кодов, каждый из которых несовместим с прочими кодами. Список может состоять из любых (в том числе отрицательных) чисел, представляющих собой валидные коды в данных. Имена переменных не могут фигурировать в этом списке.

Например, задание SOLE= DELETE 99 10, при нахождении кода 99 или кода 10 в сочетании с иными валидными кодами, удалит этот код 99 или этот код 10 у респондента. Когда же коды 99 и 10 найдутся в сочетании друг с другом, удален из них будет код 99, как упомянутый в списке *раньше*.

Задание SOLE= LEAVE 10 99, обнаружив код 10 или код 99 в сочетании с иными валидными кодами, удалит эти последние. Когда же коды 10 и 99 найдутся в сочетании друг с другом, удален из них будет код 99, а сохранен код 10, как упомянутый в списке *раньше*. Опция SOLE не заменяет собой опцию CHECK: если, допустим, код 99 набит у респондента 2 раза, и опция SOLE сохранит этот код у него, то она сохранит обе копии.

ПРИМЕР 3.

!KO\_regmrc vars= v2.1 to v2.12 /sole= LEAVE 10 /check= YES 10.

* Код 10, значивший «ничего из списка», логически несовместим с другими валидными ответами. Исследователь заказал удалить любые другие ответы, если они найдутся в сочетании с кодом 10.
* Чтобы гарантировать, что код 10 не повторяется у респондента, исследователь заказал избавиться от возможных повторений именно этого кода. Любые остальные валидные коды, если дублируются, сохранят дублирование.

**RANDOM**

Эта подкоманда (RANDOM=YES) осуществляет рандомизацию[[1]](#footnote-1) порядка, в каком идут коды у респондента слева направо. Она может пригодиться, например, если исследователь собирается использовать в своем анализе меньшее число переменных, чем их есть в наборе, даже одну единственную переменную из набора, вероятно 1-ю, в качестве представителя всего набора, - и потому желает профиль распределения ответов там приблизить к таковому целого набора. RANDOM=NO и умолчание/незадание подкоманды - не делать рандомизацию.

**CROP**

Эта опция (CROP=YES) удаляет в конце работы макроса «лишние» переменные MRC-набора, т.е. не содержащие валидных кодов. Она невозможна при SELFEXE=NO. По умолчанию/незаданию подкоманды и при CROP=NO «лишние» переменные сохраняются.

**FILLER**

Необязательная подкоманда, в которой вы можете указать макросу код-наполнитель для пустых ячеек, находящихся правее валидных данных. Наполнитель должен быть user-missing в переменных набора, но макрос не придаст ему статус пропуска, полагая, что такой статус у для него уже существует в переменных VARS. В общем, рекомендуется указывать в качестве наполнителя код, который уже является наполнителем с user-missing статусом. (Код может быть любым числом.) Неуказание кода или умолчание подкоманды оставит ячейки без валидных кодов пустыми (system-missing).

**SELFEXE**

По умолчанию/незаданию и при SELFEXE=YES макрос исполняет себя сам. При SELFEXE=NO он не исполняет себя, а ждет своего исполнения дальнейшей командой EXECUTE или командой-процедурой (или следующим пуском макроса, уже без SELFEXE=NO). Опция SELFEXE=NO дает возможность: (1) ставить макрос внутрь DO IF конструкции, благодаря чему ограничить наблюдения, им затрагиваемые; (2) ставить его, в том числе в сопровождении преобразований, под команду TEMPORARY, за счет чего сделать его эффект временным – на одну ближайшую процедуру.

Избегайте при SELFEXE=NO пускать макрос подряд для *разных* переменных:

!KO\_regmrc vars= v1 to v8 /selfexe= NO.

!KO\_regmrc vars= w1 to w6.

т.к. получите неправильный результат. В этом примере SELFEXE=NO в первом пуске это ошибка и должно быть убрано.

При вставке пуска макроса в DO IF-конструкцию может появиться предупреждение, “Command name SET. LOOP has no effect on this command”. Просто игнорируйте его.

ПРИМЕР 4.

do if region=8.

!KO\_regmrc vars= mr\_a to mr\_f /sole= DELETE 10 /selfexe= NO.

else.

!KO\_regmrc vars= mr\_a to mr\_f /check= YES 10 /selfexe= NO.

end if.

execute.

* Оба пуска макроса придают или гарантируют набору MR\_A – MR\_F правильное строение.
* У респондентов, принадлежащих к REGION=8, код 10 удаляется из набора, если он встречается в сочетании с другими кодами. У остальных респондентов с валидными данными обеспечивается, чтобы код 10 встречался не более раза.
* EXECUTE исполняет макрос. Важно, чтобы оба подусловных пуска с SELFEXE=NO занимались одними и теми же переменными VARS.

ПРИМЕР 5. Временное объединение нескольких смежных MRC-наборов.

temporary.

!KO\_regmrc vars= mrc1.1 to mrc3.4 /check= YES /sole= DELETE 99 /selfexe= NO.

select if nvalid(mrc1.1 to mrc3.4)<=5.

tables /mrgroup= $mrc mrc1.1 to mrc1.5 mrc2.1 to mrc2.8 mrc3.1 to mrc3.4 /table= $mrc.

* Три категориальных набора с общими вариантами ответа, MRC1.1…MRC1.5, MRC2.1…MRC2.8, MRC3.1…MRC3.4, располагаются в массиве один за другим. В VARS они указаны как один набор.
* Макрос придает объединенному набору правильное строение. Чтобы устранить повторение кодов, связанное с объединением наборов в один, заказано CHECK, а чтобы код «нет ответа» 99 не объединился в одну компанию с прочими кодами, заказано его в подобном случае убрать.
* Исследователь командой SELECT IF отбирает, по своему желанию, в табулирование только респондентов, давших не более 5 ответов в чищенном макросом объединенном наборе.
* Команда-процедура TABLES исполняет макрос и отбор и табулирует объединенный набор. Т.к. макрос (равно и отбор) стоял под TEMPORARY, после процедуры данные возвращаются в исходное состояние. Макрос обязан иметь спецификацию /SELFEXE=NO.

***Особые режимы***

Макрос не реагирует на взвешенность, фильтрованность (FILTER, USE), расщепленность массива данных. Кроме того, он отменяет взвешенность и расщепленность, если CROP=YES. Под TEMPORARY есть смысл его ставить лишь при SELFEXE=NO.

# МАКРОС !KO\_MRDNA: СНАБЖЕНИЕ ДИХОТОМИЧЕСКИХ НАБОРОВ МНОЖЕСТВЕННОГО ОТВЕТА ПЕРЕМЕННОЙ «НЕТ ОТВЕТА»

Version 3, May 2009 (Version 1, Apr 1999). Tested on SPSS Statistics 11, 13, 14, 26.

!KO\_mrdna caps= /\*Приставки (по числу наборов) в имена переменной "нет ответа"

/separ= *'.'* /\*ИЛИ Окавыченно разделитель, существующий во входящих переменных;

/\*если он там ничто, укажите ''

/namend= *@* /\*Окончание в имя переменной "нет ответа"

/label= *'Нет ответа' 1 'ДА' 0 'НЕТ'* /\*Опционально: ярлык переменной "нет ответа" и,

/\*если нужно, ярлыки для значений 1 и 0 в ней

/format= /\*Форматировать все переменные наборов (F1, ordinal): YES (тж п/у) или NO

/regist= NO /\*Регистрировать ли MRD набор с переменной "нет ответа": YES или NO (тж п/у);

/\*после YES можно указать окавыченные ярлыки для наборов в их порядке

/declare= YES /\*Сообщать, если создаваемая переменная "нет ответа" уже в наличии:

/\*YES или NO (тж п/у)

/sets= *v1.1 v1.2 v1.3 / v2.1 to v2.10*

/\*Один или несколько разделенных слэшем наборов двоичных переменных

/\*(без переменной "нет ответа"); набор писать поименно или диапазоном (ч-з to).

Минимум надо задать NAMEND, SETS. SETS идет последней.

Макрос создает или обновляет в двоичных наборах множественного ответа (MRD) переменные, значащие «нет ответа на вопрос». Эта переменная суть «негатив» суммы остальных переменных набора: она = 1 там, где они все = 0, и она = 0 в ином случае. Созданные новые переменные появятся в хвосте массива: вы потом сами передвинете их на нужное место.

ПРИМЕР 1.

!KO\_mrdna separ= '\_' /namend= # /regist= YES

/sets= q12\_1 to q12\_8 /q13.1\_1 to q13.1\_6 /q13.2\_1 to q13.2\_6.

* Трем наборам двоичных переменных будет добавлена, каждому, переменная «нет ответа». Ее имя будет оканчиваться на # (а если такая переменная уже есть, она обновится, перезапишется).
* SEPAR информирует макрос, что в именах переменных SETS есть разделитель \_ между «именем набора» и окончанием, индивидуальным для переменной. Таким образом, имена переменных «нет ответа» будут: *Q12#, Q13.1#, Q13.2#.*
* REGIST=YES: наборы будут зарегистрированы (или перерегистрированы) в файле данных под именами *Q12, Q13.1, Q13.2* как наборы дихотомического множественного ответа.

***Подкоманды***

**SETS**

Подкоманда должна идти последней. Укажите здесь MRD-наборы, для каждого из которых надо создать либо обновить переменную «нет ответа» (*не* указывайте здесь саму переменную «нет ответа», если она уже есть). Пишите имена переменных набора, не зарегистрированное имя набора. Если наборов несколько, разделяйте их слэшем (/). Писать можно двояко и разные наборы по-разному: перечислить переменные поименно или указать диапазон смежных переменных в массиве данных ч-з “to”. Не смешивайте оба вида записи: писать var1 var3 to var10 – в данном макросе нельзя.

MRD-набор это любое собрание числовых двоичных переменных, которые вы рассматриваете как набор «множественного ответа». Все переменные должны быть строго двоичной кодировки: 1 и 0. Пустые ячейки (system-missing) допускаются; переменная «нет ответа» будет system-missing для наблюдений, которое system-missing во всех переменных данного набора.

**NAMEND**

Окончание в имя переменной «нет ответа». Укажите символ или их сочетание. Это окончание касается всех наборов SETS. Переменная «нет ответа» будет иметь имя, в котором окончание прибавлено к имени набора. Имя набора определяется подкомандой SEPAR либо подкомандой CAPS. Если в массиве данных уже существуют переменные, одноименные переменным, которые калькулирует макрос, то те переменные будут обновлены.

**CAPS**

Задание этой подкоманды несовместимо с заданием SEPAR. В CAPS укажите для каждого из наборов SETS свою приставку, которая будет головой имени переменной «нет ответа» нем. Т.е. список приставок, по одной на каждый набор. Имя переменной «нет ответа» = приставка + NAMEND. Эта же приставка становится именем набора, под каким набор будет зарегистрирован, если REGIST=YES.

**SEPAR**

Задание этой подкоманды несовместимо с заданием CAPS. В SEPAR макросу указывается (в кавычках/апострофах) символ-разделитель, который может содержаться в именах переменных, образующих наборы SETS, и который разграничивает там общую часть, или «имя набора», и окончание, индивидуальное для переменных. Например, в именах *MRD\_1 MRD\_2* очевидно разделителем служит «\_». Если в SETS написано несколько наборов, все они должны иметь один и тот же разделитель. Существование разделителя в именах переменных не обязательно.

Макрос в действительности не проверяет, одинаковы ли у переменных набора части имен, предшествующие разделителю, и во всех ли переменных есть разделитель: он смотрит только на *первую* переменную набора (в порядке написания в SETS) и вычленяет из ее имени до-разделительную часть, которая принимается за «имя набора». Имя переменной «нет ответа» составляется как имя набора, разделитель и NAMEND.

Подробнее, существует три варианта задания SEPAR:

*Указать разделитель* – подходит, если имена в SETS содержат этот разделитель. Отсутствие разделителя в имени 1-й (в порядке указания в SETS) переменной набора вызовет ошибку, с сообщением. Окончания в именах могут быть не обязательно числами, но любыми (напр. *MRD\_S*). Перед разделителем в именах могут стоять числа (напр., *MRD1\_1*). «Имя набора» это вся часть до разделителя. Имя переменной «нет ответа» = имя набора + разделитель + NAMEND.

*Указать пустые кавычки/апострофы (не пробел!)* – подходит, если окончаниями в именах являются целые числа, а вся предшествующая, нецифровая часть имени это «имя набора» (напр., *MRD1*; не ошибка и *MRD\_1*). «Имя набора» это вся часть до 1-й цифры. Отсутствие цифры в хвосте имени 1-й переменной набора вызовет ошибку, с сообщением. Имя переменной «нет ответа» = имя набора + NAMEND.

*Опустить указание (и не задавать CAPS)* – подходит, если имена переменных в наборах не составлены как «имя набора» плюс окончание, а являются или принимаются пользователем как уникальные. Это самое универсальное задание SEPAR. За «имя набора» макрос берет целиком имя 1-й переменной набора. Имя переменной «нет ответа» = имя набора + NAMEND.

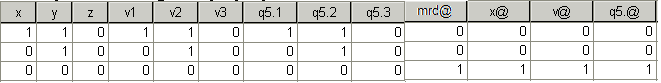
ПРИМЕР 2. Ситуации разных вариантов задания SEPAR.

!KO\_mrdna caps= mrd /namend= @ /sets= x y z.

!KO\_mrdna separ= /namend= @ /sets= x y z.

!KO\_mrdna separ= '' /namend= @ /sets= v1 v2 v3.

!KO\_mrdna separ= '.' /namend= @ /sets= q5.1 q5.2 q5.3.



* В 1-м пуске пользователь сам задает имя для набора через CAPS. Переменная «нет ответа» будет называться *MRD@*. SEPAR не задано.
* Во 2-м пуске набор *X Y Z* состоит из переменных без разделителя или цифр в именах. SEPAR не задано (либо умолчено, как и CAPS). Переменная «нет ответа» *X@* образовалась именем из целого имени первой переменной. Этот вариант не дал бы ошибки и с наборами *V1 V2 V3* или *Q5.1 Q5.2 Q5.3*, но малоприемлем с ними.
* В 3-м пуске имена в наборе *V1 V2 V3* составлены из общей буквенной приставки и индивидуального цифрового окончания. Оптимальный выбор – задать SEPAR пустыми кавычками, что даст переменную «нет ответа» *V@*.
* В 4-м пуске имена в наборе *Q5.1 Q5.2 Q5.3* имеют знак-разделитель (‘.’). Оптимальный выбор – указать этот разделитель, что даст переменную «нет ответа» *Q5.@*. Вариант SEPAR=‘’ в данном случае даст тот же результат, т.к. окончания за разделителем – целые числа.

**LABEL**

Необязательная подкоманда, позволяющая придать ярлык переменной «нет ответа»; вы можете также дать ярлыки ее значениям (1 и 0). Укажите в кавычках/апострофах ярлык для переменной, например, ‘Нет ответа’. После ярлыка, если хотите, укажите значения с их ярлыками, например: 1 ‘ДА’ 0 ‘НЕТ’.

**FORMAT**

По умолчанию и при FORMAT=YES все переменные SETS и созданные переменные «нет ответа» форматируются макросом: F1, порядковый уровень. Укажите FORMAT=NO, если этого не хотите.

**REGIST**

Необязательная подкоманда, позволяющая «зарегистрировать» в файле данных каждый набор из SETS как MRD-набор; переменная «нет ответа» также войдет в него. NO – не регистрировать. YES – зарегистрировать; после YES можно указать в кавычках/апострофах описательные ярлыки для наборов в их порядке, например: ‘Какие из перечисленных телепередач вы смотрите?’ ‘Какие радиопрограммы вы слушаете?’.

Регистрируемое имя набора – это то, которое макрос использовал в имени переменной «нет ответа»; т.е. это имя либо заданное в CAPS, либо определенное согласно SEPAR. Если имя набора оканчивается на символ “\_” или “.”, то он будет выпущен из имени регистирруемого набора.

**DECLARE**

DECLARE=YES заставляет SPSS Statistics сообщить: “Error… A variable with this name is already defined… Execution of this command stops” – в том случае, если создаваемая/обновляемая переменная «нет ответа» уже в наличии в массиве, т.е. она существовала до пуска макроса или была создана макросом в текущем пуске для более раннего набора в SETS. Данное сообщение играет роль только предупреждения, не обрывает работу и не влияет на результат – переменная в любом случае обновится.

По умолчанию и DECLARE=NO вышеприведенного сообщения не будет.

***Особые режимы***

Макрос не реагирует на взвешенность, фильтрованность (FILTER, USE), расщепленность массива данных. Он не слушается временных (под TEMPORARY) преобразований.

# МАКРОС !KO\_DONRECI: ПОВЕРКА КАТЕГОРИАЛЬНОГО НАБОРА ПЕРЕМЕННЫХ ДРУГИМИ КАТЕГОРИАЛЬНЫМИ ПЕРЕМЕННЫМИ С ТЕМИ ЖЕ КАТЕГОРИЯМИ

Version 3, Nov 2009 (Version 1, Jun 2001). Tested on SPSS Statistics 20, 22, 25.

!KO\_donreci recip= *v1 to v6* /\*"Реципиентный", поверяемый набор переменных, куда вставлять или

/\*откуда удалять значения

/donat= *w1 w2 w3 w4* /\*"Донорский", поверяющий набор переменных, по которому справляться

/values= *1 to* *8* EXCEPT *6 7* /\*Список (или поименно, или диапазоном ч-з to) интересующих неотриц

/\*значений для вставки/удаления/перекодирования;

/\*после диапазона можно перечислить, после сл EXCEPT, значения, ктр требуется

/\*исключить из него

/clean= ADD /\*Характер поверки: добавлять в реципиенты отсутствующие в них значения,

/\*присутствующие в донорах (ADD, п/у);

/\*удалять из реципиентов значения, отсутствующие в донорах (REMOVE);

/\*делать то или другое по случайному решению (CHOOSE);

/\*перекодировать в реципиентах значения, отсутствующие в донорах (RECODE список)

/valrand= /\*Рандомизовать значения в списке VALUES для респондентов: YES или NO (тж п/у)

/suffic= /\*Опционально: вставить/удалить респонденту не более стольких значений из VALUES

/\*(число или имя переменной)

/report= YES /\*Отчитаться спец переменными о количестве вставок, удалений и недовставок из-за

/\*нехватки мест: YES или NO (п/у)

/replace= *99* /\*Опционально: валидные значения в реципиентах, которые можно замещать

/\*(можно использовать слова THRU, LO, HI - как в Recode-команде)

/bases= /\*Какими респондентами заниматься: на пересечении баз RECIP и DONAT (INTERSECT)

/\*или объединив базы (UNITE, тж п/у)

/noans= *99* /\*Опционально: код "нет ответа", который вставить в первую переменную RECIP, если на

/\*выходе все RECIP оказались пусты

/selfexe= /\*Макросу исполнять себя самому (YES, тж п/у) или не исполнять (NO).

Минимум надо задать RECIP, DONAT, VALUES.

Макрос поверяет (сверяет-чистит) категориальный набор множественного ответа (MRC) или одиночную категориальную переменную данными других категориальных переменных с теми же категориями. Типичный контекст применения данного макроса – когда поверяемые переменные («реципиенты») и поверяющие переменные («доноры») относятся к вопросам анкеты, у которых общий перечень вариантов ответов, и вы желаете привести обе стороны к согласованности друг с другом, т.е. чтобы данные переменных-реципиентов не противоречили данным переменных-доноров.

Поверка – это, по вашему выбору, обогащение данными, изъятие данных или обогащение/изъятие. Еще один вариант: перекодировка:

1) Обогащение: *переменные-доноры вносят в переменные-реципиенты недостающие им значения*.

2) Изъятие: *реципиенты избавляются от лишних по сравнению с донорами значений*.

3) Обогащение/изъятие – смешанный процесс, когда добавление, изъятие и воздержание от поверки хаотично сменяют друг друга по ходу работы макроса, что ведет к *неполному и случайному уподоблению реципиентов* донорам.

4) Перекодировка: это то же, что изъятие (2), только лишние значения не стираются, а перекодируются в предзаданные значения.

Во всех случаях данные в донорах не меняются, и значения в реципиентах, которые найдены также и в донорах, сохраняются в реципиентах на своих местах. Вы можете сделать изменения данных в реципиентах временными, на одну ближайшую процедуру: для этого, в условиях п/к SELFEXE=NO, поставьте пуск макроса под TEMPORARY.

Макрос создает временные переменные с именами, содержащими пять символов *$* подряд, например, *v$$$$$.\_2*. Поэтому желательно, чтобы таких имен не было в вашем массиве. Если вы пускаете макрос в режиме SELFEXE=NO и создаете перед этим скретч-переменные, их именам также лучше не иметь пять символов *$* подряд.

ПРИМЕР 1.

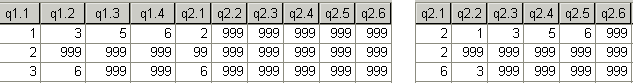
!KO\_donreci recip= q2\_1 to q2\_4 /donat= q1 /values= 2 5 6 7 /clean= REMOVE /report= YES.

!KO\_regmrc vars= q2\_1 to q2\_4.

* Внутри наблюдений сравниваются данные, конкретно – значения 2, 5, 6, 7, между переменными Q2\_1 – Q2\_4 (чистимые) и переменной Q1 (консультативная).
* CLEAN=REMOVE, т.е. если значения в Q1 у наблюдения нет, то оно стирается из Q2\_1 – Q2\_4, если там присутствует.
* Проведенные удаления посчитаны отчетными переменными, созданным п/к REPORT.
* В том случае, если переменные Q2\_1 – Q2\_4 это категориальный набор множественного ответа, то разумно после удаления из него части данных привести его к правильному виду, утрамбовав оставшиеся данные влево – что и сделано макросом !KO\_REGMRC.

ПРИМЕР 2. Было 2 вопроса на множественный ответ с общими вариантами ответа. Один вопрос – без подсказки (т.е. без показа вариантов), «Какие марки шампуней вы можете назвать?», а второй с подсказкой «Какие еще, из марок на карточке, вам знакомы?». Ответы на второй вопрос набивались так, что не включали отмеченное в первом вопросе. Однако исследователь потом захотел сделать такое включение, чтобы второй вопрос стал вопросом об общей, неважно, с подсказки или без, осведомленности о марках.

!KO\_donreci recip= q2.1 to q2.6 /donat= q1.1 to q1.4 /values= 1 to 6 /replace= 99.



* Оба вопроса – MRC-наборы, Q1.1 до Q1.4 и Q2.1 до Q2.6, с наполнителем 999 (user-missing). Код 99 значил «нет ответа». Варианты ответов на вопрос – коды от 1 до 6.
* По умолчанию, макрос выполняет обогащение: затребовано скопировать все такие ответы, которые есть в наборе «Q1», но отсутствуют в наборе «Q2», в «Q2». Кроме пропущенных значений, разрешено замещать там, что естественно, код 99.
* Результат показан отдельно справа. Все коды от 1 до 6, найденные в первом наборе и не найденные во втором, вставлены во второй. Чтобы все вставки могли осуществиться, в «Q2», разумеется, нужно достаточное число «пустых», с кодом 999, ячеек; их оказалось достаточно.

***Подкоманды***

**RECIP**, **DONAT**

В RECIP укажите поверяемые переменные («реципиенты»). Это переменные, куда макрос добавит или откуда удалит некоторые данные. В DONAT укажите поверяющие переменные («доноры»). Это переменные, по которым справляться; данные в них не будут изменены. Переменные в обоих подкомандах указывайте поименно либо диапазоном через “to”: *var* to *var* (это должны быть тогда смежные переменные в массиве), но не смешанно: *var1 var3* to *var10* писать в данном макросе нельзя. Все переменные должны быть числовые.

Переменные в каждой из этих двух подкоманд могут быть: 1) либо категориальный набор множественного ответа (MRC), весь или его часть, набор не обязательно зарегистрированный в файле; 2) либо единственная категориальная переменная; 3) либо множество самостоятельных категориальных переменных. Вариант (3) допустим в списке RECIP только в случае CLEAN= REMOVE или RECODE.

RECIP- и DONAT-переменные должны быть категориальные, с одной и той же или перекрывающейся областью значений (категорий). Т.е. реципиенты и доноры должны содержать, к примеру, ответы на такие два вопроса анкеты, в которых перечни вариантов ответов одинаковые или сходные.

**VALUES**

RECIP- и DONAT-переменные должны быть с одной и той же или перекрывающейся областью значений (категорий). В VALUES укажите, какие конкретно значения вас интересуют. Это должны быть неотрицательные числа (макрос не работает с отрицательными кодами в VALUES). Наличие этих значений в переменных-реципиентов будет сопоставлено с наличием их в переменных-донорах, и при обнаружении несоответствия по данному значению будет осуществлено действие CLEAN.

Эти значения должны в норме иметь валидный статус как в переменных-реципиентах, так и в переменных-донорах. Задать список можно либо перечислением (например, *1 2 5 16*), либо диапазоном через “to”: min to max, например, *2 to 9*. В первом случае значения не обязаны быть целыми числами. Когда же они заданы диапазоном, макрос формирует список как целые числа от min до max (оба эти числа должны быть целыми).

Вставкой ключевого слова EXCEPT после диапазона можно задать список кодов, которые вы хотите выпустить из диапазона. Например, если VALUES= *1 to 5* EXCEPT *2 4*, то список кодов, который сформирует макрос, будет *1 3 5*. Если после слова EXCEPT кодов нет, то само слово игнорируется.

**CLEAN**

Задайте действие поверки:

ADD - (тж. по умолчанию/незаданию подкоманды) добавление значений в реципиенты. Используйте эту опцию, когда ваша цель – **обогатить данные в реципиентах данными доноров**. Каждое значение из VALUES, найденное в донорах и не найденное в реципиентах, вставляется в последние. Значение вставляется респонденту (наблюдению) в одном экземпляре. Вставка осуществляется в ячейки, содержащие пропуски (system- и user-missing значения) и, по желанию, на место определенных валидных значений (см. п/к REPLACE). В поисках ячейки для вставки макрос просматривает переменные слева направо, как они указаны в RECIP. Если таких ячеек у респондента не было или больше нет, вставки ему прекращаются. Если нужно, чтобы все VALUES имели шанс быть вставлены, позаботьтесь о достаточном количестве «свободных» ячей в реципиентах, что можно сделать подключением дополнительных пустых переменных в RECIP. (Используйте подкоманду REPORT для разузнания количества недовставок.)

REMOVE - удаление значений из реципиентов. Используйте эту опцию, когда цель – **избавить реципиенты от данных, которых нет в донорах**. Каждое значение из VALUES, найденное в реципиентах и не найденное в донорах, стирается из первых. Все экземпляры этого значения будут стерты у респондента (это считается за одно удаление). При этой опции реципиенты в общем не обязаны составлять MRC-набор.

CHOOSE - добавление или удаление – по случайному выбору. Используйте эту опцию, чтобы **частично уподобить данные реципиентов данным доноров** в отношении значений VALUES. Касательно каждого очередного значения VALUES и очередного респондента (наблюдения) решение – добавлять, удалять или же оставить ситуацию как есть[[2]](#footnote-2) – принимается случайно и до действия (добавить или удалить). Следствием этого становится, что данные реципиентов согласовываются с донорами не полностью, результат зависит от случайных чисел[[3]](#footnote-3), а насыщенность реципиентов данными в итоге меняется мало. О параметре при CHOOSE см. ниже.

RECODE *список* - **перекодировка** значений в соответствующие другие. Каждое значение из VALUES, найденное в реципиентах и не найденное в донорах, заменяется отвечающим ему значением из *списка*. Все экземпляры значения будут заменены у респондента. При этой опции реципиенты в общем не обязаны составлять MRC-набор. *Список* значений должен состоять из неотрицательных чисел и быть той же длины, что список VALUES. Задать его вы можете перечислением или диапазоном – так же, как это можно в п/к VALUES, и вы можете употребить слово EXCEPT. Значения могут повторяться в *списке*. Вы можете также написать одно значение с последующим кл. словом ALL, например: CLEAN= RECODE 3 ALL. Макрос тогда перекодирует всякое значение VALUES в это единственное значение (тут 3).

Если требуется **максимально уподобить данные реципиентов данным доноров** в отношении VALUES, пустите макрос сначала с опцией REMOVE, чтобы освободить реципиенты от лишних значений и опорожнить занимаемые ими ячейки, а потом пустите с опцией ADD, чтобы добавить недостающие значения.

*Параметр при CHOOSE*. Опцию CHOOSE можно настроить в пользу преимущественно добавления или удаления. Для этого укажите число между 0 и 1 после кл. слова, например: CLEAN= CHOOSE *0.8*. В данном примере добавления будут иметь возможность совершаться чаще удалений. CHOOSE *1* эквивалентно CLEAN=ADD. CHOOSE *0* эквивалентно CLEAN=REMOVE. Слово CHOOSE без числа есть CHOOSE *0.5*. Заметьте, что чем дальше от 0.5 отступает параметр, тем больше ослабевает способность оставлять положение как есть, – в пользу более исчерпывающей, более согласовывающей поверки. Только при параметре, близком к 0.5, насыщенность реципиентов данными изменится мало. Вместо числа можно указать имя переменной, содержащий значения между 0 и 1, значения могут быть разными для разных наблюдений.

ПРИМЕР 3. Подобно ПРИМЕРУ 1, но CLEAN=RECODE.

!KO\_donreci recip= q2\_1 to q2\_4 /donat= q1 /values= 2 5 6 7 /clean= RECODE 22 55 66 77.

* Для каждого наблюдения массива сравниваются данные, конкретно – значения 2, 5, 6, 7, между переменными Q2\_1 – Q2\_4 (чистимые) и переменной Q1 (консультативная).
* CLEAN= RECODE 22 55 66 77, т.е. если значения в Q1 у наблюдения нет, то оно перекодируется в Q2\_1 – Q2\_4 так: 2 в 22, 5 в 55, 6 в 66, 7 в 77.

**SUFFIC**

Это предельно разрешенное число вставленных или удаленных значений VALUES у респондента (наблюдения массива). По умолчанию оно равно числу значений списка VALUES, – т.е. макрос пытается вставить (удалить) в реципиенты, где это нужно, все значения списка. В SUFFIC вы можете положить свое ограничение количеству вставок (удалений), указав целое положительное число. Тогда макрос, вставив (удалив) респонденту это число значений, перестает заниматься этим респондентом.

Вместо числа вы вправе указать *переменную*, содержащую неотрицательные числа; они могут быть разными для разных респондентов. Макрос внутренне округляет данные этой переменной, а пропуски в ней трактует как 0. Значение 0 означает, что ни вставок, ни удалений респонденту не будет.

Вместе с SUFFIC имеет обычно смысл употребить VALRAND=YES, чтобы в условиях ограничения числа вставок (удалений) сравнять шансы разных VALUES на вставку (удаление).

SUFFIC игнорируется при CLEAN=RECODE.

ПРИМЕР 4. То же что ПРИМЕР 2, но с ограничением числа вставок.

!KO\_donreci recip= q2.1 to q2.6 /donat= q1.1 to q1.4 /values= 1 to 6 /replace= 99 /suffic= 2.

* Разрешено добавить каждому респонденту не более 2-х значений, поэтому у 1-го респондента – см. ПРИМЕР 2 – вставлены будут только значения 1 и 3, но не 5 и 6.

**VALRAND**

По умолчанию/незаданию и при VALRAND=NO макрос обрабатывает значения в том порядке, в каком они существуют в списке VALUES. Опция VALRAND=YES рандомизует порядок в этом списке по-разному для разных респондентов, так что первыми на вставку или удаление будут претендовать то одни, то другие из значений. Эта опция имеет смысл в основном в сочетании с заданием SUFFIC или же когда свободных мест для вставки заведомо меньше, чем может потребоваться вставить данных.

VALRAND=YES игнорируется при CLEAN=RECODE.

ПРИМЕР 5. Скопировать у каждого респондента не более 10% случайно отобранных его ответов.

compute suffic= 0.1\*nvalid(var1 to var50).

!KO\_donreci recip= x1 to x16 /donat= var1 to var50 /values= 1 to 75 /suffic= suffic /valrand= YES.

* Переменные, куда будут копироваться ответы – X1 до X16. Туда будут вноситься, в ячейки без валидных данных, значения, отсутствующие в этих переменных, но присутствующие в переменных VAR1 до VAR50. Возможные для вставки значения это коды от 1 до 75.
* Может быть внесено не больше кодов, чем указывает переменная SUFFIC, посчитанная как 10% от количества валидных ответов, данных респондентом в переменных VAR1 до VAR50.
* Без VALRAND=YES только первые в списке VALUES коды (1, 2, и т.д.) имели бы шанс быть вставлены, а так все коды с 1 до 75 имеют равный шанс.

**REPLACE**

Эта опция играет роль при CLEAN= ADD и CHOOSE. Она позволяет обозначить, какие ячейки в реципиентах – в дополнение к ячейкам с пропущенными (system- и user-missing) значениями – считать «свободными» для вставки. Укажите одно или несколько числовых значений с валидным статусом в переменных-реципиентах (значения могут быть в том числе дробные и отрицательные). Можно пользоваться ключевыми словами, разрешенными в SPSS-команде RECODE: THRU, LO, HI. К примеру, REPLACE= LO THRU 10 99 означает: вставлять можно в ячейку, которая или содержит пропуск (это всегда), или имеет значение не выше 10, или имеет значение 99.

ПРИМЕР 6. Был вопрос на множественный ответ, «Какие телеканалы вы смотрите?» (Q1) и затем вопрос на один ответ, «Какой канал вы смотрите больше всего?» (Q2). За «нет ответа» закреплен код 99. Надо согласовать ответы респондентов между этими вопросами.

!KO\_donreci recip= q1.1 to q1.20 /donat= q2 /values= 1 to 15 /replace= 99.

!KO\_donreci donat= q1.1 to q1.20 /recip= q2 /values= 1 to 15 /replace= 99 /valrand= YES.

* Первый пуск макроса вносит телеканал (всякий из 15-ти), упомянутый в Q2, но не упомянутый в «Q1», в «Q1» (MRC-набор Q1.1 до Q1.20). Кроме пустых ячей, разрешено замещать код 99.
* Второй пуск делает обратную поверку, Q2 по «Q1», т.к. исследователь захотел избавиться, по возможности, от неответов в Q2. Т.к. в «Q1» может быть названо несколько каналов, VALRAND выбирает случайно, какой из них вставить в Q2 на место «нет ответа».
* Менее склонный к симулированию исследователь предпочел бы другой подход: см. ПРИМЕР 7.

**BASES**

Эта подкоманда определяет, какими наблюдениями массива (респондентами) займется макрос.

UNITE - (тж. по умолчанию/незаданию) наблюдениями, у которых есть какие-нибудь валидные значения хотя бы одном из двух наборов переменных, RECIP или DONAT. Другими словами, респондентские базы будут слиты.

INTERSECT - наблюдениями, у которых какие-нибудь валидные значения есть в обоих наборах, RECIP и DONAT. Другими словами, несовпадение респондентских баз будет учтено и сохранено.

«Какие-нибудь валидные значения» подразумевает: не обязательно из списка VALUES.

Так, если RECIP и DONAT соответствуют двум вопросам анкеты (со сходными вариантами ответов), и этими двумя вопросами опрашивались не полностью совпадающие подвыборки, то UNITE затронет чисткой объединение этих подвыборок, т.е. сольет их, а INTERSECT затронет чисткой только пересечение подвыборок. Если две подвыборки респондентов исходно совпадают, подкоманда не играет роли.

**REPORT**

Вы можете потребовать (REPORT=YES) отчет о проделанных у каждого респондента (наблюдения) удалениях и вставках значений в реципиенты. Создаются или обновляются три переменные-счетчика: REMOVE\_ (число удаленных значений либо, при CLEAN=RECODE, число перекодированных значений), ADD\_ (число добавленных значений), ADDFAIL\_ (число значений, которые не смогли быть добавлены по причине нехватки «свободных» ячей в реципиентах). Первая остается полна нулей при CLEAN=ADD, а две другие – при CLEAN=REMOVE. Все три переменные будут пусты (system-missing) у респондентов, которыми макрос не занимался. (Респонденты, которыми макрос занимается, определяются подкомандой BASES, а также каким-либо DO IF условием, внутрь которого пользователь может поместить макрос – см. подкоманду SELFEXE.) Если задано SUFFIC, сумма трех переменных-счетчиков всегда меньше или равна значению SUFFIC.

При CLEAN=CHOOSE, т.к. результат зависит от случайных чисел, показания счетчиков меняются от пуска к пуску с теми же входящими. В условиях CLEAN=CHOOSE иногда можно видеть картину, что у респондента свободные ячейки в реципиентах на выходе из макроса есть, тем не менее согласно переменной ADDFAIL\_ не все вставки были ему сделаны. Это возможно, потому что удаление-добавление CHOOSE идет динамически, и свободных мест для вставки не оказалось в *нужный* момент.

**NOANS**

Вы можете указать код (любое число), который захотите вставить в *первую* в списке RECIP переменную, если у респондента не оказалось ни одного валидного значения в реципиентах по выходе из макроса. Этот код имеет смысл «нет ответа»; его задание позволит гарантировать, что переменные-реципиенты не останутся без валидных данных у респондентов, которыми макрос занимался (код не вставляется респондентам, которыми макрос не занимался; вставка его не учитывается переменными, создаваемыми REPORT).

Вместо кода вы можете указать содержащую числовые значения *переменную*; таким образом разным наблюдениям можно назначить разные NOANS-коды.

**SELFEXE**

По умолчанию/незаданию и при SELFEXE=YES макрос исполняет себя сам. При SELFEXE=NO он не исполняет себя, а ждет своего исполнения дальнейшей командой EXECUTE или командой-процедурой (или следующим пуском макроса, уже без SELFEXE=NO). Опция SELFEXE=NO дает возможность: (1) ставить макрос внутрь DO IF конструкции, благодаря чему ограничить наблюдения, им затрагиваемые; (2) ставить его, в том числе в сопровождении преобразований, под команду TEMPORARY, за счет чего сделать его эффект временным – на одну ближайшую процедуру.

Опция SELFEXE=NO имеет следующее ограничение. Если вы пускаете макрос подряд с ней, и в *обоих* пусках VALRAND=YES, то во втором пуске VALUES обязаны быть *те же,* что в первом, - иначе будет неправильный результат. Т.е. избегайте следующего случая:

!KO\_donreci recip= … /donat= … /values= 2 3 4 5 /valrand= YES /selfexe=NO.

!KO\_donreci recip= … /donat= … /values= 1 to 10 /valrand= YES.

Правильно будет убрать SELFEXE=NO из первого пуска.

Также надо иметь в виду, что подкоманды CLEAN=CHOOSE и VALRAND=YES генерируют случайные числа, которые в условиях двух подряд пусков макроса под SELFEXE=NO будут иными, чем при двух подряд пусках без SELFEXE=NO. Поэтому результаты будут не те же самые.

При вставке пуска макроса в DO IF-конструкцию может появиться предупреждение, “Command name SET [или: ECHO]. LOOP has no effect on this command”. Просто игнорируйте его.

ПРИМЕР 7. Та же ситуация, что в ПРИМЕРЕ 6.

!KO\_donreci recip= q1.1 to q1.20 /donat= q2 /values= 1 to 15 /replace= 99.

do if nvalid(q1.1 to q1.20)=1.

!KO\_donreci donat= q1.1 to q1.20 /recip= q2 /values= 1 to 15 /replace= 99 /selfexe= NO.

end if.

execute.

* Первый пуск макроса тождествен тому что в ПРИМЕРЕ 6: вносит недостающий ответ из Q2 в набор «Q1».
* Второй пуск вносит ответ из набора «Q1» в Q2, замещая в этой переменной «нет ответа» (99) или пустоту; но это делается только в тех случаях, когда ответ – единственный валидный ответ в наборе «Q1». Потому что в противном случае – если ответов там несколько – неясно какой из них взять, и тогда лучше пусть в Q2 останется «нет ответа».
* EXECUTE исполняет второй пуск макроса. В первом пуске тоже можно было бы указать SELFEXE=NO: EXECUTE исполнила бы оба пуска.

ПРИМЕР 8.

do if region=8.

!KO\_donreci recip= q1 /donat= q2.1 q2.2 /values= 0 2 /replace= -99 /selfexe= NO.

else.

!KO\_donreci recip= q1 /donat= q3\_1 q3\_2 /values= 1 4 /replace= 0 99 /selfexe= NO.

end if.

execute.

* Для респондентов из региона 8 (это первый пуск макроса) переменная Q1 чистится переменными Q2.1, Q2.2, и чистка заключается во вставке недостающих значений 0 или 2 на место значения -99 (либо на место пропуска).
* Для прочих респондентов (второй пуск макроса) переменная Q1 чистится переменными Q3\_1, Q3\_2, и чистка заключается во вставке недостающих значений 1 или 4 на место значения 0 или 99 (либо на место пропуска).
* Чтобы сэкономить время, оба пуска исполняются за один раз внешней командой EXECUTE; сами же себя они не исполняют (в обоих: SELFEXE=NO).

***Особые режимы***

Макрос не реагирует на взвешенность, фильтрованность (FILTER, USE), расщепленность массива данных. Под TEMPORARY есть смысл его ставить лишь при SELFEXE=NO.

***Некоторые вопросы***

*В чем разница: (i) поместить пуск макроса в DO IF или (ii) указать в п/к SUFFIC переменную с определенными значениями нулевыми?* Это реальные альтернативные способы решить одну и ту же задачу: исключить некоторые наблюдения. Но это альтернативы не полностью эквивалентные. DO IF исключает некоторые наблюдения из обработки совсем, а SUFFIC с нулевыми значениями исключает такие наблюдения из основного действия макроса (подкоманда CLEAN), но не из эффектов подкоманд BASES, NOANS, REPORT. Также, CLEAN=RECODE игнорирует п/к SUFFIC.

# МАКРОС !KO\_STRMRC: РАЗВЕРТКА ТЕКСТОВОЙ ПЕРЕМЕННОЙ В КАТЕГОРИАЛЬНЫЙ НАБОР МНОЖЕСТВЕННОГО ОТВЕТА

Version 2, Jul 2004 (Version 1, Feb 2001). Tested on SPSS 11, 11.5, 13, 14.

!KO\_strmrc strvar= v1 /\*Текстовая переменная с ответами на вопрос множественного ответа

/pack= ONECHAR /\*Как набиты в ней коды: одно-значные впритык (ONECHAR);

/\*одно- и дву-значные отдельно те от других (DISPART);

/\*коды любой ширины через разграничитель (FREE)

/check= YES /\*Отсеивать возможные повторения того же кода (YES, тж п/у) или не отсеивать (NO)

/eqcase= YES /\*Если в STRVAR есть буквы, то считать ли строчные и заглавные буквы

/\*одним и тем же (YES, тж п/у) или не считать (NO)

/categs= 1 2 3 4 a b c /\*Коды ответов, использованные в переменной STRVAR, все или нужные;

/\*Если eqcase=NO, то значимо, строчными или заглавными писать в этом списке буквы;

/\*Если все коды - целые неотр числа, список можно указать диапазоном ч-з to: min to max

/\*и тогда после сл EXCEPT можно перечислить коды для выпуска из диапазона

/codes= 1 2 3 4 10 11 12 /\*Соответствующие CATEGS числовые коды (неотриц числа) в создаваемые

/\*переменные, либо SAME – взять сам список CATEG;

/\*Если опустить CODES, переменные выйдут текстовыми

/filler= 0 /\*Опционально: код-заполнитель для создаваемых переменных (он станет user-missing)

/noans= 99 /\*Опционально: код «нет ответа на вопрос» для создаваемых переменных

/limit= /\*Опционально: максимальное число ответов, какое разрешено было сделать респонденту;

/\*столько будет переменных в наборе

/separ= '.' /\*Опционально: окавыченно разделитель в имена создаваемых переменных

/regist= YES /\*Регистрировать ли созданный MRC набор: YES или NO (тж п/у);

/\*после YES можно указать окавыченный ярлык для набора.

Минимум надо задать STRVAR, PACK, CATEGS.

Данные, относящиеся к одному вопросу на множественный (неальтернативный) выбор, бывает удобно набивать в единую текстовую переменную. Макрос создаст из подобной текстовой переменной готовый к анализу категориальный набор множественного ответа (MRC). Можно сделать переменные числовыми или текстовыми, принять или отсеять повторение ответов.

***Способы набивки в текстовую переменную***

Текстовая переменная может быть набита одним из трех способов:

* Набивка *свободного* типа: коды ответов могут быть произвольной, в том числе разной, ширины (данный макрос разрешает ширину кода до 8 символов). Коды набиваются через разграничитель (например, пробел или плюс). Пример: **12 563 G6h R%#** - здесь набито через пробел 4 некоторых кода.
* Набивка типа *односимвольные коды впритык*: код может состоять только из одного символа, и набиваются коды без разграничения: **3a68S\*8** – здесь набито 7 кодов (код 8 набит дважды).
* Набивка *разделенного* типа: коды могут быть шириной в 1 или 2 символа (по кр. мере, данный макрос разрешает до 2-х символов). Все односимвольные коды идут в строке прежде двусимвольных, а между теми и другими стоит разграничитель. Пример: **36+121321** – здесь набиты коды 3, 6, 12, 13, 21. Если односимвольных кодов нет – например респондент не дал таких ответов – а есть только двусимвольные, разграничитель все равно должен присутствовать перед ними: **+121321**.

При любом типе набивки символы, из которых состоят коды, могут быть любыми значками, кроме тех что приняты за разграничители. В том случае если порядок ответов респондента важен (например, когда вопрос просил *выбрать и проранжировать* ответы), набивка разделенного типа, очевидно, не годится, т.к. в ней коды сгруппированы по их ширине[[4]](#footnote-4). С другой стороны, этот тип – самая быстрая и удобная набивка обыкновенного вопроса на множественный ответ. В самом деле: можно набивать числовые коды (типично ответы в анкетах кодируют числами 1, 2, 3, и т.д.; максимум тут можно будет использовать код 99). Также, в этой набивке не нужно вводить разграничитель – т.е. нажимать еще клавишу – после каждого кода. Эта набивка может быть даже лучше чем применение «автососкока», существующего в специальных программах по ручному вводу данных, т.к. автососкок хорош лишь тогда, когда все коды одной ширины. Впрочем, разделенная набивка требует внимательности или навыка от набивщика. Макрос !KO\_ERRDISP (см. ниже) может обнаружить ошибки, допускаемые набивщиками в разделенной набивке.

***Подкоманды***

**STRVAR**

Имя текстовой переменной, содержащей ответы на вопрос. Переменные создаваемого набора будут поименованы как имя переменной STRVAR целиком + разделитель SEPAR (если задан) + порядковый номер создаваемой переменной от 1 до LIMIT (см.). Т.о., если вы используете SPSS ниже 12-й версии – где имя переменной может быть до 8 символов – имя STRVAR сделайте достаточно коротким, чтобы все переменные могли получиться. Создаваемые переменные должны несовпадать именами с какими-л. существующими в файле переменными.

**PACK**

Укажите тип набивки в текстовой переменной:

FREE – коды разделены (между ними разграничитель). Коды могут быть шириной до 8 символов и состоять из любых символов.

ONECHAR – коды идут впритык друг к другу. Коды могут быть любыми символами единичной ширины.

DISPART – коды идут впритык друг к другу. Коды могут быть шириной до 2 символов и состоять из любых символов (типично использование одно- и дву-значных натуральных чисел). Двусимвольные коды должны идти после односимвольных. Перед первым набитым двусимвольным кодом (независимо от того, есть ли перед ним односимвольные коды) должен стоять разграничитель.

Разграничителями макрос считает *пробел* и, кроме того, символы, какие указаны в первой команде тела макроса, DEFINE, в аргументе PM…!DEFAULT. Сейчас там указан *плюс*. Значит, макрос считает разграничителями пробел и плюс. Вы можете установить там свои символы, указав их один за другим, впритык. Например, (‘.,;’) будет значить, что разграничителем считается, кроме пробела, любое из: точка, запятая, точка-с-запятой. Если хотите иметь разграничителем только пробел, уберите все находящееся между апострофами: (‘’)[[5]](#footnote-5).

Вереница символов-разграничителей в STRVAR допустима; т.е. напр., ‘+++’ будет считаться тем же что ‘+’. Встреча разграничителей в набивке ONECHAR не ведет к ошибке: ’25 A C7’ считается как ‘25AC7’. В набивке DISPART разграничитель (или их вереница) может, по правилам, встретиться не более одного раза: т.е. набивка ‘13+1422+25’ является ошибкой набивки (макрос *не сообщает* о ней).

**CATEGS**

Здесь надо указать коды ответов (все или нужные), использованные в переменной STRVAR. Это могут быть числа, буквы, прочие символы[[6]](#footnote-6), их сочетания – все что позволяет тип набивки, PACK. В случае, если данные набиты целыми неотрицательными числами, то вместо их перечисления можно указать диапазон min to max, например 1 to 20. Макрос сам составит список из чисел от min до max. В SPSS ниже 13-й версии длина списка CATEGS должна быть меньше 255 символов, включая пустоты между ними, иначе SPSS сообщит об ошибке[[7]](#footnote-7).

Если CATEGS заданы диапазоном min to max, то дальше можно, вставив слово EXCEPT, привести поименно коды, которые хотите выпустить из диапазона, - они будут исключены макросом из формируемого списка кодов. Например, CATEGS= 1 to 5 EXCEPT 2 4. В итоге список будет: 1 3 5.

**CODES**

Этот список – соответственный списку CATEGS – задает коды, которые станут значениями переменных MRC-набора. Если требуется создать переменные числовыми, необходимо либо привести поименный список неотрицательных чисел в количестве, сколько кодов в CATEGS, либо указать CODES=SAME, что означает: использовать сам список CATEGS. В последнем случае CATEGS должен состоять целиком из неотрицательных чисел. Для того же, чтобы создать переменные текстовыми, надо умолчать или незадать подкоманду CODES. Значениями переменных при этом станут сами коды CATEGS.

**CHECK**

По умолчанию/незаданию этой подкоманды и при CHECK=YES макрос следит, не повторяются ли одни и те же коды в строке переменной STRVAR, и, если повторяются, не учитывает повторные встречи кодов. В выходящем наборе один и тот же код будет встречаться не более раза. При CHECK=NO макрос допускает повторения кодов в выходящие переменные, если коды повторяются в текстовой переменной.

**EQCASE**

Эта подкоманда играет роль, если в CATEGS присутствуют буквы. По умолчанию/незаданию и при EQCASE=YES регистр букв игнорируется в текстовой переменной: код *Abc*, например, будет считаться тем же что код *ABC*; и неважно, в каком регистре вы напишете буквы в CATEGS. При EQCASE=NO регист учитывается. В этом случае проследите, чтобы коды в списке CATEGS отвечали регистром букв нужным вам кодам, набитым в текстовой переменной.

SEPAR

Это необязательный разделитель в имена выходящих переменных. Укажите символ в кавычках или апострофах.

**LIMIT**

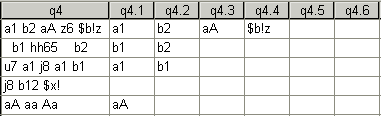
Заказ, сколько создать переменных в наборе. По умолчанию создается столько переменных, сколько кодов в CATEGS. Если респондентам разрешено было сделать не более стольких-то ответов, вы можете указать этот предел здесь: столько и выйдет переменных. С другой стороны, если было разрешено выбрать каждый вариант более раза, и следовательно вы допускаете повторение кодов (CHECK=NO), то вам может понадобиться заказать LIMIT больше, чем число кодов CATEGS, чтобы переменных создалось достаточно для размещения всех ответов.

**FILLER, NOANS**

FILLER это код-наполнитель для MRC-набора. Вы можете указать любое число (или не число, если переменные создаете текстовыми). Код-наполнитель будет иметь статус пропущенного значения в во всех переменных набора или, если это тот же код что NOANS, во всех переменных кроме первой. NOANS это код «нет ответа». Он встретится только в первой переменной набора – у респондентов, не давших ни одного ответа из списка CATEGS. Обе подкоманды необязательны. Не указывайте в качестве FILLER или NOANS коды из CATEGS или CODES.

ПРИМЕР 1. Набивка свободного типа.

!KO\_strmrc strvar= q4 /pack= FREE /check= YES /eqcase= NO /categs= a1 a2 b1 b2 aA $b!z /separ= '.'.

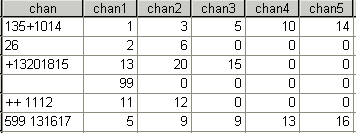


* Переменная Q4 распакована в набор из 6-ти переменных (т.к. кодов CATEGS шесть, а п/к LIMIT не задана). Переменные вышли текстовыми, т.к. п/к CODES не задана. Интересовавшие пользователя коды указаны в CATEGS. Они, если найдены в Q4, перешли в созданные переменные.
* Т.к. CHECK=YES, код “a1”, встречающийся дважды у 3-го респондента, взят в набор только один раз. Т.к. EQCASE=NO, макрос уважает регистр букв: например, коды “aA”, “aa” и “Aa” это для него не одно и то же.

ПРИМЕР 2. Набивка разделенного типа.

!KO\_strmrc strvar= chan /pack= DISPART /check= NO /categs= 1 to 20 EXCEPT 18 19 /codes= SAME /limit= 5

/filler= 0 /noans= 99 /regist= YES 'Какие каналы вы смотрите больше всего?'.



* Затребованные для распаковки коды CATEGS – натуральные числа от 1 до 20, кроме 18 и 19. В выходящих числовых переменных набора, CHAN1 – CHAN5, они будут представлены самими собой (CODES=SAME).
* В качестве наполнителя (пропущенное значение) положен код 0, а в качестве «нет ответа на вопрос» (точнее, «не найдено ничего из CATEGS») код 99.
* Повторение подряд символов-разграничителей у 5-го респондента (плюс плюс пробел) не ведет к неправильной распаковке: ошибкой было бы набить их в разных местах строки.
* Т.к. CHECK=NO, оба кода 9 (6-й респондент) взяты в выходящий набор переменных. Т.к. заказано было создать именно 5 переменных (LIMIT=5), код 17 у этого респондента остался не взятым – не хватило переменных.

**REGIST**

Необязательная подкоманда, позволяющая «зарегистрировать» в файле данных созданный набор. Зарегистрирован он будет под именем переменной STRVAR. Опция регистрации существует в SPSS 11.5 и выше с установленным модулем Custom tables. NO (тж п/у) – не регистрировать. YES – зарегистрировать; после YES можно указать в кавычках/апострофах ярлык набора (формулировку вопроса), например: ‘Какие из перечисленных телепередач вы смотрите?’.

***Особые режимы***

Макрос бессмысленно пускать под командой TEMPORARY. Он не реагирует на взвешенность, фильтрованность, расщепленность файла данных.

# МАКРОС !KO\_STRMRD: РАЗВЕРТКА ТЕКСТОВОЙ ПЕРЕМЕННОЙ В ДИХОТОМИЧЕСКИЙ НАБОР МНОЖЕСТВЕННОГО ОТВЕТА

Version 2, Sep 2004 (Version 1, Aug 2000). Tested on SPSS 11, 11.5, 13, 14.

!KO\_strmrd strvar= v1 /\*Текстовая переменная с ответами на вопрос множественного ответа

/pack= FREE /\*Как набиты в ней коды: одно-значные впритык (ONECHAR);

/\*одно- и дву-значные отдельно те от других (DISPART);

/\*коды любой ширины через разграничитель (FREE)

/eqcase= YES /\*Если в STRVAR есть буквы, то считать ли строчные и заглавные буквы

/\*одним и тем же (YES, тж п/у) или не считать (NO)

/categs= 1 AB a22 7&,5 !w:^ cd /\*Коды ответов, использованные в переменной STRVAR, все или нужные;

/\*Если eqcase=NO, то значимо, строчными или заглавными писать в этом списке буквы;

/\*Если все коды - целые неотр числа, список можно указать диапазоном ч-з to: min to max

/\*и тогда после сл EXCEPT можно перечислить коды для выпуска из диапазона

/indxs= 1 2 3 4 5 6 /\*Cоответствующий списку categs cписок окончаний (числа, буквы, и т п) в имена

/\*создаваемых переменных; или SAME - использовать сам список categs

/noans= @ /\*Окончание в имя переменной НЕТ ОТВЕТА; если она не нужна, не задавайте

/separ= '.' /\*Опционально: окавыченно разделитель в имена создаваемых переменных

/regist= YES /\*(SPSS 11.5 и выше): Регистрировать ли созданный MRD набор: YES или NO (тж п/у);

/\*после YES можно указать окавыченный ярлык для набора.

Минимум надо задать STRVAR, PACK, CATEGS, INDXS.

Данные, относящиеся к одному вопросу на множественный (неальтернативный) выбор, бывает удобно набивать в единую текстовую переменную. Макрос создаст из подобной текстовой переменной готовый к анализу двоичный, или дихотомический, набор множественного ответа (MRD). О способах набивки кодов ответов в текстовую переменную см. выше, в описании макроса !KO\_STRMRC. Переменных, одноименных создаваемым, должно не быть во входящем файле.

Макрос создает на каждый вариант ответа одну числовую переменную. Если вопрос на множественный выбор разрешал выбирать один и тот же ответ более раза и вы хотите иметь соответственно по нескольку двоичных переменных на каждый вариант ответа, воспользуйтесь !KO\_STRMRC и затем переделайте категориальный набор в двоичный с помощью !KO\_MRCMRD (“Categorical – Binary recodings”).

***Подкоманды***

Подкоманды STRVAR, PACK, CATEGS, EQCASE, SEPAR, REGIST тождественны таковым макроса [!KO\_STRMRC](#_МАКРОС_!STRMRC:_РАЗВЕРТКА).

**INDXS**

Укажите поименный список окончаний в имена создаваемых переменных. Этот список соответствен списку кодов CATEGS. В качестве окончаний можно использовать числа, буквы, иные возможные в именах символы или их сочетания. Не используйте числобуквы (1a 3b и т.п.), т.к. они не считаются макросом цельными «словами». Буквочисла (a1 b3) – можно. Вместо списка можно указать ключевое слово SAME, т.е. – использовать сам список CATEGS. Тогда последний должен быть из цельных «слов», т.е. например чисел, букв(осочетаний) или буквочисел.

ПРИМЕР 1.

!KO\_strmrd strvar= chan /pack= DISPART /categs= 1 to 20 EXCEPT 18 19 /indxs= SAME /noans= na.

* Создастся 18 двоичных переменных с именами CHAN1, CHAN2,…, CHAN17, CHAN20.
* А также 19-я переменная набора, «нет ответа», CHANNA.

**NOANS**

Эта необязательная подкоманда заказывает создать к набору переменную «нет ответа на вопрос», точнее, «не выбран ни один код из CATEGS». Укажите символ или символосочетание, на что должна оканчиваться такая переменная. Переменная создается как негатив остальных двоичных переменных: она равна 1, если они все равны 0, и равна 0 в противном случае.

***Особые режимы***

Макрос бессмысленно пускать под командой TEMPORARY. Он не реагирует на взвешенность, фильтрованность, расщепленность файла данных.

# МАКРОС !KO\_ERRDISP: ПРОВЕРКА НАБИВКИ ТИПА DISPART

Version 1, Dec 2003. Tested on SPSS 11, 11.5, 13, 14.

!KO\_errdisp vars= v1 v2 /\*Список (поименно) текстовых переменных с набивкой типа dispart

/id= /\*Опционально: одна или более переменных-идентификаторов наблюдений.

Минимум надо задать VARS

Этот подсобный макрос проверяет текстовые переменные, набитые *разделенным* способом (см. в описании макроса !KO\_STRMRC выше), на присутствие некоторых ошибок набивки. Макрос выдает по каждой из указанных текстовых переменных номера анкет (респондентов) где встречена ошибка хотя бы одного из трех видов:

* число цифр (вообще символов) среди дву-значных кодов нечетное, например: 3+10125 – здесь, предположительно, хотели набить коды 3, 10, 12, 15, но единицу в «15» забыли.
* знак-разграничитель встречен более 1 раза в разных местах строки, например, 3+1012+15.
* в отсутствие в строке разграничителя цифра (вообще символ) встречается более одного раза, что – в случае если повторения кодов не предусматривались анкетой – скорее всего означает, что забыт разграничитель или что одно-значный код повторен.

Макрос не считает ошибкой, если строка оканчивается на разграничитель или если разграничитель идет вереницей (повторяется подряд). Макрос считает разграничителем пробел (всегда) и, кроме того, плюс. Вы можете заменить плюс на любой другой символ или символы в качестве разграничителей. Для этого укажите их в макросе, в команде DEFINE, аргументе PM…!DEFAULT(‘’). Например, (‘;,’) будет значить, что разграничителями считаются, кроме пробела, точка-с-запятой или запятая[[8]](#footnote-8).

***Подкоманды***

**VARS**

Укажите поименно одну или более текстовых переменных, набитых кодами *разделенным* способом.

**ID**

Вы можете указать одну или более переменных, идентифицирующих респондентов. По умолчанию используется номер наблюдения в файле.

1. Управление зерном случайных чисел в SPSS Statistics: меню Transform – Random Number Generator. [↑](#footnote-ref-1)
2. Решение «оставить ситуацию как есть» означает оказию, когда (a) случайно выбрано решение «удалить» значение у респондента, но затем выяснилось, что это действие невыполнимо (т.к. условие «в реципиентах значение есть, а в донорах нет» ложно); либо оказию, когда (b) случайно выбрано решение «добавить» значение респонденту, но затем выяснилось, что это действие невыполнимо (т.к. условие «в реципиентах значения нет, а в донорах есть» ложно). Заметьте, что в ситуации (a) противоположне действие, добавить, могло бы иногда быть выполнено вместо того – но оно будет пропущено; и аналогично в ситуации (b) противоположне действие, удалить, могло бы иногда быть выполнено вместо того – но оно будет пропущено. [↑](#footnote-ref-2)
3. Управление зерном случайных чисел в SPSS Statistics: меню Transform – Random Number Generator. [↑](#footnote-ref-3)
4. Но порядок кодов внутри группы односимвольных и внутри группы двусимвольных может быть произвольным. [↑](#footnote-ref-4)
5. Нюанс касательно разграничителя-точки такой: не указывайте ее в одиночестве (‘.’) или последней (‘,;.’), - это приведет к неправильной работе. Можно продублировать точку (‘..’) или указать не последней (‘.,;’). [↑](#footnote-ref-5)
6. Макрос не разрешает использовать в коде ответа слэш (/). [↑](#footnote-ref-6)
7. Следовательно, при использвании тех версий нужно заранее позаботиться, чтобы текстовая переменная, набитая по свободному (FREE) способу, употребляла как можно более узкие коды, если кодов в CATEGS предполагается указать много. [↑](#footnote-ref-7)
8. Нюанс касательно разграничителя-точки такой: не указывайте ее в одиночестве (‘.’) или последней (‘,;.’), - это приведет к неправильной работе. Можно продублировать точку (‘..’) или указать не последней (‘.,;’). [↑](#footnote-ref-8)