ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ ГОРОДА МОСКВЫ

МОСКОВСКАЯ ГОРОДСКАЯ СТАНЦИЯ
ЮНЫХ НАТУРАЛИСТОВ

ГИМНАЗИЯ № 1543

ВОЛКОВА П.А., ШИПУНОВ А.Б.

# СТАТИСТИЧЕСКАЯ ОБРАБОТКА ДАННЫХ В УЧЕБНО-ИССЛЕДОВАТЕЛЬСКИХ РАБОТАХ

МОСКВА
2008

**ББК 28.5С**
**В 49**

**В 49** Волкова П.А., Шипунов А.Б., 2008. Статистическая обработка данных в учебно-исследовательских работах. — М.: Экопресс, 60 с.

**ISBN 5-88587-246-4**

В учебном пособии подробно рассматриваются возможности использования программы статистической обработки данных STATISTICA и R при проведении анализа данных, полученных в ходе учебно-исследовательской работы учащихся. Даны теоретические основы статистического анализа.

Пособие предназначено для школьников и учащихся учреждений дополнительного образования — юных натуралистов и экологов, занимающихся исследованиями , а также может использоваться педагогами, работающими в системе дополнительного образования детей.

*Редакторы:*

Багринцева Ю.А., методист МГСЮН, сотрудник Биологического факультета МГУ имени М.В. Ломоносова

Каплан Б.М., методист МГСЮН

Сиднева Е. Н., специалист-эколог МГСЮН, сотрудник Биологического факультета МГУ имени М.В. Ломоносова

*Оформление обложки* — Быков Ю.С.

*Художник* — Андреева Н. М.

---

П. А. Волкова, А. Б. Шипунов

# СТАТИСТИЧЕСКАЯ ОБРАБОТКА ДАННЫХ В УЧЕБНО-ИССЛЕДОВАТЕЛЬСКИХ РАБОТАХ

## Предисловие

Эта книга написана для тех, кто хочет научиться обрабатывать данные. Такая задача возникает очень часто, особенно тогда, когда нужно выяснить ранее неизвестный факт. Например: есть ли эффект от нового лекарства? Или: различаются ли рейтинги двух политиков? Или: как будет меняться курс доллара на следующей неделе? Многие люди думают, что этот неизвестный факт можно выяснить, если просто немного подумать над данными. К сожалению, часто это совершенно не так. Например, по опросу 262 человек, выходящих с избирательных участков, выяснилось, что 52% проголосовало за кандидата А, а 48% — за кандидата В (естественно, я упрощаю ситуацию, ведь всегда есть и проголосовавшие иначе, например, «против всех»). Значит ли это, что кандидат А победил? Подумав, многие сначала скажут «Да», а через некоторое время, я надеюсь, «Кто его знает». Но есть очень простой (с точки зрения современных компьютерных программ) «тест пропорций», который позволяет не только ответить на вопрос (в данном случае «Нет»), но и вычислить, сколько надо было опросить человек, чтобы можно было бы ответить на такой вопрос. В описанном случае это примерно 2500 человек!

В общем, если бы люди знали, что можно сделать методами анализа данных, ошибок и неясностей в нашей жизни стало бы гораздо меньше. К сожалению, ситуация в этой области далека от благополучия. Тем из нас, кто заканчивал институты, часто читали курс «Теория вероятностей и математическая статистика», однако кроме ужаса и/или тоски от длинных математических формул, набитых греческими буквами, большинство ничего из этих курсов не помнит. А ведь на теории вероятностей и основаны большинство методов анализа данных! С другой стороны, ведь совсем не обязательно знать радиофизику для того, чтобы слушать любимую радиостанцию по радиоприемнику. Значит, для того, чтобы анализировать данные в практических целях, не обязательно свободно владеть

математической статистикой и теорией вероятностей. Эту проблему давно уже почувствовали многие английские и американские авторы — названиями типа «Статистика без слез» пестрят книжные полки магазинов, посвященные книгам по анализу данных.

Тут, правда, следует быть осторожным как авторам, так и читателям таких книг: многие методы анализа данных имеют, если можно так выразиться, двойное дно. Их (эти методы) можно применять, глубоко не вникая в сущность используемой там математики, получать результаты и обсуждать эти результаты в отчетах. Однако в один далеко не прекрасный день может выясниться, что данный метод был (с позиции теории, разумеется) совершенно неприменим для Ваших данных, и поэтому полученные результаты и результатами-то назвать нельзя... Что-то похожее происходит при тестировании компьютерных программ: программа может отлично работать, выполняя все, что от нее требуется, но однажды какой-то пустяк (например, какое-то редкое слово или просто сочетание букв, набранное в окне текстового редактора) приводит к ее «зависанию» или даже к более серьезным последствиям. Дело, наверное, в том, что вероятность ошибок растет с увеличением сложности, а методы анализа данных часто очень сложны (в математическом выражении, конечно). В общем, будьте бдительны, внимательно читайте про все *ограничения* методов анализа, а при чтении примеров досконально сравнивайте их со своими данными.

Про примеры: я постарался привести как можно больше примеров, как простых так и сложных, и по возможности из разных областей жизни, поскольку читателями этой книги могут быть люди самых разных профессий. Я старался снизить объем теоретического материала, потому что я знаю — очень многие учатся только на примерах. Поскольку книга посвящена такой компьютерной программе, которая «работает на текстовом коде», логично поместить эти самые коды в отдельные текстовые файлы, а сами файлы (я буду называть их «скрипты») сделать общедоступными. Так я и поступил — все приведенные в книге примеры можно найти на моей Web-странице по адресу http://herba.msu.ru/shipunov/software/R/cbook. Там же находятся и те файлы данных, которые не поставляются вместе с программой. В вышележащей папке читатель может найти для себя еще много интересного.

О структуре книги: главы 1 и 2, по сути, целиком теоретические. Если лень читать общие рассуждения, можно сразу переходить к главе 3. Однако в первых двух главах есть много такой информации, которая позволит Вам в будущем не «наступать на грабли». В общем, решайте сами. В третьей главе самый важный — раздел 3.3, в котором объясняется, как работать с программой R. Если не усвоить этого раздела, все остальное чтение будет почти бесполезным. Советую внимательно прочитать и обязательно *проработать все примеры* из этого раздела. Главы 4 и 5 составляют ядро книги, там рассказывается про самые распространенные методы анализа данных. Наконец, в главах 6 и 7 обсуждаются более редко используемые вещи, хотя обсуждаемые там методы анализа надо, по-моему, использовать гораздо шире. Глава 8, в которой обсуждается общий порядок статистического анализа, подытоживает книгу; в ней еще раз рассказывается про методы, обсуждавшиеся в предыдущих главах. В приложении рассказано еще об одном способе работы с программой R — пакете RCommander, и приведен краткий перечень основных команд R.

# Глава 1. Что такое данные и зачем их обрабатывать?

В этой главе рассказывается о самых общих понятиях анализа данных. Статистики и математики, как и представители любой профессии, выработали свой собственный язык, которым должны, хотя бы частично, овладеть те, кто желает проникнуть в их тайны.

## *1.1. Откуда берутся данные*

«Без пруда не выловишь и рыбку из него» — говорит народная компьютерная мудрость. Действительно, если хочешь анализировать данные, надо их сначала получить. Способов получения данных много. Можно их просто выдумать, но в таком случае результатом анализа будут сведения о том, что творится в Вашей собственной голове, а не в окружающей Вас действительности. Можно взять данные (да и выводы тоже, вот и обрабатывать ничего не надо) из книг тех авторов, которым Вы доверяете — это называется «апелляция к авторитетам», а иногда и просто «плагиат». Такой подход был широко распространен в средние века, а сейчас широко распространен в средней школе. Но опытный учитель знает,

что если на вопрос существуют два ответа — правильный и неправильный, то большинство учеников спишет друг у друга неправильный ответ. Дело в том, что согласно одному из законов Мерфи, «любая проблема имеет простое, изящное и неправильное решение» — неправильный ответ проще. В средние века это приводило к еще большим казусам. Несколько сотен лет студентов учили, что у мухи четыре ноги — это было результатом ошибки одного из монахов-переписчиков трудов Аристотеля. А вот пересчитать ноги у мухи никому в голову не приходило — так велико было доверие к авторитетам!

Чтобы не уподобляться упомянутым выше персонажам, нужно использовать данные, полученные в результате *наблюдения* или *эксперимента*.

**Наблюдением** будем называть такой способ получения данных, при котором воздействие наблюдателя на наблюдаемый объект сведено к минимуму. **Эксперимент** тоже включает наблюдение, но сначала на наблюдаемый объект оказывается заранее рассчитанное воздействие. Для наблюдения очень важно это «сведение воздействия к минимуму». Если этого не сделать, мы получим данные, отражающие не «исконные» свойства объекта, а его реакцию на наше воздействие.

Вот, например, встала задача исследовать, чем питается какое-то редкое животное. Оптимальная стратегия наблюдения здесь состоит в установке скрытых камер во всех местах, где это животное обитает. После этого останется только обработать снятое, чтобы определить вид пищи. Очень часто, однако, оптимальное решение совершенно невыполнимо, и тогда пытаются обойтись, скажем, наблюдением за животным в зоопарке. Ясно, что в последнем случае на объект оказывается воздействие, и немалое. В самом деле, животное поймали, привезли в совершенно нетипичные для него условия, да и корм, скорее всего, будет непохож на тот, каким оно питалось на родине. В общем, если наблюдения в зоопарке поставлены грамотно, то выяснено будет не то, чем вообще питается данное животное, а то, чем оно питается при содержании в определенном зоопарке. К сожалению, многие (и исследователи, и те, кто потом читает их отчеты) часто не видят разницы между этими двумя утверждениями, что может привести к очень серьезным последствиям.

Вернемся к примеру из предисловия. Предположим, мы опрашиваем выходящих с избирательных участков. Часть людей, конечно, вообще окажется отвечать. Часть ответит что-нибудь, не относящееся к делу. Часть вполне может намеренно или случайно исказить свой ответ. Часть ответит правду. И все это серьезным образом зависит от наблюдателя — человека, проводящего опрос.

Даже упомянутые выше скрытые камеры приведут к определенному воздействию: они же скрытые, но не невидимые и невесомые. Нет никакой гарантии, что наше животное или его добыча не отреагирует на них. А кто будет ставить камеры? Если это люди, то чем больше камер поставить, тем сильнее будет воздействие на окружающую среду. Сбрасывать с вертолета? Сами понимаете, к чему это может привести.

В общем, из сказанного должно быть понятно, что наблюдение «в чистом виде» более или менее неосуществимо, поскольку всегда будет внесено какое-нибудь воздействие. Поэтому для того, чтобы адекватно работать с данными наблюдений, надо всегда четко представлять, как они проводились. Если воздействие было значительным, то надо представлять (хотя бы теоретически) какие оно могло повлечь изменения, а в отчете обязательно указать на те ограничения, которые были вызваны способом наблюдения. Не следует без необходимости применять экстраполяцию: это значит, что если мы увидели, что А делает Б, нельзя писать «А всегда делает Б» и даже «А обычно делает Б». Можно лишь писать нечто вроде «в наших наблюдениях А делал Б, это позволяет предположить, что Б для него — обычное дело».

У эксперимента свои проблемы. Наиболее общие из них — это точный учет воздействия и наличие контроля. Например, мы исследуем действие нового лекарства. Классический эксперимент состоит в том, что выбираются две группы больных (*как* выбрать такие группы, *сколько* должно быть человек и пр. рассмотрено в последующих разделах). Всем больным сообщают, что проводится исследование нового лекарства, но его дают только больным первой группы, остальные получают так называемое **плацебо**, внешне неотличимое от настоящего лекарства, но не содержащее ничего лекарственного. Зачем это делается? Дело в том, что если больной будет знать, что ему дают «ненастоящее» лекарство, то это скажется на эффективности лечения, потому что результат зависит не только от того, что

больной пьет, но и от того, что он чувствует. Иными словами, психологическое состояние больного — это дополнительный фактор воздействия, от которого в эксперименте лучше избавиться. Очень часто не только больным, но и их врачам не сообщают, кому дают плацебо, а кому — настоящее лекарство («двойной слепой метод»). Это позволяет гарантировать, что и психологическое состояние врача не повлияет на исход лечения.

Группа, которой дают плацебо (она называется **контроль**), нужна для того, чтобы отделить эффект, который может произвести лекарство, от эффекта какого-нибудь постороннего внешнего фактора. Известно, например, что уменьшение длины светового дня в октябре-декабре провоцирует многие нервные заболевания. Если наше исследование придется как раз на эти месяцы и у нас не будет контроля, то увеличение частоты заболеваний мы вполне можем принять за результат применения лекарства.

## *1.2. Генеральная совокупность и выборка*

«Статистика знает все» — писали Ильф и Петров в «Двенадцати стульях», имея в виду то, что обычно называют статистикой — сбор всевозможной информации обо всем на свете. Чем полнее собрана информация, тем, как считается, лучше. Однако лучше ли?

Возьмем простой пример. Допустим, фирма-производитель решила выяснить, какой из двух сортов производимого мороженого предпочитают покупатели. Проблем бы не было, если бы все мороженое продавалось в одном магазине. На самом же деле продавцов несчетное множество: это оптовые рынки и гипермаркеты, средние и малые магазины, киоски, отдельные мороженщики с тележками, те, кто торгует в пригородных поездах и т.п. Можно попробовать учесть доход от продажи каждого из двух сортов. Если они ст'оят одинаково, то б'ольшая сумма дохода должна отразить больший спрос. Представим, однако, что спрос одинаков, но по каким-то причинам мороженое первого сорта тает быстрее. Тогда потерь при его транспортировке будет в среднем больше, продавцы будут покупать его несколько чаще, и получится, что доход от продажи первого сорта будет несколько выше, чем от второго. Это рассуждение, конечно, упрощает реальную ситуацию, но подумайте, сколько других неучтенных факторов стоит на пути такого способа подсчета! Анализ товарных чеков получше,

однако многие конечные продавцы таких чеков не имеют, и поэтому в анализ не попадут. А нам-то необходимо как раз учесть спрос покупателей, а не промежуточных продавцов.

Можно поступить иначе — раздать всем конечным продавцам анкеты, в которых попросить указать, сколько какого мороженого продано; а чтобы анкеты были обязательно заполнены, вести с этими продавцами дела только при наличии заполненных анкет. Только ведь никто не будет контролировать, *как* продавцы заполняют анкеты... Вот и получит фирма большую, подробную сводную таблицу о продажах мороженого, которая ровным счетом ничего отражать не будет.

Как же поступить? Здесь на помощь приходит идея **выборочных исследований**. Всех продавцов не проконтролируешь, но ведь нескольких-то можно! Надо выбрать из общего множества несколько торговых точек (*как* выбирать — это особая наука, см. об этом ниже) и проконтролировать тамошние продажи силами самой фирмы или такими нанятыми людьми, которым можно доверять. В результате мы получим результат, который является частью общей картины. Теперь самый главный вопрос — можно ли этот результат распространить на всю совокупность продаж? Оказывается, можно, поскольку на основе теории вероятностей уже много лет назад была создана **теория выборочных исследований**. Ее-то и называют чаще всего математической статистикой, или просто статистикой.

Пример с мороженым показывает важную вещь: выборочные исследования могут быть (и часто бывают) значительно более точными (в смысле соответствия реальности), чем сплошные.

Еще один хороший пример на эту же тему есть в результатах сплошной переписи населения России 1897 г. Если рассмотреть численность населения по возрастам, то получается, что максимальные численности («пики») имеют возраста кратные 5 и в особенности кратные 10. Понятно, как это получилось. БОльшая часть населения в те времена была неграмотна, и свой возраст помнила только приблизительно, с точностью до пяти или до десяти лет. Чтобы все-таки узнать, каково было распределение по возрастам на самом деле , нужно не увеличивать данные, а наоборот, создать выборку нескольких процентов населения и провести комплексное исследование, основанное на перекрестном анализе нескольких источников: документов,

свидетельств и личных показаний. Это даст гораздо более точную картину, нежели сплошная перепись.

Естественно, сам процесс создания выборки может являться источником ошибок. Их принято называть **ошибками репрезентативности**. Однако правильная организация выборки позволяет их избежать. А поскольку с выборкой можно проводить гораздо более сложные исследования, чем со всеми данными (их называют **генеральной совокупностью**), те ошибки (**ошибки точности**), которые возникают при сплошном исследовании, в выборочном исследовании можно исключить.

## *1.3. Как получать данные*

В предыдущих разделах неоднократно упоминалось, что от правильного подбора выборки серьезным образом будет зависеть качество получаемых данных. Собственно говоря, есть два основных принципа составления выборки: повторности и рандомизация. Повторности нужны для того, чтобы быть более уверенными в полученных результатах, а рандомизация — для того, чтобы избежать отклонений, вызванных посторонними причинами.

Принцип повторностей предполагает, что один и тот же эффект будет исследован несколько раз. Собственно говоря, для этого мы в предыдущих примерах опрашивали *множество* избирателей, ловили в заповеднике *много* животных, подбирали группы из *нескольких десятков* больных и контролировали *различных* продавцов мороженого. Нужда в повторностях возникает оттого, что все объекты (даже только что изготовленные на фабрике изделия), пусть в мелочах, но отличаются друг от друга. Эти отличия способны затуманить общую картину, если мы станем изучать объекты поодиночке. И наоборот, если мы берем несколько объектов сразу, их различия как бы «взаимно уничтожаются».

Не стоит думать, что создать повторности — простое дело. К сожалению, часто именно небрежное отношение к повторностям сводит на нет результаты вроде бы безупречных исследований. Главное правило — *повторности должны быть независимы друг от друга*. Это значит, например, что нельзя в качестве повторностей рассматривать данные, полученные в последовательные промежутки времени с одного и того же объекта или с одного и того же места. Предположим, что мы хотим

определить размер лягушек какого-то вида. Для этого с интервалом в 15 минут (чтобы лягушки успокоились) ловим сачком по одной лягушке. Как только наберется двадцать лягушек, мы их меряем и вычисляем средний размер. Однако такое исследование не будет удовлетворять правилу независимости, потому что каждый отлов окажет влияние на последующее поведение лягушек (например, к концу лова будут попадаться самые смелые, или, наоборот, самые глупые). Еще хуже использовать в качестве повторностей последовательные наблюдения за объектом. Например, в некотором опыте выясняли скорость зрительной реакции, показывая человеку на доли секунды предмет, а затем спрашивая, что это было. Всего исследовали 10 человек, причем каждому показывали предмет пять раз. Авторы опыта посчитали, что у них было 50 повторностей, однако на самом деле — только десять. Это произошло потому, что каждый следующий показ не был независим от предыдущего (человек мог, например, научиться лучше распознавать предмет).

Надо быть осторожным не только с данными, собранными в последовательные промежутки времени, но и просто с данными, собранными с одного и того же места. Например, если мы определяем качество телевизоров, сходящих с конвейера, не годится в качестве выборки брать несколько штук подряд — с большой вероятностью они изготовлены в более близких условиях, чем телевизоры, взятые порознь, и стало быть, их характеристики не независимы друг от друга.

Второй важный вопрос о повторностях — сколько надо собрать данных. Есть громадная литература по этому поводу, но ответа, в общем, два: (1) столько, сколько возможно и (2) 30. Выглядящее несколько юмористически «правило 30» освящено десятилетиями опытной работы. Считается, что выборки, меньшие 30, следует называть малыми, а большие — большими. Отсюда большое значение, которое придают числу тридцать в анализе данных. Бывает так, что и 30 собрать нельзя, однако огорчаться этому не ст'оит, поскольку многие процедуры анализа данных способны работать с очень малыми выборками, в том числе из пяти и даже из трех повторностей. Следует, однако, иметь в виду, что чем меньше повторностей, тем менее достоверными будут выводы.

Существуют, кроме того, специальные методы, которые позволяют посчитать, сколько надо собрать данных для того чтобы с определенной

вероятностью высказать некоторое утверждение. Это так называемые «тесты мощности», два таких теста будут рассмотрены ниже, в главах 4 и 5.

Рандомизация — еще одно условие создания выборки, и также «с подвохом». Очень часто исследователи полагают, что выбрали свои объекты случайно (проделали рандомизацию), в то время как на самом деле их материал был подобран с большими отклонениями от случайности. Предположим, мне поручено случайным образом отобрать сто деревьев в лесу, чтобы впоследствии померить степень накопления тяжелых металлов в листьях. Как я буду выбирать деревья? Если просто ходить по лесу и собирать листья с разных деревьев, с большой вероятностью они не будут собранными случайно, потому что вольно или невольно я буду собирать листья, чем-то привлекшие мое внимание (необычностью, окраской, доступностью). Этот метод, стало быть, не годится. Возьмем метод посложнее — для этого нужна карта леса с размеченными координатами. Я выбираю случайным образом два числа, например, 123 м к западу и 15 м к югу от точки, находящейся примерно посередине леса, затем высчитываю это расстояние на местности и выбираю дерево, которое ближе всего к нужному месту. Будет ли такое дерево выбрано случайно? Оказывается, нет. Ведь деревья растут группами, поэтому у деревьев, растущих плотно (например, у елок), шанс быть выбранными окажется значительно меньше, чем у редко растущих дубов. Важным условием рандомизации, таким образом, является то, что *каждый объект должен иметь абсолютно те же самые шансы быть выбранным, что и все прочие объекты*. Как же быть? Надо просто перенумеровать все деревья, а затем выбрать сто номеров по жребию. Но это только звучит просто, а попробуйте так сделать! А если надо сравнить 20 различных лесов?.. В общем, требование рандомизации часто оборачивается весьма серьезными затратами на проведение исследования. Естественно поэтому, что часто рандомизацию осуществляют лишь частично. Например, в нашем случае можно случайно выбрать направление, протянуть в этом направлении бечевку через весь лес, а затем посчитать, скольких деревьев касается бечевка и выбрать каждое энное (пятое, пятнадцатое и т.п.) дерево так, чтобы всего в выборке оказалось 100 деревьев. Заметьте, что в данном случае метод рандомизации состоит в том, чтобы внести в исследуемую среду *такой порядок, которого там заведомо нет*. Конечно, у этого последнего метода есть недостатки, а какие — попробуйте «вычислить» сами [задача 1].

Теперь Вы знаете достаточно, чтобы ответить на еще один вопрос. В одной лаборатории изучали эффективность действия ядохимикатов на жуков-долгоносиков (их еще называют «слоники»). Для этого химикат наносили на фильтровальную бумагу, а бумагу помещали в стеклянную чашку с крышкой (чашку Петри). Жуков выбирали из банки, в которой их разводили для опытов, очень простым способом: банку с жуками открывали, и первого выползшего на край жука пересаживали в чашку с ядохимикатом. Затем засекали, сколько пройдет времени от посадки жука в банку до его гибели. Потом брали другого жука, и так повторяли 30 раз. Потом меняли ядохимикат и начинали опыт сначала. Но однажды один умный человек заметил, что в этом эксперименте самым сильным всегда оказывался тот химикат, который был взят для исследования первым. Как Вы думаете, в чем тут дело? Какие нарушения принципов повторности и рандомизации были допущены? Как надо было поставить этот опыт? [Задача 2]

Для рандомизации, конечно, существует предел. Если мы хотим выяснить возрастной состав посетителей какого-то магазина, не нужно во имя рандомизации опрашивать прохожих с улицы. Нужно четко представлять себе генеральную совокупность, с которой идет работа, и не выходить за ее границы. Помните пример с питанием животного? Если генеральная совокупность — это животные данного вида, содержащиеся *в зоопарках*, нет смысла добавлять к исследованию данные о питании этих животных *в домашних условиях*. Если же такие данные просто необходимо добавить (например, потому что данных из зоопарков очень мало), то тогда генеральная совокупность будет называться «множество животных данного вида, содержащихся *в неволе*».

Интересный вариант рандомизации используют, когда в эксперименте исследуются одновременно несколько взаимодействий. Например, мы хотим выяснить эффективность разных типов солевой засыпки тротуаров. Для этого логично выбрать (рандомизация!) несколько разных (по возрасту застройки, плотности населения, расположению и пр.) участков города и внутри каждого участка случайным образом распределить разные типы засыпок. Потом можно, например, фиксировать (в баллах или как-нибудь еще) состояние тротуаров каждый день после нанесения засыпки, можно также повторить опыт при разной погоде. Такой подход называется «блочный дизайн». Блоками здесь являются разные участки города, а повторность обеспечивается тем, что в каждом блоке повторяются одни и те

же воздействия. При этом даже не обязательно повторять однотипные воздействия по несколько раз внутри блоков, важно выбрать побольше отличающихся друг от друга блоков. Можно считать разными блоками и разные погодные условия, и тогда у нас получится «вложенный блочный дизайн»: в каждый погодный блок войдет несколько «городских» блоков, и уже внутри этих блоков будут повторены все возможные воздействия (типы засыпок).

В области рандомизации лежит еще одно коренное различие между наблюдением и экспериментом. Допустим, мы изучаем эффективность действия какого-то лекарства. Вместо того, чтобы подбирать две группы больных, использовать плацебо и т.п., можно просто порыться в архивах и подобрать соответствующие примеры (30 случаев применения лекарства и 30 случаев непримерения), а затем проанализировать разницу между группами (например, число смертей в первый год после окончания лечения). Однако такой подход сопряжен с опасностью того, что на наши выводы окажет влияние какой-то (или какие-то) неучтенный фактор, выяснить наличие которого из архивов невозможно. Мы просто не можем гарантировать, что соблюдали рандомизацию, анализируя архивные данные. К примеру, первая группа (случайно!) окажется состоящей почти целиком из пожилых людей, а вторая — из людей среднего возраста. Ясно, что это окажет воздействие на выводы. Поэтому в общем случае эксперимент всегда предпочтительней наблюдения.

## *1.4. Что ищут в данных*

Прочитав предыдущие разделы, читатель, наверное, уже не раз задавался вопросом: «Если так все сложно, зачем он вообще, этот анализ данных? Неужели и *так* не видно, что в один магазин ходит больше народу, одно лекарство лучше другого и т.п.?» В общем, *так* бывает видно довольно часто, но обычно тогда, когда либо (1) данных и/или исследуемых факторов очень мало, либо (2) разница между ними очень резка. В этих случаях, действительно, запускать всю громоздкую машину анализа данных не стоит. Однако гораздо чаще встречаются случаи, когда названные выше условия не выполняются. Давно, например, доказано, что средний человек может одновременно удержать в памяти лишь 5–9 объектов. Стало быть, анализировать в уме данные, которые насчитывают больше 10 компонентов, уже нельзя. А значит, не обойтись без каких-нибудь, пусть и самых

примитивных (типа вычисления процентов и средних величин), методов анализа данных.

Бывает и так, что внешне очевидные результаты не имеют под собой настоящего основания. Вот, например, одно из исследований насекомых-вредителей. Агрономы определяли, насколько сильно вредят кукурузе гусеницы кукурузного мотылька. Получились вполне приемлемые результаты: разница в урожае между пораженными и непораженными растениями почти вдвое. Казалось, что и обрабатывать ничего не надо — «и так все ясно». Однако нашелся вдумчивый исследователь, который заметил, что пораженные растения, различающиеся по степени поражения, не различаются по урожайности. Здесь очевидно что-то не так: если гусеницы вредят растению, то чем сильнее они вредят, тем меньше должен быть урожай. Стало быть, на какой-то стадии исследования произошла ошибка. Скорее всего, дело было так: для того, чтобы мерять урожайность, среди здоровых растений отбирали самые здоровые (во всех смыслах этого слова), ну а среди больших старались подобрать самые хилые. Вот эта ошибка репрезентативности и привела к тому, что возникли такие «хорошие» результаты. Обратите внимание, что только анализ взаимосвязи поражение-урожай (на языке анализа данных он называется «регрессионный анализ», см. главу 5) привел к выяснению истинной причины. А кукурузный мотылек, оказывается, почти и не вредит кукурузе...

Итак, анализ данных необходим всегда, когда результат неочевиден и часто даже тогда, когда он кажется очевидным. Теперь разберемся, к каким последствиям может привести анализ, что он умеет.

Во-первых, анализ данных умеет давать общие характеристики для больших выборок. Эти характеристики могут отражать так называемую центральную тенденцию, то есть число (или ряд чисел), вокруг которых, как пули вокруг десятки в мишени, «рассыпаны» данные. Всем известно, как считать среднее значение, но существует еще немало полезных характеристик «на ту же тему». Другая характеристика — это разброс, который отражает не вокруг *чего* «рассыпаны» данные, а насколько сильно они рассыпаны.

Во-вторых, можно проводить сравнения между разными выборками. Например, можно выяснить, в какой из групп больных инфарктом миокарда частота сметрей в первый год после лечения выше — у тех, к кому

применяли коронарное шунтирование или у тех, к кому применяли только медикаментозные способы лечения. «На взгляд» этой разницы может и не быть, а если она и есть, то где гарантия того, что эти различия не вызваны случайными причинами, не имеющими отношения к лечению? Скажем, заболел человек острым аппендицитом и умер после операции: к лечению инфаркта это может не иметь никакого отношения. Сравнение данных при помощи *статистических тестов* позволяет выяснить, насколько велика вероятность, что различия между группами вызваны случайными причинами. Заметьте, что гарантий анализ данных тоже не дает, за то позволяет оценить (численным образом) шансы. Кстати говоря, анализ данных позволяет оценить и упомянутые выше общие характеристики — вычислить так называемые «доверительные интервалы» (см. главу 4).

Третий тип результата, который можно получить, анализируя данные — это сведения о взаимосвязях. Изучение взаимосвязей, наверное, самый серьезный и самый развитый раздел анализа данных. Существует множество методик выяснения и, главное, проверки «качества» связей. В дальнейшем нам понадобятся сведиения о том, какие бывают взаимосвязи. Есть так называемые **соответствия**, например, когда два явления чаще встречаются вместе, нежели по отдельности (как гром и молния). Соответствия нетрудно найти, но трудно «посчитать», то есть как-то измерить. Следующий тип взаимосвязей — это **корреляции**. Корреляции показывают силу взаимосвязи, но не могут определить ее направления. Другими словами, если выяснилась корреляция между качанием деревьев и ветром, то нельзя решить, дует ли ветер оттого что деревья качаются или наоборот. Наконец, есть **зависимости**, для которых можно измерить и силу, и направление, и оценить, насколько вероятно то, что они — результат случайных причин. Кстати говоря, последнее можно, как водится в анализе данных, сделать и для корреляций, и даже для соответствий. Еще одно свойство зависимостей состоит в том, что можно *предсказать* как будет «вести» себя зависимая переменная в каких-нибудь до сих пор не опробованных условиях. Например, можно прогнозировать колебания спроса, устойчивость балок при землятресении, интенсивность поступления больных и т.п.

И наконец, анализ данных можно использовать для установления *структуры*. Это самый сложный тип анализа, поскольку для выяснения структуры обычно используются *сразу несколько характеристик*. Есть и специальное название для такой работы — «многомерная статистика».

Самое главное, на что способен многомерный анализ — это создание и проверка качества *классификации* объектов. В умелых руках хорошая классификация очень полезна. Вот, например, мебельной фабрике потребовалось выянить, какую мебель как лучше перевозить: в разобранном или в собранном виде. Рекомендации по перевозке зависят от уймы причин (сложность сборки, хрупкость, стоимость, наличие стеклянных частей, наличие ящиков и полок и т.д.). Одновременно оценить эти факторы может лишь очень умелый человек. Однако существуют методы анализа, которые с легкостью разделят мебель на две группы, а заодно и проверят качество классификации, например, ее соответствие сложившейся практике перевозок.

Существует и другой подход к результатам анализа данных. В нем все методы делятся на *предсказательные* и *описательные*. К первой группе методов относится все, что можно статистически оценить, то есть выяснить, *с какой вероятностью* может быть верным или неверным наш вывод. Ко второй — методы, которые «просто» сообщают информацию о данных без подтверждения какими-либо вероятностыми методами. В последние годы все для большего числа методов находятся способы их вероятностной оценки, и поэтому превая группа все время увеличивается.

## Ответы на вопросы

**[Ответ к задаче 1.]** В этом случае шанс быть выбранными у елок выше, чем у дубов. Кроме того, лес может иметь какую-то структуру именно в выбранном направлении, и поэтому одной такой `диагонали' будет недостаточно для того, чтобы отобразить весь лес. Чтобы улучшить данный метод, надо провести несколько `диагоналей', а расстояния между выбираемыми деревьями по возможности увеличить.

**[Ответ к задаче ***.]**

```
> prop.test(0.48*262, 262)
... p-value = 0.5581
> power.prop.test(p1=0.48, p2=0.52, power=0.8)
... n = 2451.596
```

**[Ответ к задаче 2.]** Дело в том, что первыми вылезают самые активные особи, а чем активнее особь, тем быстрее она набирает на лапки смертельную дозу ядохимиката, и, стало быть, быстрее гибнет. Это и было нарушением принципа рандомизации. Кроме того, нарушался принцип повторности: в чашку последовательно сажали жука за жуком, что не могло не повлиять на исход опыта. Для того чтобы поставить опыт правильно, надо было сначала подготовить (30 х на количество ядохимикатов) чашек, столько же листочков с бумагой, _случайным образом_ распределить ядохимикаты по чашкам, а затем перемешать жуков в банке, достать соответствующее количество и рассадить по чашкам.

# Статистическая обработка данных в школьных исследовательских работах

## 1. Еще немного необходимой теории

### 1.1. Статистические гипотезы

Итак, статистическая выборка должна быть репрезентативной (то есть адекватно характеризовать генеральную совокупность). Но как же мы можем знать, репрезентативна ли выборка, если мы не исследовали всю генеральную совокупность? Этот логический тупик называют **парадоксом выборки**. Хотя мы и обеспечиваем репрезентативность выборки соблюдением двух основных принципов ее создания (рандомизации и повторности), но некоторая неопределенность все же остается. Кроме того, если мы принимаем вероятностную точку зрения на происхождение наших данных (они получены путем случайного выбора), то все дальнейшие суждения, основанные на этих данных, будут иметь вероятностный характер. Таким образом, мы никогда не сможем на основании нашей (репрезентативной!) выборки со 100% уверенностью судить о свойствах генеральной совокупности. Мы можем лишь выдвигать гипотезы и вычислять их вероятность.

Великие философы науки (например, Карл Поппер) постулировали, что мы ничего не можем доказать, мы можем лишь что-нибудь опровергнуть (ломать -- не строить!). Действительно, пусть мы соберем 1000 фактов, подтверждающих какую-нибудь теорию, это не будет значить, что мы ее доказали. Вполне возможно, что 1001-ый (или 1 000 001-ый) факт опровергнет эту теорию. Поэтому при любом статистическом тесте выдвигаются две противоположных гипотезы. Одна -- то что мы хотим доказать (но не можем!) -- называется **альтернативная гипотеза** (ее обозначают H1). Другая -- противоречащая альтернативной -- **нулевая гипотеза** (обозначается H0). Нулевая гипотеза всегда является предположением об отсутствии чего-либо (например, зависимости одной переменной от другой или различия между двумя выборками). Стало быть, мы не можем доказать альтернативную гипотезу, а можем лишь опровергнуть нулевую гипотезу и принять альтернативную (чувствуете

разницу?). Если же мы не можем опровергнуть нулевую гипотезу, то мы вынуждены принять ее.

### 1.2. Статистические ошибки

Естественно, что когда мы делаем любые предположения (в нашем случае выдвигаем статистические гипотезы), мы можем ошибаться (в нашем случае делать статистические ошибки). Начертим простую таблицу:

| | | **для генеральной совокупности** | |
|---|---|---|---|
| | | верна H0 | верна H1 |
| **для выборки** | принимаем H0 | правильно! | статистическая ошибка второго рода |
| | принимаем H1 | статистическая ошибка первого рода | правильно! |

Если мы приняли для выборки H0 (нулевую гипотезу) и она верна для генеральной совокупности, то мы правы, и все в порядке. Аналогично и для H1 (альтернативной гипотезы). Ясно, что мы не можем знать, что в действительности верно для генеральной совокупности, и сейчас просто рассматриваем все логически возможные варианты.

Вот если мы приняли для выборки альтернативную гипотезу, а она оказалась не верна для генеральной совокупности, то мы совершили так называемую **статистическую ошибку первого рода** (нашли несуществующую закономерность). Вероятность того, что мы совершили эту ошибку (так называемое **p-value**) всегда отображается при проведении любых статистических тестов при помощи компьютерных программ. Очевидно, что если вероятность этой ошибки достаточно высока, то мы должны отвергнуть альтернативную гипотезу. Возникает естественный вопрос: какую вероятность считать достаточно высокой? Однозначного ответа на этот вопрос нет. В биологии принято соглашение считать пороговым значением 0,05 (то есть альтернативная гипотеза отвергается, если вероятность ошибки при ее принятии больше или равна 5%). В медицине ценой ошибки нередко являются человеческие жизни, поэтому там пороговое значение принято равным 0,01 или даже 0,001 (то есть решение о существовании закономерности принимается, если вероятность ошибки ничтожна).

Таким образом, решение о результатах статистических тестов принимается главным образом на основании вероятности статистической ошибки первого рода. Степень уверенности исследователя в том, что заключение, сделанное на основании статистической выборки, будет справедливо и для генеральной совокупности, отражает **статистическая достоверность.** Допустим, если вероятность статистической ошибки первого рода равна 3%, то говорят, что найденная закономерность достоверна с вероятностью 97%. А если вероятность статистической ошибки первого рода равна, например, 23%, то говорят, что достоверной закономерности не найдено.

В случае, если мы принимаем нулевую гипотезу для выборки, в то время как для генеральной совокупности справедлива альтернативная гипотеза, то мы совершаем **статистическую ошибку второго рода** (не замечаем существующей закономерности). Этот параметр характеризует так называемую **мощность статистического теста** (чем меньше вероятность статистической ошибки второго рода, то есть, чем меньше вероятность не заметить несуществующую закономерность, тем более мощным является тест).

## 2. Обработка данных

### 2.1. Как можно обрабатывать данные?

Можно обрабатывать данные вручную. Чертить графики на миллиметровке и вычислять значения статистических критериев по формулам на калькуляторе. Так делали до массового распространения компьютеров. В наше время поступать так было бы просто неразумно. Кроме того, некоторые виды статистического анализа (например, многомерные методы, которых мы не будем касаться в этом пособии) просто не могут быть проведены силами человеческого разума, оперирующего только тремя пространственными измерениями.

Можно воспользоваться программами общего назначения (например, Excel). Это не очень удобно, поскольку такие программы в общем-то не предназначены для статистической обработки данных. Какие-то функции в таких программах отсутствуют, какие-то реализованы не очень удачно.

Самое разумное -- это воспользоваться специализированными компьютерными программами для статистической обработки данных. Они бывают двух типов. С привычным пользователям Windows интерфейсом меню и кнопок (например, STATISTICA) и с привычной пользователям DOS командной строкой (например, R). Различие между этими типами программ примерно такое же, как между автоматом для продажи напитков и супермаркетом. Программы с командной строкой предоставляют пользователю гораздо больше возможностей для обработки данных, такие программы имеют и специальный язык, что позволяет пользователю самостоятельно создавать новые алгоритмы обработки данных, отвечающие его потребностям. Зато программы с интерфейсом меню и кнопок освобождают пользователя от необходимости осваивать новый командный язык и позволяют быстро провести многие стандартные виды статистической обработки данных. Думаю, что начинать знакомство со статистической обработкой данных лучше с программ с интерфейсом меню и кнопок.

Наиболее удачной и широко распространенной из них, на мой взгляд, является программа STATISTICA, на примере которой я и буду излагать обработку данных. Эта программа имеет хорошо написанную помощь, поэтому я буду по ходу изложения лишь давать указания, в каком разделе меню находится интересующий нас тип обработки, и приводить в скобках английский перевод статистических терминов, которые используются в программе. В STATISTICA 5.5, которой я рекомендую пользоваться, типы статистической обработки сгруппированы в модули. В настоящем пособии я описываю лишь некоторые наиболее часто употребимые методы статистической обработки данных. Все они реализованы в двух модулях программы STATISTICA 5.5: базовые статистики (*Basic statistics*) и непараметрические методы (*Nonparametrics/Distrib.*). Еще один описываемый метод находится в модуле дисперсионный анализ (*ANOVA/MANOVA*). Несколько лет назад появилась новая версия программы: STATISTICA 6 (деления на модули там нет), которой я не рекомендую пользоваться, так как никаких существенных улучшений там не произошло, а интерфейс стал гораздо более запутанным.

На всякий случай я буду приводить (таким вот мелким шрифтом) и краткие указания, как тот или иной тип обработки данных проделать в R. В этом пособии я не ставила перед собой цели обучения языку и идеологии R (об этом можно прочесть в многочисленных руководствах, написанных куда более компетентными в этих

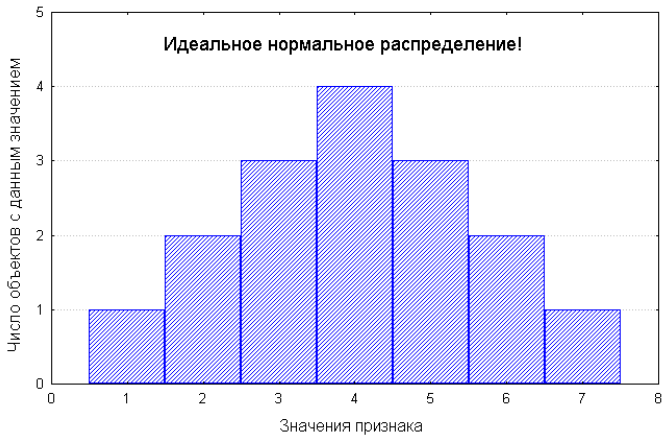
вопросах людьми). Будем считать, что ваши данные представлены объектом data (непонятно? Не читайте мелкий шрифт, пока не узнаете про работу в R больше). Изначально R был ориентирован на статистическую обработку данных, а STATISTICA -- на графическую, поэтому не удивляйтесь, что графические возможности в R без приложения особых усилий ограничены, зато проведение всяких статистических тестов реализовано более элегантно, чем в STATISTICA.

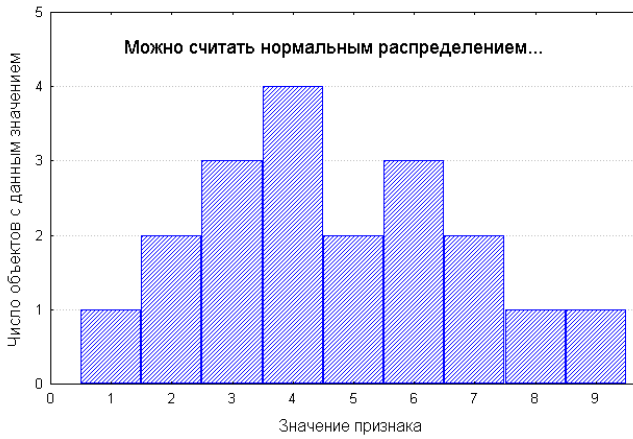
### 2.2. Как начинать работу с данными?

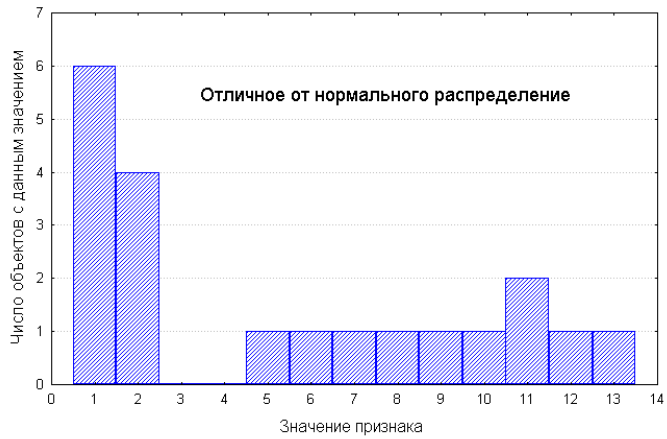
Для начала надо понять, какими типами переменных (признаков) представлены наши данные. Выделяют три основных **типа переменных**:

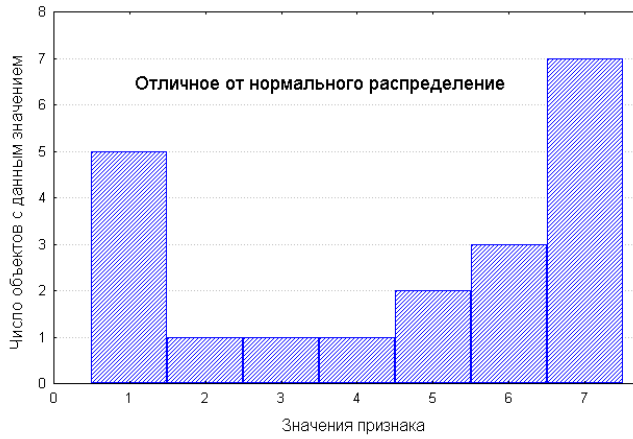
1. **непрерывные** -- представлены действительными числами (например, длина или вес)
2. **дискретные** -- представлены целыми, как правило, положительными, числами (например, число телефонных звонков, поступивших в контору за день). Понятно, что в день телефон может зазвонить, например, 6 или 7 раз, но никак не 6,3 раза. Важно, что значения дискретных данных могут быть расположены на числовой прямой и сопоставлены в терминах больше-меньше (10 телефонных звонков определенно больше, чем 6).
3. **категориальные** (например, географический регион или индекс). Важно, что значения категориальных данных не могут быть положены на числовую прямую (индексы 119256 и 114729 нельзя сравнивать в терминах больше-меньше).

Потом надо решить, как распределены данные. Различают **нормальное распределение** данных (чем больше значение признака отличается от его среднего по выборке значения, тем реже это значение встречается в выборке) и **распределение данных, отличное от нормального**.



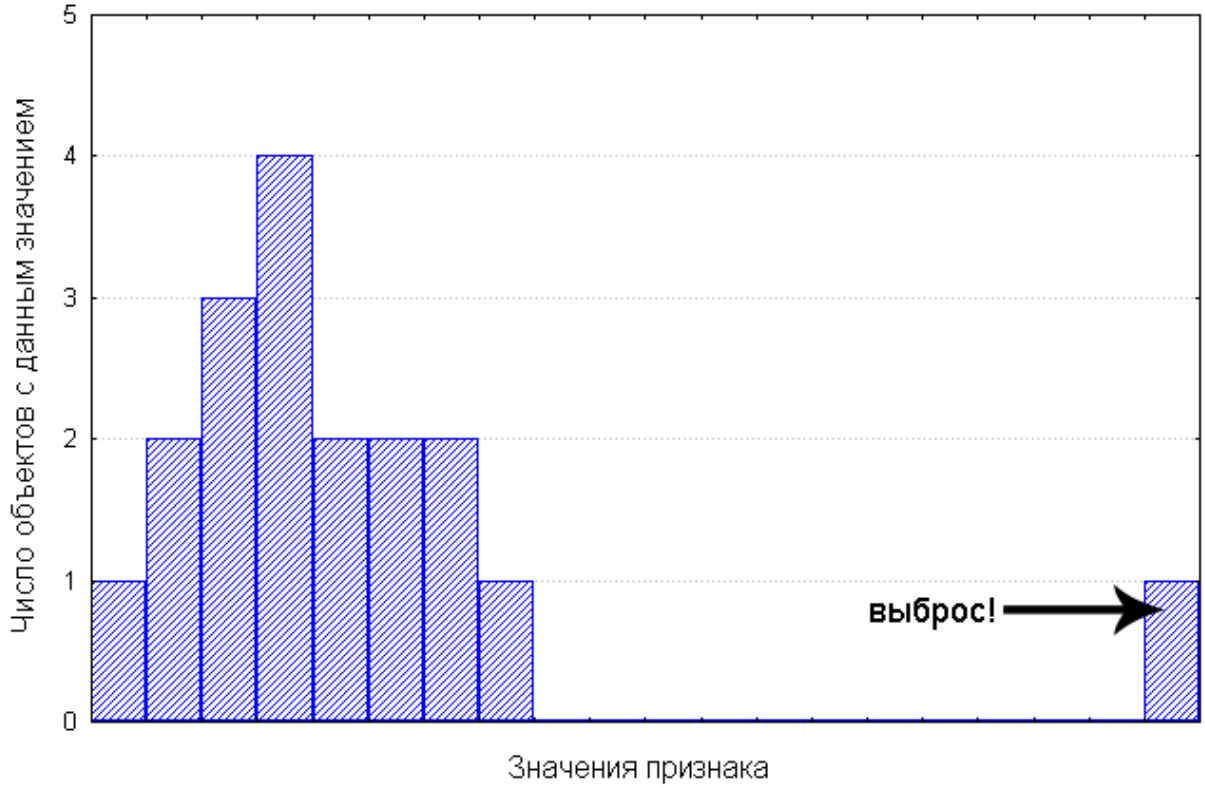




Строго говоря, на сравнительно небольших выборках, с которыми мы обычно работаем, нормальное распределение данных практически не встречается. Однако, данные, распределение которых не слишком сильно отличается от нормального, при статистической обработке данных считают нормально распределенными. Что такое "не слишком"? Четкого критерия не существует (некоторые формальные критерии, конечно, сформулированы, но они работают не слишком хорошо). В реальности каждый исследователь определяет это, опираясь на собственные ощущения.

И наконец, нужно выяснить, нет ли пропущенных данных, или **выбросов** (значений, которые очень сильно отличаются от подавляющего большинства значений исследуемого признака), или просто опечаток.




Все эти вещи способны сильно помешать вам правильно проанализировать ваши данные. С опечатками все понятно. Учтите, кстати, что если вместо цифры вы случайно ввели букву или символ, то некоторые программы, например, STATISTICA, с которой мы собираемся работать, воспринимают их как какое-нибудь (обычно довольно большое) число.

Если в ваших данных есть выбросы, нужно проанализировать причину их происхождения. Выбросы, во-первых, могут быть теми же опечатками, во-вторых, они могут быть получены в результате нарушения запланированного хода сбора данных (например, цель работы -- исследовать артериального давление у девятиклассников в спокойном состоянии. Понятно, что если какой-нибудь девятиклассник все время вертелся и

подпрыгивал вместо того, чтобы сидеть спокойно, то его артериальное давление будет существенно выше, чем у остальных). В этих случаях выбросы, естественно, удаляют из данных. Что же делать, если выброс кажется "вполне нормальным" значением? Например, вы измеряли длину листьев березы, и все листья были 5-10 см длиной и тут вам попался лист 20 см длиной! Почему он вырос таким -- тема для отдельного исследования, но из данных такое значение лучше все же исключить, потому что оно мешает увидеть нам общую картину.

Наконец, мы добрались до пропущенных данных. Они тоже могут возникнуть по нескольким причинам. Допустим, вы решили измерять черешки листьев у разных видов растений. Вполне может получиться так, что у одного листа той же березы черешок совершенно случайно оторвется, когда вы будете измерять лист. В результате, черешок останется не измеренным. В ваших данных вам придется оставить пустую ячейку. Когда вы доберетесь до листьев осок, то черешка вы там не найдете вообще. Наконец, пропущенные данные могут появиться при удалении тех самых выбросов. Как же быть с получившимися пустыми ячейками? Помните, что коварная STATISTICA и их способна заменить на числа, а подавляющее большинство типов анализа данных не способно работать с пропущенными значениями! Есть несколько выходов (я надеюсь, что ваши данные представлены в виде таблицы, где столбцами (*Variables*) являются исследуемые признаки, а строками (*Cases*) -- исследуемые объекты). Если пропущенных значений немного и они принадлежат к разным признакам, можно просто удалить содержащие их строки (необязательно физически удалять строки из таблицы, можно указать это в параметрах анализа!). Если у вас довольно много пропущенных значений находится в одном столбце, то можно удалить этот столбец. Если же ваши пропущенные значения в достаточном количестве рассеяны по всей таблице, можно попробовать заменить их на что-нибудь. Вполне естественно в случае с черешками у листьев осок принять их длину равной 0 (то есть решить, что черешок у них как бы есть, но просто очень короткий, незаметный). Понятно, что такой подход нельзя применить к листу березы с оторванным черешком. Ведь его длина не равна нулю, мы просто не знаем, какая она! Здесь пропущенное значение можно заменить на среднее значение выборки или что-нибудь вроде этого. Но такой подход нужно применять с большой осторожностью, потому что откуда мы знаем, что это был "среднестатистический лист".

### 2.3. Выяснение общих характеристик данных

(модуль *Basic statistics, Analysis → Descriptive statistics → More statistics*)

#### 2.3.1. Объем выборки (*Valid N*)

Это число наблюдений (объектов) в вашей выборке. Его принято указывать в описании методики и/или результатов вашей работы.

Нужно использовать команду str (). Например, str (data). Появится список всех переменных вашего файла данных, а начинаться он будет строчкой вроде этой: `data.frame': 20 obs. of 2 variables. Это значит, что в ваших данных 20 наблюдений и 2 переменных.

#### 2.3.2. Характеристики средней тенденции

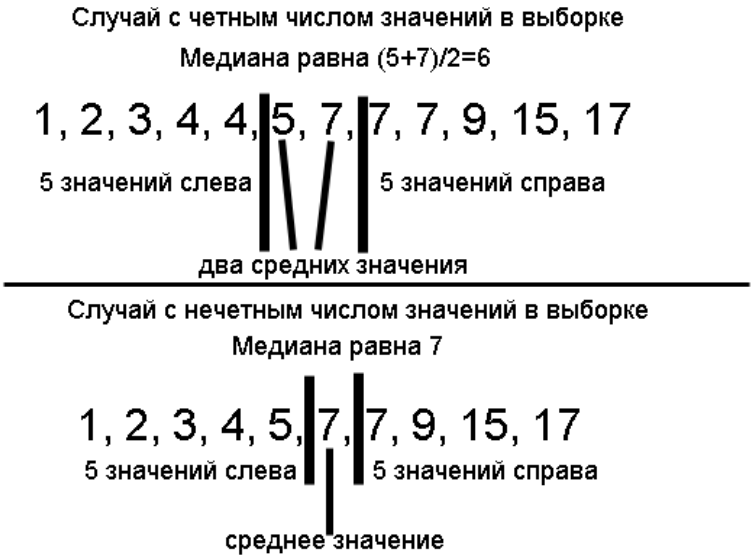
**а) Среднее арифметическое** (*Mean*)

Разумно применять для непрерывных данных с не слишком отличающимся от нормального распределением.

Используем команду mean(), например: mean(data).

**б) Медиана** (*Median*)

Представьте себе, что все значения признака выписаны в строчку в порядке их возрастания. Медианой будет считаться то значение, которое стоит в строке посередине (если признак имеет четное число значений, то медианой будет среднее арифметическое между двумя средними значениями). То есть половина значений в выборке будет больше или равна медиане, а другая половина -- меньше или равна медиане.




Медиану разумно вычислять для дискретных данных или непрерывных данных, распределение которых сильно отличается от нормального.

Используем команду median(). Например, для первой переменной наших данных: median(data[,1]).

в) **Мода** (*Mode*)

Наиболее часто встречающееся значение в выборке. Пожалуй, единственная характеристика (если не считать объема выборки), которая может быть вычислена для категориальных данных. К сожалению, вычисляется только в модуле *Nonparametrics/Distrib. (Ordinal descriptive statistics).* Если в вашей выборке два или более значения встречаются с одинаковой частотой, в графе "mode" вместо числового значения вы увидите надпись "multiple". В этом случае, узнать, что это за значения и сколько их в вашей выборке, можно, например, при помощи гистограммы (см. раздел "Визульный анализ данных"). Кстати говоря, данные с одной модой называются "**унимодальными**", а с двумя -- "**бимодальными**".

Используем команду table(). Например, для первой переменной наших данных: table(data[,1]). Появляется перечень с указанием частоты встречаемости каждого значения выборки.

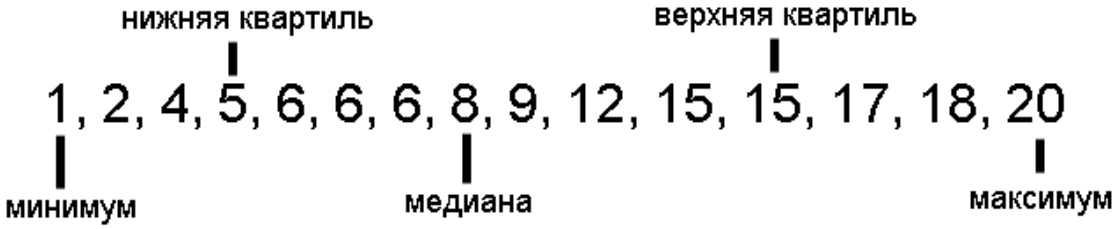
### 2.3.3. Показатели вариации данных относительно среднего

Возьмем две выборки со средним арифметическим и медианой, равными 5.
Первая выборка: 5, 5, 5, 5, 5, 5, 5, 5, 5, 5.
Вторая выборка: 1, 2, 3, 4, 5, 6, 7, 8, 9.
Ясно, что они существенно различаются между собой. Значит, одних характеристик среднего значения недостаточно для описания выборки. Нужно не только знать среднее значение, но и понимать, насколько сильно удалены от него отдельные значения в выборке.

**а) минимальное и максимальное значение** (*Minimum & maximum*)
Разумно использовать в отсутствие выбросов.

Используем команды min() и max() соответственно. Например, минимум для первой переменной наших данных: min(data[,1]).

**б) нижняя и верхняя квартиль** (*Lower & upper quartiles*)
Если при помощи медианы мы "разбивали" нашу выборку пополам, то квартили "разделяют" ее на четыре равные части. Граница между первой и второй (в порядке возрастания) частями называется нижней квартилью, а между третьей и четвертой -- верхней.




Разумно использовать для выборок с большим числом выбросов.

Используем команду summary(). Например, для первой переменной наших данных: summary(data[,1]). Появляется своеобразная таблица, в которой последовательно указаны: минимум, нижняя квартиль, медиана, среднее арифметическое, верхняя квартиль, максимум. Очень удобно!

**в) среднее квадратичное отклонение** (*Standart deviation*)
Этот параметр вычисляется так. Отклонения каждого значения в выборке от среднего арифметического возводятся в квадрат (чтобы избежать отрицательных значений) и суммируются. Полученная сумма делится на число значений в выборке (чтобы можно было сравнивать выборки разного

объема). Из получившегося числа извлекается квадратный корень, чтобы получить такую же размерность, как у значений выборки. В научных публикациях принято указывать значение среднего квадратичного отклонения всякий раз, когда вы упоминаете среднее арифметическое значение выборки.

Используем команду sd (). Например, для первой переменной наших данных: sd(data[,1]).

**2.4. Визуальный анализ данных** (в любом модуле: *Graphs* → *Stats 2D Graphs*)

Существует почти бесконечное множество разнообразных способов графического представления данных. Многие из этих способов реализованы и в программе STATISTICA. Я перечислю ниже несколько (немного!) наиболее часто употребимых и полезных типов графиков (все они двухмерные).

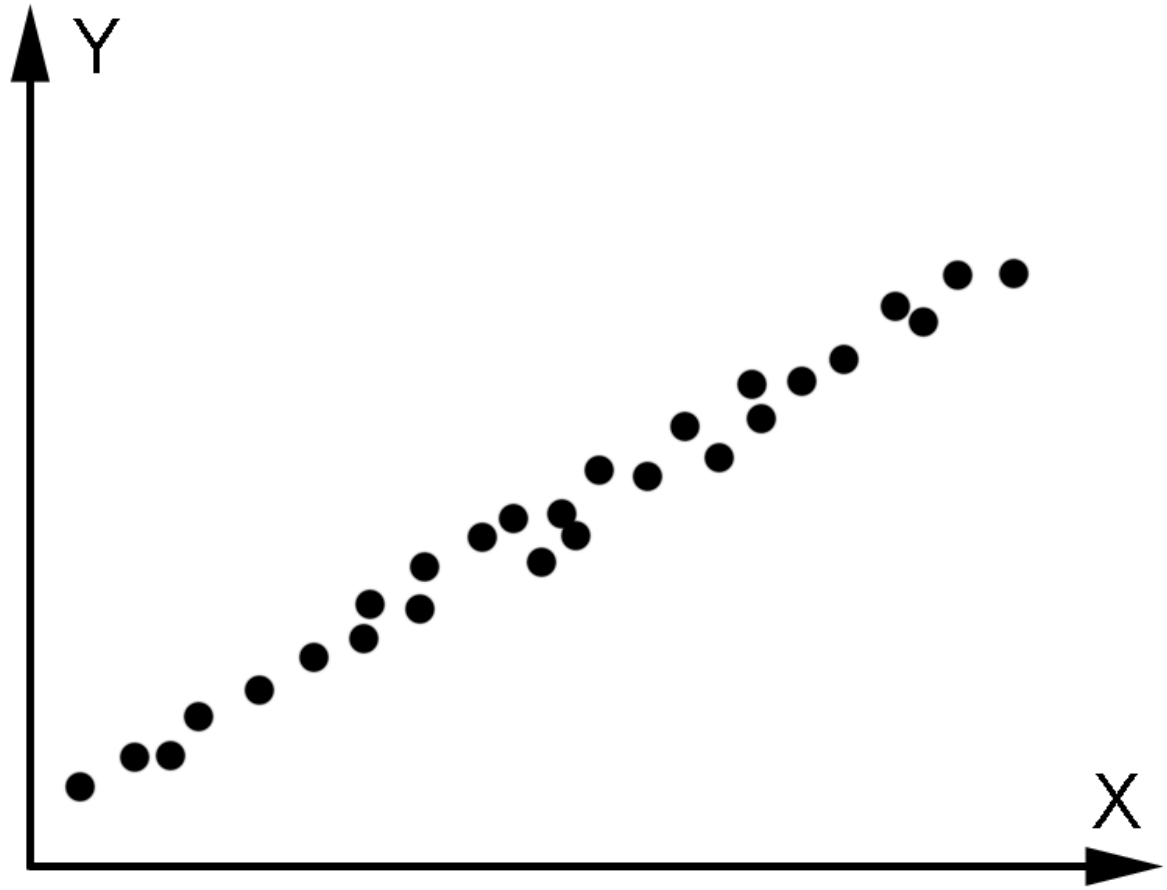
**2.4.1. Гистограмма** (*Histograms*)

Этот тип графика вы уже неоднократно видели, когда мы обсуждали типы распределения данных и выбросы. По оси абсцисс (горизонтальной!) указаны значения (или интервалы значений) признака, а по оси ординат (вертикальной!) указано, сколько раз в выборке встречаются такие значения признака (значения признака, попадающие в указанный интервал). Можно выводить на график как абсолютное число значений (*Y axis* → *N*), так и долю от общего числа значений в выборке (*Y axis* → %). Число интервалов, на которые разбивается весь диапазон значений признака, можно регулировать самостоятельно (указывать число интервалов в окошке *Categories*). Красную линию, символизирующую собой идеальное распределение данных, лучше отключать (*Fit Type* → *Off*). Обычно этот тип графика используется для того, зачем использовала его я в этом пособии (исследование типа распределения данных и поиск выбросов) или для общей характеристики выборки с большим числом значений, когда мы хотим сравнить частоты встречаемости разных значений (разных интервалов значений) между собой.

Используем команду hist(). Например, гистограмма с десятью интервалами для первой переменной наших данных: hist(data[,1], breaks=10).

### 2.4.2. **Диаграмма рассеяния** (*Scatterplots*)

Этот тип графика используется для исследования связи между двумя переменными. По оси абсцисс откладываются значения одной переменной, по оси ординат -- второй. Отдельные объекты изображатся в виде точек с координатами, соответствующими значениям этих двух переменных для конкретного объекта.



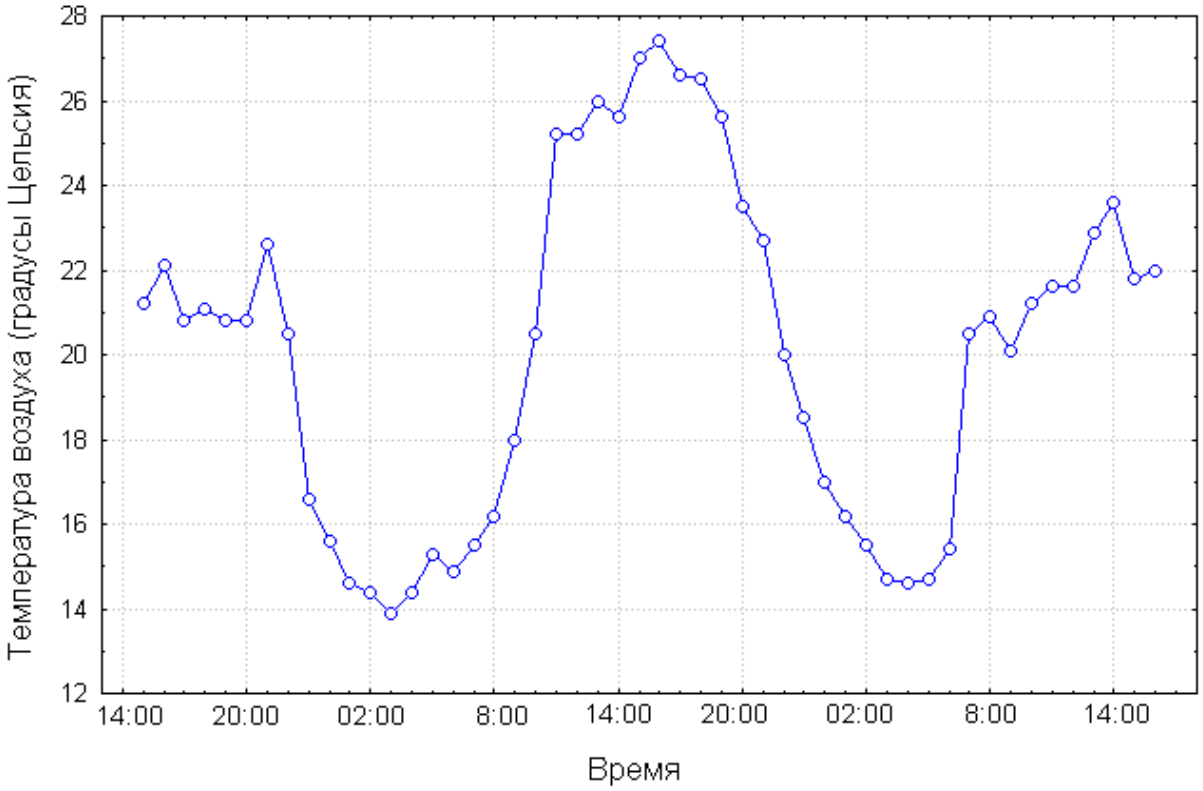

Опять же лучше отключить красную линию, символизирующую собой зависимость между переменными (*Fit* → *Off*). Можно сделать так, чтобы в

случае наложения нескольких точек размер результирующей точки отражал, сколько точек наложилось (*Graph Type* → *Frequency*).

Используем команду plot(..., type="p"). Например, отложим значения первой переменной наших данных по оси абсцисс, а второй -- по оси ординат: plot(data[,1], data[,2], type="p").

### 2.4.3. **Линия** (*Line Plots (Variables)*)

Этот тип графика имеет тот же принцип построения, что и диаграмма рассеяния, только точки соединяются отрезками в порядке их расположения в таблице данных.



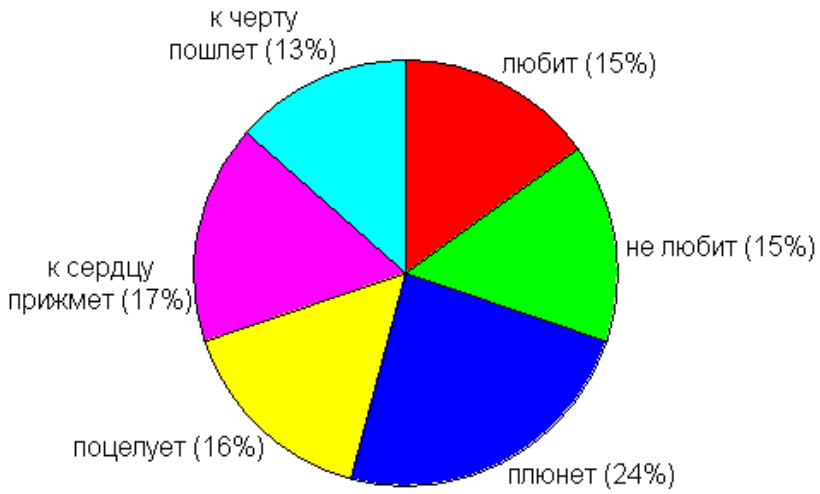

Такие графики имеет смысл использовать для исследования изменений величин во времени (динамика погодных условий, поведение животных и т.п.). В приведенном выше примере на графике изображено изменение

температуры воздуха в течение нескольких дней. Если промежутки между наблюдениями равные, то можно указывать только одну переменную, изменения во времени которой исследуются (*Graph Type* → *Regular*). В противном случае по оси абсцисс можно откладывать время (*Graph Type* → *XY Trace*). Можно одновременно исследовать изменения нескольких величин в одном масштабе (*Graph Type* → *Multiple*) или в разных (*Graph Type* → *Double-Y*).

> Используем команду plot(..., type="o"). Например, отложим значения первой переменной наших данных по оси абсцисс, а второй -- по оси ординат: plot(data[,1], data[,2], type="o")

### 2.4.4. "Пирог" (*Pie Charts: Graph Type* → *Pie Chart - Counts*)

Это своеобразный аналог гистограммы, применяющийся для характеристики выборки с небольшим числом значений. Круг делится на сектора, соответствующие значениям признака, при этом площадь секторов прямо пропорциональна частоте соответствующего значения в выборке: площадь всего круга принимается за 100% значений. Сектора можно подписывать соответствующими значениями признака (*Pie Legend* → *Text Labels*) или числом таких значений в выборке (*Pie Legend* → *Value*) или долей таких значений от общего числа значений в выборке (*Pie Legend* → *Percent*) или комбинациями этих подписей. Особенно полезно использовать этот тип графика, когда вы хотите показать частоту встречаемости какого-то значения по отношению к общему объему выборки.

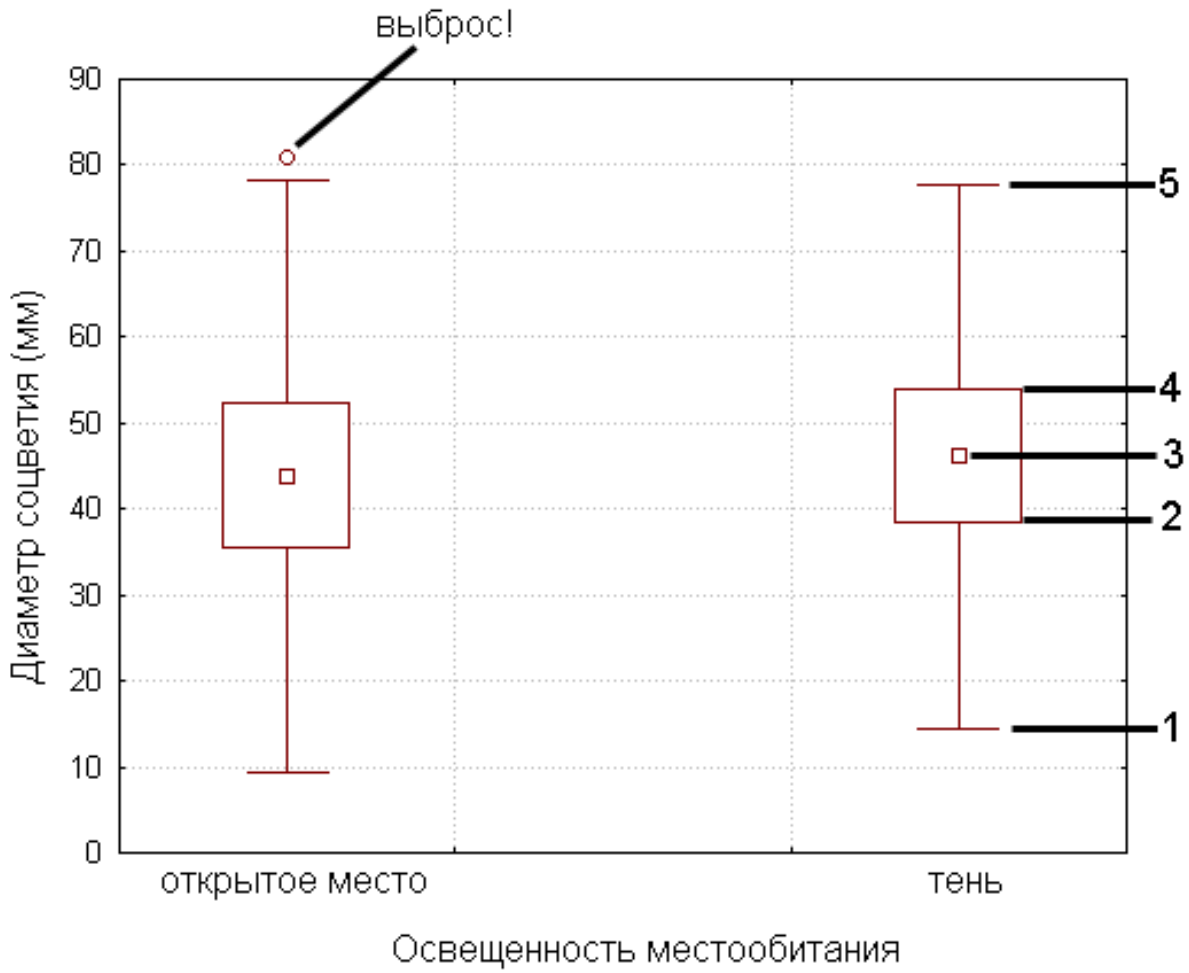



На этом примере показаны результаты гадания на ромашке (в скобках приведена частота определенного исхода гадания по отношению к общему числу гаданий).

Используем команду pie(). Например, для первой переменной наших данных: pie(data[,1]).

### 2.4.5. "Ящики с усами" (*Box plots: Graph Type → Box-Whiskers*)

Очень наглядное изображение основных характеристик выборки!




Можно указывать, что будет символизировать средняя точка (*Middle Point*) -- номер 3 на приведенном примере, верхняя (4) и нижняя (2) границы "ящика" (Box) и верхняя (5) и нижняя (1) границы "усов" (*Whisker*). На мой взгляд, наиболее разумны сочетания медианы (*Middle Points* → *Median*), нижней и верхней квартилей (*Box: Value* → *Percentiles, Coefficients* → 25) и минимума-максимума без учета выбросов (*Whisker: Value* → *Non-Outlier Range*) для дискретных переменных и среднего арифметического (*Middle Points* → *Mean*), среднего арифметического плюс-минус среднее квадратичное отклонение (*Box: Value* → *Std Dev*) и минимума-максимума без учета выбросов (*Whisker: Value* → *Non-Outlier Range*) для непрерывных

переменных. Для обоих типов переменных хорошо также показать и выбросы (*Outliers* → *Outliers*). Особенно удобно при помощи "ящиков с усами" графически сравнивать значения переменной для двух групп. На приведенном выше примере показан размер корзинки ромашки в разных условиях освещенности. Для этого нужно переменную "освещенность" указать как *Categories*, а переменную "размер корзинки" -- как *Variables*, тогда рядом в одном масштабе будут изображены два "ящика с усами" -- один для хорошо освещенного открытого местообитания, а второй -- для затененного.

> Используем команду boxplot() -- медиана, квартили, минимум-максимум. Например, для первой переменной наших данных: boxplot(data[,1]).

## 2.5. Статистические тесты

### 2.5.1. Введение

Статистических тестов существует, наверное, так же много, как и типов графиков. Главное -- понять логику их проведения, что мы и сделаем на примере нескольких самых распространенных и полезных, на мой взгляд, тестов. Тогда вы сможете без труда освоить любые статистические тесты, которые вам понадобятся в дальнейшем.

Все статистические тесты делятся на две большие группы: параметрические тесты и непараметрические. **Параметрические тесты** предназначены для обработки так называемых **параметрических данных**. Для того, чтобы данные считались параметрическими, должно одновременно выполняться три условия:

1. распределение данных близко к нормальному
2. выборка содержит не менее 30 наблюдений
3. это непрерывные данные.

Если хотя бы одно из этих условий не выполняется, данные считаются **непараметрическими** и обрабатываются **непараметрическими тестами**. Несомненным достоинством непараметрических тестов является, как ни банально это звучит, их способность работать с непараметрическими (то есть с той или иной стороны "неидеальными") данными. Зато параметрические тесты имеют бОльшую мощность (то есть при прочих равных вероятность не заметить существующую закономерность выше при использовании непараметрических тестов). Этому есть простое объяснение.

Дело в том, что дискретные (непараметрические) данные имеют свойство "скрывать" имеющиеся различия, объединяя отдельные значения в группы. Поэтому обычно стараются работать с параметрическими данными. На распределение данных мы, естественно, никак повлиять не можем. Что мы можем сделать, так это постараться иметь достаточно большой объем выборки (что, как вы помните, и увеличивает ее репрезентативность), а также работать с непрерывными данными.

Как сделать так, чтобы ваши данные были непрерывными, а не дискретными? Помните, что категориальные данные не могут быть обработаны никаким видом статистических тестов и должны быть преобразованы в дискретные или непрерывные! Можно спланировать сбор данных соответствующим образом. Например, при исследованиях размеров листьев не делить их визуально на "маленькие", "средние" и "большие", а измерить их длину и ширину при помощи линейки (на этом примере ясно, что непрерывные данные, как правило, содержат больше информации, чем дискретные). Однако иногда сбор непрерывных данных требует использования труднодоступного оборудования и сложных методик (например, если вы решите исследовать окраску цветков как непрерывную переменную, вам понадобится спектрофотометр для измерения длины волны отраженного света -- количественного выражения видимого цвета). В этом случае можно выйти из положения путем **последующего перекодирования данных** на стадии их обработки. Например, цвет можно закодировать в значениях красного, зеленого и синего каналов компьютерной цветовой шкалы RGB. Приведу еще один пример перекодирования. Предположим, вы изучаете высоту зданий в различных городах земного шара. Можно в графе "город" написать его название (категориальные данные). Это, конечно, проще всего, но тогда вы не сможете использовать эту переменную в статистическом анализе данных. Можно закодировать города цифрами в порядке их расположения, например, с севера на юг (если вас интересует географическая изменчивость высоты зданий в городе) -- получатся дискретные данные, которые можно обработать непараметрическими методами. И, наконец, каждый город можно обозначить его географическими координатами или расстоянием от самого южного города -- тогда мы получим непрерывные данные, которые (при наличии достаточного числа наблюдений и не слишком отличного от нормального распределения!) можно обработать параметрическими методами.

### 2.5.2. Различаются ли достоверно выборки?

#### 2.5.2.1. Введение

При ответе на этот вопрос при помощи описываемых ниже статистических тестов нужно всегда помнить, что эти тесты проверяют только различия по средним значениям, подразумевая, что разброс данных в выборках примерно одинаков. Например, уже упоминавшиеся выборки с одинаковыми параметрами средней тенденции и разными показателями разброса данных относительно нее
1, 2, 3, 4, 5, 6, 7, 8, 9 и
5, 5, 5, 5, 5, 5, 5, 5, 5
не будут различаться по результатам описываемых ниже тестов. Конечно, существуют тесты, которые анализируют различие в разбросе данных относительно среднего, но они используются довольно редко, и здесь мы их касаться не будем.

#### 2.5.2.2. Две выборки

Рассмотрим сначала со всех сторон наиболее часто встречающийся вариант вынесенного в заголовок этого подраздела вопроса: различаются ли достоверно ДВЕ выборки. Как вы помните, для проведения статистического теста, нам нужно выдвинуть две статистические гипотезы. Нулевая гипотеза: различий между (двумя) выборками нет. Альтернативная гипотеза: различия между (двумя) выборками есть.

Напоминаю, что ваши данные должны быть организованы в виде таблицы со строками-наблюдениями и столбцами-признаками. Сравниваемые выборки должны занимать отдельные столбцы. Например, если вы хотите узнать, различается ли достоверно рост мужчин и женщин, то в одном столбце должен быть указан рост мужчин, а в другом -- рост женщин (каждая строчка -- это один обследованный человек).

Если наши данные параметрические, то нам нужно провести **параметрический тест Стьюдента** (модуль *Basic statistics and tables: Analysis → Startup Panel...*). Причем здесь есть одна тонкость. Если переменные, которые мы хотим сравнить, были получены на разных

объектах (например, чтобы измерить рост мужчины и рост женщины, нужно как минимум два объекта -- мужчина и женщина), мы будем использовать тест Стьюдента для независимых переменных (*...t-test for independent samples,* в окошке *Input file* нужно выбрать *Each variable contains data for one group).* Если пары сравниваемых характеристик были получены на одном объекте (например, частота пульса до и после физической нагрузки измерялась у одного и того же человека), мы будем использовать тест Стьюдента для зависимых переменных (*...t-test for dependent samples, Display: Detailed table of results*). Тест для зависимых переменных более мощный. Дело здесь вот в чем. Представьте себе, что мы измеряли пульс до нагрузки у одного человека, а после нагрузки -- у другого. Тогда было бы не ясно, как объяснить полученную разницу: может быть, частота пульса увеличилась после нагрузки, а может быть, этим двум людям вообще свойственна разная частота пульса. В случае же "двойного" измерения пульса каждый человек как бы является своим собственным контролем, и разница между сравниваемыми переменными (до и после нагрузки) обуславливается только тем фактором, на основе которого они выделены (наличием нагрузки).

Если же мы имеем дело с непараметрическими данными, то нам нужно провести **непараметрический тест Вилкоксона** (модуль *Nonparametrics/Distrib.: Analysis* → *Startup Panel* (вкладка *Nonparametric stats*) → *Wilcoxon matched pairs test*).

В любом случае нужно указать пару переменных, которые вы желаете сравнить, и нажать *OK*. На экране появится таблица, содержащая множество значений, но вас будет интересовать одно единственное значение параметра *p-value* (или просто *p*) -- это вероятность статистической ошибки первого рода (вероятность найти несуществующую закономерность, если помните). Если эта вероятность больше или равна 0,05, мы вынуждены отвергнуть альтернативную гипотезу и принять нулевую об отсутствии отличий между выборками. Если *p-value* меньше 0,05, мы должны принять альтернативную гипотезу о существовании различий между выборками. Итак, еще раз кратко: *p-value* ≥ 0,05 -- достоверных различий нет, *p-value* < 0,05 -- достоверные различия есть!

Используем команды t.test() и wilcox.test(). Узнаем, есть ли достоверное различие между первой и второй переменными наших данных. Тест Стьюдента для независимых переменных: t.test(data[,1], data[,2], paired=FALSE). Тест Стьюдента для

зависимых переменных: t.test(data[,1], data[,2], paired=TRUE). Тест Вилкоксона: wilcox.test(data[,1], data[,2]). В любом случае, появится несвязанный текст, где значение p-value будет указано в третьей строчке.

Как вы помните, различия между выборками хорошо иллюстрировать при помощи графика "ящик с усами". Считается, что если "ящики" перекрываются более, чем на 1/3 своей высоты, то выборки достоверно не различаются.

А что если нам понадобится проанализировать различия между двумя выборками, значения которых представлены только нулями и единицами? Например, можно поставить вопрос: правда ли, что есть достоверная разница между частотой забывания сменной обуви мальчиками и девочками? Можно завести две колонки -- одну для мальчиков, другую для девочек -- и ставить в соответствующую колонку 0, если ученик (ученица) явились в школу без сменки, и 1, если он (она) принесли сменную обувь. Конечно же, мы получим непараметрические данные, которые будут анализироваться непараметрическим **тестом хи-квадрат** (*Chi-square test*, модуль *Nonparametrics/Distrib*.: *Analysis* → *Startup Panel* (вкладка *Nonparametric stats*) → *2 x 2 tables*). Перед нами появляется четыре окошка: пусть верхние два будут для мальчиков, а нижние два -- для девочек. В левое окошко нужно ввести число приходов в школу со сменной обувью (для каждого пола в свое окошко), а в правое -- без нее. Конечно же, совершенно не важно, какую ячейку выбрать для мальчиков, а какую для девочек, в какую заносить число нулей, а в какую -- число единиц, главное -- быть последовательным. При нажатии *OK* появляется длинная таблица, в которой нам нужна строка *Chi-square* (а не похожая на нее, будьте внимательны!), а точнее *p-level*, указанный в ней. Формулировка нулевой и альтернативной гипотез, а также ход расссуждений при выборе гипотезы точно такие же, что и в предыдущих двух тестах.

Используем команду chisq.test(). Пусть данные для мальчиков и девочек (нули и единицы, а не их число!) будут первой и второй переменными наших данных: chisq.test(data[,1], data[,2]). Значение p-value указано в третьей строчке появившегося текста.

### 2.5.2.3. Три выборки и больше

А что если теперь мы захотим узнать, есть ли различия между тремя выборками? Первое, что приходит в голову (предположим, что это

параметрические данные) -- это провести серию тестов Стьюдента: между первой и второй выборками, между первой и третьей и, наконец, между второй и третьей -- всего три теста. К сожалению, число необходимых тестов Стьюдента будет расти чрезвычайно быстро с увеличением числа интересующих нас выборок. Например, для попарного сравнения шести выборок нам понадобится провести уже 15 тестов! А представляете, как обидно будет провести все эти 15 тестов только для того, чтобы узнать, что все выборки не различаются между собой! Но главная проблема заключена не в сбережении труда исследователя (все-таки обычно нам нужно сравнить не больше 3-4 выборок). Дело в том, что при повторном проведении статистических тестов, основанных на вероятностных понятиях, на одной и той же выборке вероятность обнаружить достоверную закономерность по ошибке возрастает. Допустим, мы считаем различия достоверными при *p-value*<0,05, при этом мы будем ошибаться (находить различия там, где их нет) в 5 случаях из 100 (в 1 случае из 20). Понятно, что если мы проведем 20 статистических тестов на одной и той же выборке, то скорее всего однажды мы найдем различия просто по ошибке. Аналогичные рассуждения могут быть применены и к экстремальным видам спорта. Например, вероятность того, что парашют не раскроется при прыжке довольно мала (допустим, 1/1 000), и странно бы было ожидать, что парашют не раскроется как раз, когда человек прыгает впервые. При этом любой десантник, имеющий опыт нескольких сотен прыжков, может рассказать несколько захватывающих историй о том, как ему пришлось использовать запасной парашют.

Итак, для сравнения трех и более выборок используется (однофакторный) **дисперсионный анализ** (*ANOVA* от английского ***AN****alysis* ***Of VA****riance*)**.** Нулевая гипотеза: выборки не различаются между собой. Альтернативная гипотеза: хотя бы одна пара выборок различается между собой. Обратите внимание на формулировку альтернативной гипотезы! Результаты этого теста будут одинаковыми в случае, если различается только одна пара выборок, и в случае, если различаются все выборки. Если вы сравниваете несколько независимых переменных (вспомните тест Стьюдента), то ваши данные должны быть организованы как две переменных, в одной из которых указаны все значения всех сравниваемых выборок (например, рост брюнетов, блондинов и шатенов), а во второй -- номера выборок, к которым принадлежат значения первой переменной (например, будем ставить напротив значения роста брюнета 1, напротив роста блондина 2 и напротив роста шатена 3). Если же ваши переменные зависимые (например, частота

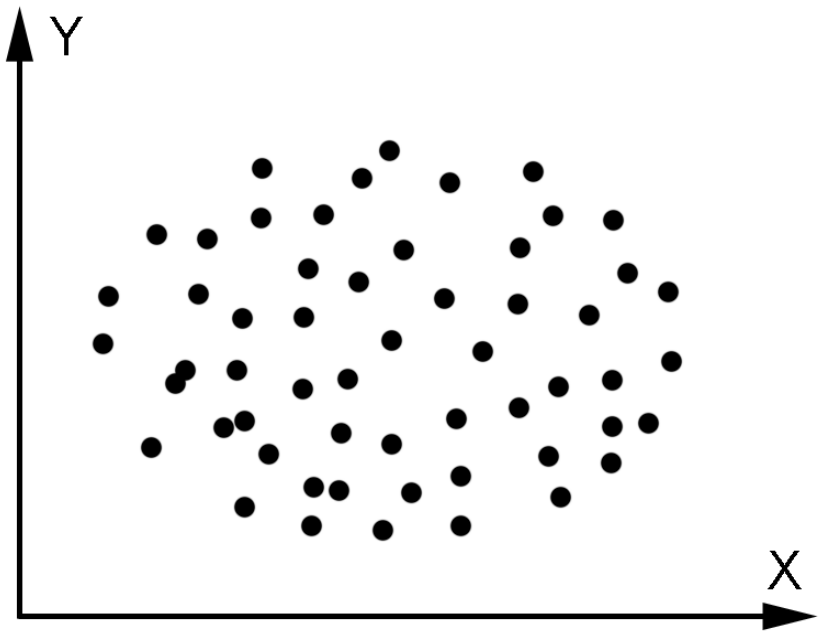
пульса после урока математики, урока физкультуры и урока немецкого языка, измеренные у каждого ученика класса), то каждая сравниваемая переменная должна занимать свой столбец (один столбец -- пульс после физкультуры, второй -- после немецкого...).

Запускаем модуль *ANOVA/MANOVA.* В случае с независимыми выборками выбираем независимую переменную (*Variables* → *Independent (factors)*) в которой содержатся номера выборок и зависимую переменную (*Variables* → *Dependent*), в которой содержатся значения выборок. В случае с зависимыми переменными, выбираем их все как зависимые, оставляя графу "независимая переменная" пустой. Теперь *OK* → *All effects*. В появившейся таблице нас, конечно же, интересует *p-value*. Если оно больше или равно 0,05, то все выборки не различаются между собой, и говорить тут больше не о чем. Если же оно меньше 0,05, то по крайней мере одна пара выборок различается. А может быть две? А может быть все выборки различаются между собой? Узнать это мы можем при помощи **Tukey test.** Вместо *All effects* нужно выбрать *Post hoc comparisons* → *Tukey honest significant difference (HSD) test*. Мы увидим таблицу, где будут указаны *p-value* для всех пар выборок. Естественно, что те пары выборок, *p-value* для которых меньше 0,05, достоверно различаются между собой (обычно они выделяются красным цветом).

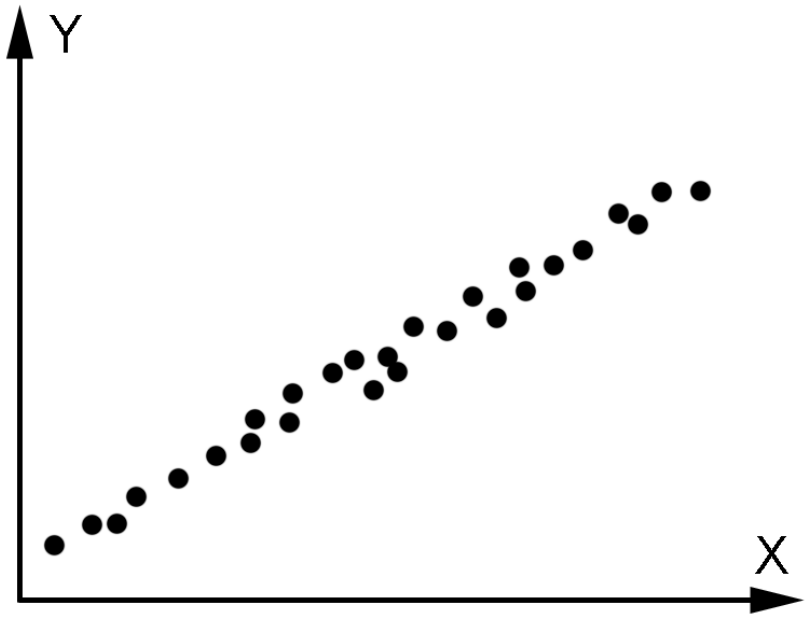
> Используем команду anova(). Пусть первая переменная наших данных -- независимая (цвет волос в первом примере или название урока во втором примере), вторая -- зависимая (рост в первом примере или частота пульса во втором примере), а третья -- номер испытуемого (во втором примере). Обратите внимание, что организация данных в R и STATISTICA в случае с зависимыми выборками будет различной! Случай с независимыми выборками (первый пример): anova(lm(data[,2] ~ data[,1])). Случай с зависимыми выборками (второй пример): anova(lm(data[,2] ~ data[,1] + data[,3])). В любом случае появляется своеобразная таблица, где условными обозначениями (их расшифровка дана в последней строке) напротив названия независимой переменной указано интересующее нас значение p-level.

### 2.5.3. Есть ли достоверная линейная связь между переменными?

Мерой линейной взаимосвязи между переменными является **коэффициент корреляции** (обозначается латинской буквой r). Значения коэффициента корреляции могут варьировать по модулю от нуля до единицы. Нулевой коэффициент корреляции говорит нам о том, что значения одной переменной совершенно не связаны со значениями другой переменной.

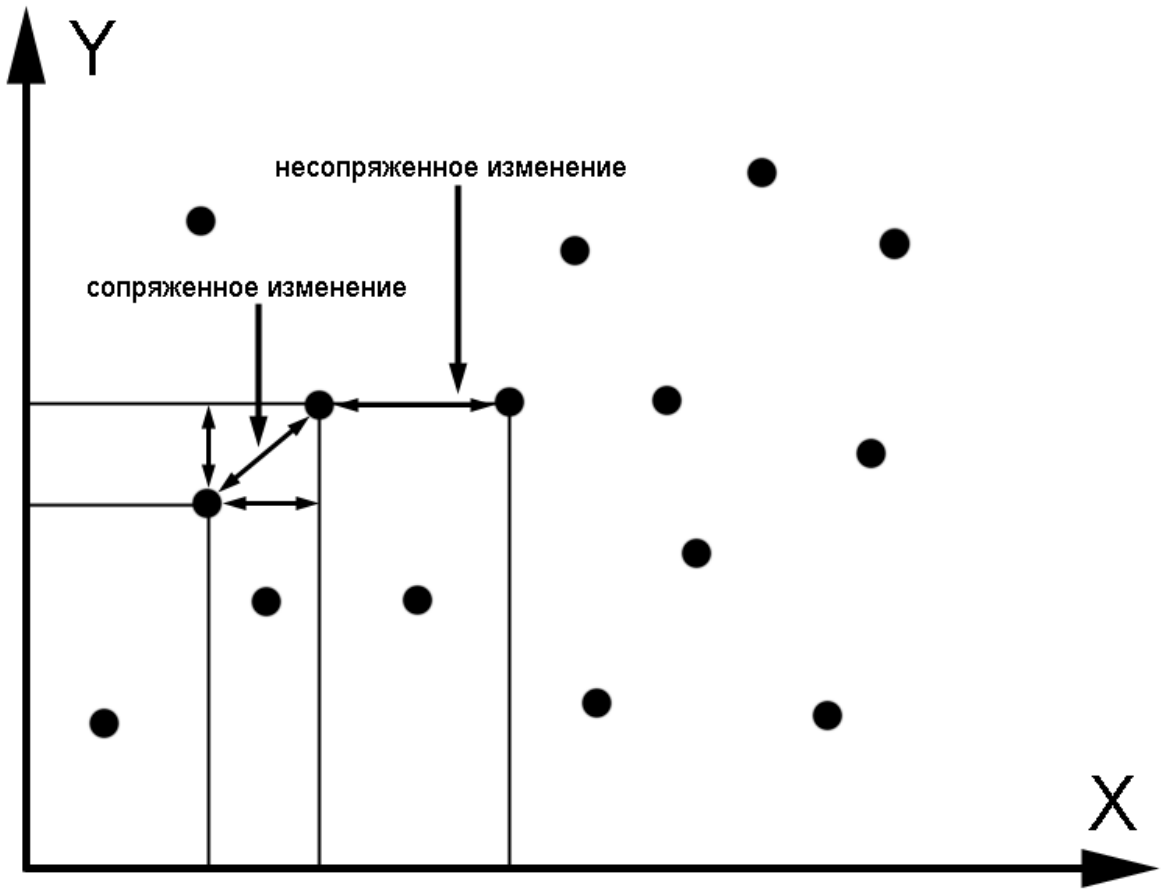



Коэффициент корреляции, равный по модулю единице, свидетельствует о четкой линейной связи между переменными (все наблюдения ложатся на прямую $y=ax+b$, где $x$ и $y$ -- наши переменные, $a$ и $b$ -- числовые коэффициенты).




Положительный коэффициент корреляции свидетельствует о положительной связи (чем больше, тем больше), отрицательный -- об отрицательной (чем больше, тем меньше).

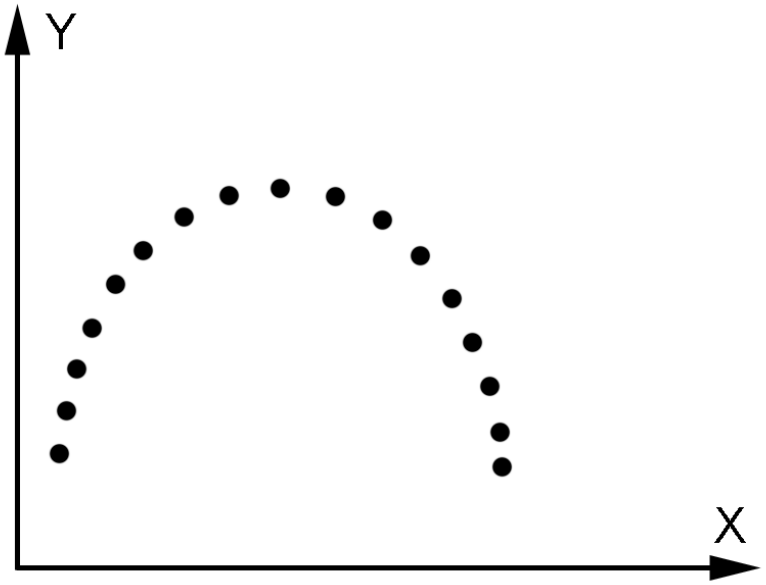
Казалось бы, из определения коэффициента корреляции следует, что если, например, он увеличится в два раза (по модулю), то и степень взаимосвязи между переменными тоже возрастет вдвое. Однако это не так. На самом деле степень взаимосвязи между переменными как таковую отражает **коэффициент детерминации** (это коэффициент корреляции, возведенный в квадрат). Эта величина показывает, какая доля изменений значений одной переменной сопряжена с изменением значений другой переменной.




Значит, если коэффициент корреляции равен 0,4, то значения переменных сопряженно изменяются в 16% случаев ($0,4^2$=0,16), а если коэффициент корреляции увеличится вдвое (0,8), то степень взаимосвязи между переменными возрастет в четыре раза ($0,8^2$=0,64).

Напоминаю, что коэффициент корреляции характеризует меру **линейной** связи между переменными. Две переменных могут быть быть очень четко взаимосвязаны, но если эта связь не линейная, а допустим, параболическая, то коэффициент корреляции будет близок к нулю. Примером такой связи может служить связь между степенью возбужденности человека и качеством решения им математических задач. Ясно, что очень слабо возбужденный человек (засыпающий) и очень сильно возбужденный (во время

футбольного матча) будет решать задачи гораздо хуже, чем умеренно возбужденный человек (на хорошо организованном уроке).




Поэтому прежде, чем оценить взаимосвязь численно (вычислить коэффициент корреляции), нужно посмотреть на ее графическое выражение (лучше всего здесь использовать диаграмму рассеяния). Существуют некоторые методы количественной оценки нелинейной связи между переменными, но мы их касаться не будем. Обращаю ваше внимание также, что речь здесь идет о **связи** между переменными, а не о **зависимости** одной переменной от другой. Если мы нашли достоверную связь между переменными А и Б, то это может значить, что А зависит от Б, Б зависит от А, А и Б зависят друг от друга, А и Б зависят от какой-то третьей переменной В, а между собой не имеют ничего общего. Например, хорошо известно, что объем продаж мороженного и число пожаров четко связаны между собой. Странно было бы предположить, что поедание мороженного располагает людей к небрежному обращению с огнем или что созерцание пожаров возбуждает тягу к мороженному. Все гораздо проще -- оба этих параметра зависят от температуры воздуха!

Итак, нулевая гипотеза: линейной связи между переменными нет. Альтернативная гипотеза: линейная связь между переменными есть.

Если данные параметрические, мы будем пользоваться **параметрическим коэффициентом Пирсона** (модуль *Basic statistics and tables*: *Analisys* → *Startup Panel* → *Correlation matrices, Display* → *Detailed table of results*). Если же наши данные непараметрические, то мы будем пользоваться **непараметрическим коэффициентом Спирмена** (модуль *Nonparametrics/Distrib.: Analysis -> Startup Panel* (вкладка *Nonparametric stats*) → *Correlations*). В общем-то нам достаточно обратить внимание на все то же значение *p-level* (вероятность найти несуществующую закономерность). Логика рассуждений здесь абсолютно такая же, как и в тестах на существование достоверных различий, такая же как и в прочих статистических тестах. Если эта вероятность (*p-value*) больше или равна 0,05, мы вынуждены отвергнуть альтернативную гипотезу и принять нулевую об отсутствии линейной связи между переменными. Если *p-value* меньше 0,05, мы должны принять альтернативную гипотезу о существовании линейной связи между переменными. Итак, *p-value* ≥ 0,05 -- достоверной линейной связи между переменными нет, p-value < 0,05 -- достоверная линейная связь есть! Надо сказать, что в отчетах и научных статьях наряду со значением p-value принято указывать и значение коэффициента корреляции.

> Используем команду cor.test(). Узнаем, есть ли достоверная связь между первой и второй переменными наших данных. Коэффициент Пирсона: cor.test(data[,1], data[,2], method="pearson"). Коэффициент Спирмена: cor.test(data[,1], data[,2], method="spearman"). Значение p-value указано в третьей строчке, значение коэффициента корреляции -- в последней.

### 3. Стандартная процедура статистического анализа

В заключение я приведу рекомендуемый порядок проведения статистического анализа данных.

а) формулировка биологической задачи (надо решить, что вы хотите узнать, например, есть ли различие между выборками, есть ли связь между величинами)

б) выбор способа статистической обработки данных (не забывайте сначала определить тип ваших данных: см. раздел "Как начать работу с данными" и провести предварительный графический анализ данных, в том числе проверить их на отсутствие выбросов и опечаток)

в) статистическая процедура (формулировка нулевой и альтернативной гипотезы, проведение расчетов, формулировка статистических выводов -- какую гипотезу вы принимаете)

г) биологическая интерпретация результата.

# Многомерный статистический анализ данных в школьных исследовательских работах

## 1. Введение

### 1.1. Зачем нужен многомерный анализ данных?

Окружающий нас мир многомерен в том смысле, что каждый объект характеризуется множеством в разной степени взаимосвязанных параметров. Исследователь снижает размерность мира, выбирая тему своего исследования, то есть, очерчивая круг параметров, которые будут его интересовать. Однако и в этом случае чаще всего одновременно изучается несколько, а то и несколько десятков и даже сотен признаков. Например, мы задались целью изучить зависимость артериального давления от возраста человека. Регистрировать только эти два параметра для каждого испытуемого было бы некорректно. Ясно, что на артериальное давление влияют другие (тоже взаимосвязанные) факторы (и некоторые даже сильнее, чем возраст), например, масса тела, наличие вредных привычек, физическая активность, наследственность и т.п., которые тоже придется учитывать при анализе зависимости артериального давления от возраста.

Основная проблема анализа таких многомерных матриц данных заключается в том, что человеческий мозг не способен одновременно оперировать более чем тремя измерениями пространства (поскольку пространственное воображение хорошо развито далеко не у всех людей, оптимально сократить число измерений до двух). Для свед**е**ния многомерных данных к двум измерениям с минимальными потерями информации была разработана специальная группа методов статистического анализа данных – многомерный анализ данных. Эти методы чрезвычайно разнообразны и основаны на довольно сложных математических расчетах. В настоящем пособии мы рассмотрим несколько самых основных и наиболее широко употребимых методов многомерного анализа данных на примере программы STATISTICA, не углубляясь, разумеется, в математические дебри.

Все примеры будут основаны на данных о размерах листьев березы[1] (каждая пронумерованная строка соответствует одному листу). Вот они:

| | ширина листа (мм) | длина листа (мм) | возраст ветки (года) | положение листа на ветке* | длина черешка (мм) | регион произ-растания** |
|---|---|---|---|---|---|---|
| 1 | 20 | 29 | 1 | 1 | 15 | 1 |
| 2 | 26 | 31 | 1 | 2 | 12 | 1 |
| 3 | 29 | 34 | 2 | 3 | 17 | 1 |
| 4 | 13 | 20 | 2 | 1 | 3 | 2 |
| 5 | 19 | 25 | 4 | 2 | 4 | 2 |
| 6 | 15 | 24 | 7 | 3 | 7 | 2 |
| 7 | 17 | 23 | 7 | 1 | 30 | 2 |
| 8 | 41 | 60 | 8 | 2 | 25 | 3 |
| 9 | 42 | 71 | 10 | 3 | 29 | 3 |
| 10 | 47 | 85 | 10 | 3 | 16 | 3 |

* 1 – в нижней части, 2 – в середине, 3 – в верхней части.

** 1 – Средняя Россия, 2 – Кольский полуостров, 3 – Сибирь

Краткие рекомендации по применению методов многомерного анализа данных в R по-прежнему даны мелким шрифтом. Пусть наши данные будут представлены объектом data.

### 1.2. Несколько практических рекомендаций

Исходные многомерные данные могут быть представлены как в виде переменных, то есть отдельных признаков объектов (более привычный нам вид), так и в виде матрицы расстояний.

Матрица расстояний представляет собой таблицу, где в первой строке и первом столбце перечислены объекты, а на пересечении строк и столбцов указаны «расстояния» между соответствующей парой объектов. Под расстояниями здесь понимается как привычное значение этого слова (примером такой матрицы могут служить таблицы расстояний между городами в туристических атласах), так и вообще любая мера различия между объектами. Например, при тестировании азбуки Морзе[2] испытуемым

[1] Конечно, таких берез не бывает, а 10 листьев вовсе недостаточно для того, чтобы обнаружить какие-либо закономерности в многомерных данных. Я просто придумала эти данные. Для примера.

[2] Система кодировки букв и цифр при помощи комбинаций из коротких и длинных сигналов. Применяется в основном в радиосвязи.

давали прослушать пары кодов и просили указать, являются ли они идентичными. Мерой различия («расстоянием») между парой кодов служило число испытуемых, считающих эту пару не идентичной. О том, как узнать об устройстве матриц расстояний в STATISTICA, можно прочитать в разделе 2.5.

Если же данные представлены переменными, то необходимо организовать файл данных так, чтобы строки представляли собой объекты, которые вы собираетесь классифицировать, а столбцы – переменные (признаки), описывающие эти объекты, на основе которых проводится классификация. Рекомендуется назвать строки короткими условными обозначениями объектов латинскими буквами, поскольку именно номера строк будут обозначать объекты на полученных графиках классификации[3].

Существует определенная проблема выбора объектов, которые вы собираетесь анализировать, и признаков, на основании которых этот анализ будет построен. На первый взгляд эта проблема кажется несколько надуманной («столько труда потратили на сбор данных, теперь надо все их и проанализировать»). Однако «лишние» признаки (то есть не вносящие существенного вклада в решение поставленной задачи) способны «маскировать» реальную структуру данных. Таким «лишним» признаком, например, может быть размер обуви в приведенном выше примере про исследование артериального давления. Включение слишком выделяющихся из общего множества объектов может также затруднить интерпретацию данных (здесь я не имею в виду выбросы, от которых нужно немедленно избавляться). К примеру, при классификации трех близких (но различающихся между собой) видов растений по морфологическим признакам включение четвертого сильно отличающегося вида непременно ухудшит разграничение этих трех видов.

И, наконец, помните, что практически все описываемые методы многомерного анализа данных работают с непрерывными и дискретными признаками, но не категориальными!

---

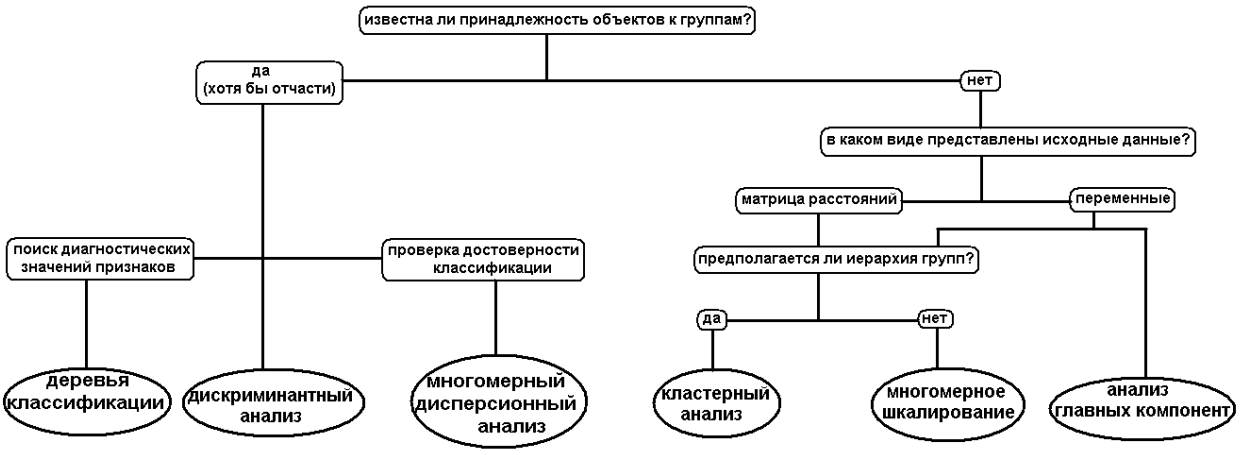
[3] Например, если вы классифицируете мебель по ее типам, столы (tables) можно назвать t1, t2, t3…, кровати (beds) – b1, b2, b3… и т.д. Для создания названий строк вам необходимо дважды щелкнуть мышкой на их номера (самый левый серый столбец таблицы данных).

## 2. Основные методы многомерного анализа данных

### 2.1. Какой метод выбрать?

Можно представить многомерные данные графически, непосредственно отражая значения переменных на графике (раздел 2.2). Несомненным достоинством этого метода является отсутствие обработки исходных данных, вследствие чего мы можем «считывать» с графика значения отдельных переменных. Однако такой метод хорош при небольшом числе признаков (не более 5), в противном случае, способы отображения разных переменных начинают смешиваться в восприятии. Кроме того, на этих графиках можно выявить только очень четко выраженные группы или закономерности.

В случае большего числа признаков и/или наличия сложной структуры данных (обычная ситуация!) нам необходимо будет воспользоваться собственно методами многомерного анализа данных. Краткий обзор применимости каждого из обсуждаемого в этом пособии методов приведен на размещенной ниже схеме.




**Дискриминантный анализ** (раздел 2.6) позволяет нам проверить сформированную теорию (о принадлежности объектов к определенным группам) и сформулировать правило распределения объектов по группам, которое можно применять для классификации объектов с неизвестной групповой принадлежностью. Последнее свойство дискриминантного

анализа (так называемая «классификация с обучением»[4]) имеет большое практическое значение. Часто бывает так, что определить принадлежность объекта к группе не представляется возможным. Например, точный диагноз больному в некоторых случаях можно поставить только после вскрытия. Однако, пронаблюдав (а в последствии вскрыв) несколько сотен больных, можно разработать надежную систему диагностики заболеваний по внешним признакам, которая позволит ставить правильный диагноз живым людям.

Прочие методы (кластерный анализ, многомерное шкалирование и анализ главных компонент) помогают нам выявить изначально неизвестную структуру в данных. Надо учитывать, что эти методы делают упор на визуальное представление результатов, а не на проверку их статистической значимости, оценка структуры данных «на глаз» весьма субъективна. Кроме того, разнообразие данных не всегда может быть сведено к двумерному пространству без существенных потерь информации. Поэтому полученные классификации желательно было бы как-нибудь проверять. Можно попробовать классифицировать данные несколькими методами и сравнить полученные классификации. Если они совпадают в общих чертах, то ваши результаты соответствуют реальному положению дел. Можно попробовать описать полученные группы (если они выявляются) – то есть при помощи описательных статистик или двухмерных графиков, или **деревьев классификации** (раздел 2.7) найти отдельные переменные, значения которых позволяют разграничить эти группы. Можно, наконец, проверить достоверность классификации при помощи **многомерного дисперсионного анализа данных** (раздел 2.8).

На самом деле, разница между **анализом главных компонент** (раздел 2.3) и **многомерным шкалированием** (раздел 2.5) велика только в теории. В реальных биологических исследованиях исходные данные в виде матрицы расстояний встречаются довольно редко, а данные в виде переменных можно без труда преобразовать в матрицу расстояний. Поэтому на практике, как правило, используют оба метода и сравнивают их результаты (или

[4] Конечно, дискриминантный анализ – это всего лишь один из множества существующих методов классификации с обучением (и не самый лучший). Однако этот метод широко распространен (может быть, потому, что он был придуман одним из первых) и реализован в STATISTICA, поэтому рассказывать о классификации с обучением я буду на примере именно дискриминантного анализа.

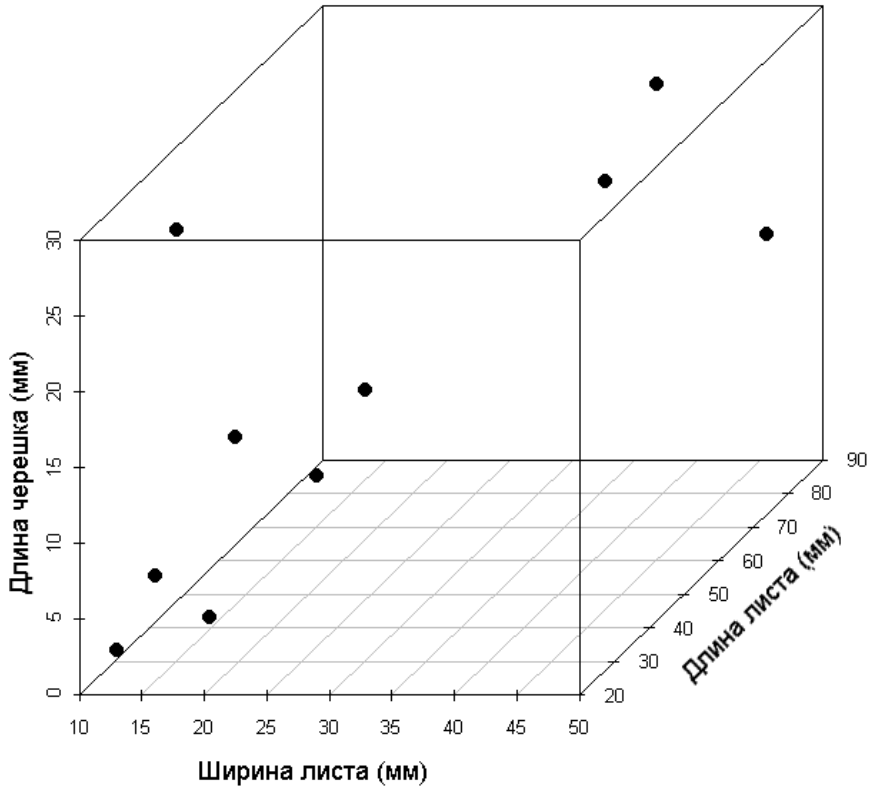
любой из методов наобум, но это уже хуже). Гораздо осторожнее нужно быть с кластерным анализом (раздел 2.4). Дело в том, что этот метод подразумевает наличие в данных структуры. Таким образом, кластерный анализ как бы «навязывает» данным структуру (хотя ее там может и не быть).

Нужно иметь в виду, что каждый метод многомерного анализа данных имеет множество вариаций, и очень часто конечный результат зависит от многих параметров. Поэтому во всех отчетах и публикациях с применением этих методов необходимо ясно указывать как минимум:

- название компьютерной программы, в которой выполнялся анализ (разные программы могут иметь разные алгоритмы реализации одних и тех же методов);
- заданные параметры (например, метод вычисления расстояния между объектами и метод кластеризации для кластерного анализа – см. ниже);
- способ определения числа групп (для тех методов, в которых оно неизвестно).

### 2.2. Графическое представление многомерных данных (из любого модуля пункт меню *Graphs*)

Если у нас есть всего три признака, то можно просто построить трехмерный график (*Stats 3D XYZ Graphs* → Scatterplots → *Graph Type: Scatterplot*). Например, такой:



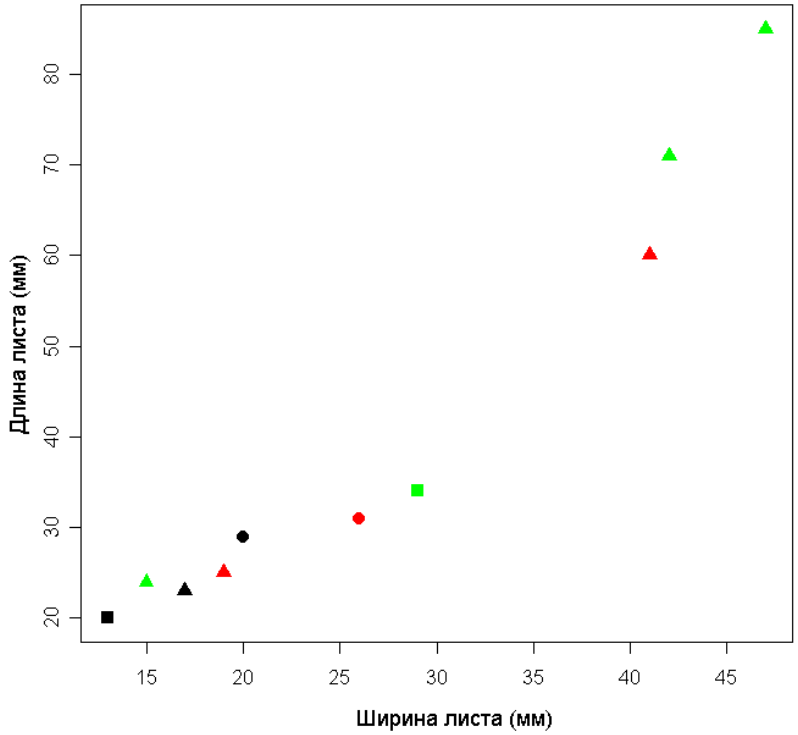

Каждый объект представлен точкой (в приведенном примере объектами служат листья березы), а значения признаков объектов отложены по осям (здесь признаки – размеры листа).

Загружаем дополнительный пакет команд: library(scatterplot3d).
Строим график: scatterplot3d(data[,1], data[,2], data[,5])

Что же делать, если одновременно вы хотите проанализировать больше трех признаков? От схемы один признак = одно измерение пространства придется отказаться. Здесь существует два самых распространенных решения. Во-первых, можно «накладывать дополнительные измерения» на обычную двумерную диаграмму рассеяния (*Stats 2D graphs → Scatterplots*: Mark selected subsets[5]).

[5] Каждое подмножество (*Subset*) -- это логические условия для выбора объектов, которые будут обозначаться одним и тем же символом. Тип символа создается автоматически, но на полученном графике можно щелкнуть два раза на обозначении символа в легенде и

Вот, например, листья березы, распределенные согласно размерам (один лист – один символ):




При этом цвет символа указывает на положение листа на ветке (черный – у основания, красный – посередине, зеленый – ближе к верхушке), а форма символа обозначает возраст ветки (круг – один год, квадрат – два года, треугольник – более двух лет).

Используем команду plot(). Тогда plot(data[,1], data[,2], pch=data[,3], col=data[,4]).

Если интересующих вас групп слишком много (например, вам хочется посмотреть, как группируются листья в зависимости от возраста ветки подробнее: от 1 года до 10 лет), можно обозначить отдельные листья не символом, а цифрой – возрастом ветки (*Stats 2D graphs → Scatterplots*:

---

изменить его. В предложенном примере с листьями березы у нас есть 9 типов символов (комбинации трех форм и трех цветов). Попробуем задать условия для подмножества «листья в основании (значение четвертой переменной равно 1) двухлетней (значение третьей переменной равно 2) ветки»: **v4=1 and v3=2**. Вот и все!

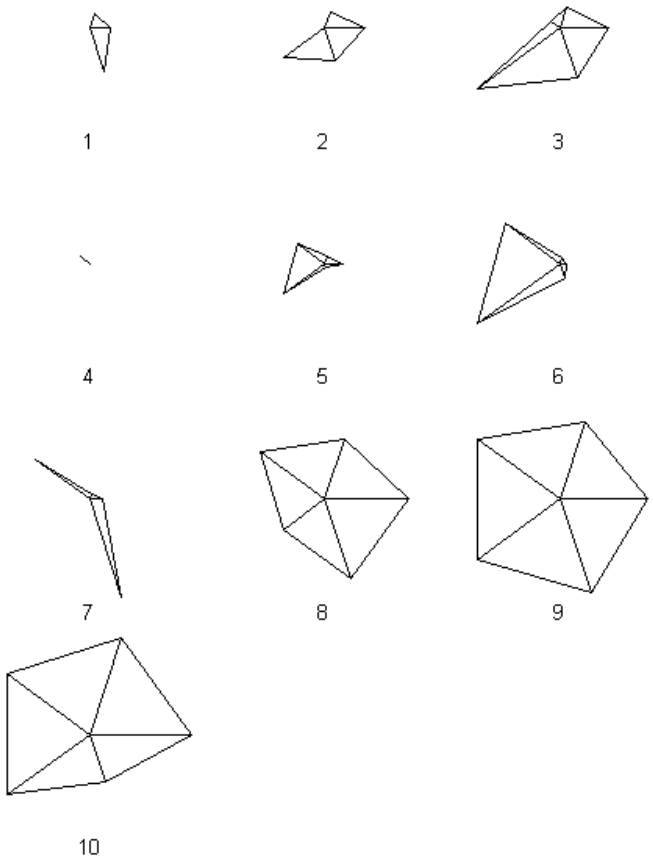
Options → DISPLAY: Case Labels: Var – указываем переменную, в которой содержится информация о возрасте ветки).

Требуемый график можно построить при помощи двух последовательных команд:

```
plot(data[,1], data[,2], type="n")
text(data[,1], data[,2], labels=data[,3])
```

Во-вторых, можно использовать самые разнообразные графики-пиктограммы, где каждая пиктограмма представляет один объект наблюдений, а ее параметры характеризуют значения признаков объекта.

Вот один из классических графиков-пиктограмм – звезды (*Stats Icon Plots → Graph Type: Stars*):

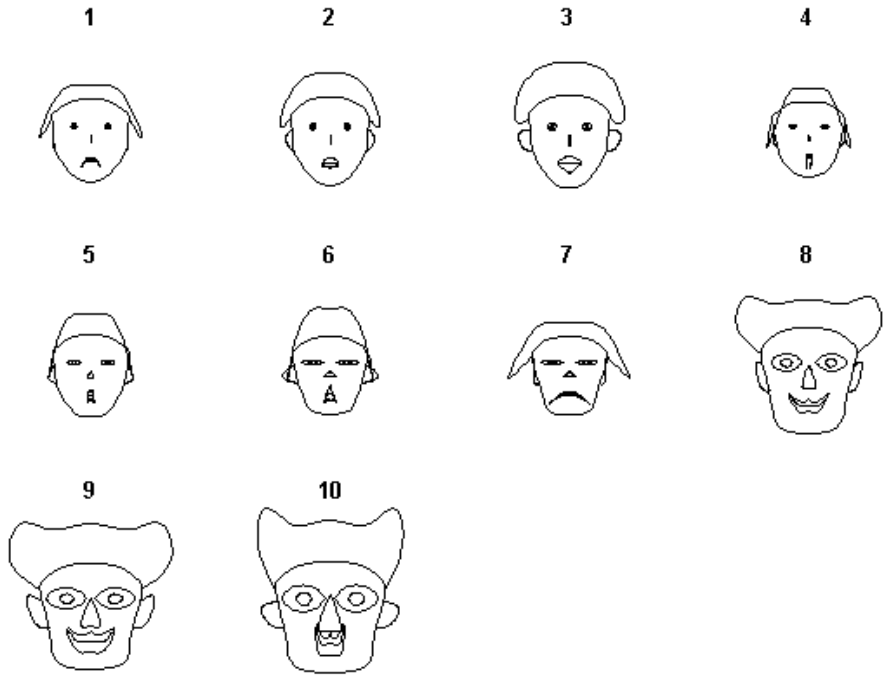



Здесь каждая пиктограмма – это один лист березы, а длины лучей соответствуют значениям разных характеристик этих листьев. Легко видеть,

например, что листья 8-10 (собранные в Сибири) чрезвычайно сходны между собой и отличаются от листьев из двух остальных регионов.

Используем команду stars(). Для пяти признаков листьев (без места сбора): stars(data[,1:5]). Признаки располагаются на лучах против часовой стрелки, начиная с правого горизонтального луча.

А вот более экзотический график-пиктограмма, так называемые «лица Чернова» (*Stats Icon Plots → Graph Type: Chernoff faces*):




Здесь каждое лицо соответствует одному исследуемому объекту (листу березы), а черты лица характеризуют значения признаков объекта. Сходство «сибирских» листьев хорошо заметно и на этом графике.

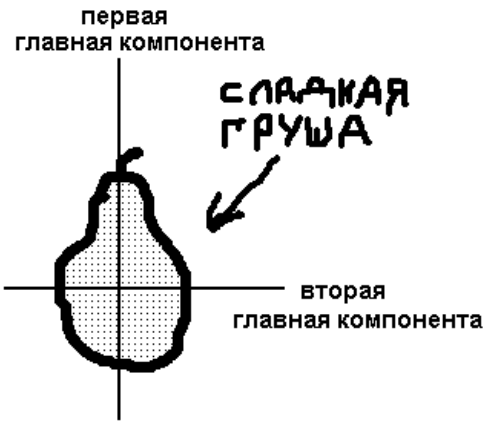
Загружаем дополнительный пакет команд: library(TeachingDemos).
Рисуем график: faces(data[,1:5]).

### 2.3. Анализ главных компонент (модуль *Factor analysis*)

Анализ главных компонент (*Principal component analysis*, PCA) – это один из наиболее широко употребимых и старых методов многомерного анализа

данных[6]. В основе этого метода лежит сведение всего множества исходных признаков к нескольким новым нескоррелированным переменным (собственно, главным компонентам), представляющим собой линейную комбинацию исходных переменных.

Это значит, что наши объекты можно представить как точки в n-мерном пространстве, где n – это число анализируемых признаков. Через полученное облако точек проводится прямая так, чтобы учесть наибольшую долю изменчивости признаков, то есть «пронизывая» это облако вдоль в наиболее вытянутой его части (визуально это можно представить себе для облака грушевидной формы в трехмерном пространстве[7]) – первая главная компонента. Затем через это облако проводится вторая, перпендикулярная первой, прямая, так чтобы учесть наибольшую оставшуюся долю изменчивости признаков – как вы уже догадались, вторая главная компонента. Эти две компоненты образуют плоскость, на которую и проецируются все точки[8].

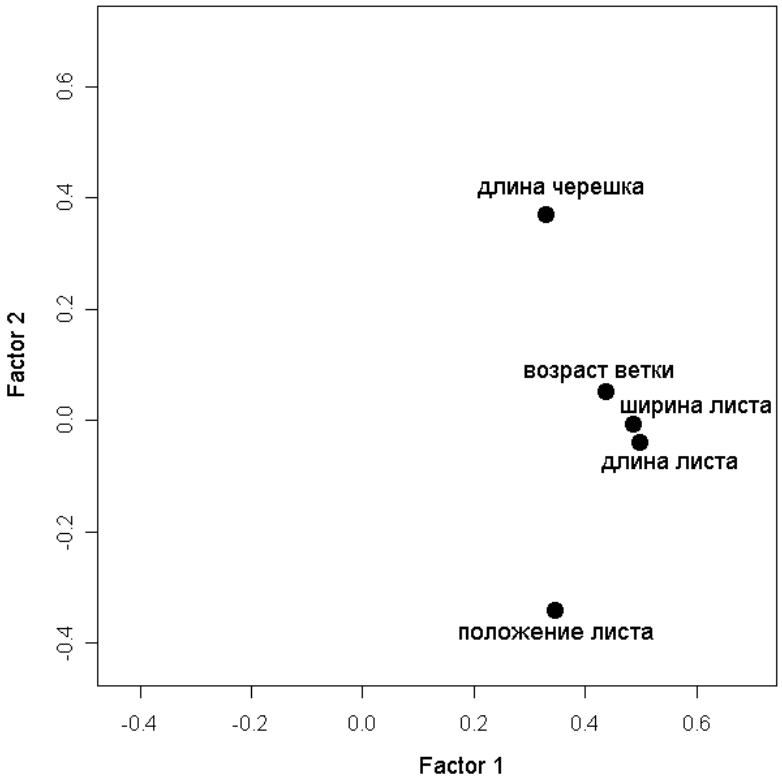



[6] Существует несколько близких методов многомерного анализа данных – это анализ соответствий (*Corresponding analysis*) и факторный анализ (*Factor analysis*). Эти методы различаются в основном теоретически, а результаты дают довольно сходные, поэтому здесь я подробно остановлюсь только на анализе главных компонент. С этими методами часто случается путаница, вот и в STATISTICA в модуле факторного анализа реализован на самом деле анализ главных компонент.

[7] Первую главную компоненту принято размещать горизонтально, а вторую – вертикально, но я не умею рисовать лежащую грушу.

[8] На самом деле, обычно главных компонент выделяется больше двух (на одну меньше, чем было исходных признаков), но основную информацию об изменчивости признаков, как правило, несут первые две компоненты.

Перед тем, как начать обработку данных, необходимо их сначала стандартизовать (из значений каждого признака нужно вычесть его среднее значение и разделить на стандартное отклонение: выделяем столбец, щелкаем правой кнопкой мыши и на выпадающем меню выбираем *Fill/Standardize Block* → *Standardize Column*). Эта операция нужна для того, чтобы размерность данных (сантиметры или километры?) и их вариабельность не влияла на результаты анализа. После этого мы выбираем переменные, на основании которых будет производиться классификация (*Startup Panel*: Variables), прочие установки оставляем по умолчанию (*Input file*: Row Data, *MD deletion*: Casewise) → OK. Выбираем собственно метод анализа главных компонент: *Extraction method: Principal components* → OK. На появившейся панели мы можем увидеть несколько важных вещей, касающихся полученной классификации. Во-первых, можно посмотреть, какую долю изменчивости признаков описывают выделенные главные компоненты (*Explained variance: Eigenvalues*, столбец таблицы *% of total variance*). Ясно, что если две первых компоненты вместе описывают очень мало изменчивости (скажем, меньше 50%), то на такую классификацию не ст**о**ит и смотреть, слишком уж она малоинформативна. Во-вторых, можно попробовать охарактеризовать полученные компоненты, понять, за что каждая из них «отвечает».




К примеру, на этом графике приведены характеристики листьев многострадальной березы (*Plot of loadings, 2D*). Чем большую координату (по модулю!) имеет признак по компоненте, тем больший вклад в нее он вносит. Как видно, первая компонента участвует в описании всех исследованных признаков. Вторая компонента «отвечает» за длину черешка и положение листа на ветке. Из этого графика можно делать выводы и о сопряженности признаков, например, интересно, что размер листа (его длина и ширина) связан с возрастом ветки. Наконец, можно получить эту же информацию о вкладе отдельных признаков каждую из компонент в табличной форме (*Factor loadings*). В этой таблице указаны коэффициенты корреляции признаков с компонентами (ясно, что чем больше коэффициент по модулю, тем больший вклад вносит признак в данную компоненту; достоверные корреляции обычно выделяются красным цветом).

Если же вы хотите увидеть классификацию своих объектов, то вам придется транспонировать файл данных, то есть перевернуть его на 90 градусов, так,

чтобы столбцы – переменные – стали строками, а строки – объекты – столбцами (в главном меню *Edit → Transpose → Data File*). В этом случае на стартовой панели нужно будет выбрать перечень классифицируемых объектов (*Variables*) и перечень признаков (*Select cases*), после этого нужно повторить все действия, описанные в предыдущем абзаце для получения графика (*Plot of loadings, 2D*).

Стандартизация признаков: scale(data[,1:5]).
Анализ главных компонент реализуется двумя последовательными командами:
data.pca <- princomp(scale(data[,1:5]))
data.p <- predict(data.pca).
Долю изменчивости, описываемую каждой из компонент, и вклад отдельных признаков в каждую из компонент можно узнать так: loadings(data.pca) – строка Proportion Var нижней таблицы и верхняя таблица соответственно. Графическое изображение вклада отдельных признаков в две первые главные компоненты: biplot(data.pca) – признаки обозначены красными стрелками. Графическое изображение доли изменчивости, описываемой каждой из компонент: plot(data.pca).
Классификация объектов на плоскости двух первых главных компонент:
plot(data.p[,1:2], type="n", xlab="PC1", ylab="PC2")
text(data.p[,1:2], labels=data[,6]) – каждый объект обозначается номером группы из шестой колонки – или просто
plot(data.p[,1:2], xlab="PC1", ylab="PC2") – каждый объект обозначается кружочком.

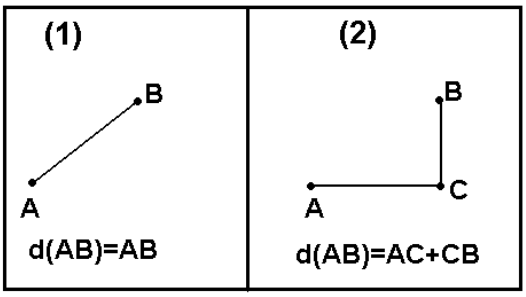
### 2.4. Кластерный анализ (модуль *Cluster analysis,* Startup panel*: joining (tree clustering)*)

Кластерный анализ основан на выделении групп сходных между собой объектов, то есть **кластеров**[9]. На сегодняшний день разработано множество методов кластерного анализа (целых семь **групп** методов), из которых в биологии обычно используются методы, основанные на последовательном объединении объектов в иерархические[10] кластеры (так называемые **агломеративные методы**). Примером такой классификации может служить Линнеевская система живого: сходные виды объединяются в рода, сходные рода – в семейства…

[9] Общепринятое или просто полезное определение этого термина отсутствует, и многие исследователи считают, что уже слишком поздно или вовсе незачем пытаться найти такое определение.
[10] Это значит, что несколько мелких (состоящих из небольшого числа объектов) кластеров объединяются в кластер среднего размера, несколько таких средних кластеров объединяются в кластер покрупнее и т.д.

При применении кластерного анализа неизбежно возникает множество насущных вопросов, на которые не существует единого ответа.

**Как измерять сходство (расстояние) между объектами**? Методов вычисления расстояний существует очень много (не забывайте, что дело происходит в многомерном пространстве). Наиболее широко употребляемыми методами для непрерывных переменных являются: **эвклидово расстояние** – *Euclidian distances* (1) и **манхеттеновское расстояние** или расстояние городских кварталов – *City-block (Manhattan)* (2). Для категориальных признаков (например, бинарных данных типа да-нет) при вычисления расстояния просто подсчитывают число параметров, которое совпадает у объектов: **коэффициент совстречаемости** (*Percent disagreement*).




Итак, пара наиболее близких объектов образовала первый кластер. **На каком основании можно присоединить к кластеру еще один объект**? Известно по крайней мере 12 методов присоединения к кластеру нового объекта, из которых наиболее распространены четыре:

- **Метод одиночной связи** (*Single Linkage*)**.** Новый объект должен иметь наибольшее сходство (по сравнению с прочими «кандидатами на присоединение») с **одним** из членов кластера. Недостатком такого метода являются большие продолговатые кластеры («гребенка»). Зато это единственный метод, который нечувствителен к изменению порядка объектов в исходном файле данных и к наличию в данных выбросов.
- **Метод полной связи** (*Complete linkage*). Сходство между новым объектом и **всеми** членами кластера должно превышать некоторое

пороговое значение (вычисляемое программой). Этот метод дает компактные кластеры и хорошо работает с группами разного размера.

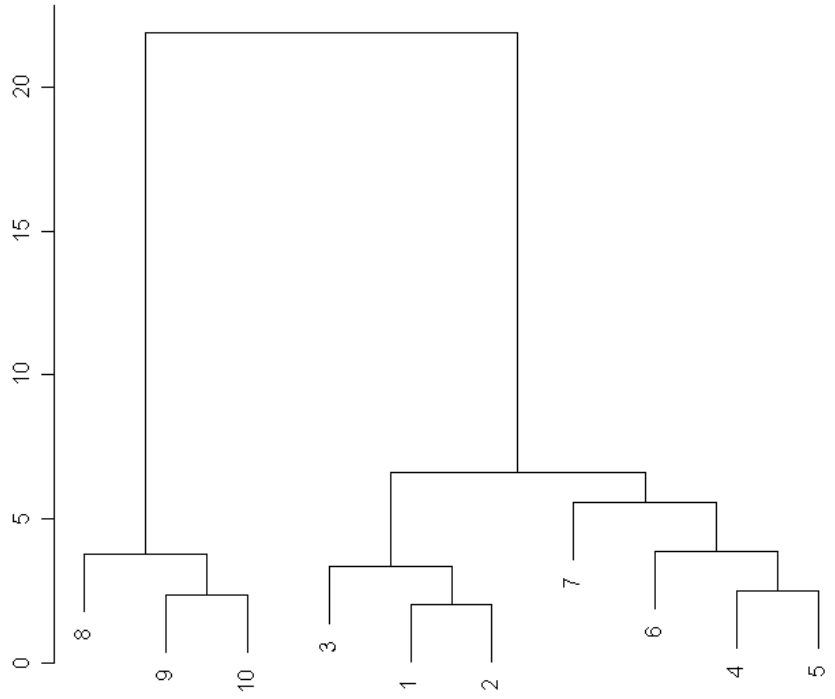
- **Метод средней связи** (*Unweighted pair-group average*)**.** Этот метод является своеобразным компромиссом между двумя предыдущими методами, потому что расстояние между новым объектом и кластером определяется как среднее арифметическое расстояний между этим объектом и всеми членами кластера. Кластеры обычно получаются довольно продолговатыми. Хорошо работает с группами разного размера, эффективно выделяет структуру, «скрытую» случайной изменчивостью признаков.
- **Метод Уорда** (*Ward's method*)**.** Объект для присоединения выбирается так, чтобы приращение суммы квадратов отклонений от средних значений признаков внутри кластера было минимальным. Позволяет получить компактные хорошо выраженные кластеры. Хорошо работает с группами сходных размеров, эффективно выделяет структуру, «скрытую» случайной изменчивостью признаков.

**Так какой же метод выбрать?** Единственно правильного ответа на этот вопрос не существует. Известно, что разные методы порождают разные классификации для одних и тех же данных. Более того, результаты кластерного анализа изменяются, если некоторые объекты исключаются из рассмотрения или, наоборот, добавляются, или даже просто меняются местами. Единственный выход из сложившейся ситуации – это попробовать несколько методов кластеризации и посмотреть, насколько стабильна полученная классификация и как она соотносится с вашими представлениями о структуре данных. По совокупности признаков наилучшей комбинацией нам представляется метод Уорда с расстояниями городских кварталов, хотя у каждого метода есть свои преимущества и недостатки.

**Как определить число кластеров?** Как правило, нас интересует не вся структура данных, а разделение объектов на несколько крупных групп, которое можно легко интерпретировать. Так на каком уровне нужно «обрезать» кластерное дерево (так называемую **дендрограмму**), чтобы получить оптимальное число групп? Эта проблема до сих пор не решена. Как правило, решение о числе кластеров принимается исследователем на основании личного опыта и визуального анализа дендрограммы. Например,

ясно, что на приведенной ниже дендрограмме можно выделить две главные группы. Существует и несколько формальных решений этого вопроса, но они малоэффективны.

Вам еще не расхотелось применять этот метод? Тогда в стартовом окне (*Startup panel: joining (tree clustering))* выбираем переменные (*Variables*), на основе которых будет производиться классификация; метод объединения кластеров (*Amalgamation (linkage) rule*) и метод измерения расстояний между объектами (*Distance measure*). Остальные установки должны быть такими: Input: *Raw data*, Cluster: *Cases (rows)*, Missing data: *Casewise deleted* → OK. Выбираем вид представления дендрограммы, наиболее часто используемой является вертикальная дендрограмма: *Vertical icicle plot.* Например, такая:




Здесь ясно видны две главные группы: одна состоит из «сибирских» листьев 8-10, а другая – из всех остальных. В этой второй группе можно в свою очередь выделить две подгруппы: «листьев средней полосы» (1-3) и «листьев Кольского полуострова» (4-7).

Вычисляем матрицу манхеттеновских расстояний:

```
data.dist <- dist(scale(data[,1:5]), method="manhattan")
```

эвклидово расстояние: … method=”Euclidian”
коэффициент совстречаемости: … method=”binary”
Классифицируем объекты методом Уорда: data.h <- hclust(data.dist, method=”ward”)
метод одиночной связи: … method=”single”
метод полной связи: … method=”complete”
метод средней связи: … method=”average”
Строим дерево: plot (data.h)

## 2.5. Многомерное шкалирование (модуль *Multidimentional Scaling*)

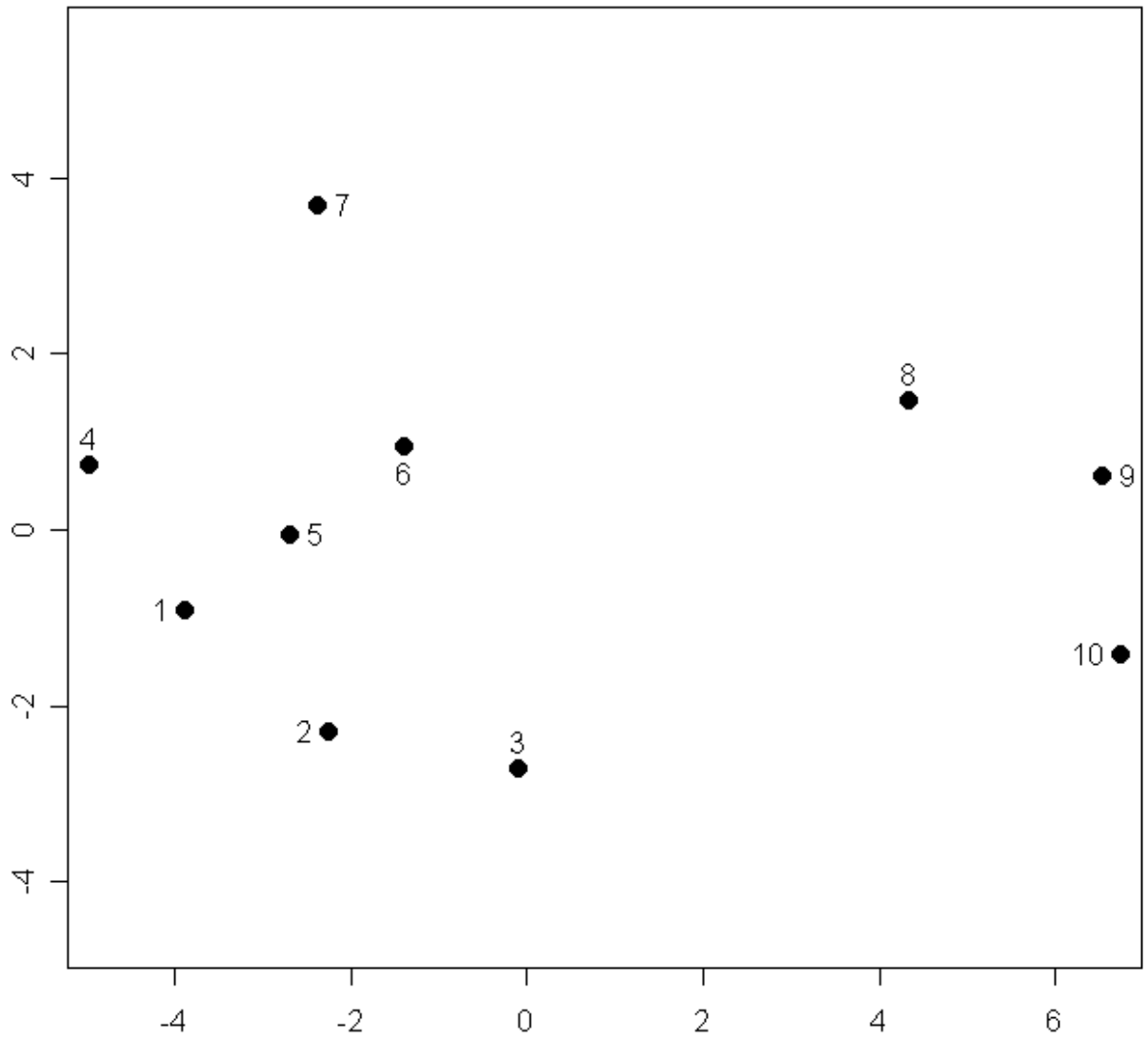
Многомерное шкалирование на основании известных расстояний между всеми парами объектов изображает взаимное расположение этих объектов[11]. Например, если обработать методом многомерного шкалирования матрицу расстояний между городами, то можно получить схему взаимного расположения этих городов.

Как вы, надеюсь, помните из раздела 2.1, многомерное шкалирование «работает» с матрицами расстояний, а не со значениями признаков. Если вы откроете в модуле многомерного шкалирования не матрицу расстояний, то появится предупреждение об этом: *“The current file is not a matrix file; Multidimentional Scaling expects a matrix input file”*. Вычислить матрицу расстояний можно только в модуле кластерного анализа (очень удобно!). Для этого нужно произвести все действия, необходимые для построения дендрограммы в кластерном анализе (см. раздел 2.4), но не выбирать тип дендрограммы. Тогда на появившейся перед вами панели вы можете посмотреть, как устроена матрица расстояний (*Distance matrix*) и сохранить ее (*Save distance matrix*). Знать устройство матрицы расстояний нужно для того, чтобы правильно ввести ее в компьютер, если ваши данные изначально имеют вид этой матрицы (см. раздел 1.2). Результаты многомерного шкалирования, точно так же как и результаты кластерного анализа, зависят от выбранного метода вычисления расстояний между объектами.

Итак, мы открываем полученную тем или иным образом матрицу расстояний. Далее выбираем в меню *Analysis → Startup panel*. На появившейся панели выбираем все «переменные» (*Variables*) – это ваши объекты, остальные параметры оставляем по умолчанию → OK → OK. На

[11] Довольно широко известный метод анализа главных координат (*Principal coordinate analysis, PCO*) является частным случаем многомерного шкалирования.

появившейся панели выбираем *Graph final configuration, 2D*, нажимаем OK и видим перед собой то, что нужно – графическое представление классификации ваших данных методом многомерного шкалирования. Вот, например, классификация наших листьев (каждый лист обозначен кружочком, цифры указывают на порядковый номер листа):




Подключаем дополнительный пакет команд: library(MASS).
многомерное шкалирование (как получить матрицу расстояний data.dist, см. в разделе «Кластерный анализ»): data.i <- isoMDS(data.dist)

визуальное представление классификации: eqscplot(data.i$points)

## 2.6. Дискриминантный анализ (модуль *Discriminant analysis, Analysis → Startup Panel*)

При дискриминантном анализе данных нам заранее известно число групп и принадлежность хотя бы части объектов к этим группам. Каждый объект может принадлежать только к одной группе, а число объектов должно превышать число признаков хотя бы на два. Ни один признак не должен являться линейной комбинацией других используемых признаков.

Выбираем переменную, в которой указан номер группы, куда принадлежат объекты, (*Variables: Grouping*), и признаки, на основании которых будет проводиться классификация объектов (*Variables: Independent*). Нажимаем ОК, на появившейся панели выбираем метод включения только тех переменных, которые дают существенный вклад в классификацию[12] (*Method: Forward stepwise*), остальные параметры оставляем по умолчанию, нажимаем ОК. Разглядывая появившуюся панель, можно увидеть много интересных и нужных вещей.

В верхней части панели на белом фоне в последней строке указано значение лямбды Вилкса (*Wilks' Lambda*). Значения этого параметра могут колебаться от 0 до 1 и характеризуют качество классификации. Значения, близкие к 0, говорят о «хорошей классификации» (это значит, что на основании выбранных переменных можно с уверенностью сказать, к какому классу относится тот или иной объект, и это хорошо согласуется с исходной классификацией объектов), значения, близкие к 1, соответственно, говорят о «плохой классификации». Можно сразу посмотреть, какие переменные вошли в модель (*Variables in the model*) – на них и была основана классификация, а какие – нет (*Variables not in the model*), как обладающие малой способностью разграничивать группы. Это тоже может быть важно, например, при планировании будущих исследований.

---

[12] Такой алгоритм является логичным и эффективным способом поиска оптимальной комбинации классифицирующих переменных, поскольку «избыточные» переменные (не несущие новой информации) способны увеличить число ошибочных классификаций. Однако нет гарантии, что полученная комбинация переменных будет оптимальной. В качестве альтернативы можно использовать для классификации все имеющиеся признаки (*Method: Standard*).

Еще один показатель качества полученной классификации – доля «неправильно» классифицированных объектов (то есть отнесенных по результатам дискриминантного анализа не в тот класс, к которому они были изначально причислены исследователем) – *Classification matrix*. В строках таблицы указаны классы, к которым объекты изначально принадлежали, а в столбцах – те классы, к которым они принадлежат по результатам дискриминантного анализа. В ячейках таблицы указано число соответствующих объектов. В первом столбце указана доля правильно классифицированных объектов (отдельно по классам и для всех объектов вместе – искомый параметр!). На основе этой таблицы можно выяснить, например, какие классы хуже всего различаются между собой (на пересечении этих классов число неправильно классифицированных объектов будет максимальным).

При анализе доли правильно классифицированных объектов нужно учитывать и общее число групп. Например, если у нас есть всего две группы, то даже при случайном отнесении объектов к группам доля правильных классификаций составит 50% (по теории вероятности), а в случае с четырьмя группами «случайных» правильных классификаций будет только 25% от общего числа объектов.

Интерпретация причины «неправильной» классификации объектов – отдельная история. В одном случае, объект может всегда быть недвусмысленно отнесен к какому-либо классу. Например, мы пытаемся найти ключевые признаки, отличающие разных домашних животных друг от друга, и анализируем кошек, собак, коров и морских свинок. Ясно, что большая доля ошибочных классификаций будет связана с неудачно выбранными классифицирующими признаками, а не с тем, что мы приняли кошку за корову в ходе сбора данных. Однако чаще всего в биологии бывает так, что принадлежность объектов к классам не так-то и легко определить (для этого, как правило, и затевается сбор данных и их обработка). Хорошим примером такой ситуации служат так называемые «сложные группы» у растений – несколько близких видов, которые плохо отличаются по внешним признакам. В этом случае ошибочные классификации могут быть объяснены неправильным определением видовой принадлежности растения при сборе материала. В любом случае детально проанализировать ошибочно классифицированные объекты можно, посмотрев на список всех объектов, в

котором ошибки классификации отмечены звездочкой, а также указаны вероятности отнесения каждого объекта к той или иной группе (*Posterior probabilities*). В этом списке есть и объекты, которые при сборе данных не были отнесены к какой-либо группе. Ясно, что при классификации объект относится к той группе, к которой он принадлежит с наибольшей вероятностью, однако часто бывает так, что вероятности отнесения объекта в две или более групп примерно одинаковы. Определять такие объекты в ту или иную группу следует с осторожностью.

Можно посмотреть, как располагаются объекты в многомерном пространстве признаков, естественно, в проекции на плоскость (*Canonical analysis & graphs → Scatterplot of canonical shores → OK*). Механизм создания такого изображения в общих чертах не отличается от используемого в анализе главных компонент.

Подключаем дополнительный пакет команд: library(MASS).
Группы, к которым отнесены объекты, и вероятности отнесения объектов к разным группам: lda(scale(data[,1:5]), data[,6], CV=T)
Собственно дискриминантный анализ:
data.lda <- lda(scale(data[,1:5]),data[,6])
Вычисляем лямбду Вилкса:
data.man <- manova(as.matrix(data[,1:5]) ~ predict(data.lda)$class)
summary(data.man, test="Wilks") – в появившийся таблице под надписью «Wilks»
Смотрим долю неправильно классифицированных объектов:
misclass(predict(data.lda)$class, data[,6])
Как ни странно, функции misclass() и каких-либо ее достойных аналогов в R нет, поэтому А.Б. Шипунову пришлось самому придумать эту функцию. Вот она:

```
misclass <- function(pred, obs) {
tbl <- table(pred, obs)
sum <- colSums(tbl)
dia <- diag(tbl)
msc <- (sum - dia)/sum * 100
m.m <- mean(msc)
cat("Classification table:", "\n")
print(tbl)
cat("Misclassification errors:", "\n")
print(round(msc, 1))
cat("Mean misclassification error:", round(m.m, 1), "\n")
}
```

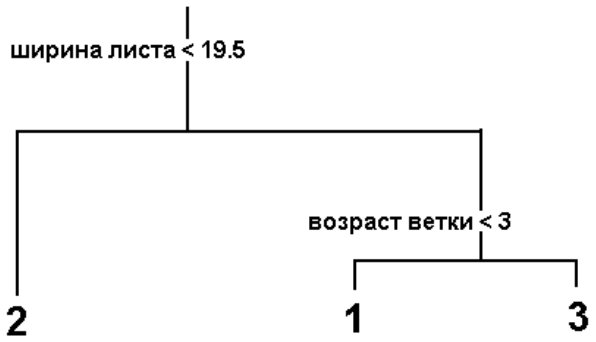
Визуальное представление классификации объектов:

```
data.lda.p <- predict(data.lda)$x
eqscplot(data.lda.p)
```

## 2.7. Деревья классификации (модуль *Classification trees*: Startup panel)

Этот метод позволяет выделить признаки, разграничивающие группы объектов, и указать значения этих признаков для каждой группы (подразумевается, что принадлежность всех объектов к группе известна). Прежде, чем применять этот метод, разумно убедиться, что ваши данные действительно разделяются на выбранные группы. Деревья классификации, в отличие от других распространенных методов многомерного анализа данных, умеют работать и с категориальными переменными.

Сначала нужно выбрать переменные (*Variables*): зависимую переменную (*Dependent variable*), в которой содержится информация о группах, куда принадлежат объекты, категориальные признаки объектов (*Categorial predictors*) и прочие признаки объектов – непрерывные и дискретные (*Ordered predictors*). Нажимаем OK два раза и видим дерево классификации. Например, такое:




Видно, что листья березы с Кольского полуострова (группа 2) отличаются от остальных более мелкими размерами, а сбор листьев в Москве (группа 1) производился с более молодых веток по сравнению с Сибирью (группа 3).

Подключаем дополнительный пакет команд: library(tree)
Проводим классификацию по пяти признакам:
data.t <- (tree(as.factor(data[,6]) ~ data[,1] + data[,2] + data[,3] + data[,4] + data[,5]))
Рисуем дерево классификации: plot(data.t)
Подписи: text(data.t)

## 2.8. Многомерный дисперсионный анализ (модуль *ANOVA/MANOVA*: Startup panel)

Многомерный дисперсионный анализ (MANOVA, от английского **M**ultivariate **AN**alysis **O**f **VA**riance) позволяет вычислить вероятность существования различий между несколькими группами объектов по совокупности их признаков. Предполагается, что принадлежность всех объектов к группам вам уже известна. Нулевая гипотеза: группы не различаются между собой по совокупности признаков. Альтернативная гипотеза: хотя бы одна пара групп различается между собой хотя бы по одному признаку. Обратите внимание на формулировку альтернативной гипотезы!

Выбираем независимую переменную (*Variables* → *Independent (factors)*), в которой указаны группы, куда принадлежат объекты, и зависимые переменные (*Variables* → *Dependent*) – то есть признаки этих объектов. Нажимаем *OK* → *All effects*. В появившейся таблице нас, как вы помните, интересует *p-value*. Если оно больше или равно 0,05, то верна нулевая гипотеза, а если меньше 0,05 – то альтернативная. Если верна альтернативная гипотеза, то можно, так же, как и для ANOVA, выяснить, благодаря каким признакам какие группы различаются между собой, при помощи *Tukey test*. Напоминаю, что для этого нужно вместо *All effects* выбрать *Post hoc comparisons* → *Tukey honest significant difference (HSD) test*, а там перебирать все признаки по одному. Для каждого признака можно увидеть таблицу, где будут указаны *p-value* для всех пар групп. Естественно, что те пары групп, *p-value* для которых меньше 0,05, достоверно различаются между собой по выбранному признаку (обычно они выделяются красным цветом).

> Для проверки достоверности классификации в целом используем две последовательные команды: summary(manova(as.matrix(data[,1:5]) ~ data[,6]), test="Wilks"). Для того чтобы исследовать различия групп по отдельным признакам, нужно использовать две другие последовательные команды: summary(aov(as.matrix(data[,1:5]) ~ data[,6])). Для каждого признака появляется своя таблица, в которой символом указано значение *p-value*.