***Generate random clusters, mixtures***

SPSS macros by Kirill Orlov

kior@akado.ru, ttnphns@gmail.com

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

All rights reserved.

*Случайные кластерные/смесные данные.* Создание случайных данных, состоящих из четких кластеров или смесей (нечетких кластеров). Можно сделать эти облака круглыми или продолговатыми, гауссовыми или платикуртичными, регулировать их размеры и тесноту соприлегания. Отдельный макрос случайно поворачивает данные в пространстве.

*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 значащую, 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.

* [!KO\_GENCLU](#_МАКРОС_!GENCLU:_ПОРОЖДЕНИЕ_СЛУЧАЙНЫ) порождает кластеры и смеси, сферические либо сфероидные (продолговатые в одном направлении).
* [!KO\_ROTCLU](#_МАКРОС_!ROTCLU:_СЛУЧАЙНЫЙ_ПОВОРОТ_() поворачивает случайным образом данные в пространстве или лишает их повернутости.

# МАКРОС !KO\_GENCLU: ПОРОЖДЕНИЕ СЛУЧАЙНЫХ КЛАСТЕРНЫХ/СМЕСНЫХ ДАННЫХ

Version 2, May 2014 (Version 1, Apr 2010). Tested on SPSS Statistics 17, 20, 27.

!KO\_genclu k= *3* /\*Число кластеров (от 2 до 100)

/p= *2* /\*Число переменных (от 1 до 50)

/n= *100* /\*(Приблизительное) число точек в кластере: одно число, либо два - min и max

/centers= EVEN /\*Инициальные центры кластеров: RANDOM (создать их случайными, тж п/у),

/\*EVEN (создать их приблиз-но равноразделенными),

/\*или окавыченное имя внешнего SAV-файла (центры заданы вручную)

/spacing= *.8* /\*Теснота между кластерами: число от 0 (тесно сдвинутые кластеры)

/\*до 1 (широко раздвинутые кластеры), или RANDOM (определять случайно, тж п/у)

/sizing= *0* /\*Одинаковость физич величины кластеров: число от 0 (одинаковые кластеры)

/\*до 1 (сильно разновеликие кластеры), или RANDOM (определять случайно, тж п/у)

/flat= /\*Уплощенность кластеров: число от 0 (нормальный эксцесс, тж п/у) до 1

/\*(плоский эксцесс, кластер есть шар), или RANDOM (определять случайно для

/\*каждого кластера)

/elong= STRETCH *1 3* /\*Опционально, для p>1: сделать кластеры продолговатыми - вытянутыми

/\*(STRETCH) или продленными (EXTEND); после кл слова укажите

/\*2 параметра продолговатости (числа от 1 до 4)

/rotat= /\*Для продолговатых кластеров: их поворот в любом направлении (FREE, тж п/у),

/\*приблизительно в одном и том же (RESTR) или вообще не поворачивать (NONE)

/fix= YES /\*Привести набл центроиды порожденных кластеров точно к инициальным центрам:

/\*YES или NO (тж п/у)

/clear= LDA /\*Опциональлно: сделать кластеры четкими (отклассифицировать точки) методом:

/\*KNN (k nearest neighbors), LDA (discriminant анализ),

/\*KMC (K-средних, classify only)

/weed= *1* /\*Для четких кластеров: прополка пограничных зон - число от 0 (не делать, тж п/у)

/\*до 1 (делать полностью).

Минимум надо задать K, P, N.

Макрос порождает случайные кластерные/смесные данные. Можно затребовать породить от 2 до 100 кластеров; число переменных (размерность пространства) – от 1 до 50. Макрос выдает новый безымянный рабочий массив данных с переменной *CLUSTER\_* (номера кластеров) и переменными *V1, V2* и т.д. (координаты точек).

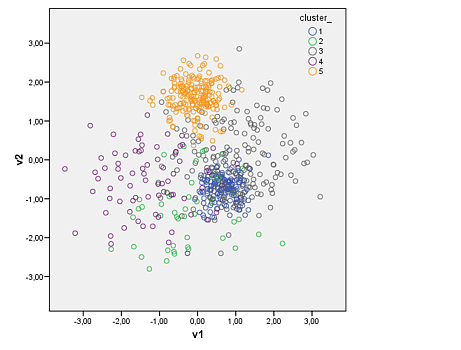
Инициальные центры кластеров – вокруг которых порождаются случайные точки – вы можете предложить макросу задать самостоятельно либо ввести свои. Вы можете позволить макросу задавать тесноту сопрелегания и размеры кластеров самостоятельно либо регулировать эти параметры по вашему желанию. По умолчанию кластеры создаются круглыми (сферическими) с нормальным распределением, но вы можете затребовать кластеры в разной и случайной степени продолговатые, а также с уплощенные (уменьшенный эксцесс вплоть до равномерного распределения). Вы можете создать кластеры нечеткими (fuzzy clusters, т.е. смесями) или четкими (clear clusters, к-рым не позволено пересекаться). Для четких кластеров можно заказать «прополку» зон их соприлегания в целях снизить густоту точек там.

Продолговатые кластеры в настоящее время доступны сфероидные, не эллипсоидные. Распределение в кластерах – не скошенное.

Для управления зерном случайных чисел в SPSS Statistics используйте меню Transform – Random Number Generators или соответствующий синтаксис.

ПРИМЕР 1.

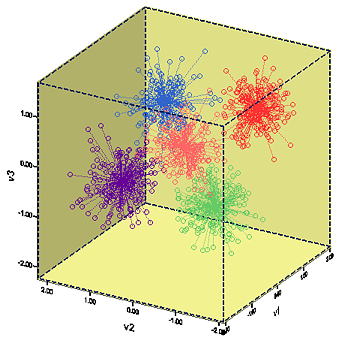
!KO\_genclu k= 5 /p= 2 /n= 50 200.



* В 2-мерном пространстве порождается 5 круглых кластеров с численностью точек от 50 до 200. Расстояния между центрами кластеров, физические размеры кластеров (величина дисперсии в них) макросу позволено выбрать случайно. Кластерам позволено остаться пересекающимися (нечеткими).

ПРИМЕР 2.

!KO\_genclu k= 5 /p= 3 /n= 200 /centers= EVEN /spacing= .8 /sizing= 0 /clear= KMC /weed= 1.



* В 3-мерном пространстве порождается 5 круглых кластеров, по 200 точек в каждом. Кластеры относительно равномерно разбросаны по пространству (CENTERS=EVEN), они одинаковой величины (SIZING=0) и довольно далеко отстоят друг от друга (SPACING=0.8). Кластеры заказано сделать непересекающимися, четкими (CLEAR=KMC) и прополоть пограничные зоны от густоты (WEED=1). Однако последние 2 подкоманды едва ли много изменили, поскольку кластеры и так практически не пересекаются, будучи в данном случае достаточно далеко отстоящими друг от друга.

***Подкоманды***

**K**

Число кластеров *k*. Укажите целое число от 2 до 100.

**P**

Число переменных (размерность пространства *p*). Укажите целое число от 1 до 50.

**N**

Число точек (наблюдений) в кластере. Укажите одно либо два целых положительных числа. Если укажете одно число, все кластеры будут равночисленными по числу точек. Если укажете 2 числа, то кластеры будут варьироваться между этими пределами по числу своих точек (не имеет значения, какое число, меньшее или большее, вы укажете первым из двух).

Учтите, что если вы заказываете прополку пограничных зон (см. п/к WEED), то число точек в кластерах может оказаться в итоге меньше, чем вы задали, поскольку часть точек в случае прополки удаляется (и оно будет тем меньше, чем теснее кластеры взаимоналагаются своими краями, - см. п/к SPACING).

**CENTERS**

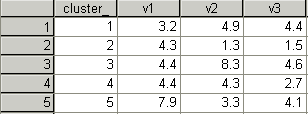
Задание инициальных центров кластеров, т.е. центров, вокруг которых будут порождаться случайные данные. Выберите одно из трех:

RANDOM - (тж. по умолчанию/незаданию подкоманды) Инициальные центры создаются как случайные точки в *p*-мерном пространстве. Одни соседние центры могут оказаться близко друг к другу, другие – в отдалении друг от друга. Другими словами, сами кластеры могут оказаться в итоге элементами кластеров более высокого порядка.

EVEN - инициальные центры создаются случайно, но приблизительно равномерно разбросанными по пространству. Используйте этот вариант, если хотите, чтобы кластеры нельзя было объединить друг с другом в кластеры высшего порядка.

*Имя файла* - пользователь задает инициальный центры сам. Укажите (в кавычках или апострофах) путь/имя внешнего SAV-файла, содержащего координаты инициальных центров.

Внешний SAV-файл с координатами центров должен иметь такое строение (см. рисунок): он должен содержать числовую переменную *CLUSTER\_* с порядковыми номерами кластеров 1, 2, …, *k* (где *k* то же число, что укажете в п/к K). И координаты центров в переменных V1, V2, …, V*p* (где *p* то же число, что укажете в п/к P). Никаких других переменных или лишних строк файл должен не иметь. Несоблюдение этих правил вызовет ошибку.



**SPACING**

Параметр, определяющий тесноту соприлегания соседних кластеров друг к другу. (Если кластерам позволено пересекаться – п/к CLEAR не задана – то это есть степень, в какой соседние кластеры взаимонакладываются друг на друга своими краями, образуя смесь.) Укажите одно из следующего:

*Число от 0 до 1* - 0 создает кластеры, все соседние (ближайшие) из которых будут тесно соприлегать друг к другу. Напротив, 1 создает кластеры, все соседние из которых будут далеко отстоять друг от друга. Вы можете указать любое число между 0 и 1. (Значения ровно 0 или 1 указывайте без десятичных знаков, т.е. напр. указать 0.0 или 1.0 – это неправильно.)

RANDOM - (тж. по умолчанию/незаданию подкоманды) теснота между двумя данными соседними кластерами определяется макросом случайно. Одни соседние кластеры могут оказаться тесно соприлегающими, тогда как другие соседние кластеры – весьма разделенными.

**SIZING**

Параметр, определяющий физическую разновеликость кластеров, неодинаковость их ст. отклонений, или «радиусов». Укажите одно из следующего:

*Число от 0 до 1* - 0 создает все кластеры одинаковыми по величине; 1 создает кластеры сильно разновеликие: радиус самого большого кластера будет относиться к радиусу самого маленького кластера как 3:1. Вы можете указать любое число между 0 и 1. (Значения ровно 0 или 1 указывайте без десятичных знаков, т.е. напр. указать 0.0 или 1.0 – это неправильно.)

RANDOM - (тж. по умолчанию/незаданию подкоманды) отношение размеров между самым большим и самым маленьким кластером макрос задаст сам как случайное число между 1 и 3.

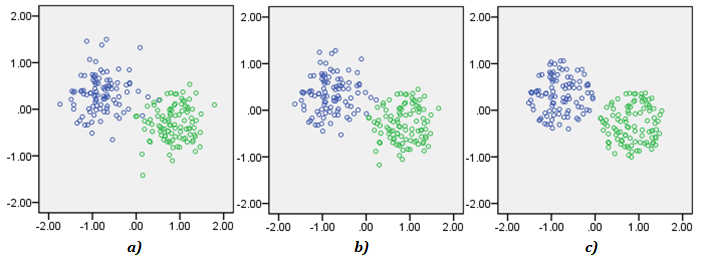
**FLAT**

Эта подкоманда не действует при ELONG=EXTEND. Параметр, определяющий эксцесс в распределении внутрикластерных точек (см. рисунок). По умолчанию и при FLAT=0 распределение в кластерах нормальное. Чтобы сделать распределение более туповершинным, укажите одно из следующего:

*Число от 0 до 1* - степень туповершинности, одинаковая для всех кластеров. Чем ближе число к 1, тем сильнее туповершинность. При FLAT=1 распределение достигает сферического равномерного, т.е. кластеры это гипершары.

RANDOM - степень туповершинности, случайно варьирующая от кластера к кластеру. Параметр от 0 до 1 макрос задаст для каждого кластера сам, как случайное число.

Внутрикластерная многомерная дисперсия в уплощенном кластере будет приблизительно такой же, как при FLAT=0, а в большой серии симуляций кластеры, порожденные под разным значением FLAT (и равными всеми прочими условиями) можно считать имеющими одинаковую дисперсию.



1. FLAT=0 (нормальное распределение)
2. FLAT=0.5
3. FLAT=1 (равномерный шар – круг)

**ELONG**

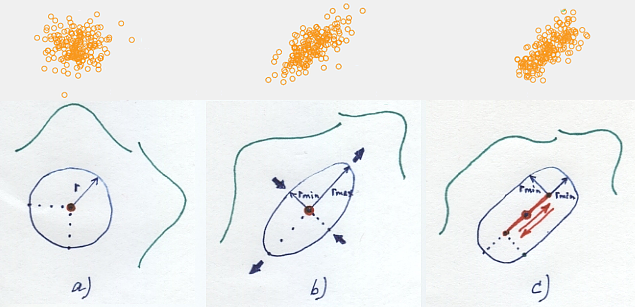
По умолчанию/незаданию этой подкоманды кластеры создаются круглыми, сферическими. ELONG (подкоманда дозволена, если *p*>1) позволяет сделать кластеры продолговатыми. Кластеры получают продолговатость только по какому-то одному направлению в пространстве; по всем остальным направлениям они остаются сферическими. Иными словами, каждый кластер представляет собой, в идеале, облако с единственной доминирующей главной осью. Укажите нужный вид продолговатых кластеров:

STRETCH *min max* - вытянутые кластеры. Такой кластер получается из круглого вытягиванием дисперсии по одному (2-полюсному) направлению пространства и ее сжатием по прочим измерениям пространства (при том если круглый кластер был с нормальным распределением точек, то он и сохранит нормальное распределение). Плотность точек на единицу меры по одному направлению уменьшена, а по остальным направдениям увеличена.

EXTEND *min max* - продленные кластеры. Такой кластер имеет туповершинное распределение по направлению своей продолговатости и нормальное распределение по прочим, перпендикулярным, направлениям. Кластер более компактный, плотный, чем вытянутый кластер. Продленный кластер порождается так же, как круглый, только в ходе порождения точек центр все время случайным образом бегает по отрезку одной прямой, а не стоит на месте.

Параметры *min* и *max* – это два числа в диапазоне от 1 до 4. Они определяют степень продолговатости кластеров – чем больше число, тем продолговатее кластер (1 эквивалентно круглому кластеру). Если вы укажете два разных числа, кластеры будут иметь неодинаковую степень продолговатости, причем для каждого кластера степень продолговатости будет случайной величиной между min и max. Если укажете два одинаковых числа – тогда все кластеры будут иметь одинаковую степень продолговатости. Неважно, в каком порядке указывать числа – min max или max min.

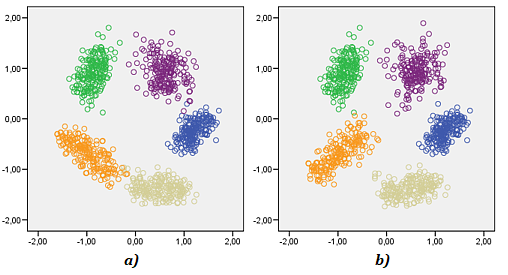
Продолговатый кластер макрос порождает приблизительно с той же величиной многомерной суммарной дисперсии, какая имеется у этого кластера, когда он круглый, если обстоятельства порождения те же самые (т.е. когда все прочие спецификации, кроме п/к ELONG, в макросе те же и зерно случайных чисел пользователь задал то же). Три вида кластеров сравниваются на рисунке ниже.



1. Круглый нормально распределенный кластер. Точки кластера порождаются под стандартным отклонением r по всем направлениям.
2. Вытянутый кластер. Подобен круглому кластеру, но вытянут по одному 2-полюсному направлению и сжат по второму направлению, т.е. точки кластера порождаются под ст. отклонением rmax и rmin, соответственно.
3. Продленный кластер. При порождении точек имеет подвижный центр, находящийся в случайном месте на отрезке прямой вдоль одного 2-полюсного направления. Точки кластера порождаются под ст. отклонением rmin во все стороны, аналогично тому как в круглом кластере.

**ROTAT**

Эта подкоманда действует, только если задана п/к ELONG. Продолговатые кластеры продолговаты вдоль одной случайной ориентации в пространстве; или, другими словами, они случайно повернуты своей 1-й главной осью в пространстве. Если вы хотите, чтобы этот поворот был совершенно произвольным, выберите ROTAT=FREE (тж. по умолчанию/незаданию п/к). Тогда разные кластеры могут оказаться повернуты независимо друг от друга и как угодно. Если вы укажете ROTAT=RESTR, тогда 1-е главные оси всех кластеров будут в той или иной мере сонаправленны, и все кластеры будут представлять собой в той или иной мере положительно коррелированные облака (см. рисунок). Если же вы отказываетесь от поворота вообще, укажите ROTAT=NONE. Первая главная ось всех кластеров будет параллельна оси переменной *V1*. (Вы можете совершить поворот позднее макросом !KO\_ROTCLU.)



1. ROTAT=FREE. Кластеры продолговаты в произвольном направлении.
2. ROTAT=RESTR. Все кластеры в той или иной степени сонаправлены.

**CLEAR**

Кластеры всегда порождаются сначала как нечеткие (fuzzy) – это кластеры, которые могут пересекаться, взаимонакладываться, если они достаточно тесны друг к другу. П/к CLEAR позволяет вам сделать такие кластеры более четкими (clear), разграничив их в области пересечения. При этом некоторые точки могут поменять свое кластерное членство. Выберите метод:

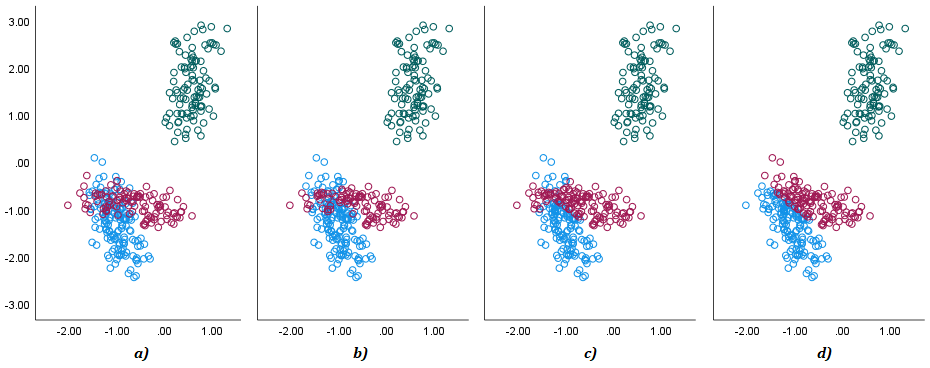
KNN - классификация методом K-ближайших соседей. Для каждой точки определяются ее ближайшие соседи (по евклидовым расстояниям), и их кластерная принадлежность спрашивается. Точка будет приписана к кластеру, который преобладает среди упомянутых соседей. Этот метод размежевывает кластеры щадяще, неполно. Используется SPSS команда KNN.

LDA - классификация методом дискриминант. Делается линейный дискриминантный анализ с экстракцией из данных всех дискриминант и затем классификацией точек ими под диктовку их раздельных ковариационных матриц. Иными словами, это есть приписание точек к ближайшим для них кластерам согласно махаланобисовым расстояниям в пространстве дискриминант. Этот метод сильно размежевывает кластеры, но в случае существенно продолговатых накрест пересекающиеся кластеров последние могут сохранить факт пересечения. Используется SPSS команда DISCRIMINANT.

KMC - классификация методом К-средних. Делается кластеризация K-средних без итераций: только приписание точек к ближайшим для них центрам, в качестве которых выступают инициальные центры, под какими точки порождались. Это наиболее радикальный метод размежевания, но он может не годиться для кластеров существенно продолговатых, поскольку оперирует евклидовыми расстояниями (они не учитывают возможную ковариационную неодинаковость кластеров). Используется SPSS команда QUICK CLUSTER.

Для метода KNN вы можете задать число ближайших соседей. Добавьте подкоманду /NEIGH=*число*. По умолчанию *число* равно 3. Для метода LDA вы можете затребовать априорную вероятность, отражающую населенность кластера точками. Добавьте подкоманду /PRIORS=SIZE. По умолчанию априорная вероятность для кластеров одинакова.

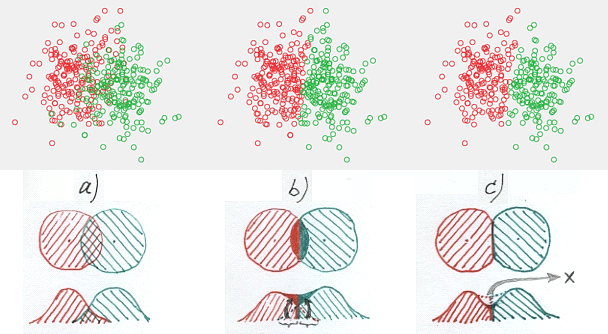
Методы LDA и KMC могут изменять форму кластеров значительно. Если два круглых кластера сильно пересекались, они, став четкими, окажутся полукруглыми. С другой стороны, если кластеры создавались продолговатыми, они при KMC иногда могут «подцепить сбоку» чужие точки и стать более круглыми (такова природа метода K-средних, который изотропен по всем направлениям и избирает оставлять после себя приблизительно круглые кластеры).



1. Нечеткие кластеры, смесь (CLEAR не задано).
2. CLEAR=KNN. Щадящее размежевание.
3. CLEAR=LDA. Сильное размежевавние, но пересечение осталось (из-за продолговатости кластеров).
4. CLEAR=KMC. Кластеры разграничены до соприлегания.

**WEED**

Подкоманда действует только при заданной подкоманде CLEAR и заказывает «прополку» пограничных зон размежеванных кластеров от «излишней» густоты точек. Укажите число от 0 до 1. WEED=0 (тж. по умолчанию/незаданию подкоманды) означает не делать прополку после размежевания кластеров. WEED=1 означает сделать полную прополку. (Значения ровно 0 или 1 указывайте без десятичных знаков, т.е. напр. указать 0.0 или 1.0 – это неправильно.) Суть отсутствия прополки и полной прополки показана на рисунке ниже. Прополка удаляет точки, изменившие свое кластерное членство из-за п/к CLEAR.



1. Пересекающиеся (нечеткие) кластеры, смесь.
2. Размежеванные (четкие) кластеры, без прополки. Точки, пришедшие в данный кластер из другого, повышают плотность кластера в пограничной зоне. Два кластера, хотя четкие, по-прежнему вместе представляют собой смешанное распределение (mixed distribution), почти лишенное бимодальности.
3. Размежеванные (четкие) кластеры, с полной прополкой. Точки, пришедшие в данный кластер из другого, все удалены. Кластеры вместе представляют собой явно бимодальное распределение.

Если вы укажете для WEED число *между* 0 и 1, будет сделана частичная прополка, при которой из каждого кластера будет удалена только часть точек, пришедших в него из других кластеров, – их доля, равная указанному значению WEED. Точки-кандидаты на удаление будут выбраны из них случайно, однако базово больший шанс быть удаленными получают те точки, которые дальше отстоят от инициального центра данного кластера в направлении инициального центра того кластера, откуда пришли.

Всякое значение WEED >0 означает, что, если какие-либо кластеры до своего размежевания пересекались, часть точек из них будет вычеркнута. В итоге, наполненность кластеров точками окажется часто меньше, чем то ожидалось согласно заданной пользователем подкоманде N. Поэтому пользователю, очевидно, имеет смысл завышать (давать «с запасом») задание для п/к N в том случае, если планируется CLEAR плюс WEED близкое к 1, а SPACING близко к 0. Иногда, при 0<WEED<1, некоторые кластеры могут в итоге иметь больше точек, чем им запланировано согласно N.

***Особые режимы***

Поскольку макрос не имеет входящих данных, ему безразличны такие вещи как взвешенность, расщепленность или фильтрованность рабочего массива данных.

***Алгоритм***

Алгоритм придуман автором макроса.

k - число кластеров; p - число переменных (размерность пространства).

Этап I. Порождение инициальных центров кластеров.

-Если заказано создать центры случайными, порождаем их как k точек из равномерного распределения.

-Если заказано создать центры примерно равноразделенными, то порождаем 100\*k\*p случайных точек, причем

половина из равномерного (прямоугольного) распределения, а половина из нормального распределения; оба

распределения должны иметь одни и те же параметры средней и ст. отклонения. Это облако точек подвергаем

кластеризации K-СРЕДНИХ с дефолтными настройками и требованием выделить k кластеров. Центры последних

и есть искомые инициальные примерно равнораспределенные центры. (Создание смеси точек из прямоугольного

и нормального распределений потребовалось потому, что кластеризация только лишь квадратного облака

имеет тенденцию дать центры кластеров, расположенные более или менее крестообразно, а кластеризация

только лишь круглого, вроде нормального, облака имеет тенденцию дать центры, расположеннные

кольцеобразно. Нам желательно уравновесить эти две тенденции.)

Этап II. Задание размеров кластеров и степени тесноты между соседними кластерами.

r - размер, или "радиус" кластера; это величина ст. отклонения в нем.

m - отношение r\_max/r\_min (т.е. отношение r между самым большим и самым малым из порождаемых кластеров),

Значение m задается один раз пользователем либо генерируется случайно, в диапазоне от 1 до 3.

q - теснота соприлегания между двумя соседними кластерами, именно, между каждыми двумя кластерами, один

из к-рых является ближайшим соседом второго. q измеряется в количестве r, укладывающихся между

центрами 2-х кластеров. Желательное (не во всех соседних кластерах в итоге выдерживаемое) значение q

задается, в диапазоне от 0.4 до 8.4, либо единожды пользователем, либо заново для каждых двух

соседних кластеров случайным порождением.

Цель этапа - вычислить такие r кластерам, к-рые учитывали бы заданные параметры m и q.

1. Вычислим евклидовы расстояния между всеми k инициальных центров.

2. Для каждого центра, найдем его ближайшего соседа, т.е. другой центр, отделенный от него наименьшим

расстоянием (расстояние до ближайшего соседа назовем d). Создадим табличку: номер центра п/п, номер

его ближ. соседа, d; вот пример таблички:

Центр Бл.Сосед D

1 4 4.56

2 6 2.97

3 1 5.48

... ... ...

k 2 6.55

3. Вычислим среднегеометрическую d (назовем ее d\_mean).

4. -Если m=1, т.е. все кластеры должны быть одного размера, то их r= d\_mean/q. Конец этапа.

-Если m>1, то r\_min= 2\*d\_mean/[q\*(m+1)] и r\_max= r\_min\*m. (r\_min и r\_max, посчитанные впервые,

запомним как r\_min1 и r\_max1.) Отдадим r\_max центру, у к-рого d максимально (наибольшее значение

в столбце D таблички). Отдадим r\_min центру, у к-рого d минимально (наименшее значение в столбце D).

(Таких центров, с мин. d, в действительности будет два, - неважно, какому из них отдать r\_min.)

5. Пробегаем список центров (1-й столбец таблички). Если центр i еще не имеет r, а центр-его ближайший

сосед (j) имеет, вычислим r для центра i: r(i)= 2\*d/q - r(j), где d - расст. между i и его ближ.

соседом j. Вычисленное r(i) нельзя выпустить за пределы от r\_min1 до r\_max1, поэтому, если

r(i)<r\_min1, то пусть r(i)= r\_min1; если r(i)>r\_max1, то пусть r(i)= r\_max1.

6. Повторяем пробежки (п. 5) и вычисление r для центров, пока не обнаружится, что после очередной

пробежки число центров с вычисленным r не изменилось.

-Если число центров с r =k, конец этапа.

-Если число центров с r <k, тогда удалим из таблички все ряды (т.е. центры), для которых r вычислен,

и идем опять к п. 3. В п. 4, при вычислении новых r\_min и r\_max надо будет учесть, что они не имеют

права выйти за пределы от r\_min1 до r\_max1, поэтому если r\_min<r\_min1, пусть r\_min= r\_min1;

если r\_max>r\_max1, пусть r\_max= r\_max1.

Этап III. Порождение кластеров.

-Порождаем каждый кластер как случ. данные из стандартного нормального распредедения. Если кластер должен

быть более туповершинным, чем нормальный, умножаем координаты каждой точки на коэффициент сжатия, равный

1 + flat \* [CDF.CHISQUARE(D,df=p)^(1/p) \* sqrt(p+2) / sqrt(D) - 1], где D – квадратное отклонение точки

от центра кластера; flat – параметр (от 0 до 1) заданный пользователем или выбранный случайно.

-Если кластеры должны быть круглыми, то:

1. Порождаем каждый кластер как случ. данные из стандартного нормального распредедения.

-Если кластер должен быть более туповершинным, чем нормальный, умножаем координаты каждой точки на

коэффициент сжатия, равный 1 + flat \* [CDF.CHISQUARE(D,df=p)^(1/p) \* sqrt(p+2) / sqrt(D) - 1], где

D – квадратное отклонение точки от центра кластера; flat – параметр (от 0 до 1) заданный

пользователем или выбранный случайно.

2. Придаем кластеру ст. отклонение (умножив его данные на r) и центр (прибавляем координаты инициального

центра.

-Если кластеры должны быть вытянутыми в одном случайном направлении, то:

1. Делаем то же, что пункт 1 для круглых кластеров.

2. Для каждого кластера имеем (задано или порождаем) коэф-т продолговатости u (число от 1 до 4) и

порождаем матрицу случайного ортогонального поворота rm. Последняя есть ортобазис Грама-Шмидта

матрицы размером pxp, состоящей из случ. чисел из нормального распределения

(причем если ROTAT=RESTR, 1-й столбец этой матрицы должен содержать только положит. числа,

а полученный из матрицы ортобазис rm транспонируем).

3. Придаем кластеру ст. отклонение, равное r\_уменьшенное\*u, для первой переменной (V1), и равное

r\_уменьшенное, для остальных переменных; r\_уменьшенное= sqrt[p\*r^2/(p-1+u^2)].

4. Данные каждого кластера поворачиваем в пространстве, умножая на заготовленную для них случайную

матрицу поворота rm (пункт 2).

-Если кластеры должны быть продленными в одном случайном направлении, то:

1. Для каждого кластера имеем (задано или порождаем) коэф-т продолговатости u (число от 1 до 4) и

порождаем координаты некоторой точки ("направляющая точка"), отклоняющейся от центра кластера

в случайном направлении пространства (причем если ROTAT=RESTR, то все ее координаты

делаем положительными).

2. Требуемое расстояние dcd направляющей точки до центра вычислим как (e - основание нат. логарифма):

dcd= e^(.4+1.42/u)\*r\_уменьшенное\*(u-1), где r\_уменьшенное= см. ф-лу выше. Пропорционально изменим

координаты точки так, чтобы привести ее отстояние от центра кластера к величине dcd.

3. Порождаем каждый кластер как случ. данные из нормального распределения с параметрами: средняя=

координата бегающего центра; ст. отклонение= r\_уменьшенное. Бегающий центр это точка, находящаяся

в случайном месте на отрезке от направляющей точки до точки, зеркально симметричной ей относительно

инициального центра кластера. Бегающий центр изменяем при порождении каждого очередного данного.

Если кластеры заказано создать нечеткими - конец работы.

(Нечеткие кластеры могут перекрываться своими перифериями, если кластеры достаточно тесны друг к другу.)

Этап IV. Делание кластеров четкими и прополка их пограничных зон.

Этот этап может быть излишен, если кластеры заказано создать широко раздвинутыми, т.к. тогда они

наверняка не перекрываются.

1. Дадим (i) кластерному анализу K-средних (команда QUICK CLUSTER) либо

(ii) линейному дискриминантному анализу (команда DISCRIMINANT) либо (iii) анализу K-ближайших соседей

(команда KNN) отклассифицировать порожденные точки (наблюдения) к тем кластерам, к которым они ближе.

(Примечание 1: QUICK CLUSTER делается в режиме Classify only, т.е. без итераций, и инициальные

кластерные центры ждя кдассификации – это инициальные центры порождения точек.

Примечание 2: DISCRIMINANT пускается с п/к CLASSIFY=SEPARATE, т.е. классифицируем без допущения, что

ковариации в кластерах сходны, и классификация делается каноническими дискриминантами.) Если прополка

пограничных зон (участков, где кластеры пересекались) не требуется – конец работы.

2. Прополка пограничных зон означает удаление части порожденных наблюдений: численность в кластерах

будет меньше, чем заказано пользователем в подкоманде N.

-Если требуется полная прополка, то удаляются все точки, кластерная принадлежность к-рых изменилась

при классификации в п. 1.

-Если требуется частичная прополка (требуется удалить из кластера долю 0<weed<1 точек, пришедших в

него из других кластеров в результате классификации п. 1):

2.1. Для каждой точки, пришедшей в "наш" кластер из данного "чужого" кластера, вычислим расстояние

точки до инициального центра нашего кластера (d1), расстояние ее до инициального центра чужого

кластера (d2), а также расстояние между инициальными центрами нашего и чужого кластеров (d3).

2.2. Имея эти 3 дистанции, вычислим длину проекции точки на линию, соединяющую центры нашего и

чужого кластеров: (d1^2+d3^2-d2^2)/sqrt(d3^2)/2. Умножим эту величину на случайное положит.

число. Проделаем описанное для всех точек, пришедших в наш кластер из данного чужого кластера

(случайное число каждый раз порождая новое).

2.3. Удалим из всех точек, пришедших в наш кластер из данного чужого, долю=weed точек с наибольшими

получившимися таким образом значениями.

Т.е. мы удалили из кластера нужную долю (weed) точек, бывших прежде "чужими", выбрав из них

случайно, однако с тем, чтобы точки, лежащие дальше от центра нашего кластера в направлении к

центру своего бывшего кластера, имели базово больший шанс быть удаленными.

# МАКРОС !KO\_ROTCLU: СЛУЧАЙНЫЙ ПОВОРОТ (КЛАСТЕРНЫХ) ДАННЫХ

Version 2, Dec 2010 (Version 1, Nov 2010). Tested on SPSS Statistics 13, 15, 17.

!KO\_rotclu vars= *v1 v2* /\*Переменные, представляющие координаты точек; минимум две, поименно и/или ч-з to

/rotat= FREE\_PAR /\*Требуемый поворот: на любой угол, свой у каждого кластера (FREE, тж п/у);

/\*на любой угол, один на все кластеры (FREE\_ANG);

/\*на любой угол, делающий кластеры параллельными своими гл осями (FREE\_PAR);

/\*в положит-й и отрицат-й квадранты пространства, угол свой у каждого кластера (RESTR);

/\*в положит-ый и отрицат-ый квадранты пространства, кластеры параллельными гл осями (RESTR\_PAR);

/\*устранить повернутость (UNROT), это поворот в гл компоненты

/pc= DIRECT /\*Особенность предварит-го поворота в гл компоненты: разрешить противонаправленность

/\*гл осей (ORIENT, тж п/у); разрешить только сонаправленность (DIRECT).

Минимум надо задать VARS.

Макрос берет данные, состоящие из одной или более групп, или «кластеров», наблюдений (это могут быть действительно кластеры или совершенно иначе сформированные группы) и поворачивает каждый кластер в пространстве осей-переменных вокруг своего центроида (многомерной средней). Поворот – на случайный угол, зависящий от посеянного зерна случайных чисел. Есть также возможности устранить повернутость, придать кластерам одинаковую повернутость или сонаправленность, и т.п.

Входящими являются многомерные данные рабочего массива, в котором должна присутствовать группирующая числовая переменная *CLUSTER\_* с кодами групп (кластеров). Коды могут быть любыми числами. Массив *должен быть сортирован* по возрастанию значений этой переменной. Переменная может быть и константой (что значит что ваши данные состоят из одного кластера). Макрос выводит повернутые данные как новый рабочий массив данных. Во входящих данных (включая переменную *CLUSTER\_*) недопустимы пропуски.

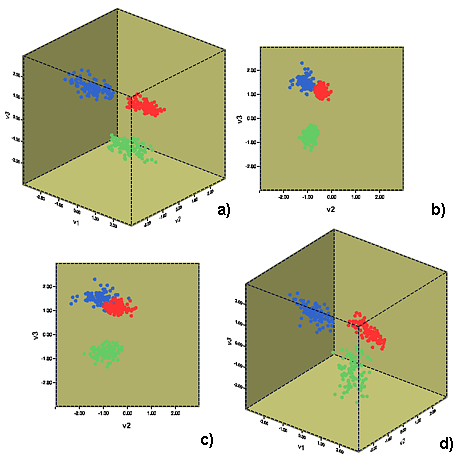
ПРИМЕР 1. Создание эллипсоидных повернутых кластеров. Макрос !KO\_GENCLU (см. выше) может порождать продолговатые кластеры, однако это сфероидные кластеры: они продолговаты только в одном направлении многомерного пространства, оставаясь в остальном сферическими. Другими словами, в каждый кластер заложена исходно однофакторная модель (в терминологии факторного анализа), сколько бы переменных ни было создано. Эллипсоидные (следовательно многофакторные) облака-кластеры можно сделать, модифицировав вручную кластеры, порожденные в !KO\_GENCLU, а затем случайно повернув их в пространстве макросом !KO\_ROTCLU.

!KO\_genclu k= 3 /p= 3 /n= 100 /spacing= .8 /sizing= .4 /elong= EXTEND 3 3 /rotat= NONE.

aggregate /outfile= \* mode= addvariables /break= cluster\_ /v2\_mean= mean(v2).

compute v2= (v2-v2\_mean)\*2+v2\_mean.

!KO\_rotclu vars= v1 v2 v3.



* !KO\_GENCLU породил 3 трехмерных кластера. Кластеры в данном случае продолговатые (по типу продленных). Они продолговаты все параллельно друг другу и оси V1, поскольку поворот не заказывался (ROTAT=NONE), - см. рис. a. Кластеры сфероидные: в проекции V2-V3 они круглые (рис. b).
* Исследователь решил увеличить в 2 раза стандартное отклонение во всех кластерах по переменной V2. Для этого он выяснил средние кластеров по этой переменной (команда AGGREGATE), перевел ее данные в отклонения от средний, умножил на 2 и вернул среднюю обратно (команда COMPUTE). Теперь кластеры эллипсоидные: они вытянуты не только вдоль V1, но немного и вдоль V2 (рис. c).
* Наконец, исследователь придал эллипсоидным кластерам случайную повернутость в пространстве макросом !KO\_ROTCLU (рис. d).

***Подкоманды***

**VARS**

Укажите минимум две числовые переменные, являющие собой собственно данные – координаты наблюдений. Список можно писать поименно и/или ч-з to.

**ROTAT**

Задайте особенности поворота. Следующие три варианта не накладывают ограничения на итоговую ориентацию кластеров в пространстве: угол их повернутости может оказаться любым случайным.

FREE - (тж. по умолчанию/незаданию) Кластеры поворачиваются независимо друг от друга, каждый кластер на свой угол.

FREE\_ANG - Поворот всех кластеров на один и тот же угол. Кластеры, следовательно, сохраняют исходную повернутость относительно *друг друга*.

FREE\_PAR - Поворот кластеров, который делает их облака параллельными друг другу: 1-е главные оси кластеров параллельны, 2-е главные оси их параллельны, и т.д.

Следующие два варианта тоже делают случайный поворот, но накладывают такое ограничение на итоговую повернутость, чтобы первая главная ось каждого кластера была ориентирована непременно между «углом чисто отрицательных» и «углом чисто положительных» координат пространства. Если кластеры продолговатые, они, стало быть, все станут положительно коррелированными облаками.

RESTR - Кластеры поворачиваются независимо друг от друга, каждый кластер на свой угол.

RESTR\_PAR - Поворот кластеров, который делает их облака параллельными друг другу: 1-е главные оси кластеров параллельны, 2-е главные оси их параллельны, и т.д.

Следующий вариант отменяют повернутость входящих кластеров. Это замена данных на их главные компоненты: 1-я главная ось каждого кластера станет параллельна 1-й оси пространства (т.е. 1-й переменной), 2-я главная ось – 2-й оси пространства (2-й переменной), и т.д.

UNROT - Поворот кластеров, который делает их главные оси параллельными друг другу и параллельными осям пространства.

Если данные состоят из единственного кластера, то FREE и FREE\_ANG дадут один и тот же результат, RESTR и RESTR\_PAR дадут один и тот же результат (при одном итом же зерне случайных чисел, разумеется).

**PC**

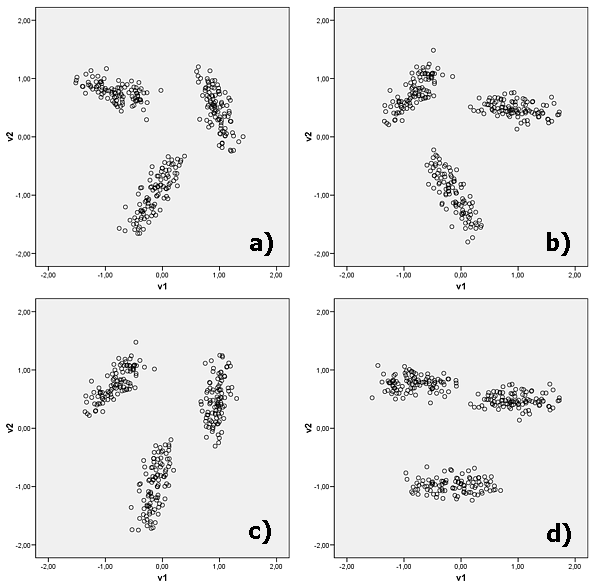
Подкоманда не действует при ROTAT=FREE или ROTAT=FREE\_ANG. В остальных случаях перед приданием кластерам окончательной повернутости с каждым из них проводится анализ главных компонент: исходные данные заменяются на главные компоненты. П/к PC задает нюансы этого.

ORIENT - (тж по умолчанию/незаданию) Углы поворота переменных (осей пространства) в главные оси облака могут быть от 0° до 180°. Если угол поворота n-й переменной в n-ю главную ось у одного кластера окажется меньше 90°, а у другого больше 90°, значит кластеры станут смотреть в одну сторону исходно противоположными краями. Словом, соответственные главные оси кластеров ориентируются параллельно, но не обязательно сонаправлено.

DIRECT - Углы поворота переменных в главные оси облака могут быть от 0° до 90°, для всех кластеров, поэтому кластеры не смогут повернуться друг к другу исходно противоположными краями. Словом, соответственные главные оси кластеров не просто параллельны, но сонаправлены.

Сочетание ROTAT=UNROT с PC=DIRECT – единственное задание, результаты при котором не зависят от случайных чисел.

На рисунках ниже – иллюстрация некоторых поворотов.



* a) Входящие данные, т.е. до поворота макросом.
* b) ROTAT=FREE\_ANG. Поворот всех кластеров на один и тот же случайный угол (здесь примерно на 50° против часовой стрелки).
* c) ROTAT=RESTR + PC=DIRECT. Ограниченный поворот: все кластеры ориентированы своей 1-й главной осью между левым нижним и правым верхним углами пространства. Обратите внимание, что те полюса всех кластеров, которые смотрели во входящих данных вправо, все по-прежнему смотрят вправо (т.к. PC=DIRECT).
* d) ROTAT=UNROT + PC=ORIENT. Устранение повернутости кластеров: их главные оси параллельны друг другу и осям пространства. Заметьте, что по сравнению с входящими данными не все кластеры сонаправлены, т.е. развернуты в одну сторону; левый верхний кластер повернулся более чем на 90°, тогда как другие – менее чем на 90°; это возможно при PC=ORIENT.

***Особые режимы***

Не используйте расщепление массива данных (SPLIT FILE) при работе с макросом. Макрос слушается фильтрации или отбора наблюдений (FILTER, SELECT IF, USE). Взвешивания не слушается.