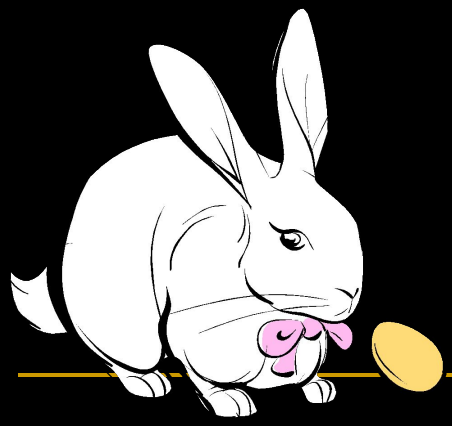
Краткий обзор дискриминантного, факторного, кластерного анализов.



Дискриминантный анализ

У нас есть зверьки разного возраста, у которых измеряли 20 показателей. По каким из них лучше всего определяется возраст?

Собирали данные про школьников 11-го класса (20 разнокачественных переменных); после этого школьники поступили в ВУЗ, колледж или вообще никуда не поступили. Какие показатели лучше всего предсказывают судьбу школьника?

Для решения таких задач создан

ДИСКРИМИНАНТНЫЙ АНАЛИЗ (discriminant function analysis)

Основная идея:

Мы измерили целый набор переменных, и у нас **ИЗНАЧАЛЬНО ЕСТЬ ГРУППЫ**.

Мы хотим понять, **чем отличаются** между собой эти группы (на основе данных переменных).

(скажем, когда мы потом измерим эти переменные у новой особи, мы сможем с известной вероятностью отнести её к той или иной группе).

Дискриминантный анализ

Суть анализа:

Очень близок ANOVA. Проверяет, отличаются ли группы на основе **СРЕДНИХ ЗНАЧЕНИЙ** переменных. (Пример про мужчин и женщин, которые высокого и низкого роста). Если в ANOVA переменная одна, мы считаем F-статистику на основе внутригрупповой и межгрупповой дисперсий. Когда переменных много (MANOVA и дискриминантный анализ) – создают матрицу дисперсий.

Строим «**Модель**» - способ определения, к какой группе относится данное измерение.
Переменные включаем в модель по одной, начиная с той, которая лучше всех разделяет группы (Forward stepwise analysis) (Backward stepwise analysis – наоборот, сначала в модели все переменные и их по одной убирают).

Дискриминантный анализ

На каждом шаге (для каждой переменной) считается статистика F, т.е. мы сравниваем группы по этой переменной.

F to enter: показывает, насколько хорошо группы отличаются по этой переменной (для Forward stepwise analysis) Можно задать минимальное значение, ниже которого переменная не будет включена в модель (когда анализ дойдёт до соответствующего шага, он остановится).

F to remove: то же самое; показывает, насколько «плохо» группы отличаются по этой переменной (для Backward stepwise analysis).

(нельзя использовать эти статистики в качестве результатов ANOVA)

Дискриминантный анализ

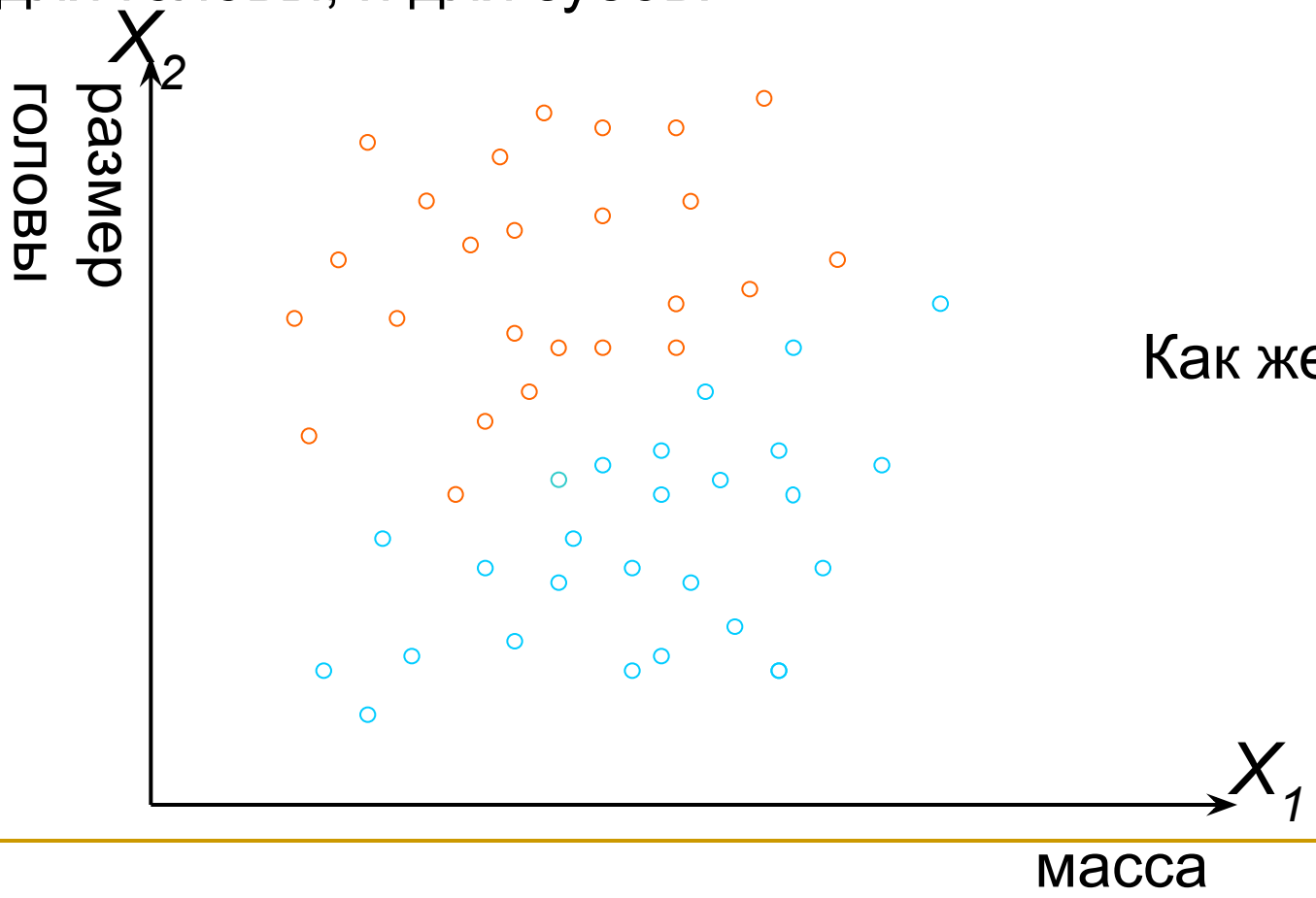
Мы изучаем лемуров на Мадагаскаре.
У нас 3 вида лемуров, мы поймали зверьков разных видов, взвесили, померили голову и зубы.

Вопрос: по какой из переменных мы лучше всего отличим виды?



Дискриминантный анализ

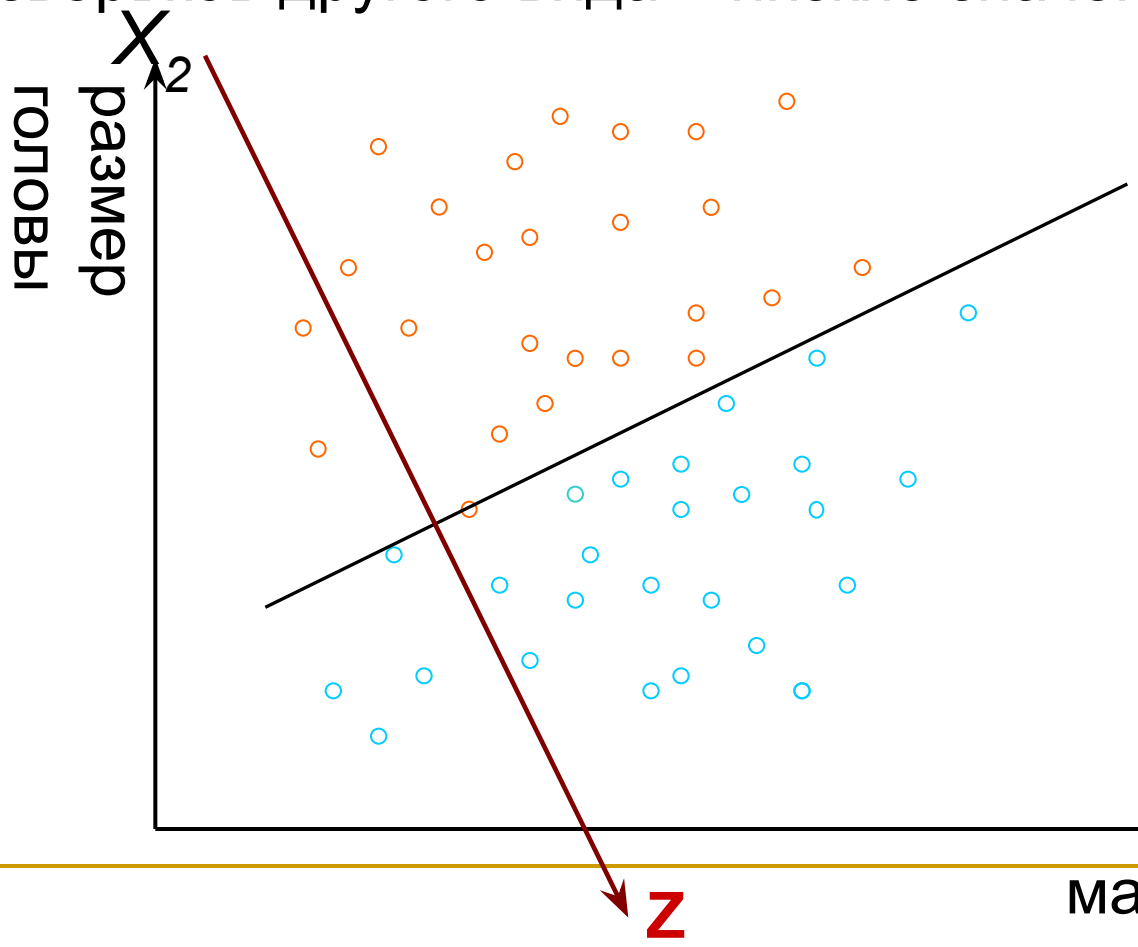
Оказалось, что, несмотря на то, что средние значения для каждой переменной у разных видов отличаются, их распределения сильно перекрываются и для массы, и для головы, и для зубов!



Как же быть?

Дискриминантный анализ

Переменная Z (дискриминантная функция) строится таким образом, чтобы как можно больше зверьков одного из видов получили высокие значения Z, и как можно больше зверьков другого вида – низкие значения Z.



Поиск такой переменной ведётся на основе ANOVA и регрессионного анализа

Дискриминантный анализ

Создание дискриминантной функции

Из выбранных нами переменных (на основе F to enter) рассчитываем новую переменную Z (дискриминантную функцию) –линейную комбинацию исходных переменных, которая наилучшим образом разделит группы (напр., виды).

Если группы две: получается одно уравнение Group = a + b_1x_1 + b_2x_2 + ... + b_mx_m.

Когда групп много, получают **несколько дискриминантных функций**, «перпендикулярных» друг другу. Чем больше коэффициент при переменной, тем лучше она разделяет группы (не говорит, какие именно).

$$Z = \lambda_1 X_1 + \lambda_2 X_2 + \lambda_3 X_3 + \ldots$$

X_i - исходные переменные

Дискриминантный анализ

Программа сама выбирает «лучшую» дискриминантную функцию и строит её первой, потом «лучшую» из оставшихся возможных, и.т.д. – всего k-1 или j-1 функций (k – число групп, j – число переменных, выбирают меньшее из этих чисел).

Выбор и построение функций осуществляется с помощью Канонического анализа (Canonical analysis) – это один из вариантов регрессионного анализа.

Коэффициенты в дискриминантной функции (b или β) соответствуют тому, какой вклад вносит данная переменная в разделение групп.

Дискриминантный анализ

Интерпретация дискриминантных (=канонических) функций:

- ✔Каждую дискриминантную функцию характеризует **Root** (канонический корень), и мы можем проверить, сколько функций в нашем анализе действительно помогает различить группы, и какую часть изменчивости они объясняют (и исключить недостоверные).
- ✔**standardized *b* coefficient** – позволяют оценить вклад каждой из переменных в различение групп данной дискриминантной функцией.
- ✔Структура факторов (**factor structure** coefficients) – позволяет понять, насколько какие переменные коррелируют с дискриминантными функциями.

Дискриминантный анализ

Теперь, когда мы построили такую функцию, мы сможем поймать зверька неизвестного вида, измерить у него X_1 и X_2 , рассчитать значение Z на основе уже посчитанных коэффициентов, и с некоторой точностью причислить его к тому или другом виду.

Дискриминантный анализ

Классификация:

Теперь можно **предсказать**, к какой группе относится та или иная особь, и оценить точность этого предсказания!
Строятся **классификационные функции** (для каждой группы), и можно для каждой особи посчитать их и отнести в ту или иную группу.
Можно провести на основе уже посчитанных функций классификацию тестовой выборки.

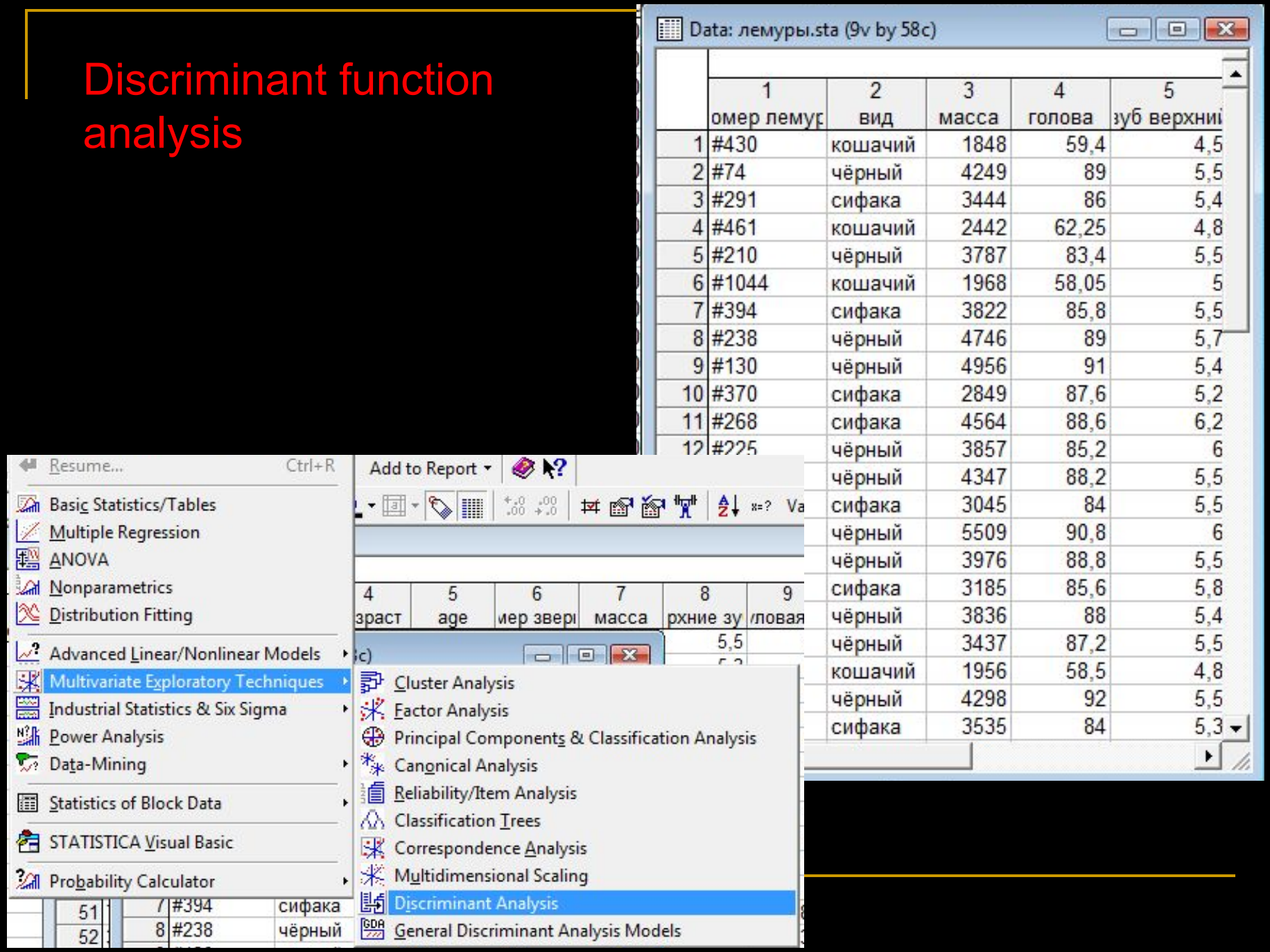
Дискриминантный анализ

Итак:

Дискриминантная функция рассчитывается только для тех измерений, для которых *известно, к какой группе они принадлежат* (т.е., только для тех особей, для которых вид известен).

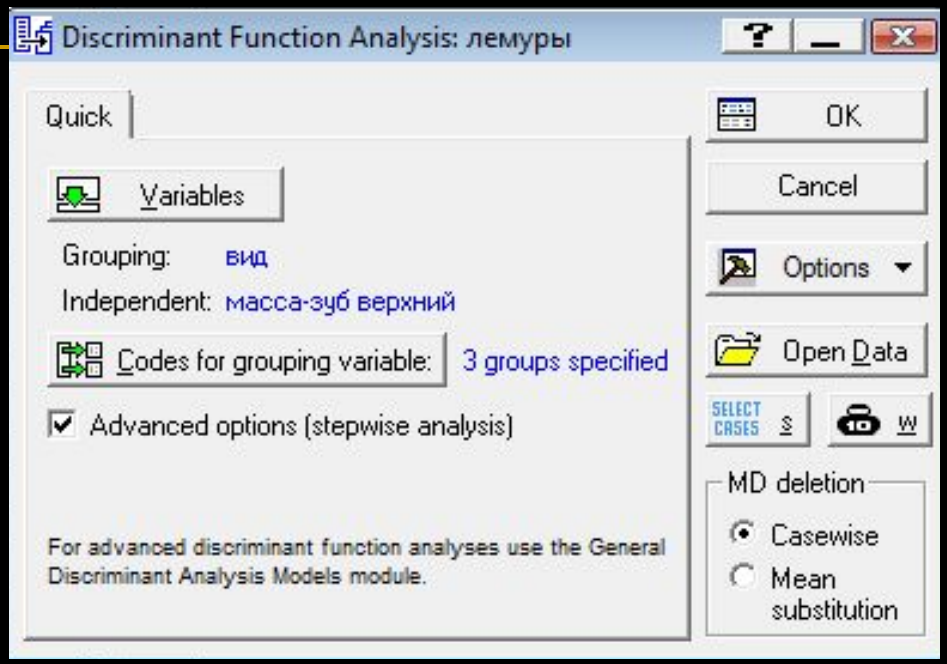
Если у нас есть набор признаков, и мы их на основе хотим создать группы (например, поделить вид на подвиды), это – задача для другого анализа! (для количественной таксономии, numerical taxonomy).

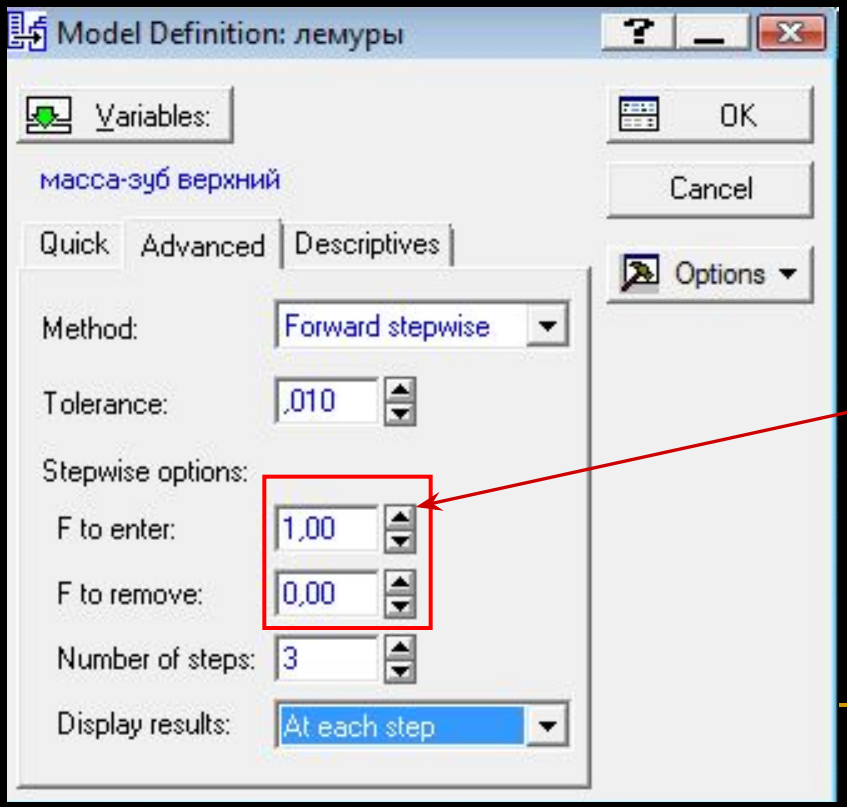
Discriminant function analysis



Ступень 1: создание модели

Выберем переменные для анализа.

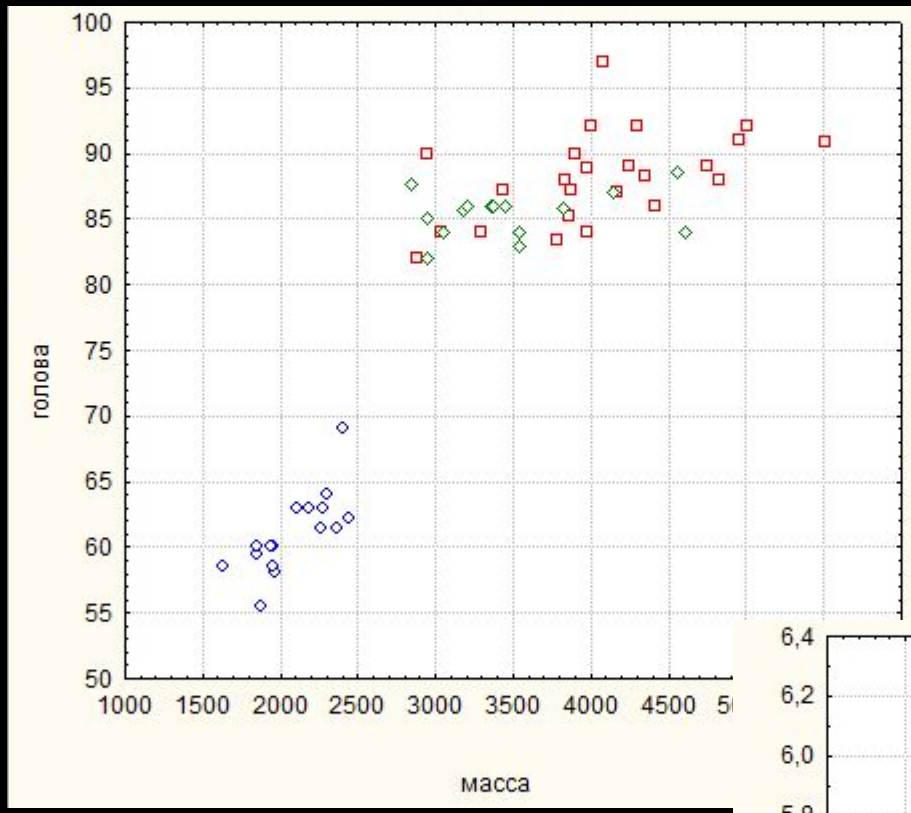




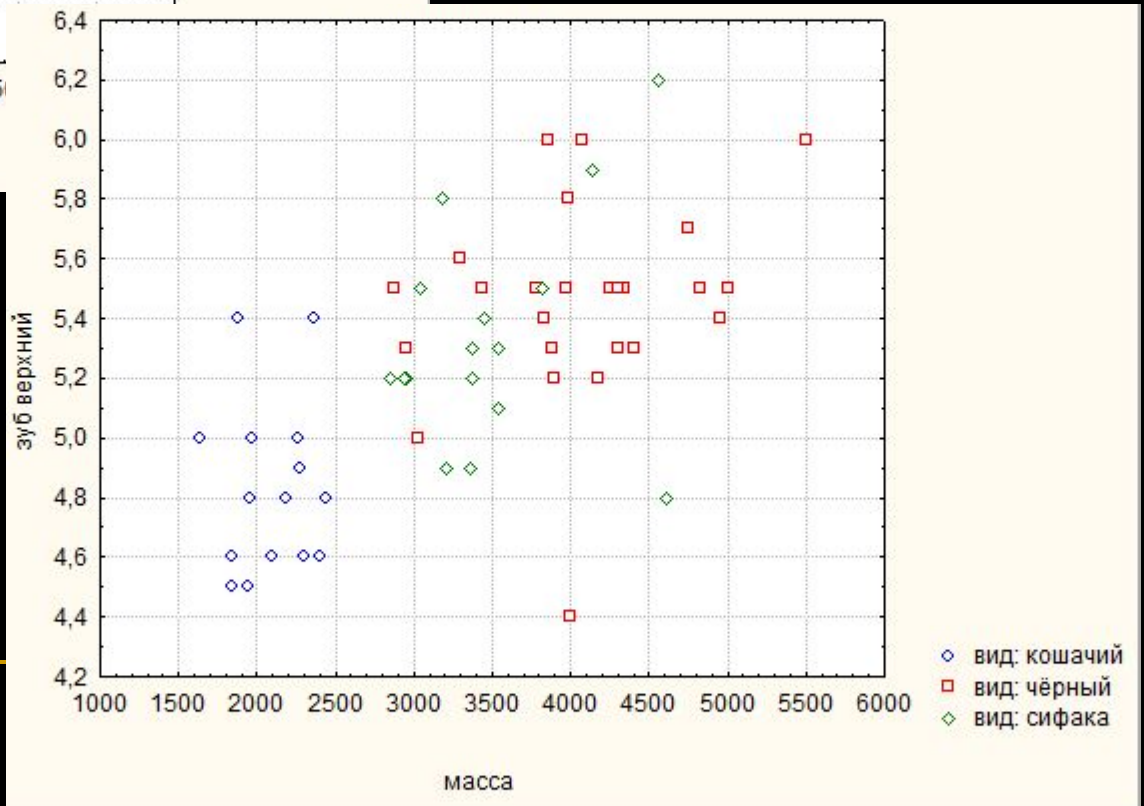
Выберем пошаговый анализ.

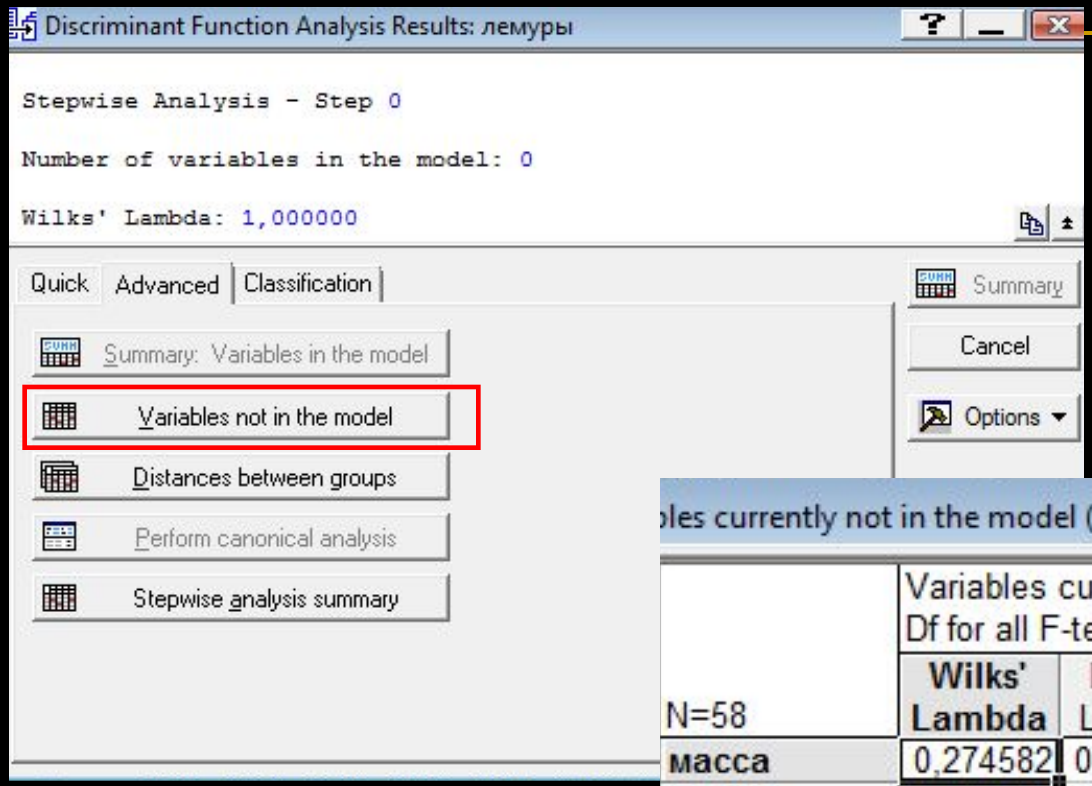
Критерии, по которым мы будем включать переменные для построения дискриминантной функции.

Толерантность – позволяет задать минимальный необходимый вклад переменной по сравнению с другими переменными, т.е., исключить избыточные переменные.



Прежде чем приступить к анализу, посмотрим, есть ли разделение на группы по нашим переменным.





Предварительный анализ переменных:

насколько по ним вообще различаются группы (на основе ANOVA)

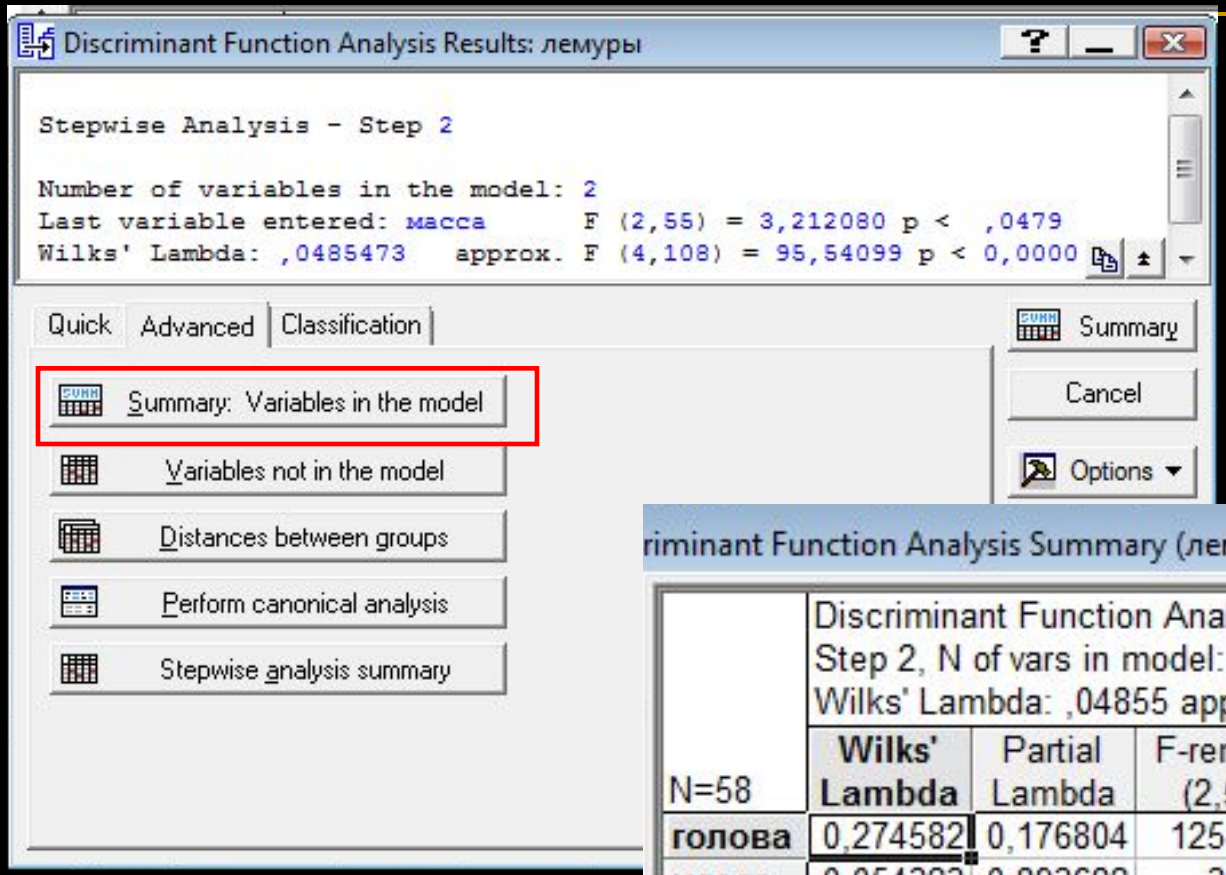
les currently not in the model (лемуры)

Variables currently not in the model (лемуры)
Df for all F-tests: 2,55

N=58	Wilks' Lambda	Partial Lambda	F to enter	p-level	Toler.	1-Toler. (R-Sqr.)
масса	0,274582	0,274582	72,6522	0,000000	1,000000	0,00
голова	0,054323	0,054323	478,7336	0,000000	1,000000	0,00
зуб верхний	0,578130	0,578130	20,0671	0,000000	1,000000	0,00

Wilk's lambda – статистика, оценивает мощность дискриминации модели после введения в неё переменной. Чем она меньше – тем больше вклад

F to enter – статистика для оценки достоверности вклада переменной в дискриминацию.



Пройдём Шаг 1 и Шаг 2. Можно посмотреть, какие переменные уже включены в анализ.

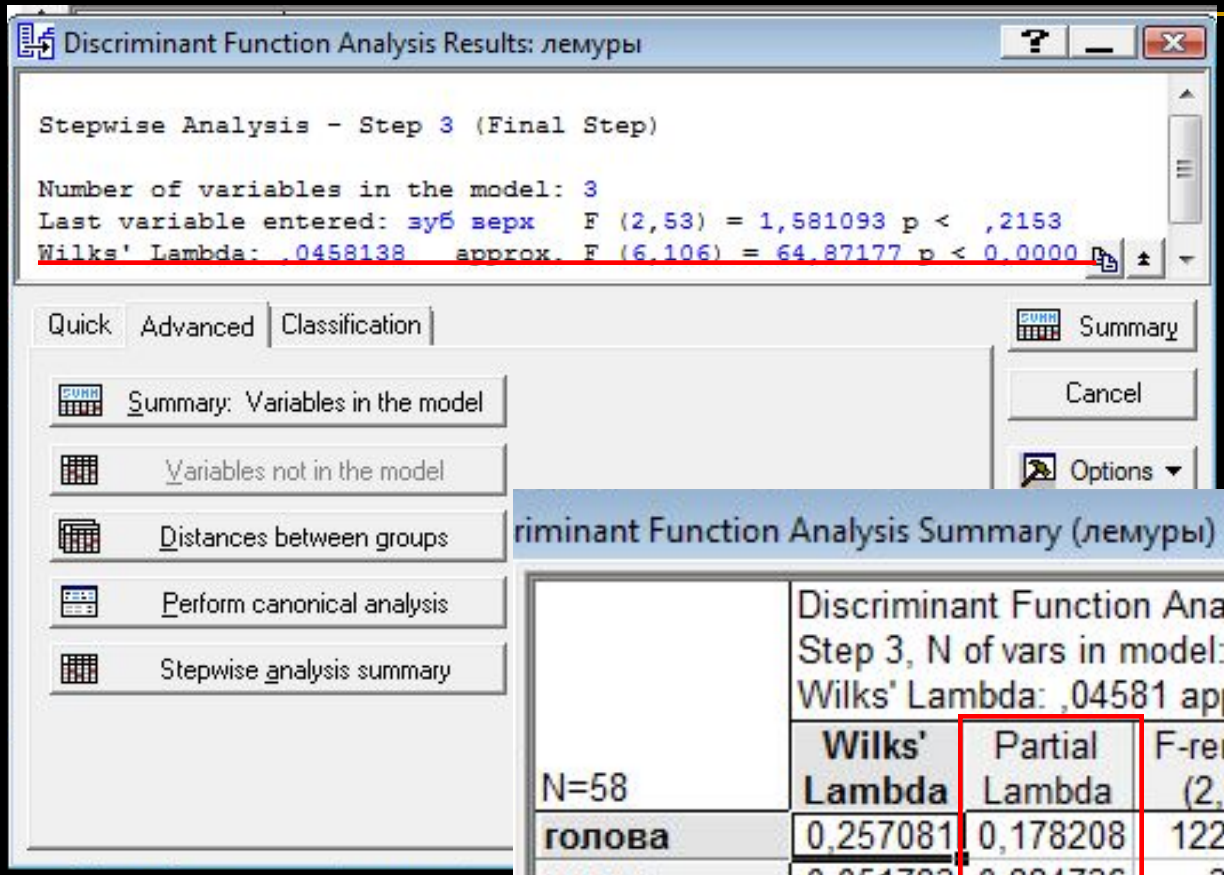
riminant Function Analysis Summary (лемуры)

Discriminant Function Analysis Summary (лемуры)
Step 2, N of vars in model: 2; Grouping: вид (3 grps)
Wilks' Lambda: ,04855 approx. F (4,108)=95,541 p<0,0000

N=58	Wilks' Lambda	Partial Lambda	F-remove (2,54)	p-level	Toler.	1-Toler. (R-Sqr.)
голова	0,274582	0,176804	125,7112	0,000000	0,792212	0,207788
масса	0,054323	0,893682	3,2121	0,048078	0,792212	0,207788

Partial lambda - статистика для вклада переменной в дискриминацию между совокупностями. Чем она меньше, тем больше вклад переменной.

Переменная Голова лучше помогает различать виды, чем Масса.



Последний Шаг 3: дискриминация между видами значима

riminant Function Analysis Summary (лемуры)

Discriminant Function Analysis Summary (лемуры)
Step 3, N of vars in model: 3; Grouping: вид (3 grps)
Wilks' Lambda: ,04581 approx. F (6,106)=64,872 p<0,0000

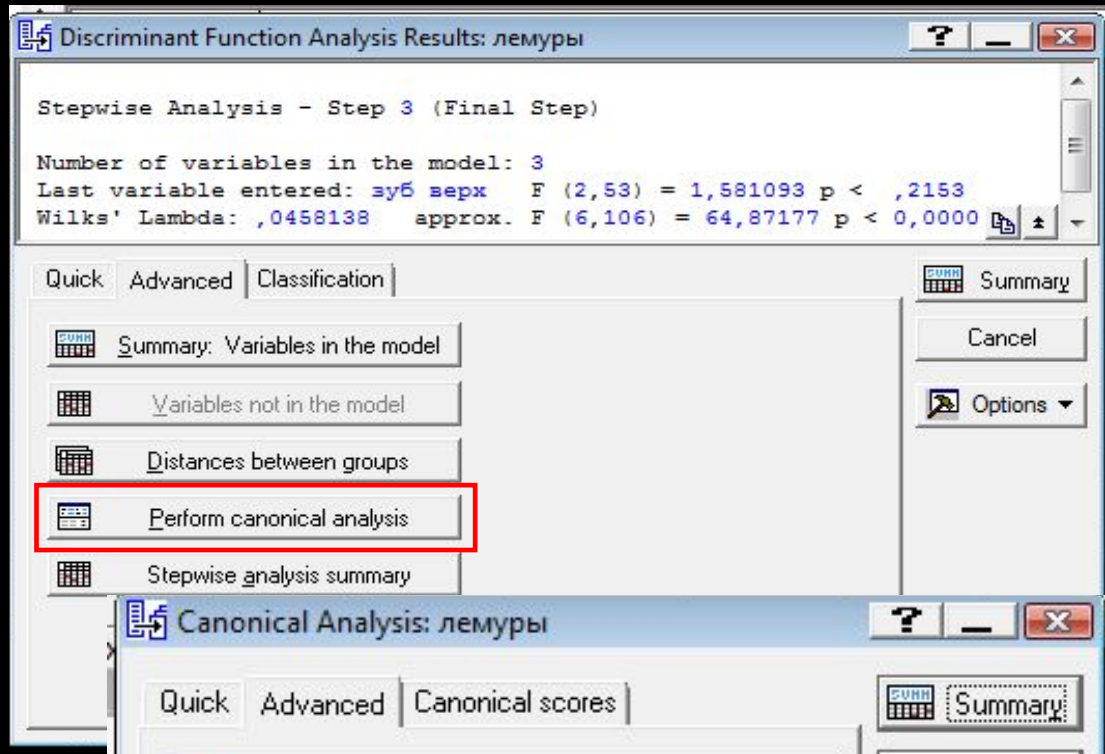
N=58	Wilks' Lambda	Partial Lambda	F-remove (2,53)	p-level	Toler.	1-Toler. (R-Sqr.)
голова	0,257081	0,178208	122,2028	0,000000	0,775488	0,224512
масса	0,051783	0,884736	3,4524	0,038955	0,728760	0,271241
зуб верхний	0,048547	0,943696	1,5811	0,215300	0,919741	0,080259

Partial lambda: Переменная Голова даёт вклад больше всех, а вклад Зуба – недостоверный.

Ступень 2: создание дискриминантной функции

Предпримем канонический анализ

Дискриминантных функций у нас 2

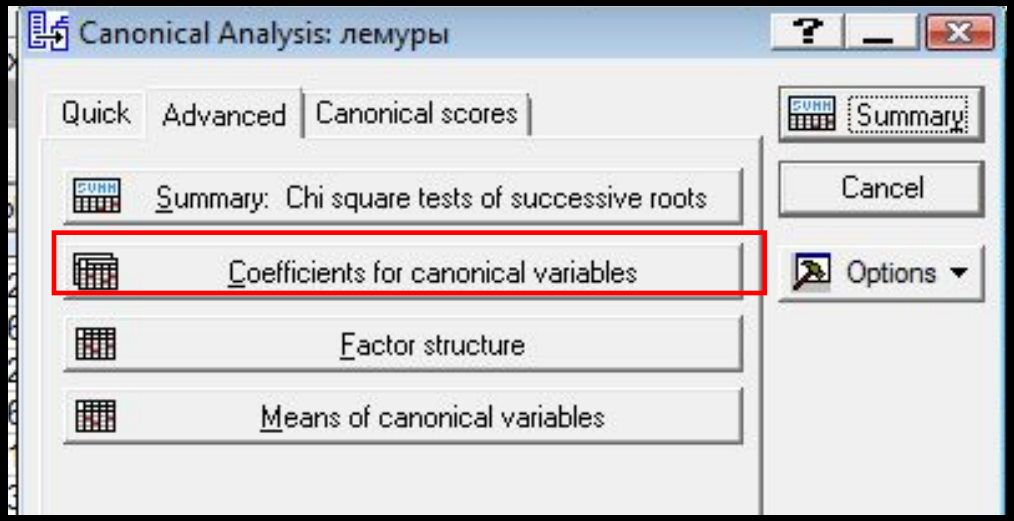


Square Tests with Successive Roots Removed (лемуры)

Roots Removed	Chi-Square Tests with Successive Roots Removed (лемуры) Eigen-value	Canonicl R	Wilks' Lambda	Chi-Sqr.	df	p-level
0	18,62227	0,974186	0,045814	166,4911	6	0,000000
1	0,11238	0,317850	0,898972	5,7512	2	0,056382

Значимой оказалась только первая функция (root)

Посмотрим, какой вклад внесли переменные в различение групп нашими дискриминантными функциями.



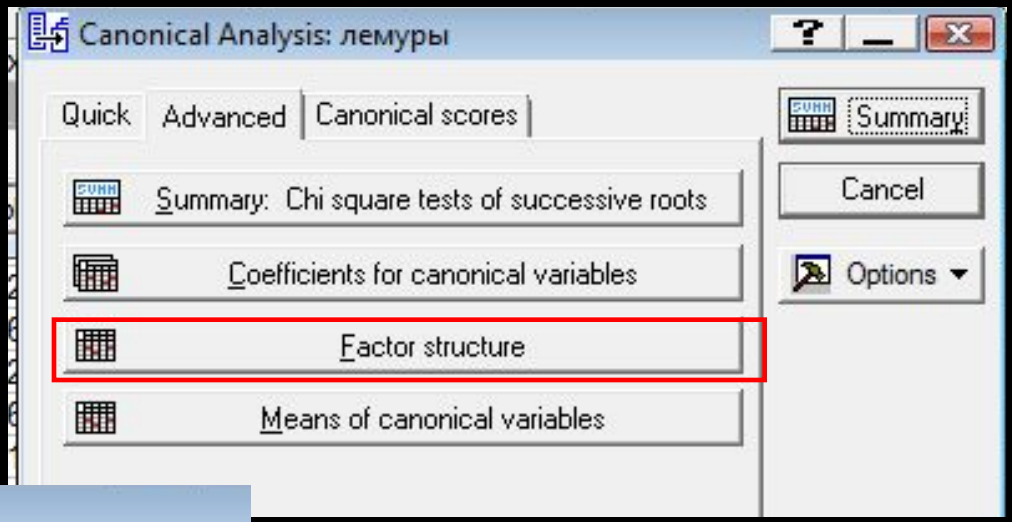
ardized Coefficients (лемуры)

Variable	Standardized Coefficients (лемуры) for Canonical Variables	
	Root 1	Root 2
голова	-1,04774	0,42091
масса	0,17011	-1,13742
зуб верхний	-0,25299	0,06861
Eigenval	18,62227	0,11238
Cum.Prop	0,99400	1,00000

Standardized coefficients – коэффициенты для сравнения значимости. «Голова» лучше всех позволяет различать группы

Первая функция объясняет 99,4% изменчивости

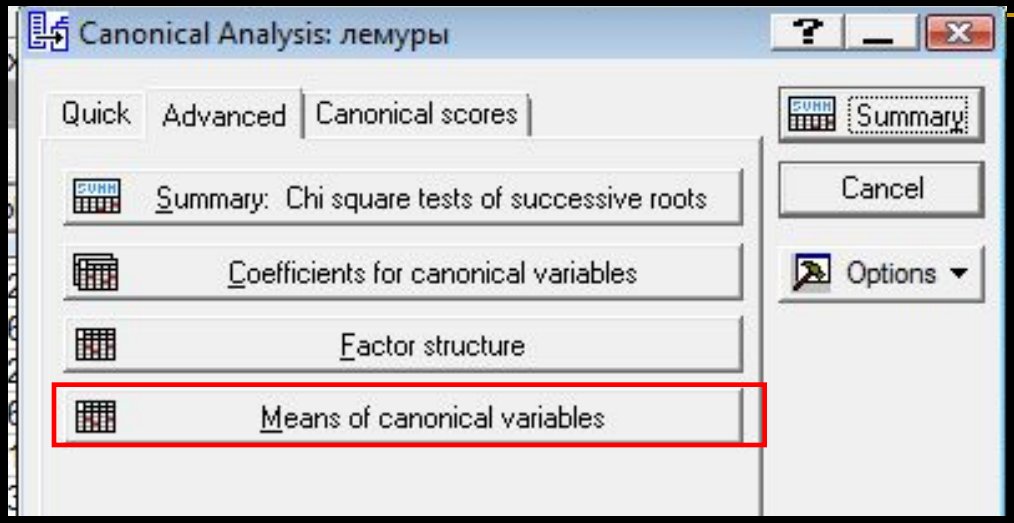
Структура факторов (дискриминантных функций)



or Structure Matrix (лемуры)

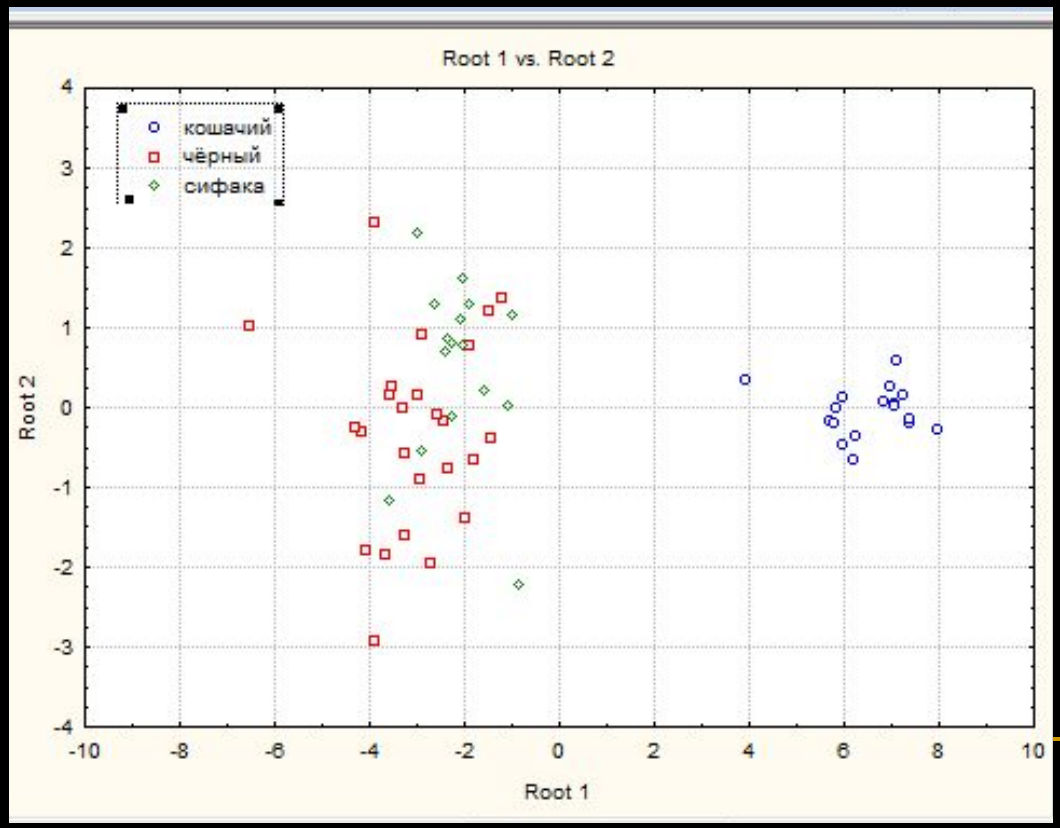
Variable	Factor Structure Matrix (лемуры) Correlations Variables - Canonical Roots (Pooled-within-groups correlations) Root 1	Root 2
голова	-0,966831	-0,098483
масса	-0,369679	-0,928690
зуб верхний	-0,197236	-0,216579

Наибольший вклад в первую функцию вносит Голова (она сильнее всего коррелирует с ней).



Means of Canonical Variables (лемуры)

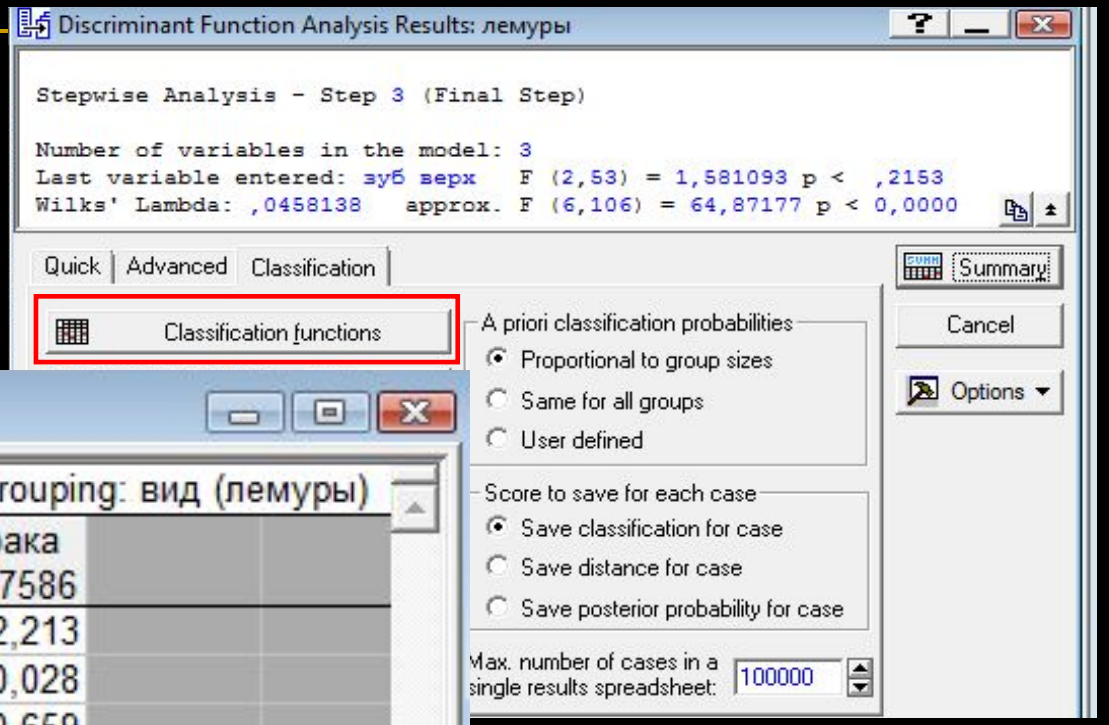
Group	Means of Canonical Variabl	
	Root 1	Root 2
кошачий	6,49871	-0,046369
чёрный	-3,06030	-0,290091
сифака	-2,12316	0,502534



Мы можем посмотреть на разницу средних значений функций между группами.

Кошачий лемур сильно отличается от других видов по значения первой функции

Ступень 3: классификация



ification Functions; grouping: вид (лемуры)

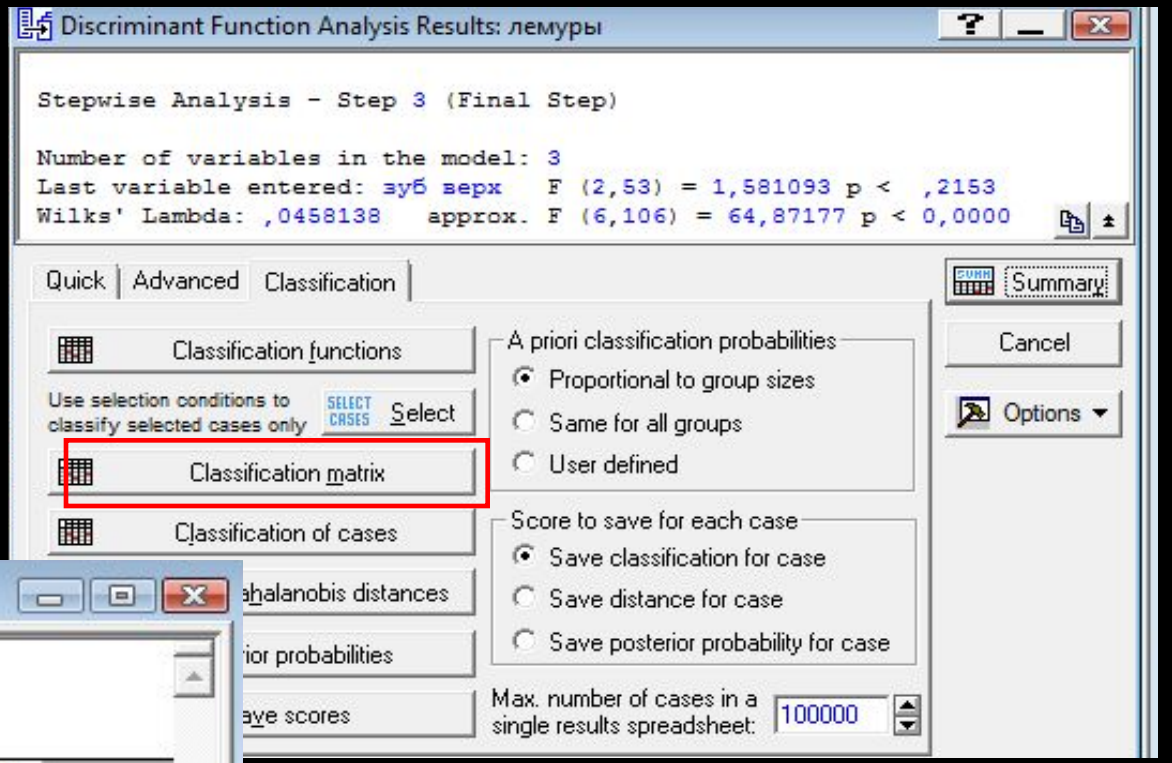
Classification Functions; grouping: вид (лемуры)

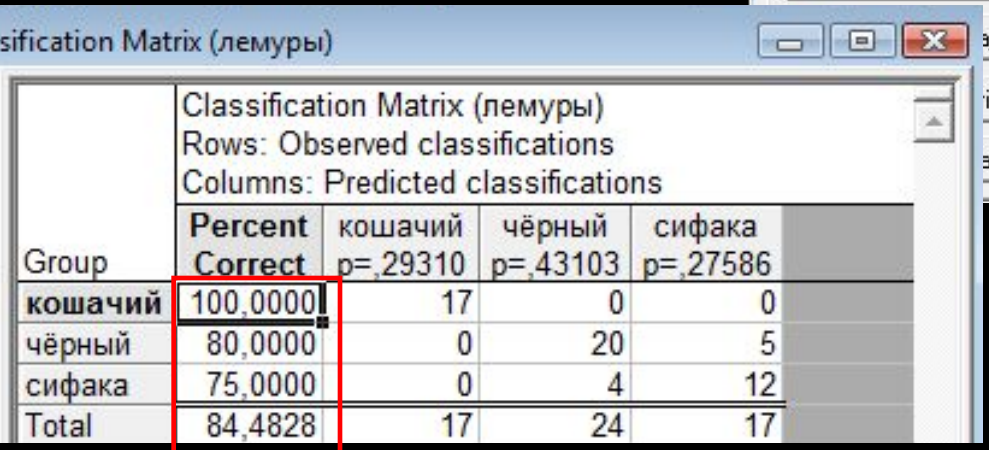
Variable	кошачий p=,29310	чёрный p=,43103	сифака p=,27586
голова	9,068	12,433	12,213
масса	-0,024	-0,026	-0,028
зуб верхний	53,984	61,209	60,659
Constant	-382,876	-662,947	-636,011

Функции классификации : мы получаем для них коэффициенты, и можем <u>классифицировать новых</u> лемуров: взять новую особь, посчитать для неё функцию для каждой группы, и отнести её в ту группу, для которой значение будет наибольшим!

Значения *p* – вероятности случайного причисления лемура к той или иной группе, исходя из размеров группы.

Можно посмотреть, сколько лемуров правильно и неправильно причислено к той или иной группе на основе функций классификации.

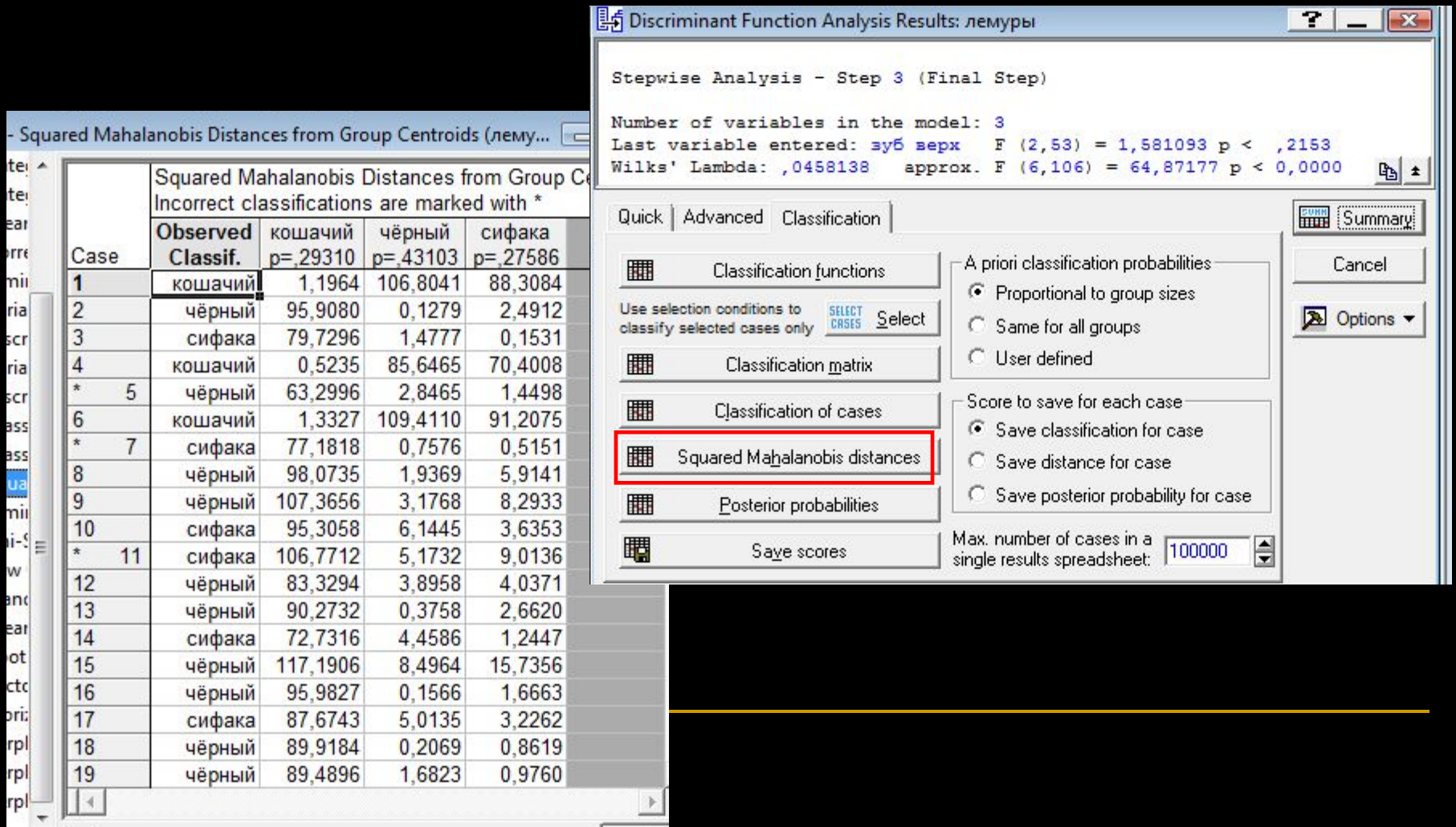




Group	Percent Correct	кошачий p=,29310	чёрный p=,43103	сифака p=,27586
кошачий	100,0000	17	0	0
чёрный	80,0000	0	20	5
сифака	75,0000	0	4	12
Total	84,4828	17	24	17

Теперь можно взять других особей (они должны стоять в той же таблице) и посмотреть процент правильного причисления в группы

На основе дистанций Махаланобиса от каждого измерения до центра группы можно посмотреть, к какому виду тот или иной лемур причисляется. Неправильные причисления помечены звёздочками



Требования к выборкам для проведения дискриминантного анализа

1. Внутри групп должно быть многомерное *нормальное распределение* (оценка – на основе построения гистограмм частот);

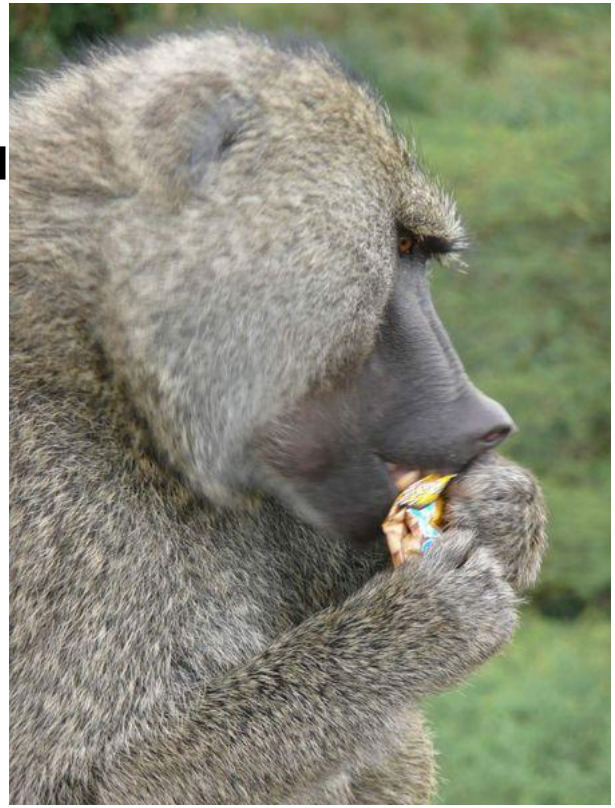
2. Гомогенность внутригрупповых *дисперсий* (не очень критичное требование);

3. Не должно быть корреляции *средних* значений и *дисперсий* в группах;

4. Не должно быть чрезмерно коррелирующих друг с другом переменных.

ФАКТОРНЫЙ АНАЛИЗ

Мы много лет изучаем пищевые предпочтения павианов и разработали комплексные оценки того, как они относятся к разным типам пищи.
Павианы едят разную еду, поэтому типов пищи – 10.
Но реальных факторов, определяющих эти предпочтения, наверняка меньше.



Мы хотим узнать, сколько (и каких) факторов определяют пищевые предпочтения павиана.

Итак,

Мы хотим

Найти те факторы, которые определяют изменчивость (объясняют действие) большого количества измеренных нами реальных переменных.

Подразумевается, что таких факторов гораздо меньше, чем исходных переменных.

Цели факторного анализа в биологии:

1. Преобразование взаимодействия многих переменных во взаимодействие небольшого числа факторов.

   → Уменьшение числа переменных в анализе (что, например, уменьшит эффект множественных сравнений).

2. Выявление реальных действующих факторов (причинно-следственных связей), лежащих в основе биологических корреляций, или просто выявление структуры взаимосвязи переменных.

Например, поиск трендов в морфологии из корреляций многих морфологических признаков.

Поясняющий пример:

Мы изучаем кроликов. Сначала взвешиваем каждого из 100 кроликов на безмене, потом на весах с гирьками, потом на электронных кухонных весах.

Потом мы хотим исследовать влияние питания на вес кроликов.
Неужели мы возьмём в анализ все три переменные? Ведь, очевидно, вес кролика – только **одна** его характеристика, а не три. Скорее всего, мы захотим превратить все переменные в одну.

Факторный анализ:

1. Анализ главных компонент (principal component analysis);
   - Основная идея: получить факторы, объясняющие как можно больше общей изменчивости; больше подходит, если основная цель – сократить число переменных в анализе;

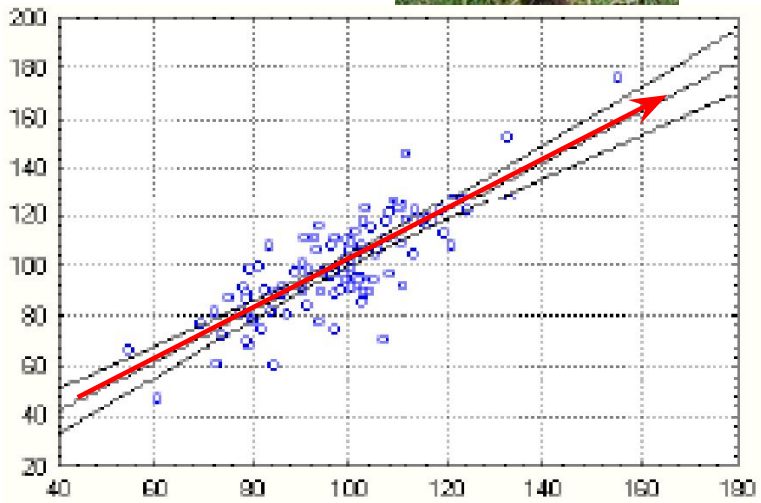
2. Анализ главных факторов (principal factor analysis)
   - Основная идея: для каждой переменной используется только доля изменчивости, общая с другими переменными; больше подходит для поиска структуры переменных, определения их иерархии.

Анализ главных компонент

Подразумевается, что наши реально измеренные переменные являются линейными комбинациями этих подлежащих факторов.

Факторы (главные компоненты) находят на основании матрицы корреляции переменных – на основе **линий регрессии**.

Процедура анализа подобна вращению, максимизирующему дисперсию исходного пространства переменных.



Примерно так будет проходить новая ось OX.

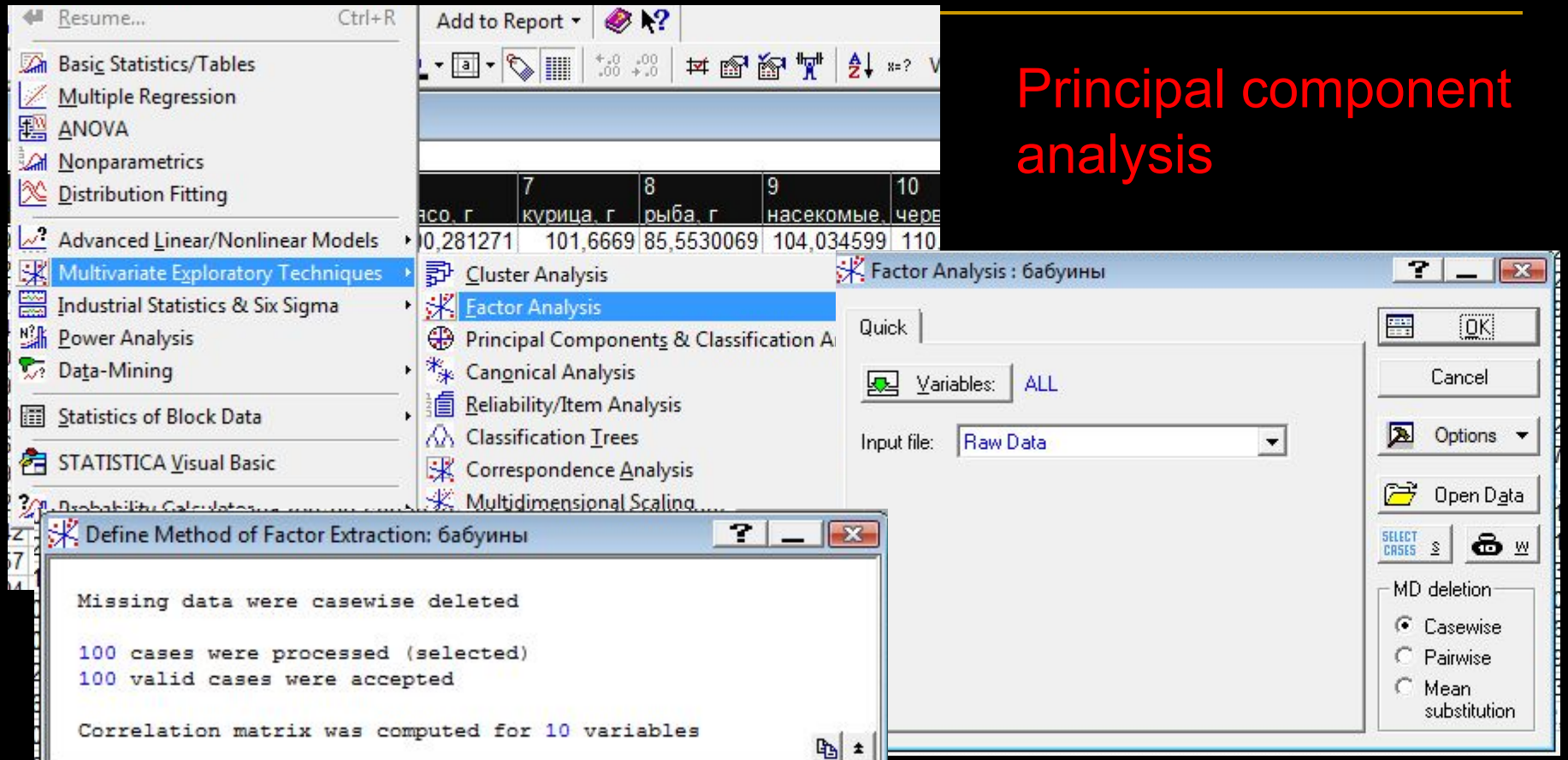
После выделения первого фактора выделяется следующий, который должен тоже максимизирует оставшуюся дисперсию и т.д. – все факторы будут **ортогональны.**

Итак, мы изучаем питание павианов. Типов пищи у павианов 10:

апельсины,
бананы,
яблоки,
помидоры,
огурцы,
мясо,
курица,
рыба,
насекомые,
червяки.

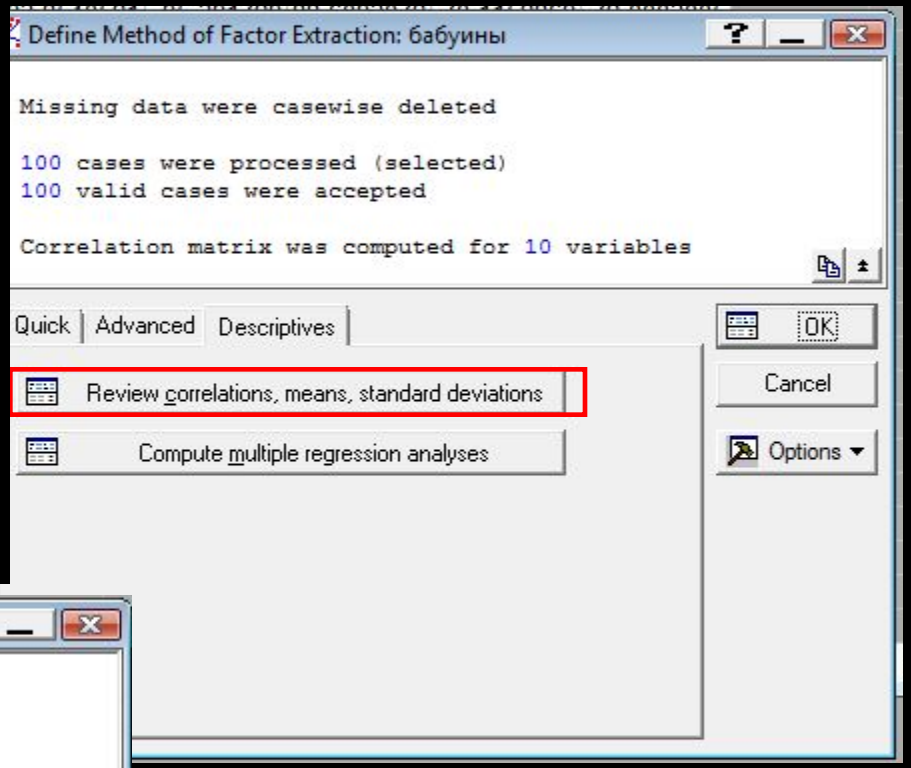
Сколько факторов скрывается за разными предпочтениями павианов в еде?

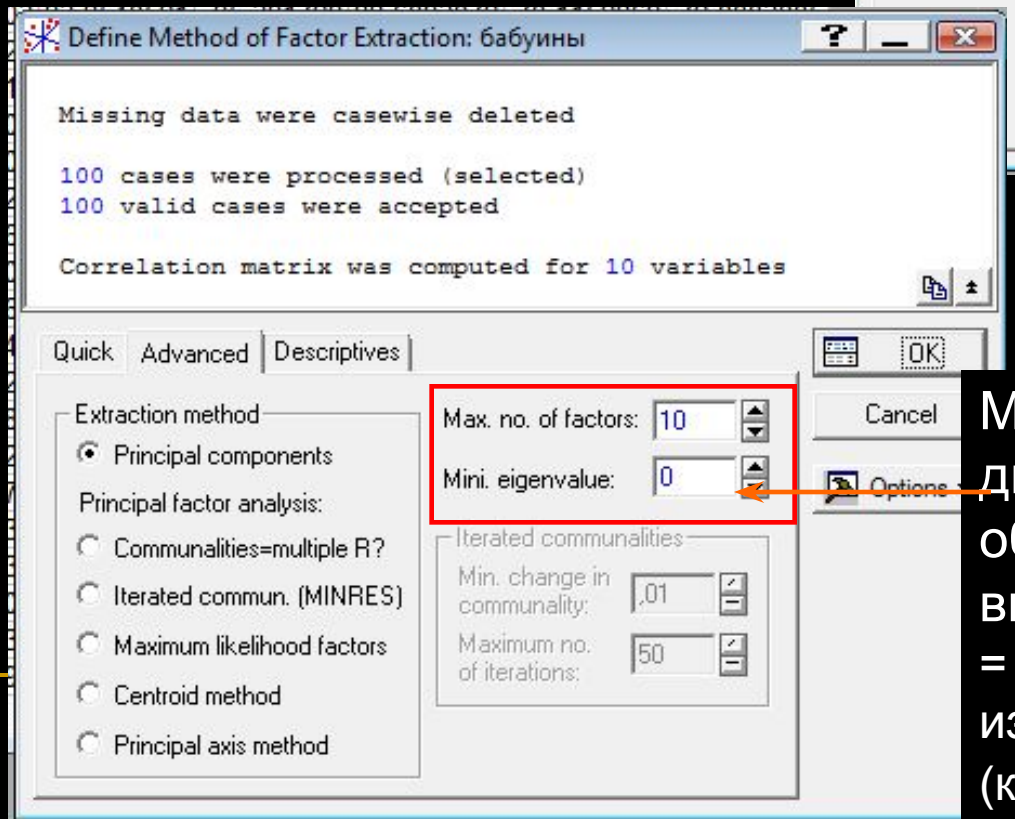
Principal component analysis



(прежде, чем проводить факторный анализ, рекомендуется построить матрицу корреляций: исключить переменные, слишком сильно коррелирующие с другими)

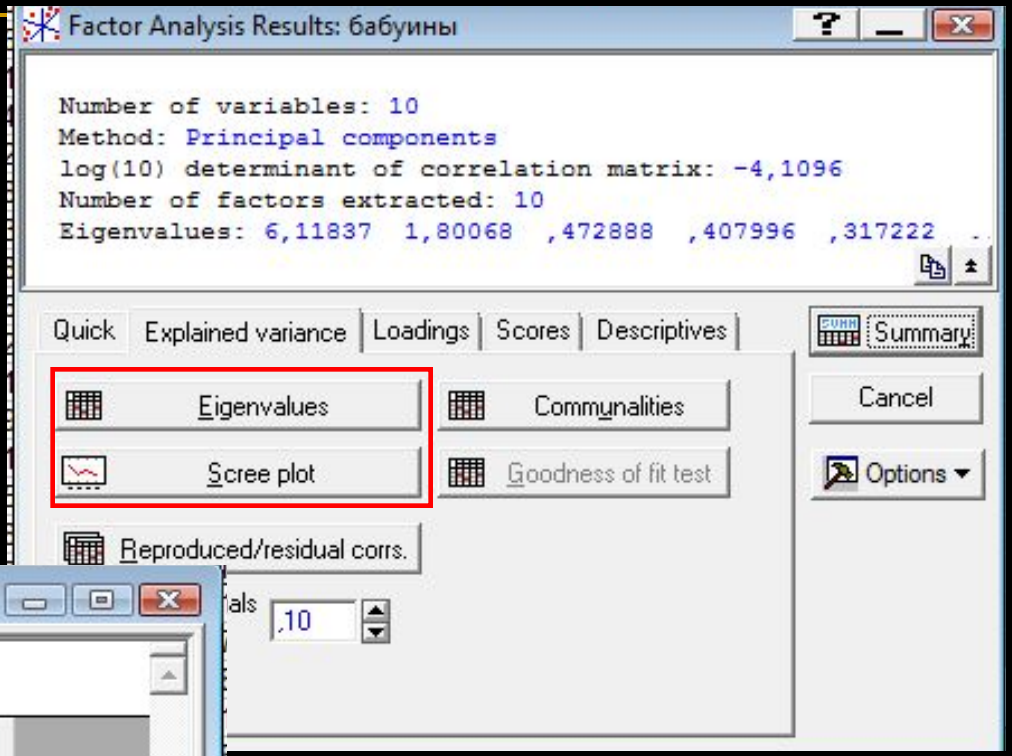
Просмотрим матрицу корреляций:
Не должно быть слишком сильно коррелирующих друг с другом переменных (иначе матрица не может быть транспонирована: *matrix ill-conditioning*)

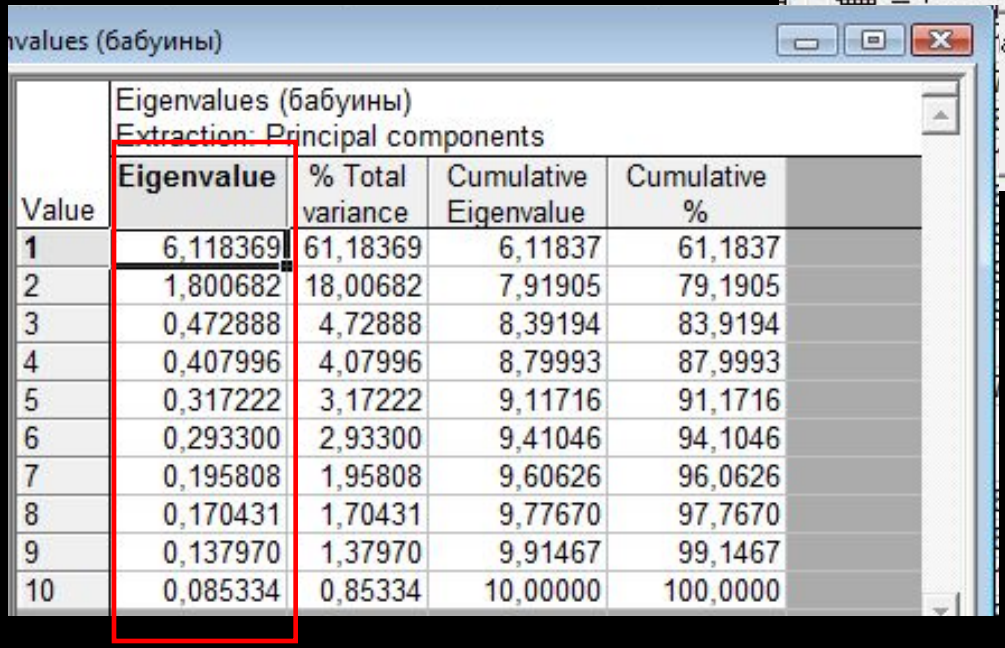




Можно задать min количество дисперсии, которое должен объяснять фактор, чтобы его включили в анализ (обычно min = 1, что соответствует случайной изменчивости одной переменной (критерий Кайзера))

Собственные значения (eigenvalues)– определяют, какую долю общей дисперсии объясняет данный фактор.

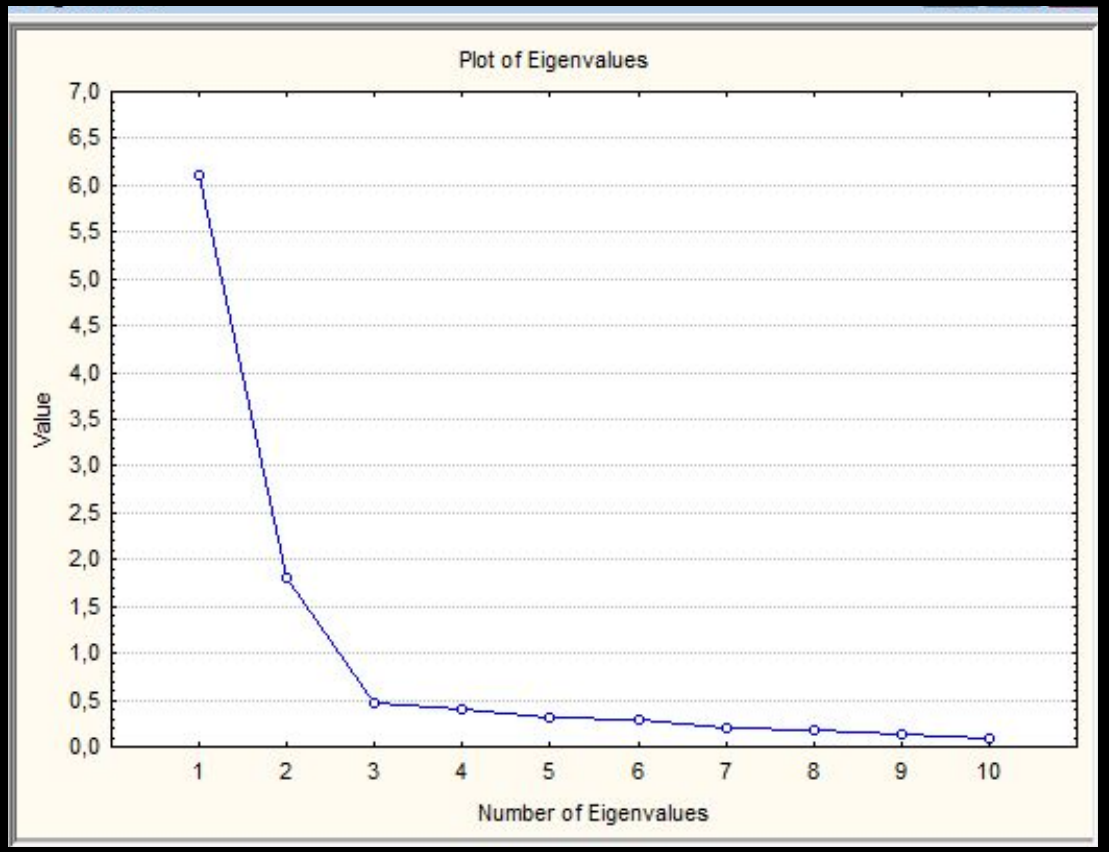




nvalues (бабуины)

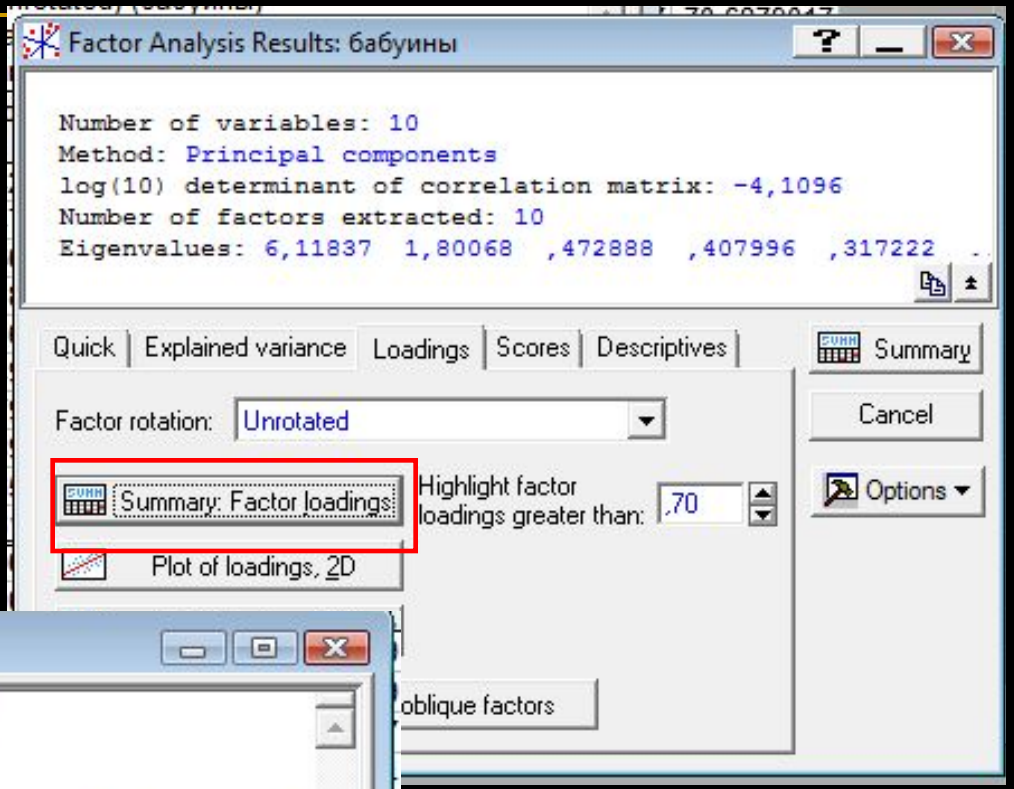
Eigenvalues (бабуины)
Extraction: Principal components

Value	Eigenvalue	% Total variance	Cumulative Eigenvalue	Cumulative %
1	6,118369	61,18369	6,11837	61,1837
2	1,800682	18,00682	7,91905	79,1905
3	0,472888	4,72888	8,39194	83,9194
4	0,407996	4,07996	8,79993	87,9993
5	0,317222	3,17222	9,11716	91,1716
6	0,293300	2,93300	9,41046	94,1046
7	0,195808	1,95808	9,60626	96,0626
8	0,170431	1,70431	9,77670	97,7670
9	0,137970	1,37970	9,91467	99,1467
10	0,085334	0,85334	10,00000	100,0000



Этот график показывает, что первые два фактора лучше остальных, они объясняют большую часть общей изменчивости (the scree test).

Посмотрим, как полученные факторы связаны с реальными переменными

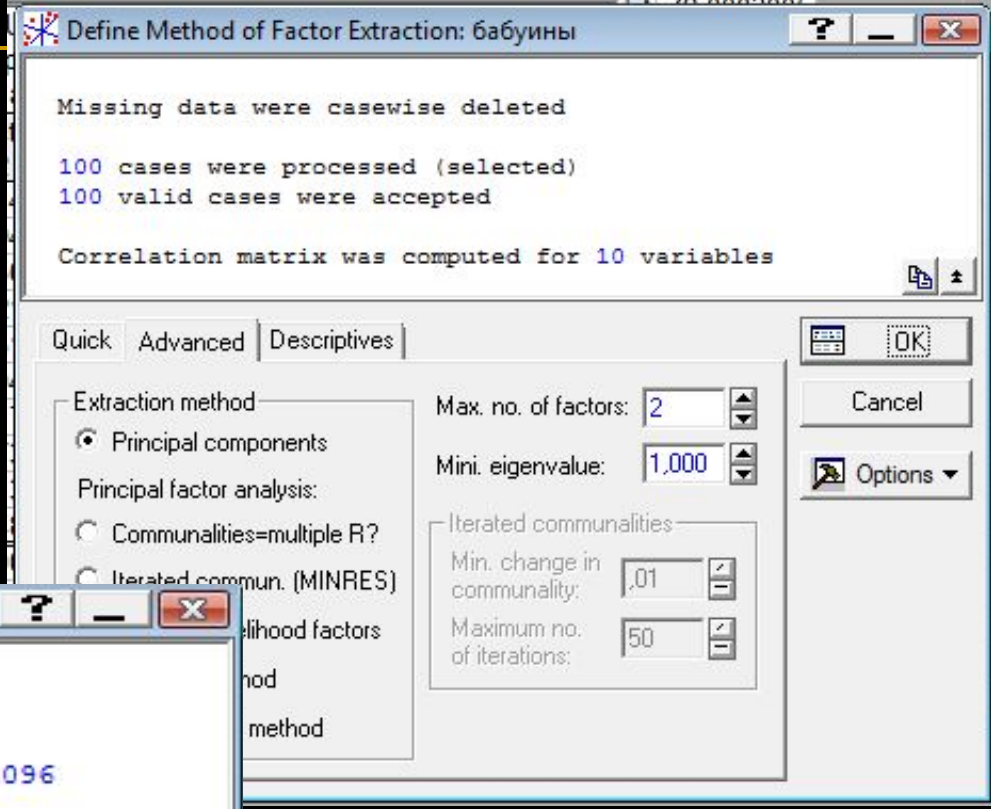


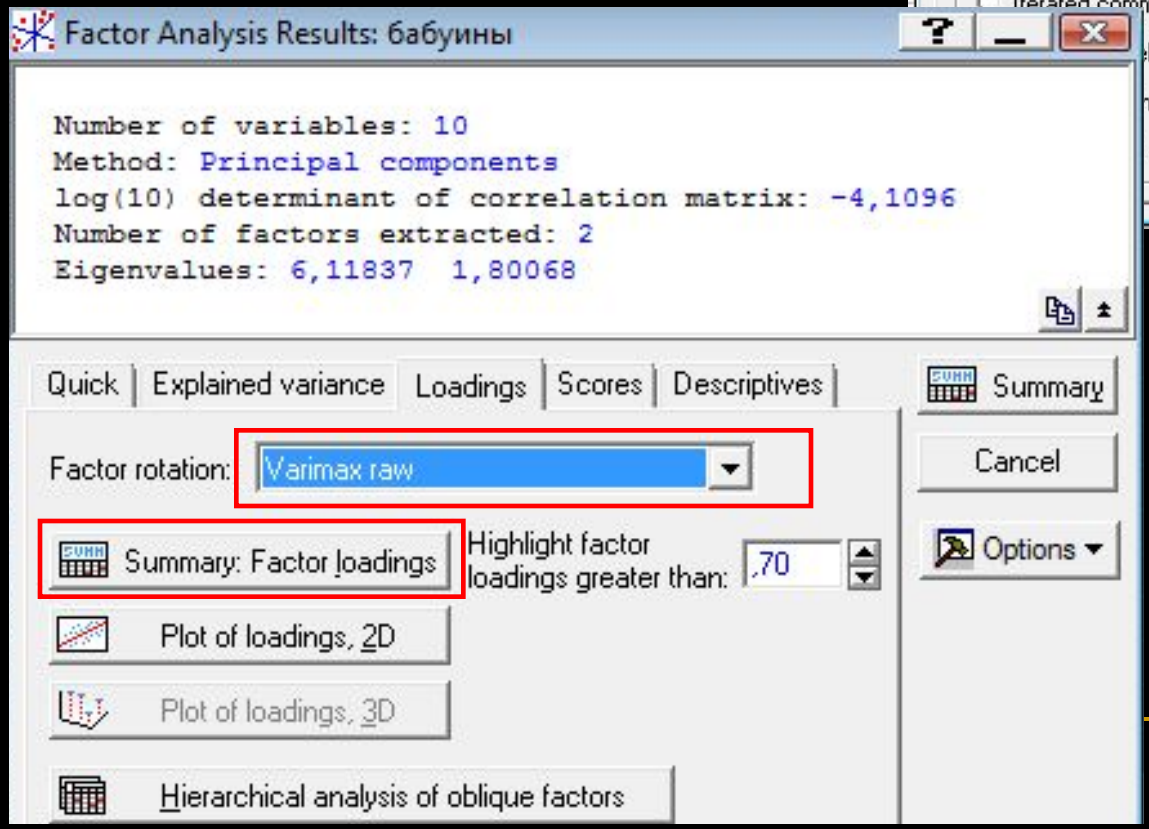
tor Loadings (Unrotated) (бабуины)

Factor Loadings (Unrotated) (бабуины)
Extraction: Principal components
(Marked loadings are > ,700000)

Variable	Factor 1	Factor 2	Factor 3	Factor 4	Factor 5	Fa
апельсины, г	-0,652601	0,514217	0,301687	0,439108	-0,013701	0,1
бананы, г	-0,756976	0,494770	-0,078826	-0,211795	-0,090859	0,1
яблоки, г	-0,745706	0,456680	-0,104749	0,030826	-0,204913	-0,4
помидоры, г	-0,941630	-0,021835	0,012653	0,001861	0,120655	0,0
огурцы, г	-0,875615	0,051643	0,099675	-0,324541	-0,015852	0,0
мясо,г	-0,576062	-0,604977	0,490999	-0,114927	-0,112513	-0,1
курица, г	-0,671289	-0,617962	-0,125776	0,159963	0,225012	-0,1
рыба,г	-0,641532	-0,573925	-0,268572	0,152709	-0,362524	0,1
насекомые, г	-0,951516	0,013513	-0,050164	0,026706	0,076795	0,0
червяки, г	-0,900333	0,048154	-0,151805	-0,034832	0,226647	-0,0
Expl.Var	6,118369	1,800682	0,472888	0,407996	0,317222	0,2
Prp.Totl	0,611837	0,180068	0,047289	0,040800	0,031722	0,0

Можно выбрать два фактора, расположить в их пространстве переменные; потом повернуть факторы (оси координат) так, чтобы максимизировать изменчивость переменных по ним.





После вращения факторов их структура становится более ясной:

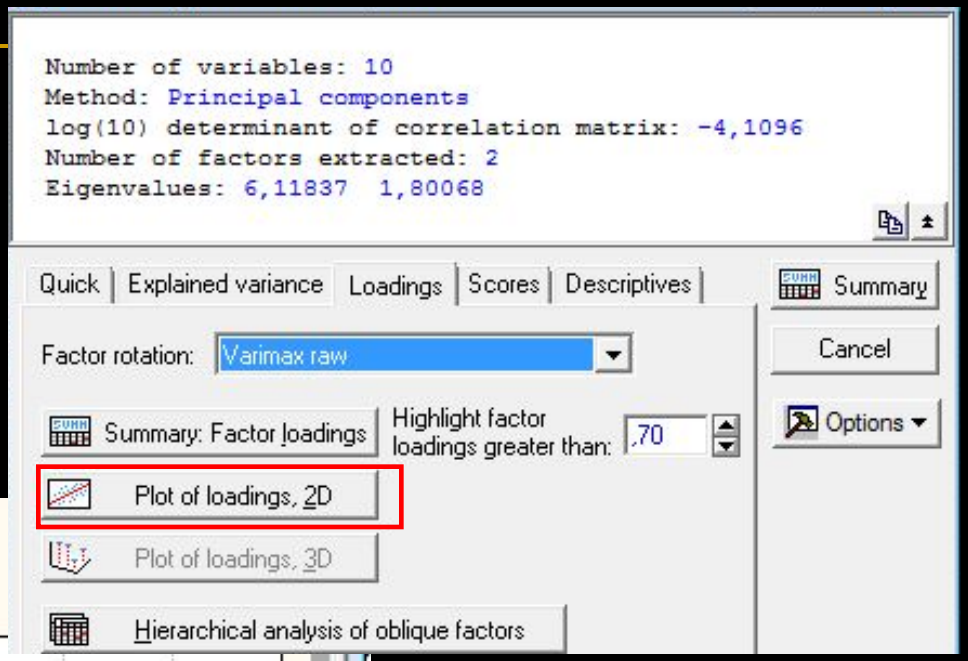
r Loadings (Varimax raw) (бабуины)

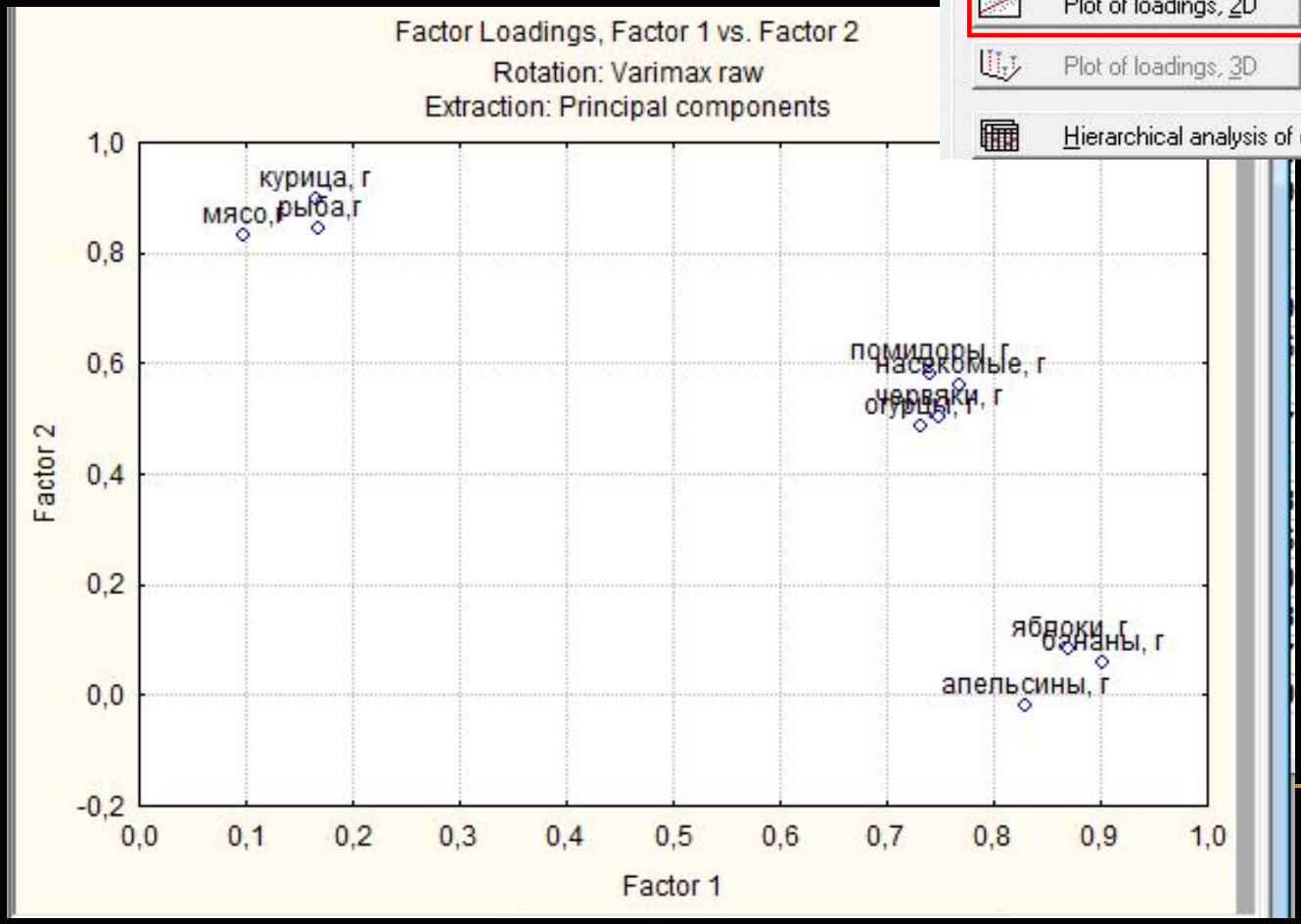
Variable	Factor Loadings (Varimax raw) (ба Extraction: Principal components (Marked loadings are > ,700000) Factor 1	Factor 2
апельсины, г	0,830623	-0,019320
бананы, г	0,902408	0,058905
яблоки, г	0,870524	0,082595
помидоры, г	0,739857	0,582885
огурцы, г	0,731191	0,484489
мясо,г	0,097371	0,829676
курица, г	0,165722	0,897242
рыба,г	0,168370	0,844159
насекомые, г	0,768988	0,560555
червяки, г	0,748861	0,502121
Expl.Var	4,561544	3,357507
Prp.Totl	0,456154	0,335751

Фактор 1 в основном связан с растительной пищей, фактор 2 – с животной.

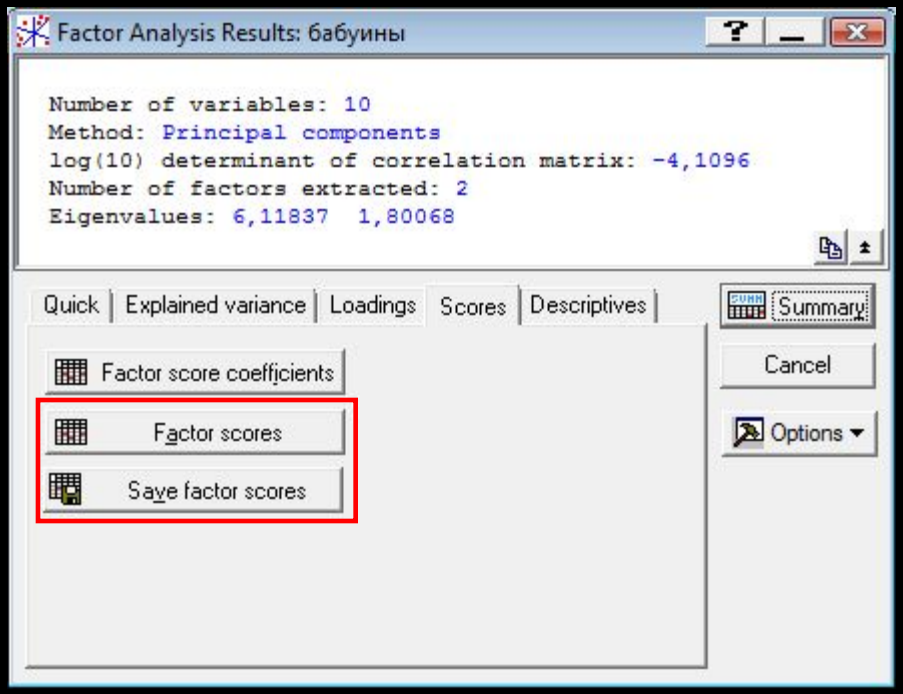
Итак, пищевые предпочтения павианов составлены из двух основных факторов – отношением к животной и растительной пище.

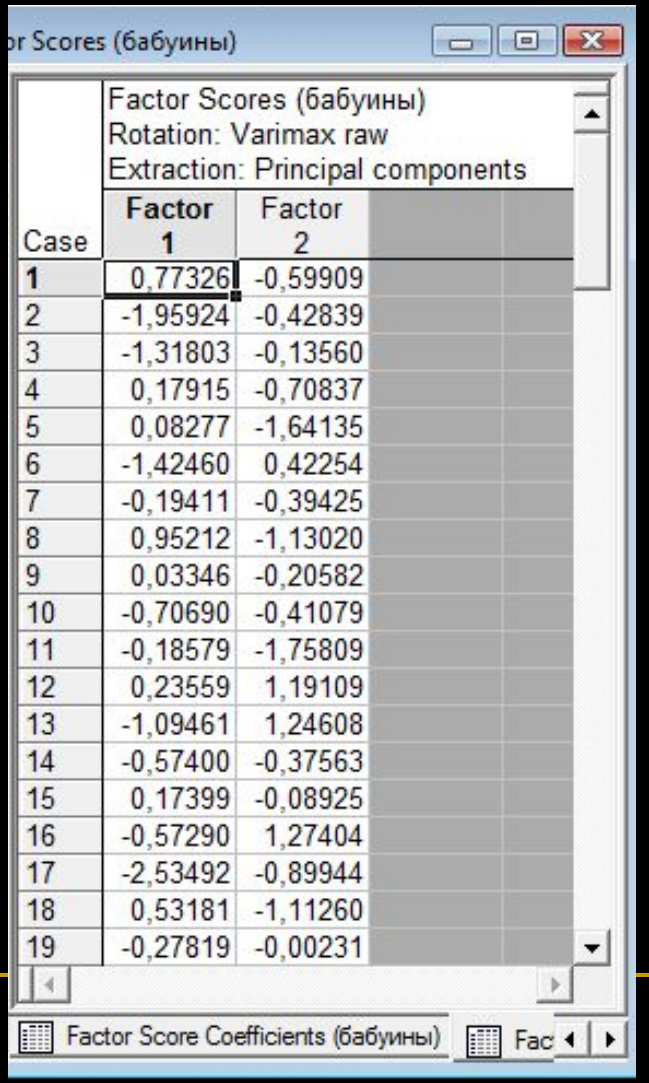
Посмотрим, как исходные переменные расположились в пространстве новых факторов



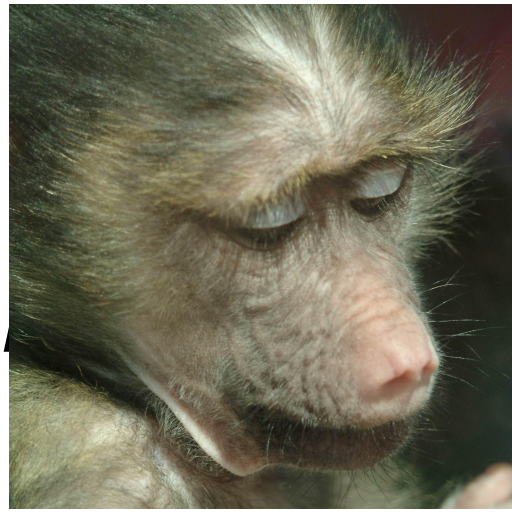


Если мы в дальнейшем хотим проводить анализ связи питания павианов с другими переменными, мы можем заменить наши 10 переменных на полученных два фактора.





Требования к выборкам для проведения факторного анализа



1. Внутри групп должно быть многомерное *распределение* (оценка – на основе гистограмм частот);
2. Гомогенность *дисперсий* (для метода главных компонент; не очень критичное требование);
3. Связь переменных должна быть *линейной*;
4. Размер выборки не должен быть меньше 50, оптимальный – ≥100 наблюдений.
5. Между переменными должна быть *ненулевая корреляция*, но коэффициентов корреляции, близких *единице*, тоже быть не должно.

Если распределение не нормальное, связь переменных нелинейная, выборка небольшая:

Многомерное шкалирование (*Multidimentional scaling*)

На основе сходства (любых дистанций!) между наблюдениями позволяет расположить их в пространстве нескольких новых факторов так, чтобы факторы объясняли как можно больше изменчивости.

Но если данные более-менее удовлетворяют требованиям факторного анализа, лучше проводить его, т.к.:

1. Факторный анализ - гораздо более мощная процедура, намного лучше оценивает связи исходных переменных;

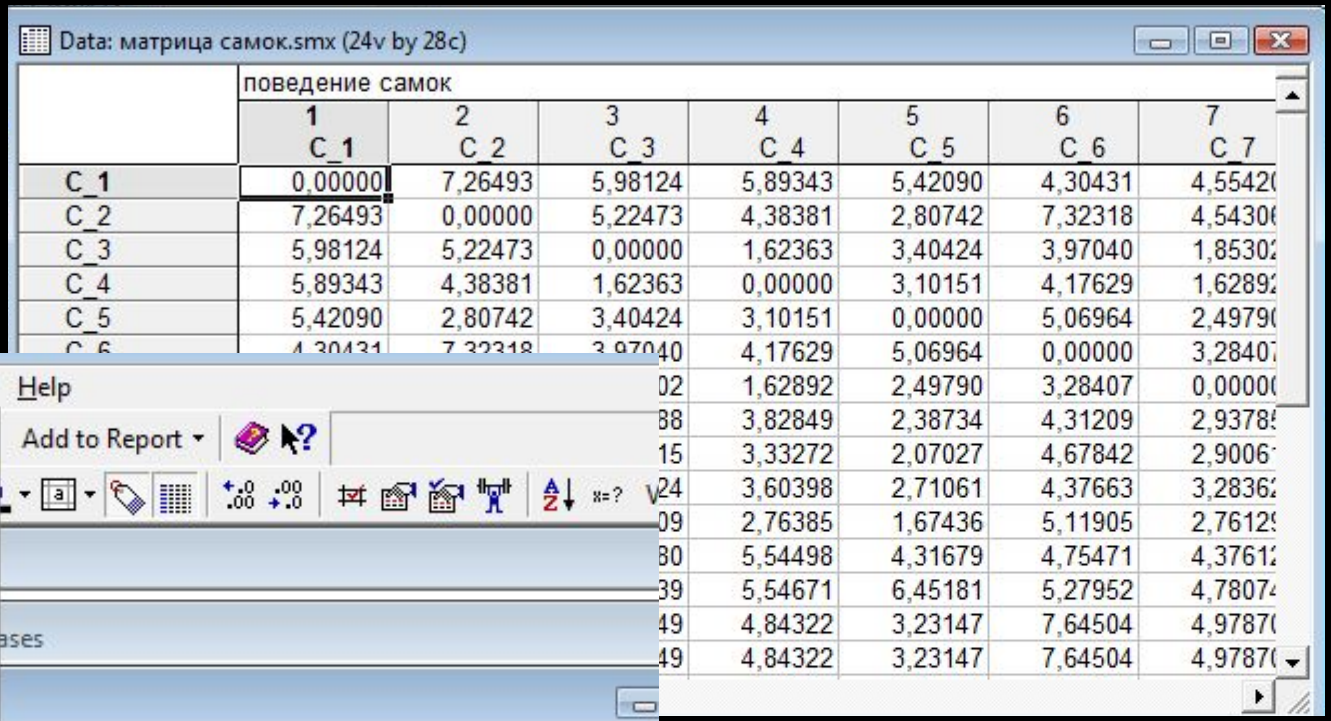
2. Результаты гораздо проще интерпретировать: в многомерном шкалировании очень трудно объяснить, что же значат полученные факторы.

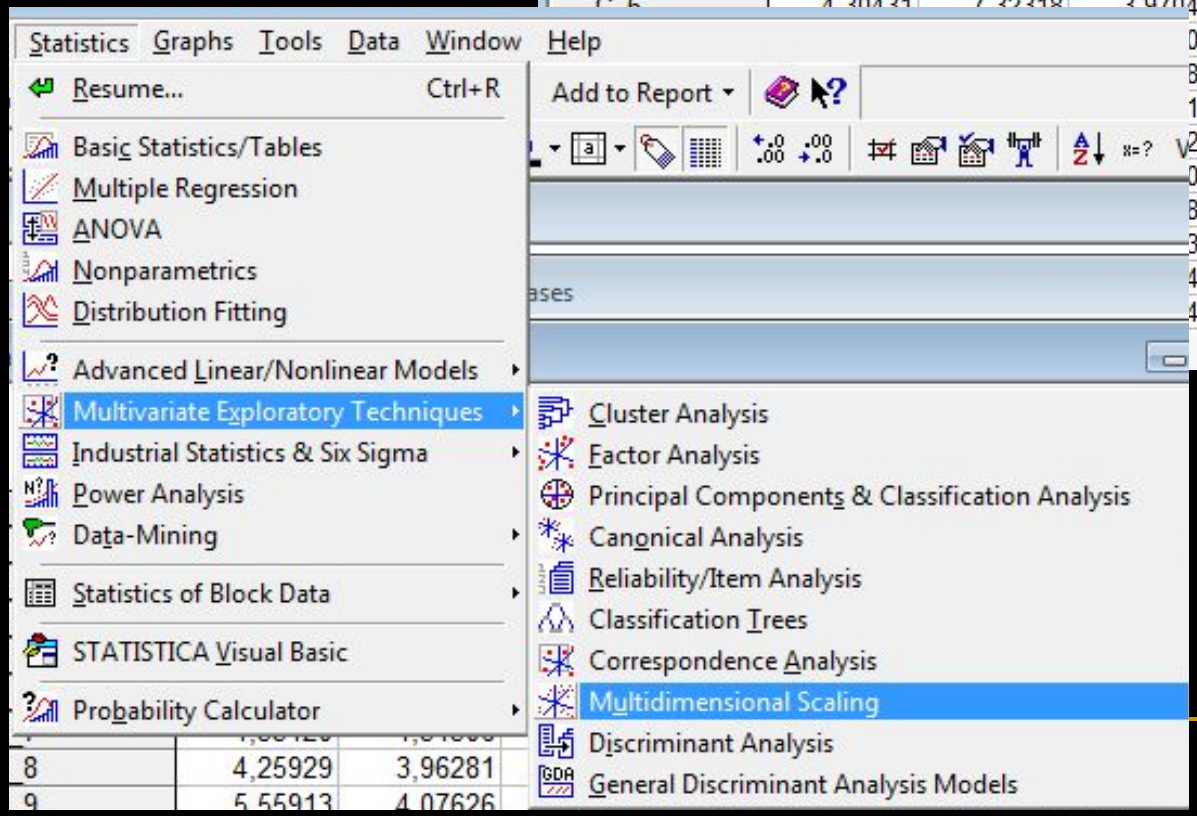
Это просто уменьшение числа переменных, а не статистический метод

Мы наблюдаем поведение молодых сурков. У нас есть 15 переменных, описывающих социальное поведение. Это частоты контактов, которые имеют распределение, далёкое от нормального.

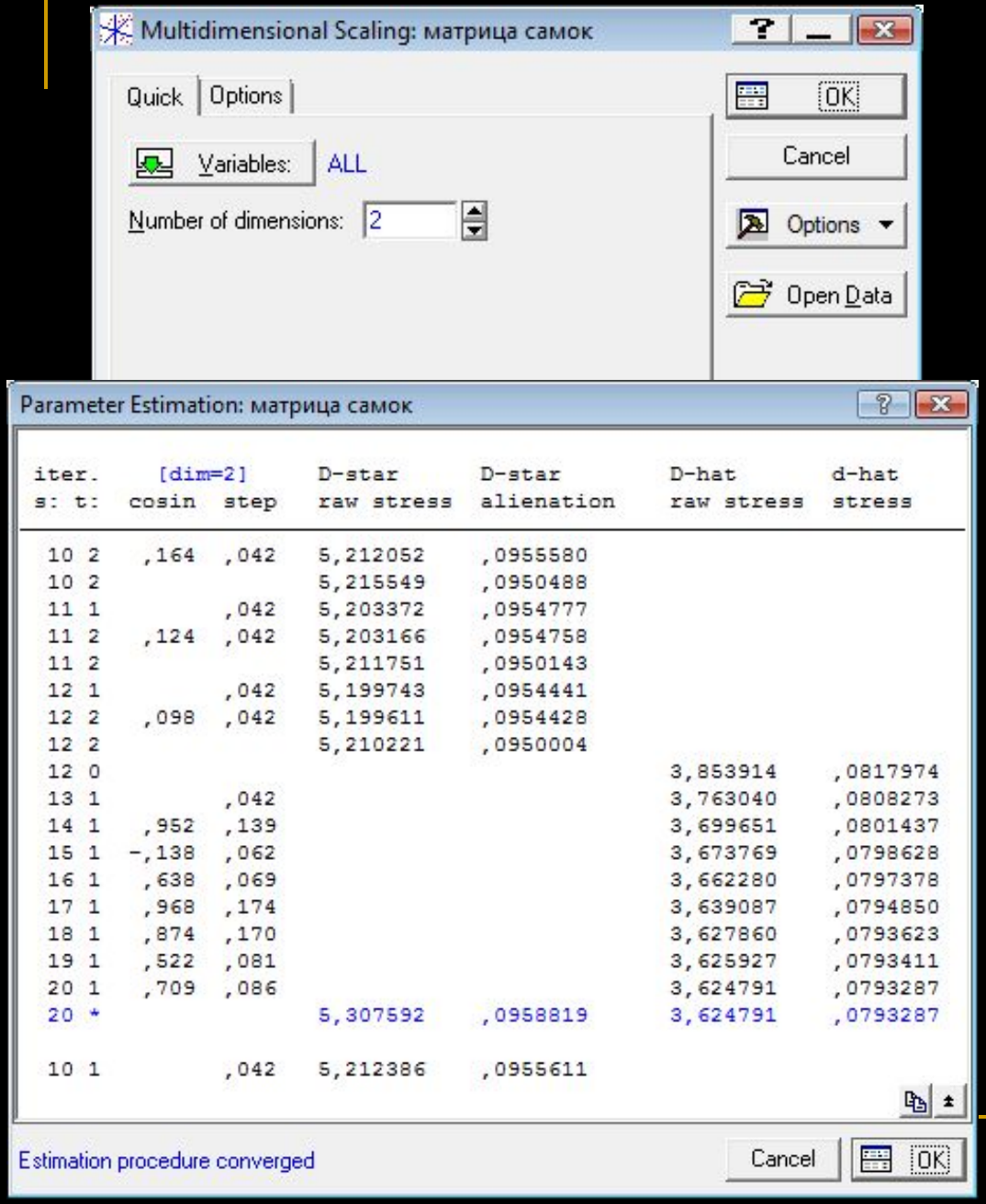
Мы хотим из 15 переменных получить 2-3, которые бы хорошо объясняли изменчивость в выборке.

Данные для анализа должны быть представлены **МАТРИЦЕЙ ДИСТАНЦИЙ** (как её получать – рассказ дальше)

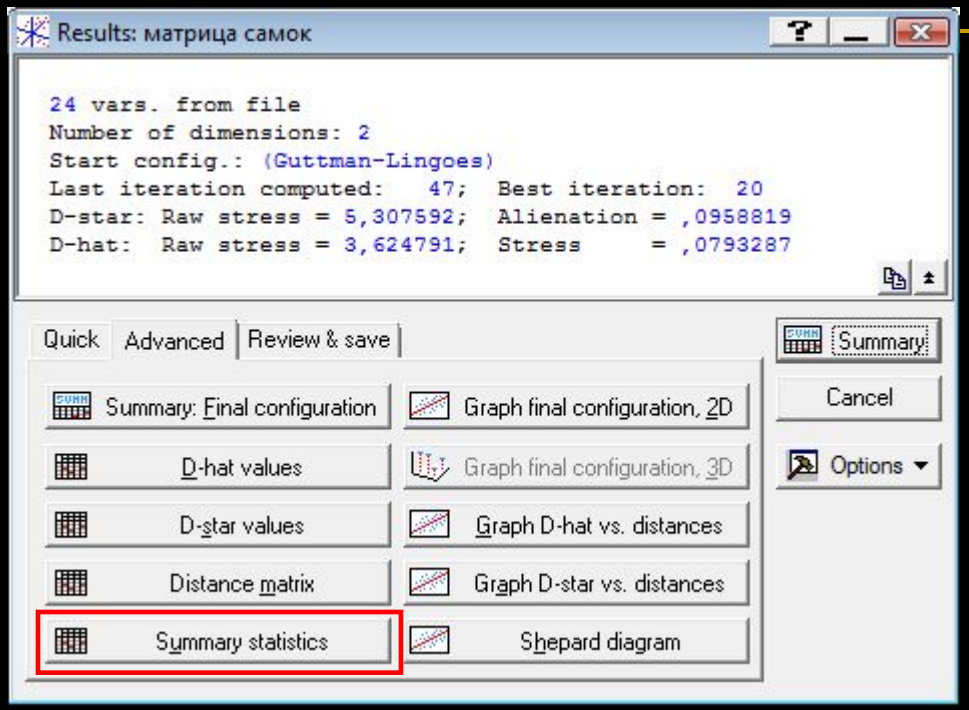




Число измерений (строк) не может быть больше 90



Программа вращает наши наблюдения в пространстве так, чтобы расстояния между ними в полученной модели лучше всего соответствовали исходным расстояниям между наблюдениями (чем больше измерений в модели, тем лучше модель будет отражать реальность, но тем она будет сложнее)



Мы получили итоговую конфигурацию. Посмотрим, насколько она хороша.

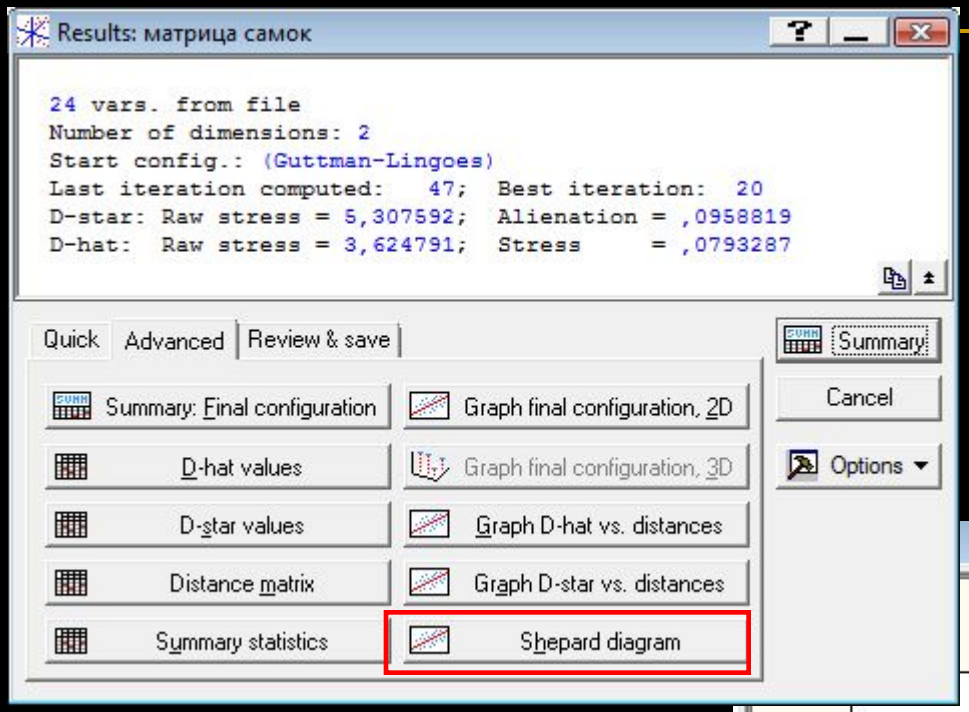
Configuration (матрица самок)

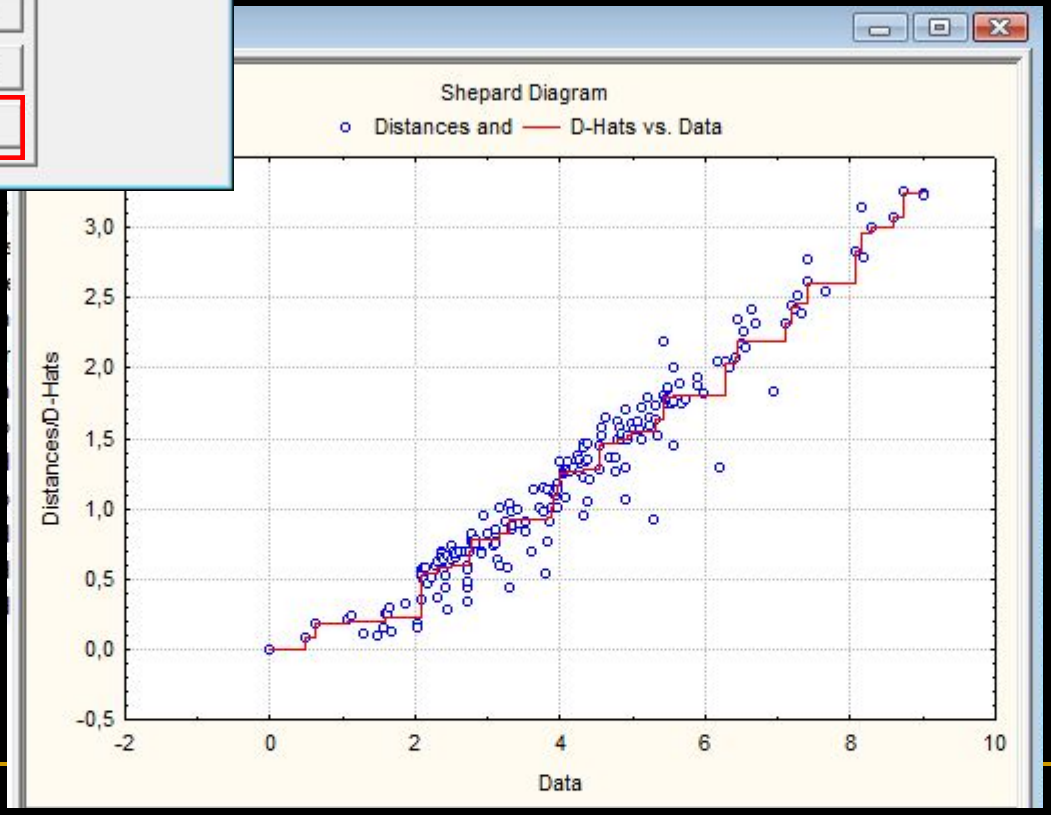
Final Configuration (матрица самок)
D-star: Raw stress = 5,307592; Alienation = ,0958819
D-hat: Raw stress = 3,624791; Stress = ,0793287

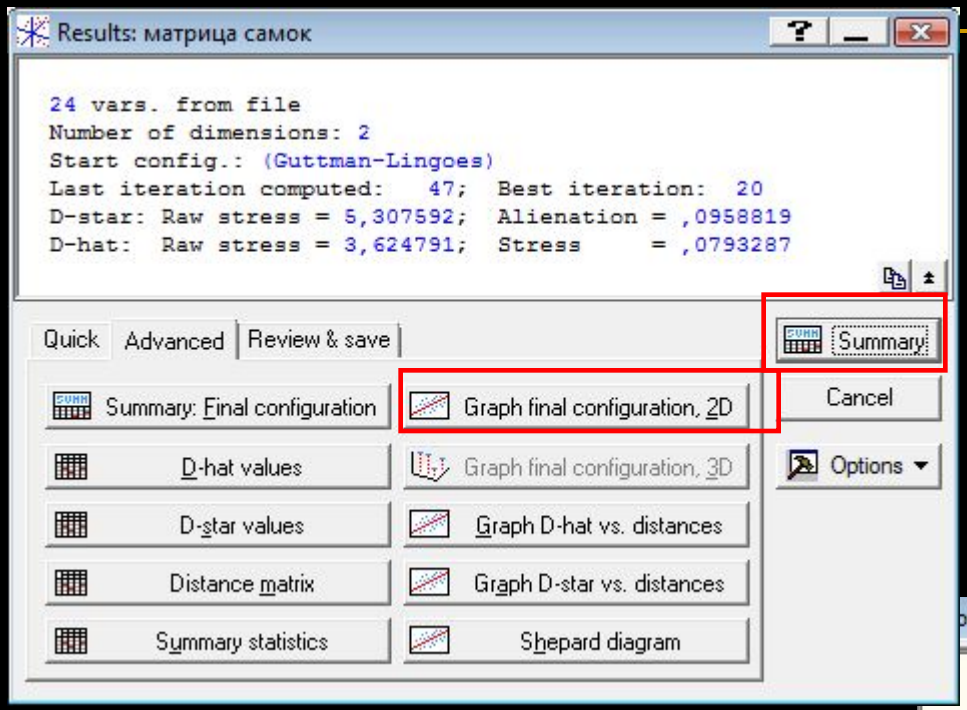
	Distance	D-star	D-hat
D(22,15)	0,000243	0,000074	0,000163
D(15,14)	0,000087	0,000083	0,000163
D(22,16)	0,000328	0,000085	0,000163
D(19,14)	0,000083	0,000087	0,000163
D(22,19)	0,000074	0,000156	0,000163
D(19,16)	0,000254	0,000169	0,000167
D(22,14)	0,000156	0,000172	0,000167
D(16,14)	0,000172	0,000243	0,000167
D(19,15)	0,000169	0,000254	0,000167
D(16,15)	0,000085	0,000328	0,000167
D(21, 2)	0,077300	0,077300	0,077300
D(16, 2)	0,179191	0,099765	0,179191
D(15, 2)	0,179237	0,115304	0,179237
D(22, 2)	0,179369	0,122014	0,179328
D(14, 2)	0,179287	0,160012	0,179328
D(19, 2)	0,179330	0,160621	0,179330
D(24, 5)	0,214672	0,179066	0,198856
D(22,21)	0,239922	0,179191	0,198856
D(21,15)	0,239840	0,179237	0,198856
D(21,19)	0,239898	0,179287	0,198856
D(21,14)	0,239872	0,179330	0,198856
D(21,16)	0,239812	0,179369	0,198856
D(17, 7)	0,115304	0,214672	0,198856
D(24,11)	0,099765	0,239812	0,198856
D(20,11)	0,160621	0,239840	0,198856

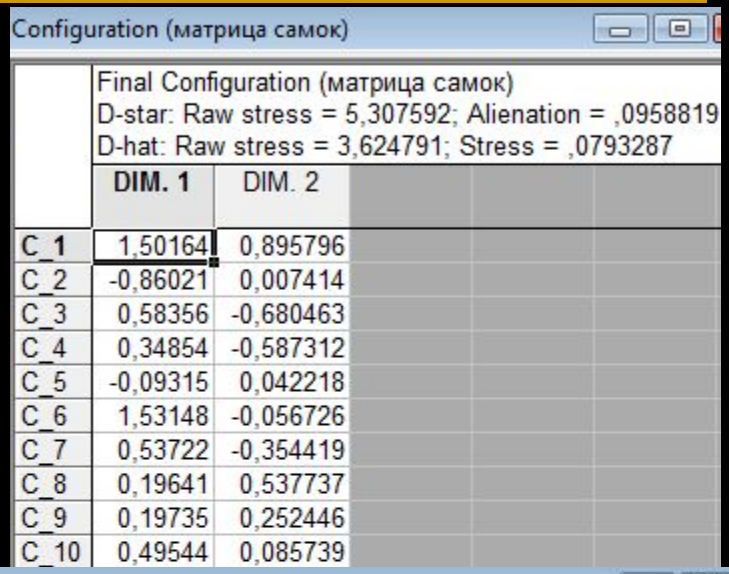
D-star и D-hat – вычисленные программой дистанции между измерениями; расстояния упорядочены по ним. Distance – реальные дистанции, должны стоять в том же порядке.

Диаграмма Шепарда покажет, хорошо ли модель согласуется с исходными данными: чем ближе точки к красной линии, тем лучше.

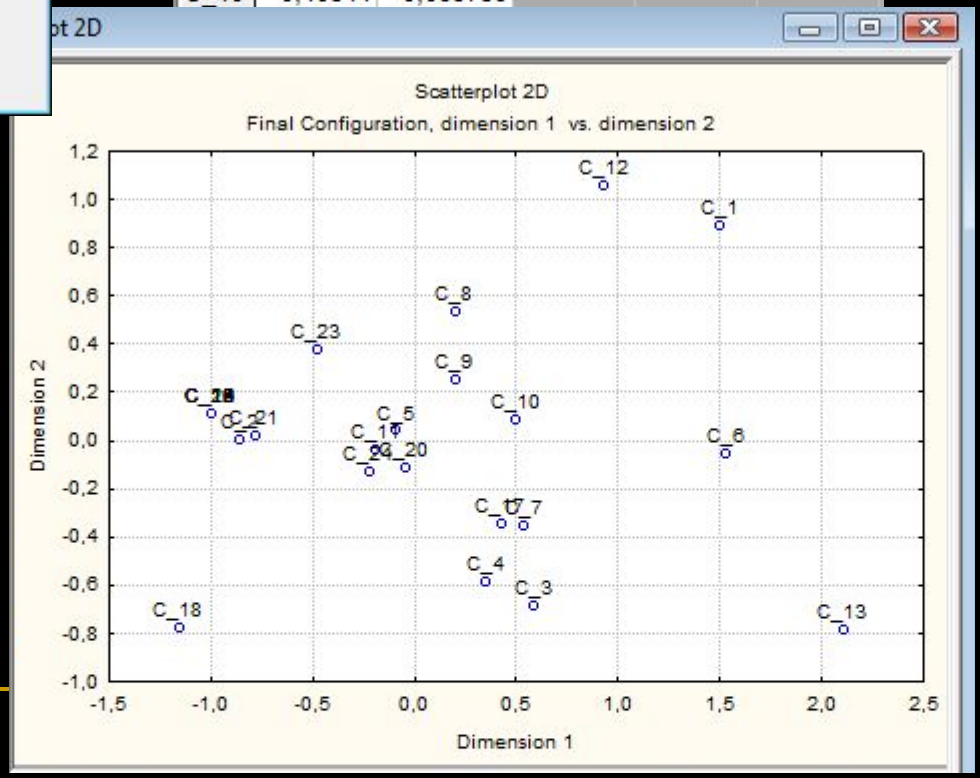








	DIM. 1	DIM. 2
C_1	1,50164	0,895796
C_2	-0,86021	0,007414
C_3	0,58356	-0,680463
C_4	0,34854	-0,587312
C_5	-0,09315	0,042218
C_6	1,53148	-0,056726
C_7	0,53722	-0,354419
C_8	0,19641	0,537737
C_9	0,19735	0,252446
C_10	0,49544	0,085739



Наконец, получим значения новых переменных для наших наблюдений и построим картинку, где они расположены в пространстве этих переменных

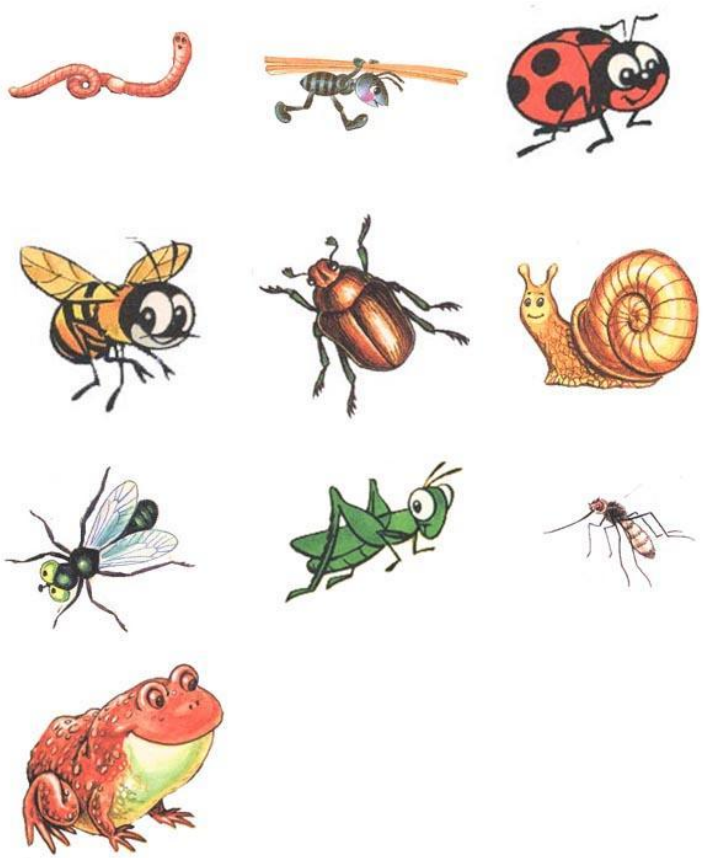
Интерпретация результатов многомерного шкалирования –
исключительно на основе картинки, где наблюдения расположены в пространстве новых переменных.

Посмотреть, какая исходная переменная какой вклад вносит в полученные переменные, нельзя.

КЛАСТЕРНЫЙ АНАЛИЗ

Это вообще *не статистический метод*, а чисто описательная математическая процедура группировки и классификации данных.

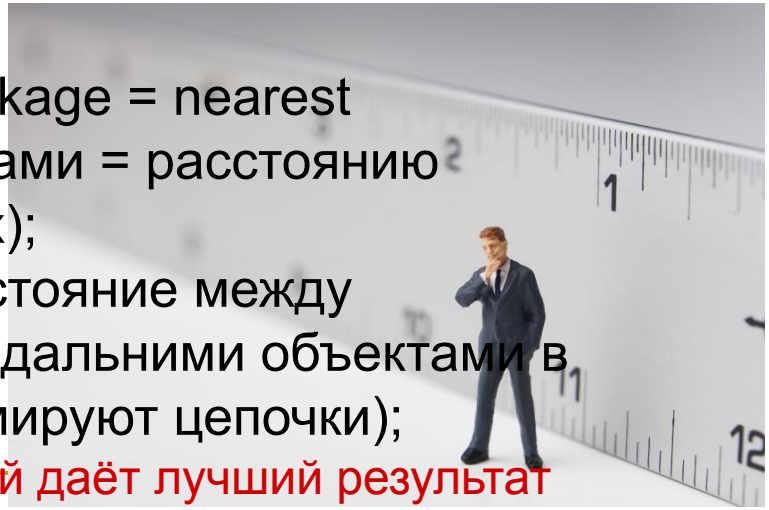
Здесь вообще неприменима проверка статистической значимости



Классификация: программа начинает с кластеров, содержащих не более одного элемента; потом – не больше двух, и.т.д, и в конце в одном большом кластере оказываются все элементы.

Идея анализа –

1. Рассчитываются *дистанции* между измерениями в пространстве исходных переменных;
   - Евклидовы дистанции;
   - Квадрат евклидова расстояния (если хотим увеличить вес отдельных больших разностей);
   - Манхэттенское расстояние (если хотим уменьшить вес отдельных больших расстояний)
   - …
2. На основе этих дистанций разными способами объекты объединяют в *кластеры*
   - Метод ближайшего соседа (Single linkage = nearest neighbor; расстояние между кластерами = расстоянию между ближайшими объектами в них);
   - Полная связь (Complete linkage; расстояние между кластерами определяется наиболее дальними объектами в них; не годится, если кластеры формируют цепочки);
   - В целом, можно выбирать метод, который даёт лучший результат

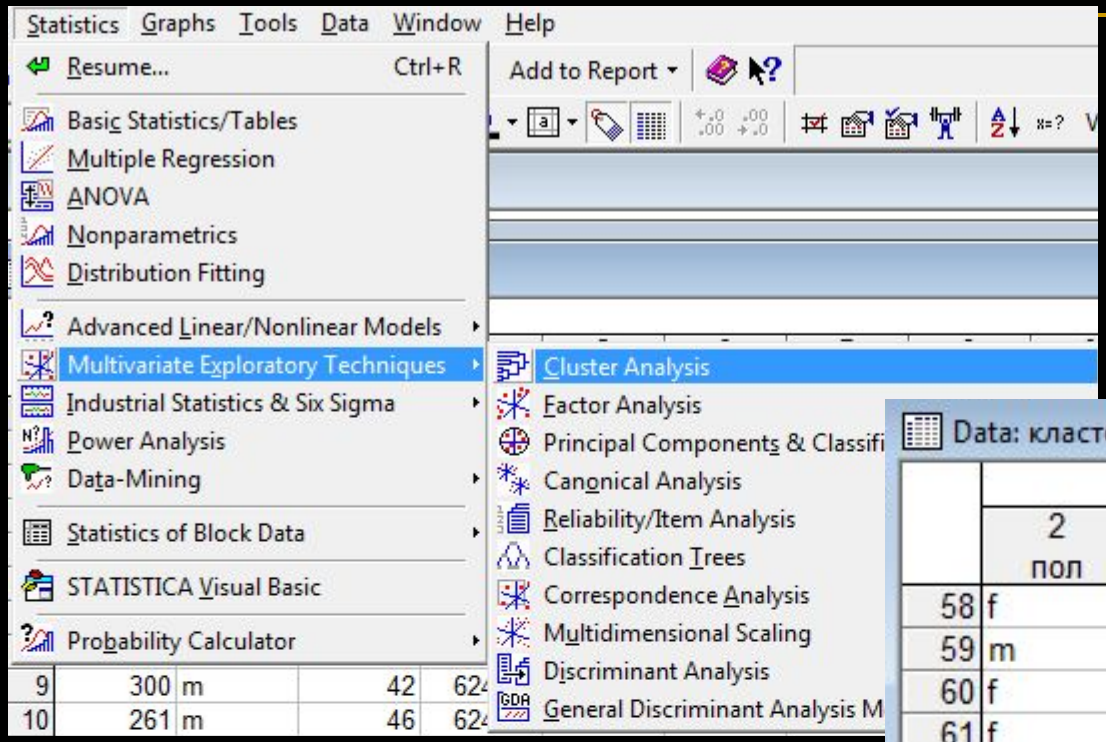


Основной результат – получение иерархического дерева

Пример.

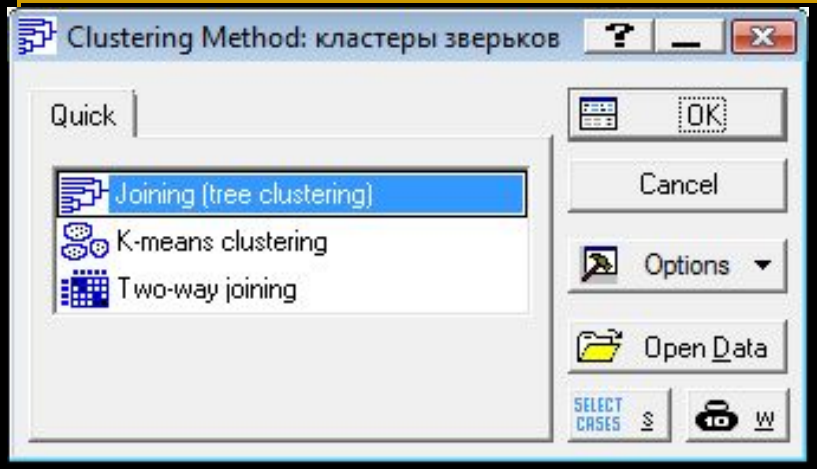
У нас есть молодые лемуры, которые после расселения заняли дупла в лесу. Известны координаты каждого дупла. Мы хотим узнать, формируют ли зверьки пространственные кластеры?

Cluster analysis

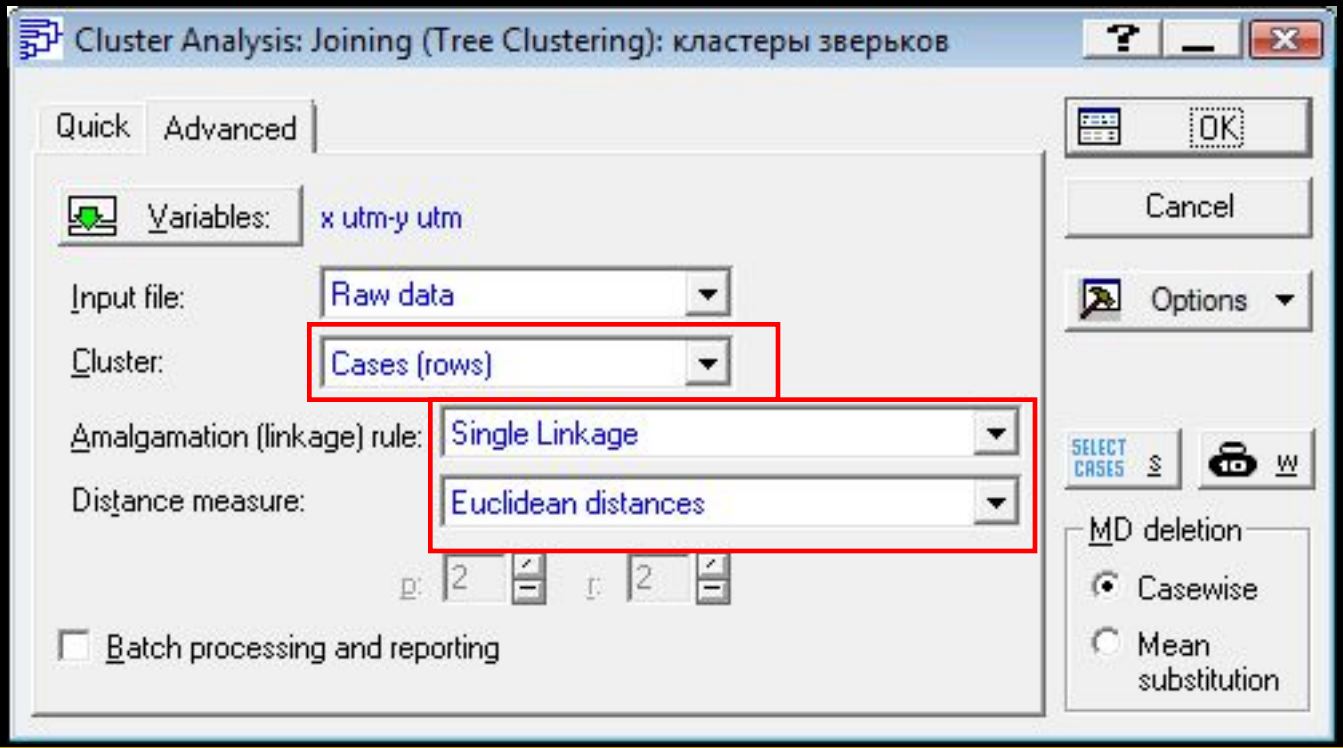


Data: кластеры зверьков.sta (10...

	2 пол	3 выводок	4 x utm	5 y utm
58	f	50	625033,3	5621474
59	m	53	624866,6	5621661
60	f	53	624605,9	5621511
61	f	53	624820,1	5621786
62	f	57	624623,2	5621484
63	f	58	624973	5621409
64	m	60	624534	5621505
65	m	60	624576,5	5621477
66	m	60	624605	5621523
67	m	60	624598,9	5621513
68	f	65	624861,6	5621745
69	f	66	624780	5621546
70	m	66	624756,1	5621710
71	m	66	624708,5	5621498
72	m	67	624761,3	5621831

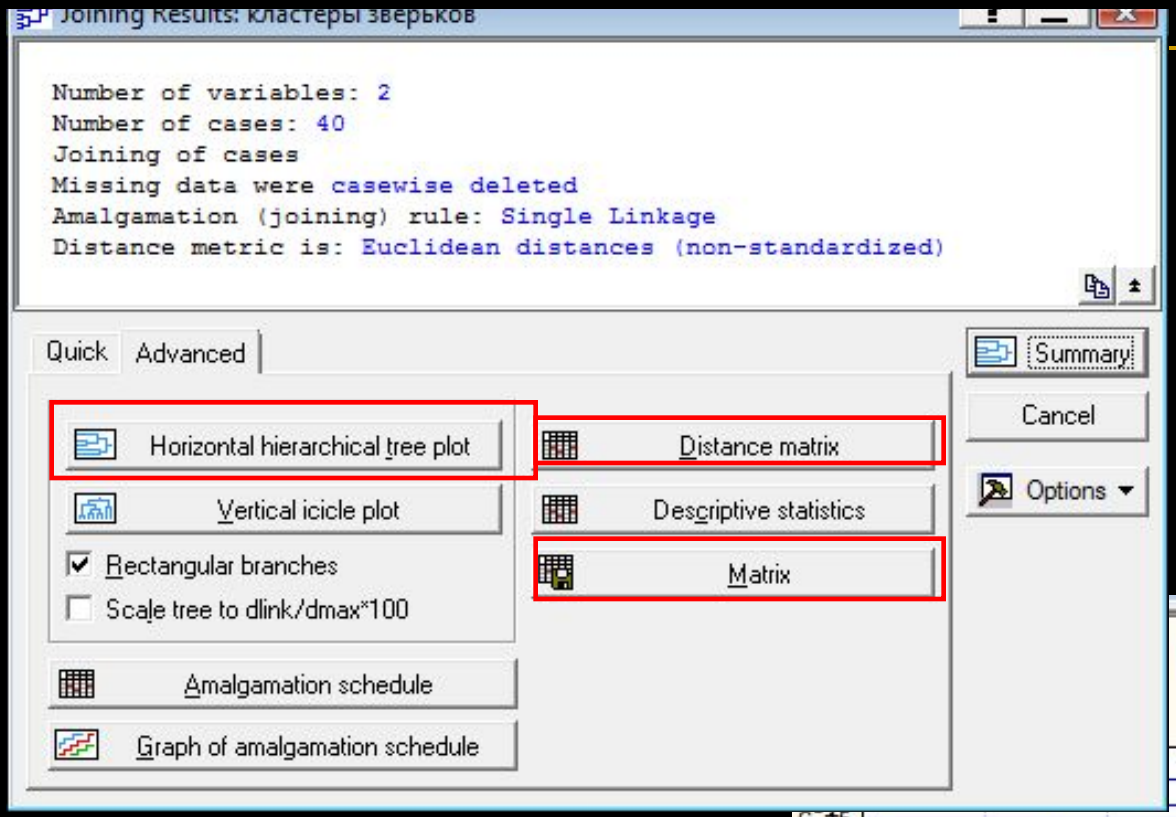


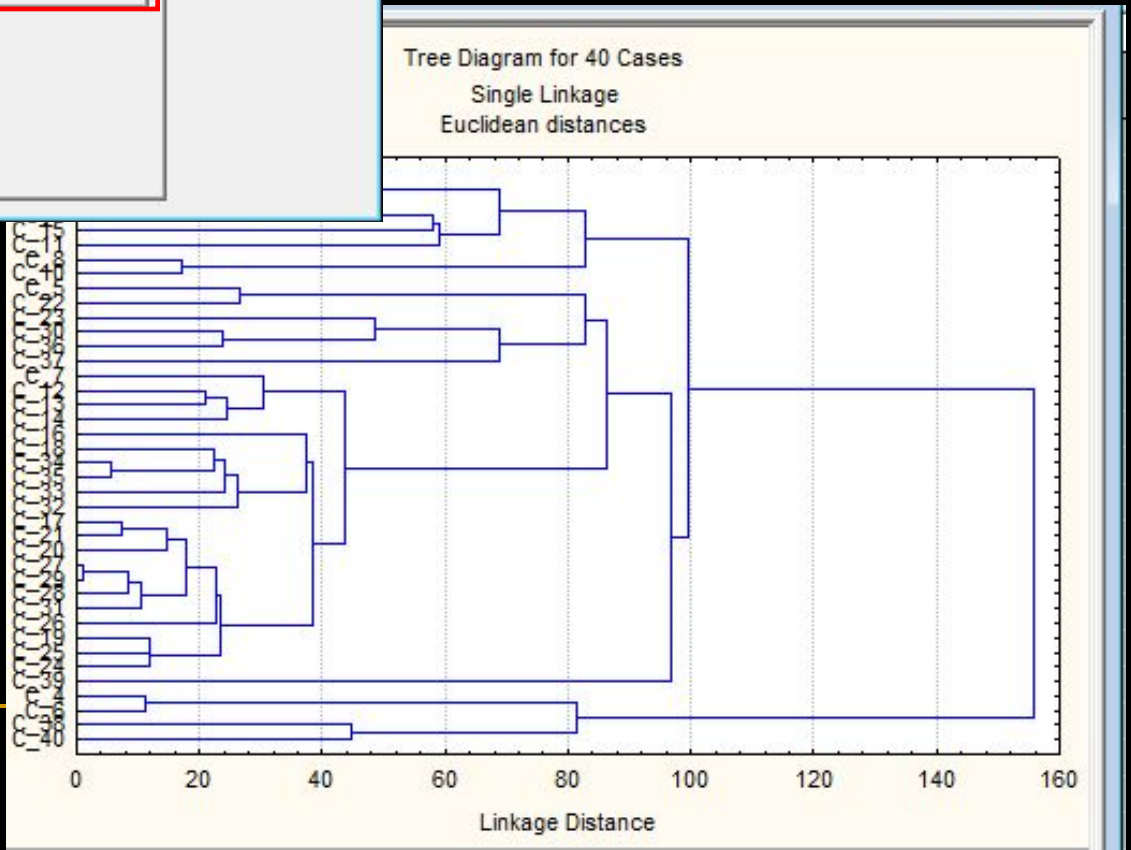
Мы будем рассматривать древовидную кластеризацию; Кластеры будем строить на основе евклидовых дистанций методом ближайшего соседа.



Можно получить матрицу дистанций между наблюдениями (например, для многомерного шкалирования)

Можно нарисовать деревья разного вида и посмотреть, на каких уровнях выделяются кластеры





Посмотрим, на каких расстояниях какие особи объединяются в кластеры

Joining Results: Кластеры зверьков

```
Number of variables: 2
Number of cases: 40
Joining of cases
Missing data were casewise deleted
Amalgamation (joining) rule: Single Linkage
Distance metric is: Euclidean distances (non-standardized)
```

Quick | Advanced

Summary

Cancel

Options

Horizontal hierarchical tree plot

Vertical icicle plot

Rectangular branches

Scale tree to dlink/dmax*100

Distance matrix

Descriptive statistics

Matrix

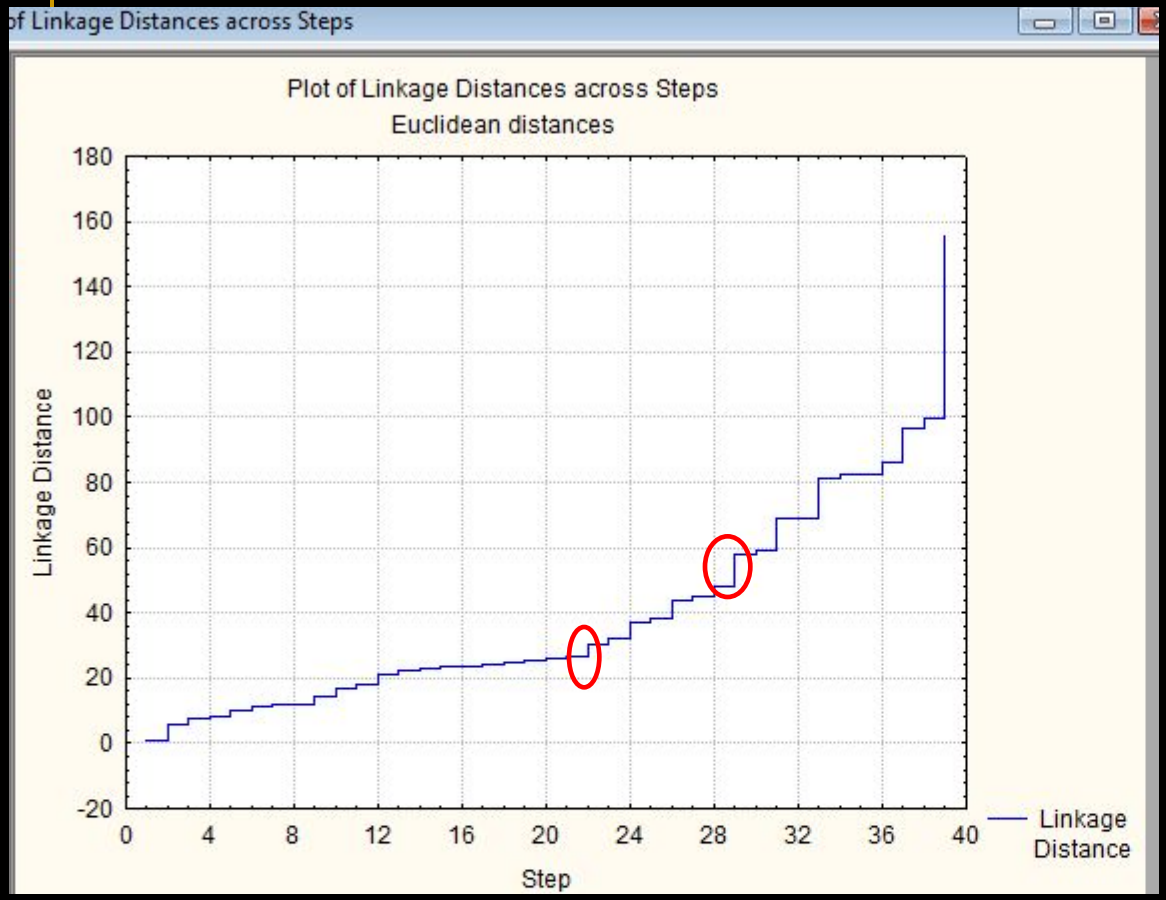
Amalgamation schedule

Graph of amalgamation schedule

algamation Schedule (кластеры зверьков)

Amalgamation Schedule (кластеры зверьков)
Single Linkage
Euclidean distances

linkage distance	Obj. No. 1	Obj. No. 2	Obj. No. 3	Obj. No. 4	Obj. No. 5	Obj. No. 6	Obj. No. 7	Obj. No. 8	Obj. 9
,9409459	C_27	C_29							
5,693835	C_34	C_35							
7,507034	C_17	C_21							
8,316082	C_27	C_29	C_28						
10,32054	C_27	C_29	C_28	C_31					
11,15430	C_4	C_6							
11,94573	C_19	C_25							
12,02269	C_19	C_25	C_24						
14,62312	C_17	C_21	C_20						
17,06330	C_8	C_10							
17,96975	C_17	C_21	C_20	C_27	C_29	C_28	C_31		
21,02993	C_12	C_13							
22,31342	C_18	C_34	C_35						
22,85211	C_17	C_21	C_20	C_27	C_29	C_28	C_31	C_26	
23,30388	C_17	C_21	C_20	C_27	C_29	C_28	C_31	C_26	



По этому графику можно посмотреть, на каком расстоянии происходят скачки в дистанциях присоединения. Если такие скачки есть, значит, есть и кластеры соответствующего размера

Дискриминантный анализ

У нас есть *исходно существующие группы*. Мы ищем переменные, которые лучше всего их разделяют.

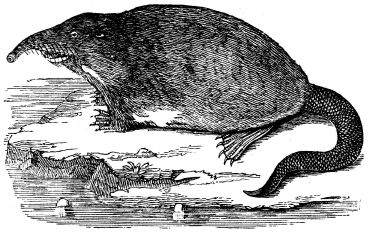
Кластерный анализ

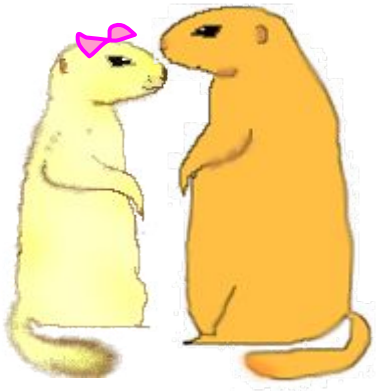
У нас есть несколько *переменных*. Мы на основе них хотим классифицировать выборку – проверить, не объединяются ли наблюдения в группы.

Факторный анализ; многомерное шкалирование

У нас есть несколько *переменных*. Мы хотим классифицировать их или уменьшить их число

Это было последнее занятие нашего семинара!





Спасибо за внимание!

Моя почта: ninavasilieva@gmail.com
(Нина Александровна Васильева)