# Introductory Biostatistics with R

Dylan Z. Childs, Bethan J. Hindle and Philip H. Warren

2022-11-15

# Contents

# IX Frequency Data and Non-parametric Tests 291

# Overview

This book provides a self-contained introduction to the analysis of biological data using the R programming language. Topics include principles of experimental design principles of frequentist statistics, simple statistical tests, analysis of variance, regression, analysis of categorical data, and non-parametric statistics.

## Why do a data analysis course?

To do science yourself, or to understand the science other people do, you need to understand the principles of experimental design, data collection, statistical analysis and scientific presentation. That does not mean becoming a maths wizard or a computer genius. It means knowing how to make sensible decisions about designing studies and collecting data and then interpreting those data correctly. Sometimes the methods required are straightforward, sometimes more complex. You aren't expected to get to grips with all of them, but what we hope to do in the course is to give you a good practical grasp of the core techniques that are widely used in biology. You should then be equipped to use these techniques intelligently and, equally importantly, know when they are inappropriate and when you need to seek help to find the correct way to design or analyse your study.

With some work on your part, you should acquire a set of skills that you can apply in practicals, field courses, and project work. These same skills will almost certainly also be useful in other situations, whether doing biology, or something completely different. We live in a world increasingly flooded with data, and people who know how to make sense of this are in high demand.

## Programming prerequisites

This book assumes you have used R and RStudio before and that you are comfortable with the material given in the associated Exploratory Data Analysis with R book. What follows is a quick summary of what you need to know.

## R and RStudio

R is a programming language designed first and foremost to support **data analysis**. We use it to carry out all the statistical analyses covered in this book. R is a fully-fledged programming language, meaning that we have to interact with it by writing computer code. Essentially, R sits and waits for instructions in the form of text (R code). A user can type in those instructions, or they can be sent to it from another helper program. This book assumes you have written R code before and know how to 'run it' via these two routes.

We strongly recommend that you use a helper program when working with R. We like RStudio. RStudio makes R a more pleasant and productive tool. RStudio provides a Source Code Editor for working with R code. It also provides a visual, point-and-click means of accessing various language-specific features, such as install packages (see below).

⚠ **R and RStudio are not the same things**

RStudio is a different program from R—it is installed separately and occupies its own place in the Programs menu (Windows PC) or Applications folder (Mac). You can run R without RStudio but you cannot run RStudio without R.

## Using packages

R packages extend the basic functionality of R so that we can do more with it. They bundle together R code, data, and documentation in a standardised way that makes it easy to use and share with others. This book uses a subset of the tidyverse ecosystem of packages:

- The **readr** package for reading data into R
- The **dplyr** package for data manipulation
- The **ggplot2** package for making plots

You need to understand R's package system to use these. This means knowing that **installing** a package and **loading** the package are different operations. We only have to install a package onto our computer once, but we have to load the package every time we want to use it in a new R session. If that is confusing, revise the package system chapter Exploratory Data Analysis with R book.

Installing a package can be done via the `install.packages` function, e.g. use this code to install the **dplyr** package:

```
install.packages("dplyr")
```

Alternatively, use the simple RStudio menu interface to install packages. This is accessed via the **Install** button in the packages tab of the lower right window in RStudio. Loading and attaching a package so that it can be used in a given session happens via the `library` function, e.g.

```
library("dplyr")
```

⚠ **What goes in your script?**

We should not routinely leave `install.packages` lines in our scripts. Why? Because we don't usually need to install a package every time we run a script—installing packages is a slow 'do once' operation. We do usually place one or more `library` lines at the start of a script to ensure the required packages available in the current R session every time we run the script.

## Working with data

All the examples use data sets that are stored in CSV files. You can get these by emailing the lead author. We assume you know how to use a function like `read_csv` from the **readr** package to get those data into R. Remember, when using `read_csv` we end up with a 'tibble': the **tidyverse** version of a data frame. You could also use the base R `read_csv` function to read CSV data into a standard base R data frame—the differences don't matter in this book.

Tibbles and data frame are a table-like objects with rows and columns. They collect together different variables, storing each of them as a named column. We always assume that the data inside these objects are organised according to tidy data conventions. Tidy data has a specific structure that makes it easy to manipulate, model and visualise. A tidy data set is one where each variable is only found in one column, and each row contains one unique observation.

## Data wrangling with dplyr

We assume you know how to manipulate data frames and tibbles using the **dplyr** package. The **dplyr** package makes it easy to manipulate 'rectangular data' that lives in tibbles and data frames. **dplyr** is orientated around a few core functions, each designed to do one thing well. These **dplyr** functions are sometimes referred to as its 'verbs' to reflect the fact that they 'do something' to data. For example:

- `select` obtains a subset of variables,
- `mutate` constructs new variables,

- `filter` obtains a subset of rows,
- `arrange` reorders rows, and
- `summarise` calculates information about groups.

Notice that the names are chosen to reflect what each function does to the input data. We'll routinely use these—along with a few additional functions such as `rename` and `group_by`—to create new variables, subset data based on some criteria, and calculate summaries for different subsets of data.

## Plotting with ggplot2

We assume you know how to create data visualisations using the **ggplot2** package. A visualisation is created by defining one or more layers, where each layer is associated with some data and a set of rules for how to display the data. We assume you know how to use the **ggplot2** 'grammar' to set up:

1) one or more **layers**,
2) a set of **scales**,
3) a **coordinate system**, and
4) a **facet** specification for multi-panel plots (optional).

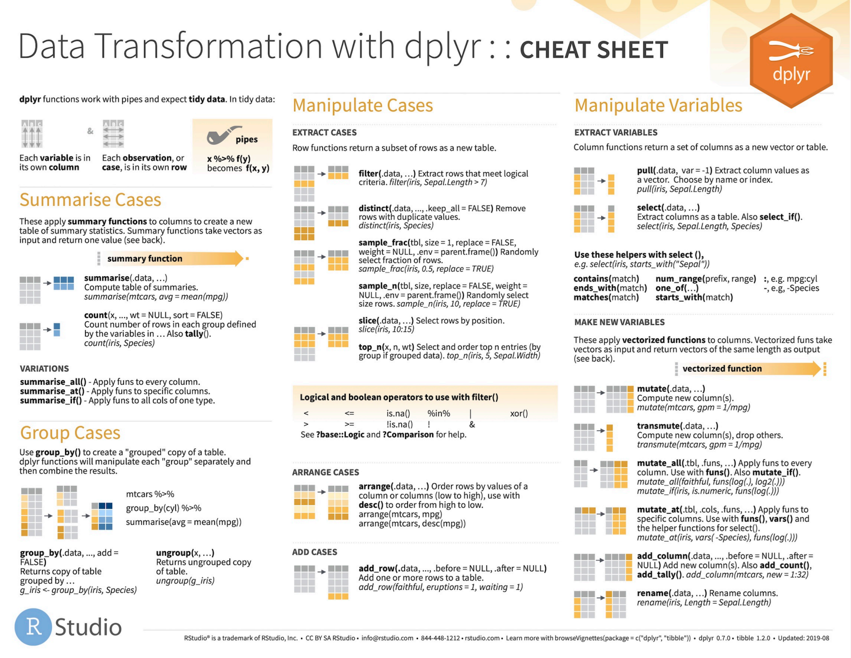
We assume you know how to make basic plots to aid exploratory data analysis. You should be able to:

- visualise single-variable distributions using dot plots and histograms (numeric variables) and bar plots (categorical variables);
- visualise associations between pairs of variables using scatter plots (numeric vs numeric), bar plots (categorical vs categorical), and box plots (numeric vs categorical).

We also assume you know how to make more informative plots by defining additional aesthetic mappings (e.g. `colour` and `fill`), or using faceting to construct a multi-panel plot according to the values of categorical variables.

### A dplyr cheat sheet

The developers of RStudio have produced very usable cheat sheats that summarise the core packages in the **tidyverse** ecosystem.



Data Transformation with dplyr : : **CHEAT SHEET**

dplyr

**dplyr** functions work with pipes and expect **tidy data**. In tidy data:

Each **variable** is in its own **column** & Each **observation**, or **case**, is in its own **row**

**pipes**
**x %>% f(y)** becomes **f(x, y)**

Summarise Cases

These apply **summary functions** to columns to create a new table of summary statistics. Summary functions take vectors as input and return one value (see back).

**summary function**

**summarise**(.data, …)
Compute table of summaries.
*summarise(mtcars, avg = mean(mpg))*

**count**(x, ..., wt = NULL, sort = FALSE)
Count number of rows in each group defined by the variables in … Also **tally**().
*count(iris, Species)*

**VARIATIONS**

**summarise_all()** - Apply funs to every column.
**summarise_at()** - Apply funs to specific columns.
**summarise_if()** - Apply funs to all cols of one type.

Group Cases

Use **group_by()** to create a "grouped" copy of a table. dplyr functions will manipulate each "group" separately and then combine the results.

mtcars %>%
group_by(cyl) %>%
summarise(avg = mean(mpg))

**group_by(**.data, ..., add = FALSE**)**
Returns copy of table grouped by …
*g_iris <- group_by(iris, Species)*

**ungroup(**x, …**)**
Returns ungrouped copy of table.
*ungroup(g_iris)*

Manipulate Cases

**EXTRACT CASES**

Row functions return a subset of rows as a new table.

**filter(**.data, …**)** Extract rows that meet logical criteria. *filter(iris, Sepal.Length > 7)*

**distinct(**.data, ..., .keep_all = FALSE**)** Remove rows with duplicate values.
*distinct(iris, Species)*

**sample_frac(**tbl, size = 1, replace = FALSE, weight = NULL, .env = parent.frame()**)** Randomly select fraction of rows.
*sample_frac(iris, 0.5, replace = TRUE)*

**sample_n(**tbl, size, replace = FALSE, weight = NULL, .env = parent.frame()**)** Randomly select size rows. *sample_n(iris, 10, replace = TRUE)*

**slice(**.data, …**)** Select rows by position.
*slice(iris, 10:15)*

**top_n(**x, n, wt**)** Select and order top n entries (by group if grouped data). *top_n(iris, 5, Sepal.Width)*

**Logical and boolean operators to use with filter()**

```
<     <=    is.na()    %in%    |    xor()
>     >=    !is.na()   !       &
```

See **?base::Logic** and **?Comparison** for help.

**ARRANGE CASES**

**arrange(**.data, …**)** Order rows by values of a column or columns (low to high), use with **desc()** to order from high to low.
arrange(mtcars, mpg)
arrange(mtcars, desc(mpg))

**ADD CASES**

**add_row(**.data, ..., .before = NULL, .after = NULL**)**
Add one or more rows to a table.
*add_row(faithful, eruptions = 1, waiting = 1)*

Manipulate Variables

**EXTRACT VARIABLES**

Column functions return a set of columns as a new vector or table.

**pull(**.data, var = -1**)** Extract column values as a vector. Choose by name or index.
*pull(iris, Sepal.Length)*

**select(**.data, …**)**
Extract columns as a table. Also **select_if()**.
*select(iris, Sepal.Length, Species)*

**Use these helpers with select (),**
*e.g. select(iris, starts_with("Sepal"))*

| | | |
|---|---|---|
| **contains(**match**)** | **num_range(**prefix, range**)** | **:**, e.g. mpg:cyl |
| **ends_with(**match**)** | **one_of(**…**)** | -, e.g, -Species |
| **matches(**match**)** | **starts_with(**match**)** | |

**MAKE NEW VARIABLES**

These apply **vectorized functions** to columns. Vectorized funs take vectors as input and return vectors of the same length as output (see back).

**vectorized function**

**mutate(**.data, …**)**
Compute new column(s).
*mutate(mtcars, gpm = 1/mpg)*

**transmute(**.data, …**)**
Compute new column(s), drop others.
*transmute(mtcars, gpm = 1/mpg)*

**mutate_all(**.tbl, .funs, …**)** Apply funs to every column. Use with **funs()**. Also **mutate_if()**.
*mutate_all(faithful, funs(log(.), log2(.)))*
*mutate_if(iris, is.numeric, funs(log(.)))*

**mutate_at(**.tbl, .cols, .funs, …**)** Apply funs to specific columns. Use with **funs()**, **vars()** and the helper functions for select().
*mutate_at(iris, vars( -Species), funs(log(.)))*

**add_column(**.data, ..., .before = NULL, .after = NULL**)** Add new column(s). Also **add_count()**, **add_tally()**. *add_column(mtcars, new = 1:32)*

**rename(**.data, …**)** Rename columns.
*rename(iris, Length = Sepal.Length)*

R Studio

RStudio® is a trademark of RStudio, Inc. • CC BY SA RStudio • info@rstudio.com • 844-448-1212 • rstudio.com • Learn more with browseVignettes(package = c("dplyr", "tibble")) • dplyr 0.7.0 • tibble 1.2.0 • Updated: 2019-08

Download these, print out a copy, and refer to it when you need to do something with **dplyr** or **ggplot2**.

# How the book is formatted

We have adopted several formatting conventions in this book to distinguish between normal text, R code, file names, and so on. You need to be aware to make best use of the book.

## Text, instructions, and explanations

Normal text—instructions, explanations and so on—are written in the same type as this document. We will tend to use bold for emphasis and italics to highlight specific technical terms when they are first introduced. In addition:

- `This typeface` is used to distinguish R code within a sentence of text: e.g. "We use the `mutate` function to change or add new variables."
- A sequence of selections from an RStudio menu is indicated as follows: e.g. **File** **New File** **R Script**

- File names referred to in the general text are given in upper case in the normal typeface: e.g. MYFILE.CSV.

At various points, you will come across text in different coloured boxes. These are designed to highlight stand-alone exercises or little pieces of supplementary information that might otherwise break the flow. There are three different kinds of boxes:

**Action!**

This is an **action** box. We use these when we want you to do something. Do not ignore these boxes.

**Information!**

This is an **information** box. These aim to offer a discussion of why something works the way it does.

**Warning!**

This is a **warning** box. These usually highlight a common 'gotcha' that might trip up new users.

## R code and output

We try to illustrate ideas using snippets of real R code where possible. Stand alone snippets are formatted like this:

```
tmp <- 1
print(tmp)
```

```
## [1] 1
```

The lines that start with **##** show us what R prints to the screen after it evaluates some R code; they show the output. The lines that **do not** start with **##** show us the R code; they show us the input. So remember, the absence of **##** shows us what we are asking R to do. Otherwise, we are looking at something R prints in response to these instructions.

# Part I

# Collecting and Using Data

# Chapter 1

# The scientific process

*There is something fascinating about science. One gets such a wholesale return of conjecture out of a trifling investment of fact.*

Mark Twain

*To do science is to search for patterns, not simply to accumulate facts.*

Robert MacArthur

## 1.1 Stages in the scientific process

Science is about asking, and answering, the right questions. A number of distinct stages occur within this process: making observations, asking questions, formulating hypotheses, and testing predictions. Collectively these are the building blocks of what is known as the scientific method. Exactly how they fit together, and what the philosophical and practical limitations of different approaches are, have been the subject of much debate by philosophers of science. We're not going to tackle those issues here, but instead try to extract a general working framework for the process of a typical scientific investigation.

### 1.1.1 Observations

**Observation** — *information, or impression, about events or objects.*

In general the questions we ask are not generated by pure abstract thought, but are a result of observations about the natural world. These may take the form of direct observations we make ourselves, patterns that crop up in data collected for other purposes, in non-specific surveys, and the accumulated works of other people.

Such observations of biological systems will lead almost automatically into asking questions.

### 1.1.2 Questions

**Question** — *what it is that you want to know; the scope of your investigation.*

It is important to keep the question sufficiently focussed. The overall aim of a particular study should be to answer one or (perhaps) two questions. The question of why the tropics are more diverse than the temperate regions is a valid and fascinating question, but it's probably not be a good choice for a final year project or even a PhD.

The next stage is to formulate an hypothesis.

### 1.1.3 Hypotheses

**Hypothesis** — *an explanation proposed to account for observed facts.*

In general, in biology, the critical distinguishing feature of an hypothesis is that it specifies some biological process, or processes, which might account for the observations made. Formulating hypotheses requires more than just a restatement of the question – it usually embodies some mechanism (though this may not be fully understood) and it will often draw on additional information. One question will also often generate more than one hypothesis.

Once we have formulated a set of hypotheses, we need to derive the predictions that follow.

### 1.1.4 Predictions

**Prediction** — *what you would expect to see if the hypothesis was true.*

Hypotheses are about proposing explanations, but they might not be directly testable. That is, they may not tell us what data to collect, or what pattern to expect in the data. To be able to test an hypothesis we need to make some predictions from that hypothesis. These will be determined both by what we expect to see and what it is possible, or practical, to measure. A prediction is not simply a rephrasing of the hypothesis – it should more or less lead to a statement of the experiment to conduct or observation to make, and type of data to collect.

The ideal prediction is one that is unique to the hypothesis it is based on, so if the prediction is true only one of the hypotheses could have been responsible. It may not always be possible to generate such 'clean' predictions, in which case, some combination of predictions may need to be considered. Additionally, several processes may be operating at the same time, which makes hypothesis testing harder still, because it may be necessary to consider two or more hypotheses, and their corresponding predictions, in combination.

Science is hard!

## 1.2 An example

#### Observations

Imagine, while observing a stream one day, we notice that the freshwater 'shrimps' (*Gammarus*) seem to occur almost entirely under stones; we rarely seem to see them in a patch of open stream bed. Having made observations, it may be necessary to collect some more data to check that this phenomenon is not just a one-off event, or a false impression. Look under a few more stones, watch the same species another day, or in another place, check the literature for similar observations by others.

#### Question

Why does *Gammarus* spend most of its time under stones?

#### Hypotheses

*Gammarus* occur under stones because:

- they need to shelter from the current
- their food (leaf litter) gets trapped and accumulates under stones
- they are subject to predation by visually hunting fish and need to remain out of sight

#### Predictions

Taking each hypothesis in turn, we expect to observe:

- *Shelter hypothesis:* a greater proportion of *Gammarus* should be found in the open in streams with slow flow, or in slower flowing areas of a stream.
- *Food hypothesis: Gammarus* should not aggregate under stones from which all leaf litter has been removed; *Gammarus* should aggregate on patches of leaf litter tethered in the open part of the stream bed.

- *Predation hypothesis: Gammarus* should aggregate under stones more in streams where fish are present than where they are not; *Gammarus* may spend less time under stones at night.

If we want to tease apart our hypotheses we may need to refine these predictions. For example, *Gammarus* may be under stones because it prefers the sheltered environment, but also because food accumulates there. In this case we might expect that *Gammarus* will show a weak aggregative response to shelter alone, or food alone, and a stronger one to them both together.

## 1.3 Hypothesis testing

Once we have firmed up our hypotheses and predictions we are in the position to collect data to evaluate our ideas. On the basis of these data we will either accept or reject our various hypotheses. The important thing to realise about the process of hypothesis testing is that, in science especially, hypotheses are either rejected, or not rejected, but an hypothesis can rarely, except in trivial cases, be proved.

Why is this? Since we cannot be sure that we've thought of all the possible hypotheses to explain an observation, finding evidence that supports a prediction does not guarantee that the underlying hypothesis is the only one which could have produced the effect we find. On the other hand, if we find evidence that directly contradicts the prediction(s) from an hypothesis, we can be certain (assuming the prediction and data are not flawed) that the hypothesis cannot be true.

An hypothesis which predicted that all conifers should be evergreen could be supported by numerous observations of different conifer species in forests around the world, but is conclusively refuted by the first larch tree we encounter.

Having tested our hypothesis, by examining the evidence that its predictions are true, we may accept it as the best current explanation of the observations, or we may reject it as an explanation, and turn to other hypotheses. The same procedure must then be repeated for these hypotheses.

This basic cycle of proposing hypotheses and then seeking evidence potentially capable of falsifying them, is, in essence, the idealized model of the scientific process famously proposed by the philosopher of science Karl Popper (1902-1994). It is often termed *falsificationism*.

## 1.4 Don't we ever know anything for sure?

The method presented here provides a view of science as one in which we suggest hypotheses, then test them, trying to reject them by finding conclusive counter-evidence, then replacing them with new hypotheses. It all sounds a bit frustrating.

In fact of course we do 'accept' hypotheses all the time – that is, we fail to reject them over and over again. These hypotheses become more accepted and in some sense become regarded as 'true' if repeated attempts to test them all fail to provide good counter evidence. In other words, we have some ideas that are doing pretty well in terms of resisting falsification, and we use these as our best estimates of the truth, with the proviso that it is still possible a better idea will come along in due course.

The simple process of falsification described above also presents a picture of scientists as neutral, objective creatures, rationally proceeding through cycles of setting up hypotheses, testing them, rejecting them, setting them aside and starting over again. Of course this is not a true reflection of the complex, often messy, business involved in trying to figure out how the world works. Philosophers of science have argued long and hard about how far from this idealized process real science actually is. Various alternative philosophies suggest more 'realistic' processes. For example…

- Thomas Kuhn's view of science as periods of relative stasis, where people work within an accepted paradigm (a set of views about how things work) despite accumulating evidence that doesn't always support the paradigm, until finally it is upset by a 'revolution' which rejects the entire paradigm, and proposes a new view. This is where the phrase 'paradigm shift' originated.
- Imre Lakatos proposed some resolution of Kuhn's views, suggesting that scientific ideas were grouped together in 'research programmes' concerned with particular endeavours, and that within these there may be core ideas that are not challenged, but other related ideas that are being challenged and adjusted by falsification, and that together these make each research programme progress.

This is a very over-simplified sketch of some important ideas. These are well worth reading about, but in practice most philosophical arguments are more focused on how whole areas of science develop. When thinking about how to construct a study of one problem, the basic falsification cycle is a pretty good approach to have in mind. Keep in mind that the process laid out here is not a strict set of rules, but outlines an approach to scientific investigation which is widely considered to provide a rigorous and productive system. As with all such systems understanding the 'normal' process is a prerequisite for constructively breaking the rules.

# Chapter 2

# Data and variables

> *The truth is the science of nature has been already too long made only a work of the Brain and the Fancy. It is now high time that it should return to the plainness and soundness of Observations on material and obvious things.*
>
> Robert Hooke (1665)
>
> *The plural of anecdote is not data.*
>
> Roger Brinner

## 2.1 "Observations on material and obvious things"

As Hooke's observation suggests, science cannot proceed on theory alone. The information we gather about a system stimulates questions and ideas about it and, in turn, can also allow us to test these ideas. In fact, the idea of measuring and counting things is so familiar that it is easy to start a project without giving much thought to something as mundane as the nature, quantity and resolution of the data we intend to collect. Nonetheless, the features of the data we collect determine the types of analyses that can be carried out and the confidence we can have in the conclusions drawn. In the end, no level of analytical sophistication will enable us to extract information that is not there to begin with.

So what is there to say about data? The first point to note is that there are many different sorts of data. We focus on simple kinds of data in this book, such as quantities ('how much') or classifications ('what kind'), but not everything can be thought of in simple terms. Examples of more complex data types include spatial maps of species' occurrence, the DNA sequences of individuals, and networks of feeding relationships among species (i.e. food webs) or gene activation (gene regulatory networks).

The data we collect in a study can be organised into one or more statistical **variables**. In its loosest sense, statisticians use the word 'variable' to mean any characteristic measured or experimentally controlled on different items or objects. People tend to think of variables as numeric quantities, such as cell size, but nothing stops us working with non-numeric variables, such as developmental stage.

Collectively, a set of related variables is referred to as a **data set** (or just 'the data'). Confused? Consider a concrete example: the spatial maps of species' occurrence mentioned above. A minimal data set might comprise two variables containing the x and y position of sample locations, a third variable denoting the presence/absence of a species, and one or more additional variables containing information about the environmental factors we measured.

**Data and variables in R**

Data sets are often stored in data frames when working with R. A data frame can be thought of as having rows and columns. Each column in a data frame has to be one of R's many types of 'vector'. Typically, these are simple things, like numeric or character vectors, but other kinds of vector can be used (e.g. dates and intervals of times). The data it contains are said to be tidy when each column of a data frame corresponds to a statistical variable, and each row corresponds to a single observation (a cell, a tissue sample, an animal, etc).

This simple connection between abstract statistical concepts and concrete objects in R is not a coincidence. Its structures and abstractions reflect the fact that R was designed first and foremost to analyse data.

**The data are or the data is?**

Strictly speaking, the word data is the plural of datum (a single piece of information), so we should say "the data are…" not "the data is…". That said, the use of the word in the singular has become widespread, despite the objections of Grammar Nazis, and to be perfectly honest, either usage is fine these days. We lean towards plural usage with 'data', but both forms appear in this book…

## 2.2 Revision: Types of variable

When designing an experiment or carrying out some data analysis, it is vital to understand what sort of data we need, or have, as this affects what we can do with it. The variables in a data set are often classified as being either **numeric** or **categorical**:

- Numeric variables have values that describe a measurable quantity as a number, like 'how many' or 'how much'.
- Categorical variables have values that describe a characteristic of an observation, like 'what type' or 'which category'.

Numeric variables can be further characterised according to the type of scale they are measured on: **interval** vs. **ratio** scales. Categorical variables can be further characterised according to whether or not they have a natural order: **nominal** vs. **ordinal** variables.

This is probably the minimal classification scheme we need to learn if we want to analyse our data well. Let's take a closer look at this scheme.

### 2.2.1 Nominal (categorical) variables

Nominal variables arise when observations are recorded as categories that have no natural ordering relative to one other. For example:

| **Marital status** | **Sex** | **Colour morph** |
|---|---|---|
| Single | Male | Red |
| Married | Female | Yellow |
| Widowed | | Black |
| Divorced | | |

Variables of this type are common in surveys where, for example, a record is made of the 'type of thing' (e.g. the species) observed.

### 2.2.2 Ordinal (categorical) data

Ordinal variables occur where observations can be assigned some meaningful order but the exact 'distance' between items is not fixed, or even known. For example, if we are studying the behaviour of an animal when it meets a conspecific, it may not be possible to obtain quantitative data about the interaction, but we might be able to score the behaviours in order of aggressiveness:

| **Behaviour** | **Score** |
|---|---|
| initiates attack | 3 |

| | |
|---|---|
| aggressive display | 2 |
| ignores | 1 |
| retreats | 0 |

Rank orderings are a type of ordinal data. For example, the order in which runners finish a race (1st, 2nd, 3rd, etc.) is a rank ordering. It doesn't tell us whether it was a close finish or not but still conveys important information about the result.

In both situations, we can say something about the relationships between categories: in the first example, the larger the score, the more aggressive the response; in the second example, the greater the rank, the slower the runner. However, we can't say that the gap between the first runner and the second was the same as between the second and third and we can't say that a score of 2 is twice as aggressive as a score of 1.

⚠ **How should you encode different categories in R?**

We always have to define a coding scheme to represent the categories of nominal/ordinal variables in computerised form. It was once common to assign numbers to different categories (e.g. Female=1, Male=2). This method was sensible in the early days of computer-based analysis because it allowed data to be stored efficiently. This efficiency argument is no longer relevant. What's more, there are good *practical* reasons to avoid coding categorical variables as numbers:

- Numeric coding makes it harder to understand our raw data and interpret the output of a statistical analysis of those data. This is because we have to remember which number is associated with each category.
- Numeric codes are arbitrary and should not be used as numbers in mathematical operations. In the above example, it is meaningless to say 2 ["male"] is twice the size of 1 ["female"].

R usually assumes that a variable containing numeric values is meant to be treated as a number. If we use numeric coding for a categorical variable, R will try to treat the offending variable as a number, leading to confusing errors.

The take home message: Always use words (e.g., 'female' vs. 'male'), not numbers, to describe categories in R.

### 2.2.3 Ratio scale (numeric) variables

Ratio scale numeric variables have a true zero and a known and consistent mathematical relationship between any points on the measurement scale. For example, there is a temperature scale that has a true zero: the Kelvin scale. Zero K is absolute zero, where matter has no thermal energy whatsoever. Temperature measurements in degrees K are on a ratio scale, i.e. it **does** make sense to say that 60 K is twice as hot as 30 K. Ratio scale numeric variables are quite familiar, because the physical quantities that crop up in science are usually measured on a ratio scale: length, weight, and numbers of objects are good examples.

### 2.2.4 Interval scale (numeric) variables

Interval scale variables take values on a consistent numeric scale, but where that scale starts at an arbitrary point. Differences between values on an interval scale are meaningful, whereas ratios are not. The temperature on the Celsius scale is a good example of interval data. We can say that 60º C is hotter than 50º C, and we can say that the difference in temperature between 60º C and 70º C is the same as that between -20º C and -10º C. We cannot say that 60º C is twice as hot as 30º C. This is because temperature on the Celsius scale has an artificial zero value—the freezing point of water. The idea becomes obvious when we consider that temperature can equally well be measured on the Fahrenheit scale (where the freezing point of water is 32º F).

**Continuous or discrete?**

A common confusion with numeric data concerns whether the data are on continuous or discrete scales. Many biological ratio data are discrete (i.e. only certain discrete values are possible in the original data). Count data are an obvious example, e.g. the number of eggs found in a nest, the number of mutations in a cancer cell, or the number of heartbeats counted in a minute. These can only comprise whole numbers, ‘in between’ values are not possible.

However, the distinction between continuous and discrete data is often not clear cut – even ‘continuous’ variables such as weight are made discrete in reality by the fact that our measuring apparatus is of limited resolution (i.e. a balance may weigh to the nearest 0.01 g). So… keep in mind that the fact that data look (or really are) discrete does not mean they are necessarily truly discrete data.

### 2.2.5 Why does the distinction matter?

The measurement scale affects what we can do mathematically to a numeric variable. We can add and subtract interval scale variables, but we can not divide or multiply them and arrive a meaningful result. In contrast, we can add, subtract, multiply or divide ratio scale variables. Put another way, differences and distances are meaningful for both interval and ratio scales, but ratios are only meaningful for ratio variables (yes, the clue is in the name).

### 2.2.6 Which is best?

All types of data can be useful, but it is important to be aware that not all types of data can be used with all statistical models. This is one very good reason for why it is worth having a clear idea of the statistical tools we intend to use when designing a study.

In general, ratio variables are best suited to statistical analysis. However, biological systems often cannot be readily represented as ratio data, or the work involved in collecting good ratio data may be vastly greater than the resources allow, or the question we are interested in may not demand ratio data to achieve a perfectly satisfactory answer. It is this last question that should come first when thinking about a study. What sort of data do we need to answer the question we are interested in? Suppose it is clear at the outset that data on a rank scale will not be sufficiently detailed to enable us to answer the question. In that case, we must either develop a better way of collecting the data, or abandon that approach altogether. If we know the data we're able to collect cannot address the question, we should do something else.

An obvious but important point: we can always convert measurements taken on a ratio scale to an interval scale, but we cannot do the reverse. Similarly, we can convert interval scale data to ordinal data, but we cannot do the reverse. That said, it's good practice to avoid such conversions as they inevitably result in a loss of information.

## 2.3 Accuracy and precision

### 2.3.1 What do they mean?

The two terms accuracy and precision are used more or less synonymously in everyday speech, but they have quite distinct meanings in scientific investigation.

**Accuracy** – how close a measurement is to the true value of whatever it is you are trying to measure.

**Precision** – how repeatable a measure is, irrespective of whether it is close to the actual value.

Imagine we are measuring an insect's weight on an old and poorly maintained balance, which measures to the nearest 0.1 g. In that case, we might weigh the same insect several times and each time get a different weight — the balance is not very precise, though some of the measurements might happen be quite close to the real weight. By contrast, we could be using a new electronic balance, weighing to the nearest 0.01g, but which has been incorrectly zeroed so that it is 0.2 g out from the true weight. Repeated weighing here might yield results that are identical but all incorrect — the balance is precise, but the results are inaccurate. The analogy often used is with shooting at a target:

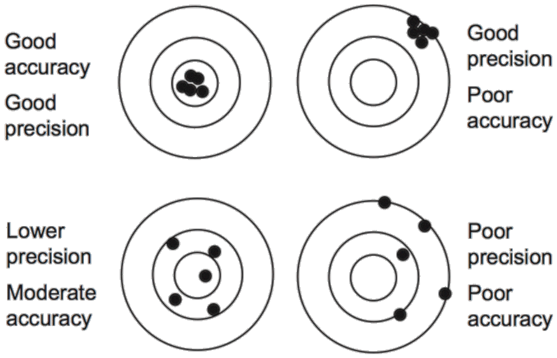


Figure 2.1: Accuracy and precision

It's important to know how accurate and how precise our data are. The ideal is the situation in the top left target in the diagram. Still, high precision is not possible in many circumstances, and it is usually preferable to make measurements whose accuracy we can be reasonably confident (bottom left), than more precise measurements, whose accuracy may be suspect (top right). Taking an average of the values for the bottom left target would produce a value pretty close to the centre; taking an average for the top right target wouldn't help the accuracy at all.

### 2.3.2 Implied precision – significant figures

It's worth being aware that when we state results, we're making implicit statements about the precision of the measurement. The number of significant figures we use suggests something about the precision of the result. A result quoted as 12.375 mm implies the measurement is more precise than one quoted as 12.4 mm. A value of 12.4 actually measured with the same precision as 12.735 should properly be written 12.400. When quoting results, look at the original data to decide how many significant figures to use—generally, the same number of significant figures will be appropriate.

These considerations do not apply in the same way when working with discrete data, e.g. precision of measurement is not an issue in recording the number of eggs in a nest. We use 4 not 4.0, but since 4 eggs implies 4.0 eggs it's fine to quote average clutch size from several nests as 4.3 eggs. However, even with discrete data, if numbers are large then obviously precision is an issue again … a figure of 300 000 ants in a nest is likely to imply a precision of plus or minus 50 000. A figure of 320987 ants implies an improbably precise measurement.

### 2.3.3 How precise should measurements be?

The appropriate precision to use when making measurements is largely common sense. It will depend on practicality and the use to which we wish to put the data. It may not be possible to weigh an elephant to the nearest 0.001g, but if we want to know whether the elephant will cause a 10-tonne bridge to collapse, then the nearest tonne will be good enough. On the other hand, if we want to compare the mean sizes of male and female elephants, then the nearest 100 kg may be sufficient, but if we plan to monitor the progress of a pregnant female elephant, then the nearest 10 kg or less might be desirable.

As a rough guide aim, where possible, for a scale where the number of measurement steps is between 30 and 300 over the expected range of the data. For example, in a study of the variation in shell thickness of dogwhelks on a 300 m transect up a shore, it would be adequate to measure the position of each sampling point on the transect to the nearest metre, but shell thickness will almost certainly need to be measured to the nearest 0.1 mm.

### 2.3.4 Error, bias and prejudice

Error is present in almost all biological data, but not all error is equally problematic. Usually, the worst form of error is bias. Bias is a systematic lack of accuracy—i.e. the data are not just inaccurate—but all tend to deviate from the true measurements in the same direction (situations B and D in the 'target' analogy above). Thus there is an important distinction in statistics between the situation where the measurements differ from the true value at random and those where they differ systematically. Measurements lacking some precision, such as the situation illustrated in C, may still yield a reasonable estimate of the true value if the mean of a number of values is taken.

Avoiding bias in the collection of data is one of the most important skills in designing scientific investigations. Some forms of bias are obvious, others more subtle and hard to spot:

- *Non-random sampling.* Many sampling techniques are selective and may result in biased information. For example, pitfall trapping of arthropods will favour collection of the very active species that encounter traps most frequently. Studying escape responses of an organism in the lab may be biased if the process of catching organisms to use in the study selected for those whose escape response is poorest.

- *Conditioning of biological material.* Organisms kept under particular conditions for long periods may become acclimatised to those conditions. Similarly, if stocks are kept in a laboratory for many generations, their characteristics may change through natural selection. Such organisms may give a biased impression of the behaviour of the organism in natural conditions.
- *Interference by the process of investigation.* Often the process of making a measurement itself distorts the characteristic being measured. For example, it may be hard to measure the level of adrenalin in the blood of a small mammal without affecting the adrenalin level in the process. Pitfall traps are often filled with a preservative, such as ethanol, but the ethanol attracts some species of insect that normally feed on decaying fruit and use the fermentation products as a cue to find resources.
- *Investigator bias.* Measurements can be strongly influenced by conscious or unconscious prejudice on the part of the investigator. We rarely undertake studies without an initial idea of what we are expecting. For example, rounding up 'in between' values in the samples you are expecting to have large values and rounding down where a smaller value is expected, or having another 'random' throw of a quadrat when it doesn't land in a 'typical' bit of habitat.

The ways in which biases, conscious and unconscious, can affect our investigations are many, often subtle, and sometimes serious. Sutherland (1994) gives an illuminating catalogue of how biases affect our perception of the world and the judgements we make about it. The message is that the results we get from an investigation must always be judged and interpreted with respect to the nature of the data used to derive them—if the data are suspect, the results will be suspect too.

# Chapter 3

# Learning from data

> Statistics is the science of learning from data, and of measuring, controlling, and communicating uncertainty; and it thereby provides the navigation essential for controlling the course of scientific and societal advances.
>
> Davidian and Louis (2012)

The particular flavour of statistics covered in this book is called **frequentist statistics**. From a practical perspective, it is perfectly possible to apply frequentist tools by learning a few basic 'rules' about how they work. However, it is easier to apply a technique if we understand at least roughly how it works.

The goal of the next few chapters is to provide an overview of frequentist concepts without getting bogged down with the underpinning mathematics. We aim to arrive at a conceptual feel for the essential ideas. These can be difficult to understand at first. This is fine. We'll return to them over and over again as we introduce different statistical models and tests.

We're going to start by laying out a somewhat simplified overview of the steps involved in 'doing frequentist statistics' in this chapter. We'll also introduce a few key definitions along the way. Later chapters will then drill into important concepts like sampling variation, standard errors, null hypotheses and *p*-values.

## 3.1 Populations

When a biologist talks about a population they mean a group of individuals of the same species that interact. What does a statistician mean when they talk about populations? The word has a very different meaning in statistics. Indeed, it is a much more abstract concept: a statistical population is any group of items or objects that share certain attributes or properties. This is best understood by example:

- The readers of this book could be viewed as a statistical population. This book is targeted at undergraduate students interested in biology. That group are primarily in their late teens and early 20s, and they tend to have similar educational backgrounds and career aspirations. As a consequence of these similarities, these students tend to be more similar to one another than they would be to a randomly chosen inhabitant of say, the US or Germany.
- The different areas of peatland in the UK can be thought of as a statistical population. There are many peatland sites in the UK, and although their precise ecology varies from one location to the next, they are also very similar in many respects. Peatland habitat is characterised by low-growing vegetation and acidic soils. If you visit two different peatland sites in the UK, they will seem quite similar compared to, for example, a neighbouring calcareous grassland.
- A population of plants or animals (as typically understood by biologists) can also be thought of as a statistical population. Indeed, this is often the kind of population ecologists are interested in studying. The individuals in a biological population share common behaviours and physiological characteristics, and much of biology is concerned with learning about these properties of organisms (e.g. development, physiology and behaviour).

We conceptualise populations as fixed and unknown entities within the framework of frequentist statistics. The goal of statistics is then to learn something about populations by analysing data collected from them. It is important to realise that the investigator defines 'the population', and 'the something we want to learn about' can be anything we know how to measure.

Consider the examples again:

- A social scientist might be interested in the political attitudes of undergraduates, so they choose to survey a group of students at university.
- A climate change scientist might measure the mass of carbon stored in peatland areas at sites across Scotland and northern England.
- A behavioural ecologist might want to understand how much time beavers spend foraging for food, so they might study a Scottish population.

So how does the process of learning about a population from data work?

## 3.2 Learning about populations

The three examples discussed above involve very different kinds of populations, questions and data. Nonetheless, there are fundamental commonalities in how those studies could be addressed. The process can be broken down into distinct steps:

### Step 1: Refine your questions, hypotheses and predictions

We discussed this step in the scientific process chapter. The key point is that we should never begin collecting data until we've set out the relevant questions, hypotheses and predictions. This might seem obvious, but it is surprising how often people don't get these things straight before diving into data collection. Collecting data without a clear scientific objective and rationale is guaranteed to waste a lot of time and energy.

### Step 2: Decide which population(s) is (are) important

The second step is to decide which population(s) we should study. What constitutes 'the population' will be obvious in some kinds of study. In the three cases above, the corresponding populations we choose to study could be undergraduate students at a university, peatland habitats from across the UK, and beavers in Scotland.

It is fairly easy to define what we mean by 'a population' in those examples. However, the problem of deciding which population(s) to study can be more subtle than it first appears. What happens if we're planning an experiment? Imagine that we want to test a prediction that nutrient addition reduces biodiversity in chalk grasslands. We might set up an experiment with two kinds of plots:

1. manipulated plots where we add fertiliser, and
2. control plots where we do nothing.

Comparing these in some way would allow us to assess the impact of adding nutrients. That's easy enough to understand.

What can we say about the statistical populations in this example? First, there are two of them—control and manipulated communities—and they are completely defined by the experimental design. The nutrient addition plots don't 'exist' until we do the experiment. That's not hugely intuitive. What's more, the weird mental contortion that a frequentist does is to imagine that the experimental plots are part of some larger, unobserved population of nutrient addition plots.

The important point is that statistical populations are something the investigator defines. They might be 'real', undergraduates at university, or they might be something that doesn't exist in a meaningful way, like a population of not-yet-realised experimentally manipulated plots. In either case, we can use the same statistical techniques to learn something general about 'the population(s)'.

### Step 3: Decide which variables to study

The next step is to decide which attributes of the items that comprise our population(s) need to be measured. This comes down to deciding which variable(s) to measure. We discussed some of the considerations in the data and variables chapter. In the examples above, the appropriate variables would be things like a standardised measure of political attitude, the mass of stored carbon per unit area, or the body mass of individuals.

This step is often straightforward, though some effort may be required to pick among different options. There isn't much ambiguity associated with a physical variable like body mass, but something like 'political attitude' needs careful thought. Can we quantify this by studying just one thing, like voting patterns? Probably not. Maybe we need to measure several things and aggregate those in some way.

### Step 4: Decide which population parameters are relevant

Once we have decided which variables to study, we have to decide which **population parameters** are relevant. A population parameter is simply a numeric quantity that describes the variable(s) we want to learn about. A population parameter defines a property of that variable in the population—it's not a property of the actual data.

A simple population parameter familiar to many people is the population mean. We often study means because they allow us to answer questions like, "how much, on average, of something is present?". Quite a number of the statistical tests boil down to asking questions of population means. That doesn't mean we can't ask questions about other kinds of population parameters:

- Statistical genetics aims to partition *variability* among individuals—we want to know how much phenotypic variation is due to genetic vs non-genetic sources. In this case, it is population variances we want to learn about.
- Sometimes we want to understand how two or more aspects of the population are *associated* with one another. In this situation, something called a correlation coefficient might be the right parameter to focus on.

Once we have identified the population parameter we care about, we need a way to learn something about it. That takes us to the next step.

### Step 5: Gather a representative sample

We would not need statistics if we could measure every member of a population. In that case, we would simply calculate the parameter of interest from an exhaustive sample. However, we always have limited time and money to invest in any problem in the real world. This means we usually have to work with a limited **sample** of the population. A sample is a subset of a population that has been selected to be representative. That word 'representative' is really important in this context. A representative sample is one that reflects the characteristics of the entire population.

It is really important to use sampling methods that lead to representative samples. Why? It's very hard to infer anything useful about a population from a non-representative sample. For example, if we aim to understand the reproductive characteristics of our favourite study organism, but we only sample old-aged individuals, it will be impossible to generalise our findings when reproductive performance changes with age.

Collecting larger samples does not solve this kind of problem. For example, a sample that contains mostly old-age individuals doesn't tell us much about younger age groups. People often assume that 'more data = better data' but that is not be true when the data are not representative of the population. Try to remember that in your own work.

Methods for obtaining useful samples and calculateing meaningful estimates are an important part of statistics. They fall under the banners of experimental design and sampling theory. These are large, technical topics that are well beyond the scope of this book. However, we will touch on a few of the more important aspects later.

#### Step 6: Estimate the population parameter

After acquiring a representative sample from a population we can calculate something called a **point estimate** of the population parameter. Remember, that parameter is ultimately unknown. A point estimate is a number that represents our 'best guess' at its true, unknown value. For example, if we are interested in the population mean, the obvious point estimate to use is the 'the average' we learn about at school.

There are many different kinds of point estimates we can calculate. What we choose to estimate depends on the question we're answering. If we care about variability we might calculate one or more sample 'variances'. Alternatively, we might calculate some kind of measure of an effect or change, like the difference between two sample means.

(By the way, people often say 'estimate' instead of 'point estimate', for the simple reason that saying or writing 'point estimate' all the time soon becomes tedious. We use both terms in this book.)

### Step 7: Estimate uncertainty

A point estimate is useless on its own. Why? Because it is derived from a limited sample of the population. Even if we collect a huge representative sample, its composition won't exactly match that of the population. This means any point estimate we derive from the sample will be imperfect in the sense that it won't exactly match the true population value.

There is always some uncertainty associated with estimates population parameters. Our job as scientists is to to find a way to *quantify that uncertainty*. This part of the frequentist method is probably the hardest to understand so we'll spend some time considering it in the sampling error chapter.

### Step 8: Answer the question!

Once we have our point estimates and measures of uncertainty, we're finally in the position to start answering questions.

Imagine that we want to answer a seemingly simple question, e.g. "Are there more than 200 tonnes of carbon per hectare stored in the peatland of the Peak District?" We might sample several sites, measure the stored carbon at each of these, and then calculate the mean of these measurements. What should we conclude if the sample mean is 210 t h$^{-1}$? We can't say much until we have a sense of how reliable that mean is likely to be. To answer our question, we have to assess whether the difference we observe (210 - 200 = 10) was just a fluke.

The tools we'll learn about in this book are designed to answer many different kinds of scientific questions. Nonetheless, they all boil down to the same general question: is the pattern I see 'real', or is it instead likely to be a result of chance variation? To tackle this, we combine point estimates and measures of uncertainty in various ways. Statistical software like R will do the hard work for us, but we still just have to learn how to interpret the results it gives us.

## 3.3 A simple example

The best way to get a sense of how all this fits together is by working through an example (it is worth paying attention to this because we'll use it again in later chapters). We're only going to focus on steps 1-6 for the moment. The last two steps are sufficiently challenging that they need their own chapters.

Imagine we're interested in a plant species that is phenotypically polymorphic. There are two different 'morphs': a purple morph and a green morph. We can depict this situation visually with a map showing where purple and green plants are located on a hypothetical landscape:

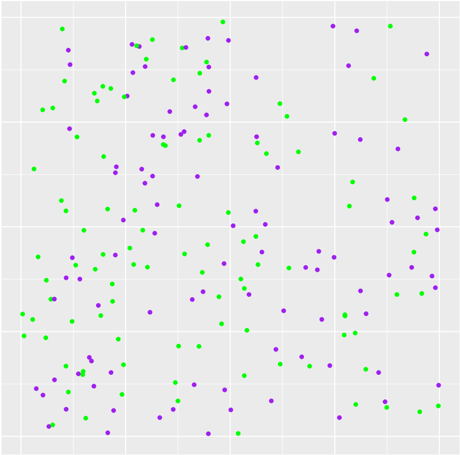


Figure 3.1: Stylised landscape showing a population of purple and green plants

These are idealised data generated using a simulation in R but proceed as though these they real.

### Step 1: Refine your questions, hypotheses and predictions

Imagine we had also previously studied a different population of this species that exhibits the same polymorphism. We think the two populations were once connected but habitat loss has isolated them. Our previous studies with the old neighbouring population have shown:

- A single gene with two alleles controls the colour polymorphism: a recessive mutant allele ('P') confers the purple colour and the dominant wild-type allele ('G') makes plants green. Genetic work has shown that the two alleles are present in a ratio of about 1:1 in the neighbouring population.
- There seems to be no observable fitness difference between the two morphs in the neighbouring population. What's more, about 25% of plants are purple. This suggests that the two alleles are in Hardy-Weinberg equilibrium. These two observations suggest that there is no selection operating on the polymorphism.

Things are different in the new study population. The purple morph seems to be about as common as the green morph, and preliminary research indicates purple plants seem to produce more seeds than green plants. Our **hypothesis** is, therefore, that purple plants have a fitness advantage in the new study population. Our **prediction** is that the frequency of the purple morph will be greater than 25% in the new study population, as selection is driving the 'P' allele to fixation.

(This isn't the strongest test of our hypothesis, by the way. Really, we need to study allele and genotype frequencies, not just phenotypes. It is good enough for illustrative purposes.)

#### Step 2: Decide which population is important

This situation is made up, so consideration of the statistical population is a little artificial. In the real world, we would consider various factors, such as whether or not we can study the whole population or must restrict ourselves to one sub-population. Working at a large scale should produce a more general result, but it could also present a significant logistical challenge.

#### Step 3: Decide which variables to study

This step is easy. We could measure all kinds of different attributes of the plants—biomass, height, seed production, etc—but to study the polymorphism, we only need to collect information about the colour of different individuals. This means we are going to be working with a nominal variable that takes two values: 'purple' or 'green'.

#### Step 4: Decide which population parameters are relevant

The prediction we want to test is about the purple morph frequency, or equivalently, the percentage, or proportion, of purple plants. Therefore, the relevant population parameter is the frequency of purple morphs in the wider population. We need to collect data that would allow us learn about this unknown quantity.

#### Step 5: Gather a representative sample

A representative sample in this situation is one in which every individual on the landscape has the same probability of being sampled. This is what people mean when they refer to a 'random sample'. Gathering a random sample of organisms from across a landscape is surprisingly hard to do. Fortunately, it is very easy to do in a simulation. Let's seen what happens if we sample 20 plants at random…

The new plot shows the original population of plants, only this time we've circled the sampled individuals in red. We would only 'see' this sample of plants in the real world.

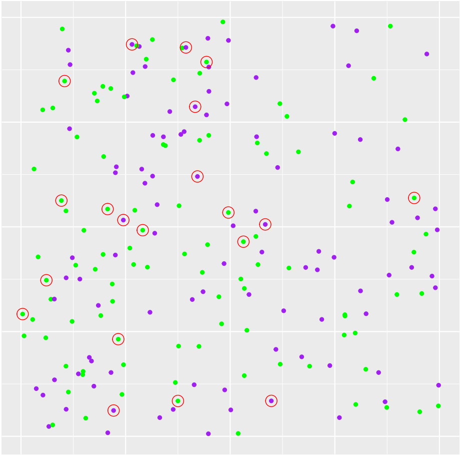


Figure 3.2: Sampling plants. Sampled plants are circled in red

### Step 6: Estimate the population parameter

Estimating a frequency from a sample is simple enough though we can express frequencies in different ways. We'll use a percentage. We found 12 green plants and 8 purple plants in our sample, which means our point estimate of the purple morph frequency is 40%. This is certainly greater than 25%—the value observed in the original population—but it isn't that far off when you consider this is from a small sample.

## 3.4 Now what?

Maybe the purple plants aren't at a selective advantage after all? Or perhaps they are? We'll need to develop a rigorous statistical test of some kind to evaluate our prediction. Before we can do this we need to delve into a few more concepts. The first, sampling error, is needed describe the uncertainty in the purple morph frequency estimate (step 7). We'll look at this topic in the next chapter. After that, we'll be in the position to answer our question about differences among the two populations (step 8).

# Part II

# Statistical Concepts

# Chapter 4

# Sampling error

In the previous chapter, we introduced the idea that point estimates of a population parameter are always imperfect in that they won't exactly match its true value. This uncertainty is unavoidable. This means it is not enough to have estimated something to 'do statistics'—we have to also know about uncertainty of the estimate. We use the machinery of statistics to quantify this uncertainty. Once we have pinned down the uncertainty, we can start to provide meaningful answers to our scientific questions.

We will arrive at the 'getting to the answer step' in the next chapter. First, we have to develop the uncertainty idea by considering sampling error, sampling distributions and standard errors.

## 4.1 Sampling error

For now, let's continue with the plant polymorphism example from the previous chapter. We had taken one sample of 20 plants from our hypothetical population and found that the frequency of purple plants in that sample was 40%. This is a point estimate of purple plant frequency based on a random sample of 20 plants.

What happens if we repeat that same sampling process to arrive at a new, completely independent sample? This is what the population looks like, along with this new sample highlighted with red circles:

This time we ended up sampling 11 green plants and 9 purple plants, so our second estimate of the purple morph frequency is 45%. This estimate is quite different from the first one. It is lower than that seen in the original study population. Our hypothesis that the purple morph is more prevalent in the new population is beginning to look a little shaky now.

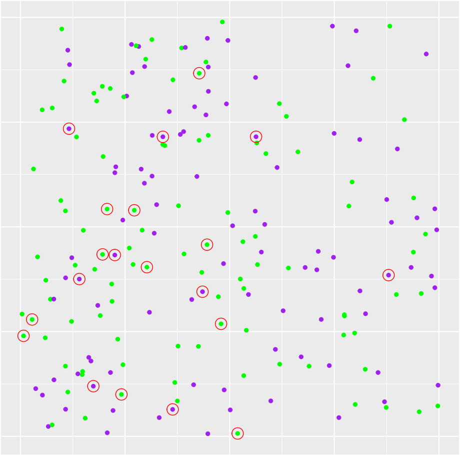


Figure 4.1: Plants sampled on the second occasion

Nothing about the study population changed between the first and second samples. What's more, we used a completely reliable sampling scheme to generate these samples—there was nothing biased or incorrect about the way individuals were sampled. The two different estimates of purple morph frequency arise from nothing more than chance variation in the sampling process.

This variation–present whenever we observe a sample—has a special name. It is called **sampling error**. The word 'error' has nothing to do with mistakes or problems. In statistics, it used to refer to the variability or 'noise' in some process. Sampling error is the reason why we have to use statistics. Every time we estimate something, there's some sampling error lurking in the estimate. If we ignore it, we risk drawing incorrect conclusions about the world.

(By the way, another name for sampling error is 'sampling variation'. We'll use both terms in this book because they are both widely used)

It is important to realise that sampling error is not really a property of any particular sample. It is a consequence of the nature of the variable(s) we're studying and the sampling method used to investigate the population. We can start to understand what all this means by considering something called a sampling distribution.

## 4.2 Sampling distributions

We can use our simple simulation to explore sampling error. This time, rather than taking one sample at a time, we will simulate thousands of independent samples. The number of plants sampled ('n') will always be 20. Every sample will be drawn from the same population, i.e. the population parameter (purple morph frequency) will never change across samples. This means any variation we observe is due to nothing more than sampling error.

Here is a graphical summary of one such repeated simulation exercise:

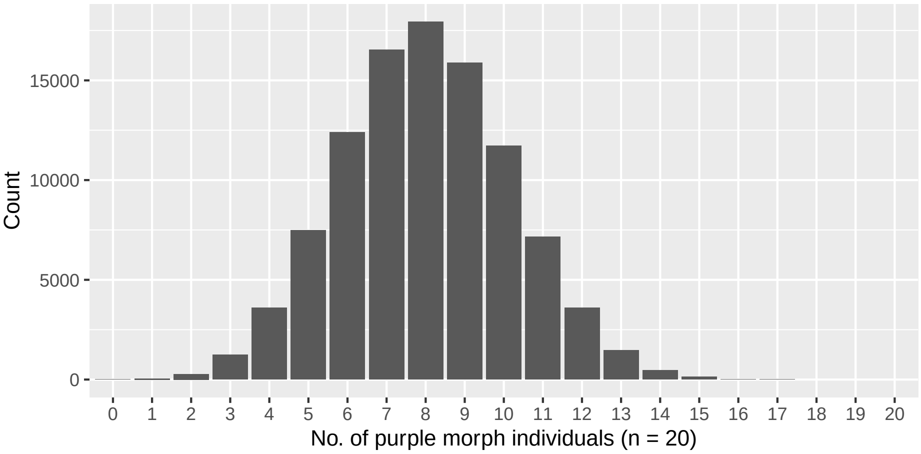


Figure 4.2: Distribution of number of purple morphs sampled (n = 20)

This bar plot summarises the results from taking 100000 independent samples of our population. We took 20 individuals from our hypothetical population and calculated the number of purple morphs seen each time. The bar plot shows the number of times we found 0, 1, 2, 3, … purple individuals, all the way up to the maximum possible (20). Notice that we're summarising the distribution of purple morph counts rather than percentages at this point (either is fine).

This distribution has a special name. It is called a **sampling distribution**. To work this out, we have to postulate values for the population parameters, and we have to know how the population was sampled. This can often be done using mathematical reasoning. However, we used a brute-force simulation to approximate the sampling distribution of purple morph counts that arises when we sample 20 individuals from the hypothetical population.

What does this particular sampling distribution show? It summarises the range of outcomes we can expect to observe when we repeat the same sampling process over and over again. The most common outcome is eight purple morphs, which would yield an estimate of 8/20 = 40% for the purple morph frequency.

Although there is no way to know this without being told, a 40% purple morph frequency is the number used to simulate the original hypothetical data set. It turns out that in this instance, the true value of the parameter we're trying to learn about also happens to be the most common estimate we expect to see under repeated sampling.

Returning to our question—what is the purple morph frequency—we now have our answer. The purple morph frequency is 40%. Ok, but we obviously 'cheated' here because we used information from 1000s of samples to arrive at that answer. In the real world we typically have to work with only one sample. We'll see out how to use that one sample to learn the purple morph frequency later.

What use is the sampling distribution idea? It turns out that sampling distributions are key to 'doing frequentist statistics'. To understand why we need to look a bit more carefully at that sampling distribution we made. Look at its spread along the x-axis. The range of outcomes is roughly 2 to 15, corresponding to estimated purple morph frequencies of 10-75%. This shows that when we sample 20 individuals, the sampling error for a frequency estimate can be quite large.

The sampling distribution we summarised above is only relevant for the case where 20 individuals are sampled and the frequency of purple plants in the population is 40%. If we change either of those two numbers and rerun the repeated sampling process we will end up with a different sampling distribution. That's what was meant by, "It <sampling error> is a consequence of the nature of the variable(s) we're studying and the sampling method used to investigate the population."

Once we know how to construct the sampling distribution for a particular problem, we can start making statements about sampling error to quantify uncertainty, and eventually, answer scientific questions. Fortunately, we don't have to work any of the details out for ourselves—a lot of clever statisticians have already done the hard work for us!

## 4.3 The effect of sample size

One of the most important aspects of a sampling scheme is the **sample size** —often denoted 'n'. This is the number of observations of objects, items, etc, in a sample. To get a sense of why sample size matters, we'll repeat in our sampling exercise using two new sample sizes. First we'll use a sample size of 40 individuals, then we'll take a sample of 80 individuals. As before, we'll take a total of 100000 repeated samples from the population.

Here's a graphical summary of the resulting sampling distributions:

We plotted each of them over the full range of possible outcomes, i.e. the x axis runs from 0-40 and 0-80, respectively, in both plots. This is important because it allows us to compare the spread of each sampling distribution relative to the range of all possible outcomes. What do these two plots tell us about the effect of changing sample size?

The range of outcomes in the first plot (n = 40) is roughly 6 to 26, corresponding to estimated purple morph frequencies in the range of 15-65%. The range of outcomes in the second plot (n = 80) is roughly 16 to 48, corresponding to

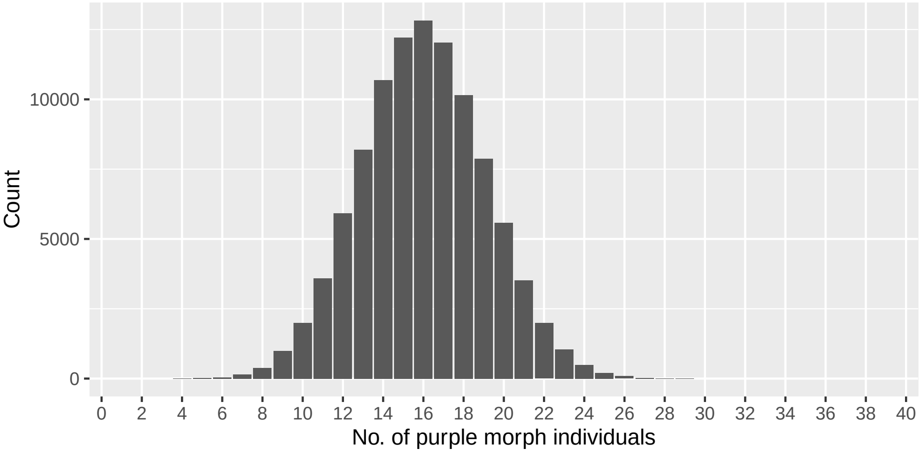


Figure 4.3: Distribution of number of purple morphs sampled (n = 40)

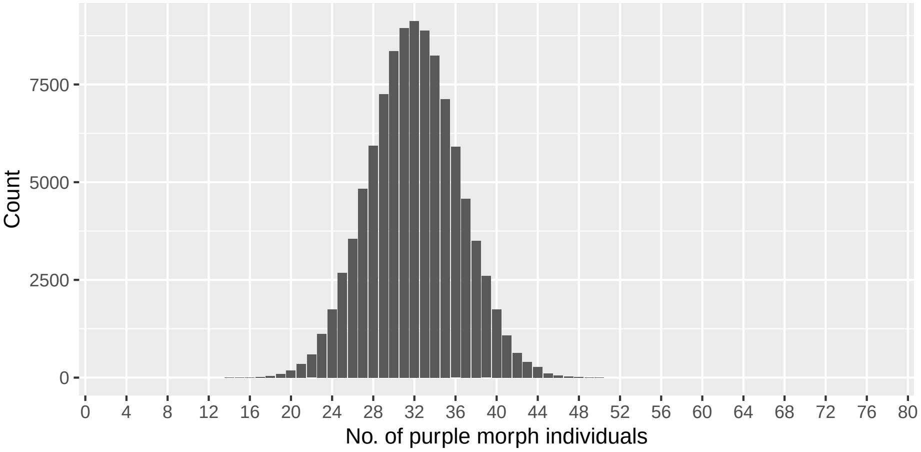


Figure 4.4: Distribution of number of purple morphs sampled (n = 80)

estimated frequencies in the range of 20-60%. The implications of this basic assessment are clear: we reduced sampling error by increasing the sample size. This makes intuitive sense. The composition of a large sample should more closely approximate that of the true population than a small sample.

How much data do we need to collect to accurately estimate a frequency? Here is the approximate sampling distribution of the purple morph frequency estimate when we sample 500 individuals:

```
## Warning: Removed 2 rows containing missing values (geom_bar).
```

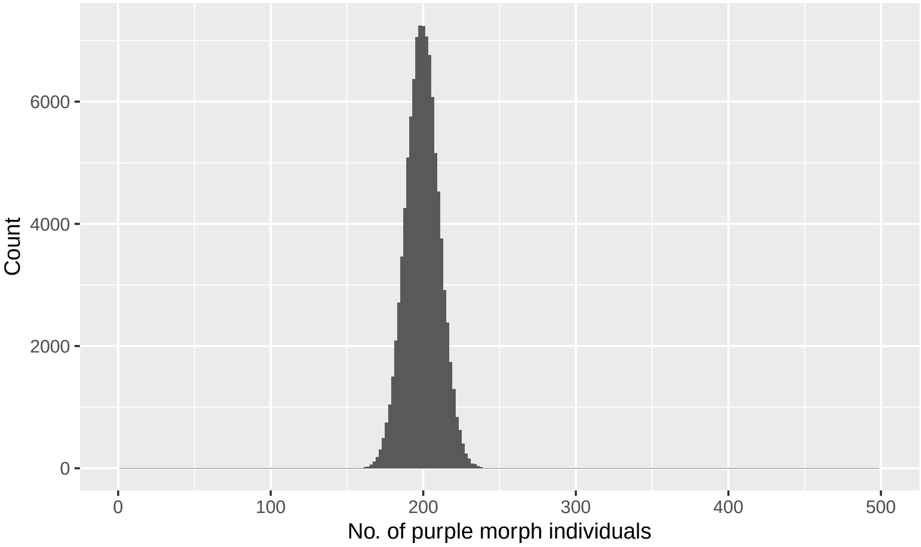


Figure 4.5: Distribution of number of purple morphs sampled (n = 500)

This time the range of outcomes is about 160 to 240, corresponding to purple morph frequencies in the 32-48% range. This is a big improvement over the smaller samples we just considered, but still, even with 500 individuals in a sample, we still expect quite a lot of uncertainty in our estimate. The take-home message is that we need a lot of data to reduce sampling error.

## 4.4 The standard error

We've not been very careful about how we quantify the spread of a sampling distribution so far—we just estimated the range of purple morph counts by looking at histograms. This is fine for an investigation of general patterns, but to make rigorous comparisons, we need a quantitative measure of sampling error variation. One such quantity is called the **standard error**.

The standard error is quite a simple idea, though its definition is a bit of mouthful: the standard error of an estimate is the standard deviation of the estimate's sampling distribution. Don't worry if that is confusing! Everyone finds the definition of standard error confusing when they first encounter it. The key point is that it is a measure of the spread, or dispersion, of a distribution. That distribution is the sampling distribution associated with some kind of estimate.

(It is common to use a shorthand abbreviations such "SE", "S.E.", "se" or "s.e." in place of 'standard error' when referring to the standard error in text.)

It's much easier to make sense of something abstract, like a standard error, by looking at a concrete example. Let's calculate the standard error of the estimated purple morph frequency. It is possible to do this using mathematical reasoning; however, it is much easier to understand what's happening if we use 'brute force' simulations in R.

We'll show you the code to do this but there's no need to commit any of it to memory. It's included for readers who like to see how things work. We start by specifying the number of samples to take (`n_samples`), the sample size to use for each sample (`sample_size`), and the value of the population morph frequency (`purple_prob`). We set up the simulation by creating these variables:

```r
# number of independent samples
n_samples <- 100000
# sample size to use for each sample
sample_size <- 80
# probability a plant will be purple
purple_prob <- 0.4
```

This sets up a simulation to calculate the expected standard error when the purple morph frequency is 40% and the sample size is 80, using $10^5$ samples. The next step requires a bit more knowledge of R and probability theory:

```r
# repeat the sampling/estimation procedure many times
raw_samples <- rbinom(n = n_samples, size = sample_size, prob = purple_prob)
# convert results to %
percent_samples <- 100 * raw_samples / sample_size
```

The details of the R code are not important here. A minimum of A-level statistics is needed to understand what the `rbinom` function is doing, but in a nutshell, it is the workhorse that runs the simulation. Mostly, we're showing the code to demonstrate that seemingly complex things are often easy to do in R.

What matters here is the result. This is stored the result in a vector called `percent_samples`. Here are its first 50 values:

```
##  [1] 33.75 35.00 42.50 41.25 46.25 35.00 42.50 45.00 43.75 40.00 41.25 52.50
## [13] 43.75 38.75 38.75 47.50 42.50 40.00 43.75 42.50 36.25 42.50 43.75 43.75
## [25] 35.00 42.50 35.00 37.50 41.25 40.00 45.00 40.00 41.25 38.75 46.25 45.00
## [37] 31.25 40.00 47.50 42.50 37.50 41.25 31.25 45.00 32.50 36.25 40.00 37.50
## [49] 40.00 45.00
```

These numbers are all part of the sampling distribution of morph frequency estimates. Each value is one possible estimate of the purple morph frequency when we sample 80 individuals from a population where the true value is 40%. There are $10^5$ such estimates stored in `percent_samples`.

How do we then calculate the standard error? This is *defined* as the standard deviation of the sampling distribution that we just constructed with a simulation. So all we need to do now is calculate their standard deviation using the `sd` function:

```
sd(percent_samples)
```

```
## [1] 5.494663
```

The standard error is about 5.5. Why is this number useful? It gives us a means to reason about the variability in a sampling distribution. When a sampling distribution is 'well-behaved', then roughly speaking, about 95% of estimates are expected to lie within a range of about four standard errors.

Not convinced? Look at the second bar plot we produced, where the sample size was 80 and the purple morph frequency was 40%. What is the approximate range of simulated values? How close is this to $4 \times 5.5$? Quite close!

In summary, the standard error quantifies the variability of a sampling distribution. We said in the previous chapter that a point estimate is largely useless without an associated measure of uncertainty. The standard error is one such measure.

## 4.5 What is the point of all this?

Why have we just spent so much time looking at what happens when we take repeated samples from a population with known properties? When we collect data in the real world, we only have the luxury of working with a small number of samples—usually just one, in fact. We also won't know anything much about the population parameter of interest in that situation. This lack of knowledge is the reason for collecting data in the first place!

The short answer is that before we can use frequentist statistics, we need to have a sense of:

- how point estimates behave under repeated sampling (i.e. **sampling distributions**),

- and how **sampling error** and **standard error** relate to sampling distributions.

Once we understand those ideas, we can start to make sense of how frequentist statistical tests work. That's what we'll do in the next few chapters.

# Chapter 5

# Statistical significance and *p*-values

Frequentist statistics works by answering the following question: *what would have happened* if we repeated a data collection exercise many times, *assuming that the population remains the same* each time. This is the idea we used to generate sampling distributions in the previous chapter. The details of this procedure depend on what kind of question we are asking.

What is common to every frequentist technique is that we ultimately have to work out what some sort of sampling distribution looks like. Once we've done that, we can evaluate how likely a particular result is. This naturally leads to the most important ideas in frequentist statistics: *p*-values and statistical significance.

## 5.1 Estimating a sampling distribution

Let's carry on with the plant polymorphism example. Our ultimate goal is to determine if the purple morph frequency is likely to be greater than 25% in the new study population. The suggestion above is that we will need to determine the sampling distribution of purple morph frequency estimates to get to this point.

At first glance, this seems like an impossible task because we only have access to a single sample. The solution to this problem is surprisingly simple: use the one sample to approximate the population somehow, then work out what the sampling distribution of our estimate should look like by 'taking samples' from this approximation.

We'll unpack this idea a bit more before we try it out for real.

### 5.1.1 Overview of bootstrapping

There are many ways to use a sample to approximate the population. One of the simplest is to *pretend the sample is the true population.* We can then draw new samples from this pretend population to get at a sampling distribution. This may sound like 'cheating', but it turns out this is a perfectly valid way to construct approximate sampling distributions.

We'll try to understand how this works by using a physical analogy based on our plant morph example. Imagine that we have written down the colour of every sampled plant on a different piece of paper and then placed these bits of paper into a hat. We then do the following:

1. Pick a piece of paper at random, record its value (purple or green), put the paper back into the hat, and shake the hat about to mix up the bits of paper.

(The shaking here is meant to ensure that each piece of paper has an equal chance of being picked.)

2. Now pick another piece of paper (we might get the same one), record its value, put that one back into the hat, and shake everything up again.
3. Repeat this process until we have a recorded new sample of colours that is the same size as the real sample. We have now have generated a 'new sample' from the original one.

(This process is called 'sampling with replacement'. Each artificial sample is called a 'bootstrapped sample'.)

4. For each bootstrapped sample, calculate whatever quantity is of interest. In our example, this is the proportion of purple plants sampled.
5. Repeat steps 1-4 until we have generated a large number of bootstrapped samples. About 10000 is sufficient for most problems.

Although it seems like cheating, this procedure really does approximate the sampling distribution of the purple plant frequency. It is called **bootstrapping** (or 'the bootstrap').

The bootstrap is quite a sophisticated technique developed by statistician Bradley Efron. We're not going to use it to solve real data analysis problems and there's no need to learn how to do it. We're introducing bootstrapping because it provides an intuitive way to understand how frequentist methodology works without getting stuck into any challenging mathematics.

### 5.1.2 Doing it for real

No one carries out bootstrapping using bits of paper and hat. Generating 10000 bootstrapped samples via such a method would obviously take a very long time!

Luckily, computers are very good at carrying out repetitive tasks quickly. We're going to work through how to implement the bootstrap for our hypothetical example.

> The best way to understand what follows is by actually running the example code line by line. You are encouraged to do this, but it is certainly not essential. You can gain an adequate understanding bootstrapping by simply reading through the example.

### Set up and read the data

Assume we had sampled 250 individuals from the new plant population. A data set representing this situation is stored in the Comma Separated Value (CSV) file called 'MORPH_DATA.CSV'. Let's assume these data have been read into a tibble called `morph_data`, e.g.

```
morph_data <- read_csv(file = "MORPH_DATA.CSV")
```

The data set looks like this:

```
morph_data
```

```
## # A tibble: 250 x 2
##    Colour Weight
##    <chr>   <dbl>
##  1 Green    714.
##  2 Green    693.
##  3 Green    556.
##  4 Purple   654.
##  5 Green    673.
##  6 Green    661.
##  7 Green    445.
##  8 Green    482.
##  9 Green    648.
## 10 Green    858.
## # ... with 240 more rows
```

This contains 250 rows and two variables: `Colour` and `Weight`. `Colour` is a categorical variable and `Weight` is a numeric variable. The `Colour` variable contains the colour of each plant in the sample. What is that `Weight` variable all about? Actually… we don't need this now but we will use it in the next chapter.

### Running the bootstrap

Now that we understand the data, we're ready to implement bootstrapping. We are going to use a few programming tricks that beginners may not have come across before. We'll explain these as we go, but there's no need to learn them. Focus on the 'why'—the logic of what we're doing—rather than the 'how'.

We want to construct an approximate sampling distribution for the frequency of purple morphs. That means the variable that matters is `Colour`. Rather than work with this inside the `morph_data` data frame, we're going to pull it out using the `$` operator and assign it a name (`plant_morphs`):

```
# pull out the 'Colour' variable
plant_morphs <- morph_data$Colour
```

Next, we'll take a quick look at the values of `plant_morphs`:

```
# what is the set of values 'plant_morphs' can take?
unique(plant_morphs)
```

```
## [1] "Green"  "Purple"
```

```
# show the first 100 values
head(plant_morphs, 50)
```

```
##  [1] "Green"  "Green"  "Green"  "Purple" "Green"  "Green"  "Green"  "Green"
##  [9] "Green"  "Green"  "Green"  "Green"  "Green"  "Purple" "Green"  "Green"
## [17] "Purple" "Purple" "Green"  "Green"  "Green"  "Green"  "Green"  "Purple"
## [25] "Green"  "Green"  "Green"  "Green"  "Purple" "Purple" "Green"  "Green"
## [33] "Green"  "Purple" "Purple" "Green"  "Green"  "Green"  "Green"  "Purple"
## [41] "Green"  "Purple" "Green"  "Green"  "Purple" "Purple" "Green"  "Green"
## [49] "Green"  "Green"
```

The last line printed out the first 50 values of `plant_morphs`. This shows that `plant_morphs` is a simple character vector with two categories describing the plant colour morph information.

Next, we calculate and store the sample size (`samp_size`) and the point estimate of purple morph frequency (`mean_point_est`) from the sample:

```
# get the sample size form the length of 'plant_morphs'
samp_size <- length(plant_morphs)
samp_size
```

```
## [1] 250
```

```
# estimate the frequency of purple plants as a %
mean_point_est <- 100 * sum(plant_morphs == "Purple") / samp_size
mean_point_est
```

```
## [1] 30.8
```

The code in the point estimate calculation says, “add up all the cases where `plant_morphs` is equal to ‘purple’, divide that total by the sample size to get the proportion of purple plants in the sample, then multiply the proprtion by 100 to turn it into a percentage.” We find that 30.8% of plants were purple among our sample of 250 plants.

Now we’re ready to start bootstrapping. For convenience, we’ll store the number of bootstrap samples we want in `n_samp` (i.e. 10000 in this case):

```
# number of bootstrapped samples we want
n_samp <- 10000
```

Next we need to work out how to resample the values in the `plant_morphs` vector. The `sample` function can do this for us:

```
# resample the plant colours
samp <- sample(plant_morphs, replace = TRUE)
# show the first 50 values of the bootstrapped sample
head(samp, 50)
```

```
##  [1] "Green"  "Purple" "Purple" "Green"  "Green"  "Purple" "Green"  "Purple"
##  [9] "Green"  "Green"  "Green"  "Green"  "Purple" "Purple" "Green"  "Green"
## [17] "Purple" "Purple" "Green"  "Green"  "Green"  "Green"  "Green"  "Green"
## [25] "Green"  "Green"  "Purple" "Purple" "Purple" "Purple" "Green"  "Green"
## [33] "Green"  "Purple" "Green"  "Green"  "Purple" "Green"  "Purple" "Green"
## [41] "Green"  "Green"  "Purple" "Purple" "Green"  "Purple" "Green"  "Purple"
## [49] "Purple" "Purple"
```

The `replace = TRUE` ensures that we sample with replacement—this is the ‘putting the bits of paper back into the hat’ part of the process.

The new `samp` variable now contains exactly one bootstrapped sample of the 250 plants in the real sample. We only need to extract one number from this—the frequency of purple morphs:

```
# calculate the purple morph frequency in the bootstrapped sample
first_bs_freq <- 100 * sum(samp == "Purple") / samp_size
```

That’s one bootstrapped value of the purple morph frequency. Fine, but we need $10^4$ values. We don’t want to have to keep doing this over an over ‘by hand’—

making `second_bs_freq`, `third_bs_freq`, and so on—because this would be very slow and boring to do.

As we said earlier, computers are very good at carrying out repetitive tasks. The `replicate` function can replicate any R code many times and return the set of results. Here is some R code that repeats what we just did `n_samp` times, storing the resulting bootstrapped values of purple morph frequency in a numeric vector called `boot_out`:

```
boot_out <- replicate(n_samp, {
  samp <- sample(plant_morphs, replace = TRUE)
  100 * sum(samp == "Purple") / samp_size
})
```

The `boot_out` vector now contains a bootstrapped sample of frequency estimates. Here are the first 30 values rounded to 1 decimal place:

```
head(boot_out, 30) %>% round(1)
```

```
##  [1] 31.6 23.6 27.6 28.4 32.0 28.0 34.0 28.0 32.8 33.2 34.0 25.2 34.0 30.0 28.8
## [16] 31.6 30.0 30.0 29.2 28.0 31.2 27.6 29.6 30.4 32.8 29.6 30.4 37.2 34.4 32.0
```

(We used the pipe `%>%` to make a code a bit more readable—remember, this won't work unless the **dplyr** package is loaded.)

### Making sense of the bootstrapped sample

What has all this achieved? The numbers in `boot_out` represent the values of purple morph frequency we can expect to find if we repeated the data collection exercise many times, assuming that the purple morph frequency is equal to that of the actual sample. This is a bootstrapped sampling distribution!

We can use this bootstrapped sampling distribution in a number of ways. Let's plot it first get a sense of what it looks like. A histogram is OK here because we have a reasonably large number of possible cases:

What the most common values in our bootstrapped sample? The centre of the distribution looks to be round about 30%. We can be a bit more precise by calculating its mean:

```
mean(boot_out) %>% round(1)
```

```
## [1] 30.8
```

This is essentially the same as the point estimate of purple morph frequency from the true sample. In fact, it is guaranteed to be the same if we construct

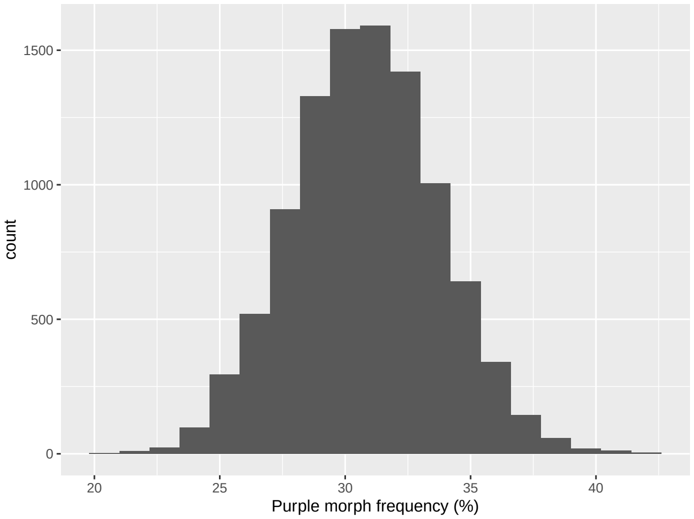


Figure 5.1: Bootstrapped sampling distribution of purple morph frequency

a large enough sample because we're just resampling the data used to estimate that frequency.

A more useful quantity is the standard error (SE) of our estimate. This is *defined* as being the standard deviation of the sampling distribution. We can calculate that by applying the `sd` function to the bootstrapped sample:

```
sd(boot_out) %>% round(1)
```

```
## [1] 2.9
```

The standard error is a very handy quantity. Remember, it is a measure of the precision of an estimate. For example, a large SE implies that our sample size was too small to reliably estimate the population mean; a small SE means we have a reliable estimate. Once we have the point estimate of a population parameter and its standard error we're able to start asking questions like, "is the true value likely to be different from 25%."

It is standard practice to include the standard error when we report a point estimate of some quantity. Whenever we report a point estimate, we really should also report its standard error, like this:

> The frequency of purple morph plants (n = 250) was 30.8% (s.e. ± 2.9).

Notice we also report the sample size. More on that later in the book.

## 5.2 Statistical significance

Now back to the question that motivated all the work in the last few chapters. Is the purple morph frequency greater than 25% in the new study population? The first thing to realise is that we can never answer a question like this definitively from a sample. We have to carry out some kind of probabilistic assessment instead. To make this assessment, we're going to do something that looks odd at first glance.

> ⚠ **Don't panic!**
>
> The ideas in the next section are very abstract, and you probably won't 'get' them straight away. That is fine—these concepts take time to absorb and understand.

### Carrying out the assessment

We need to make two assumptions to arrive at our probabilistic assessment of whether or not the purple morph frequency is greater than 25%:

1. Assume the true value of the purple morph frequency in our new study population is 25%, i.e. we'll assume the population parameter of interest is the same as that of the original population that motivated this work. In effect, we're pretending there is no difference between the populations.

2. Assume that the form of sampling distribution that we generated would have been the same if the 'equal population' hypothesis were true. That is, the expected 'shape' of the sampling distribution would not change if the purple morph frequency was 25%.

That first assumption is an example of a **null hypothesis**. The null hypothesis is an hypothesis of 'no effect' or 'no difference'. We're going to revisit this idea many times in future chapters.

The second assumption is necessary for the reasoning below to work. This can be shown to be a pretty reasonable assumption in many situations (we don't want to get lost in the details though, so this will have to remain a "trust us" situation).

Now we ask a question: if the purple morph frequency in the population really is 25%, what would its corresponding sampling distribution look like? This is called the **null distribution**—the distribution expected under the null hypothesis.

If the second assumption is valid, we can actually construct the null distribution from our bootstrapped distribution as follows:

```
null_dist <- boot_out - mean(boot_out) + 25
```

All we did here was shift the bootstrapped sampling distribution along until the mean is at 25%. Here's what that null distribution looks like, along with the original observed estimate of the purple morph frequency:

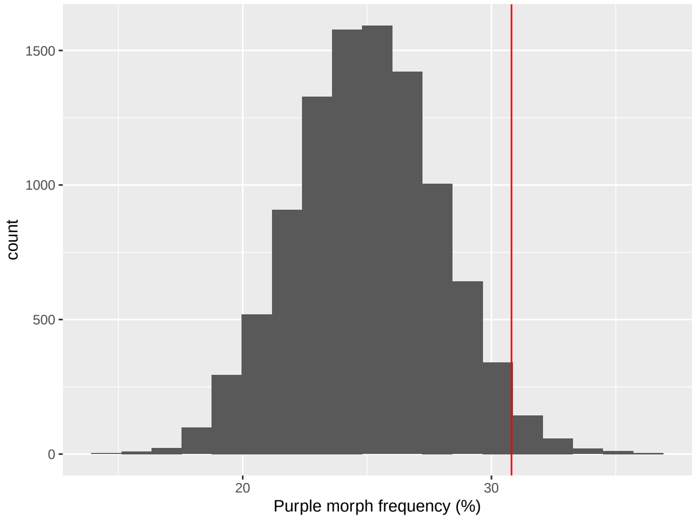


Figure 5.2: Sampling distribution of purple morph frequency under the null hypothesis

The red line shows where the point estimate from the true sample lies. What does this tells us? It looks like the observed purple morph frequency would be quite unlikely to have arisen through sampling variation if the population frequency really was 25%. We can say this because the observed frequency (red line) lies at the end of one 'tail' of the sampling distribution over on the right.

We need to be able to make a more precise statement than this though. Instead of 'eyeballing' the distribution, we can quantify how often the values of the bootstrapped null distribution ended up greater than the observed estimate:

```
p_value <- sum(null_dist > mean_point_est) / n_samp
p_value
```

```
## [1] 0.0238
```

This number (generally denoted '*p*') is called a ***p*-value**.

### Interpreting the p-value

What are we supposed to do with the finding $p = 0.0238$? This is the probability of obtaining a result equal to, or 'more extreme', than that which was actually observed, *assuming that the hypothesis under consideration (the null hypothesis) is true.* The null hypothesis is an hypothesis of no effect (or no difference), and so a low $p$-value can be interpreted as evidence for an effect being present.

It's worth reading that a few times because it is not at all intuitive…

In our example, it appears that the purple morph frequency we observed is fairly unlikely to occur if its frequency in the new population really was 25%. In biological terms, we take the low $p$-value as evidence for a difference in purple morph frequency among the populations, i.e. the data support the prediction that the purple morph is present at a frequency greater than 25% in the new study population.

One important question remains: How small does a $p$-value have to be before we are happy to conclude that the effect we're interested in is probably present? In practice, we do this by applying a threshold, called a **significance level**. If the $p$-value is less than the chosen significance level, we say the result is said to be **statistically significant**. Very often, we use a significance level of $p < 0.05$ (5%).

Why do we use a significance level of $p < 0.05$? The short answer is that this is just a convention. Nothing more. There is nothing special about the 5% threshold other than the fact that it's the one most often used. Statistical significance has nothing to do with biological significance. Unfortunately, many people are very uncritical about their use of this arbitrary threshold, to the extent that it can be very hard to publish a scientific study if it doesn't contain 'statistically significant' results.

## 5.3 Concluding remarks

We just carried out a type of statistical test called a **significance test**. It was a bit convoluted reasoning, but the chain of reasoning we just employed underlies all the significance tests we use in this book. The precise details of how to construct such tests will vary from one problem to the next, but ultimately, when using frequentist ideas we always…

1. assume that there is actually no 'effect' (the **null hypothesis**), where an effect is expressed in terms of one or more population parameters,

2. construct the corresponding **null distribution** of the estimated parameter by working out what would happen if we were to take frequent samples in the 'no effect' situation,

(This is why the word ‘frequentist’ is used to describe this flavour of statistics.)

3. then compare the estimated population parameter to the null distribution to arrive at a ***p*-value**, which evaluates how frequently the result, or a more extreme result, would be observed under the hypothesis of no effect.

We used the bootstrap to operationalise that process for our example (i.e. make it real). Bootstrapping is certainly a useful tool, but it is also an advanced technique that can be difficult to apply in many settings. We won’t use it anymore—the bootstrap was introduced here to demonstrate how frequentist reasoning works.

We will focus on simple, ‘off-the-shelf’ statistical tools in this book. The good news is we don’t need to understand the low-level details to use these tools effectively. As long as we can identify the null hypothesis and understand how to interpret the associated *p*-values we should be in a good position to apply them. These two ideas—null hypotheses and *p*-values—are so important, we’re going to consider them in much greater detail over the next two chapters.

# Chapter 6

# Statistical comparisons

## 6.1 Making comparisons

Scientific inquiry often requires us to evaluate predictions about differences between populations. The simplest such case involves just two populations, e.g.

‘Does enzyme activity differ among control and drug-treated cell lines?’

‘Do maize plants photosynthesise at different rates at 25°C and 20°C?’

‘Do purple and green plants differ in their biomass?’

In this setting, we’re evaluating whether or not two *statistical populations* are different in some way. To do this, we have to step through the same kind of process discussed in the last few chapters.

This chapter demonstrates how to compare two populations by employing ideas like null hypotheses and *p*-values. The goal is not really to learn how to compare populations—instead, we want to continue developing our understanding of significance tests and evaluate predictions.

As we do this, we’re going to introduce a new component of the frequentist machine called a ‘test statistic’. This is an important idea—it will crop up every time we introduce a new type of statistical test.

### 6.1.1 A new example

Let’s continue with the familiar purple and green plants example. This time, instead of working with morph frequencies, we’ll consider plant size. We introduce this by stepping through the process introduced in the learning from data chapter:

**Question, hypothesis and prediction:** We think the purple and green morph plants might be different in some way. Specifically, we want to address the question: Why have the purple morphs increased in frequency in the new population? Our working hypothesis is that purple plants are fitter than green plants. Since fitness is closely related to seed production, and seed production is typically positively correlated with plant size, we predict that purple morphs should be larger in the new population.

**Statistical populations:** What statistical populations are we interested in? This time, we will conceive of each morph as its own statistical population. That means there are now two distinct statistical populations in play: a purple plant population and a green plant population. This change of focus from our earlier analysis where we considered all the plants as one population is perfectly valid. Statistical populations are not really fixed things—they are defined by the problem we're tackling.

**Variables:** Which variable should we study? An obvious way to address a prediction of size differences of plants is to measure the weight of individuals of each morph in our samples. To be more precise, 'dry weight biomass' is the standard, reliable measure of how 'big' a plant is. Dry weight mass is a numeric variable measured on a ratio scale (zero means 'nothing').

**Population parameters:** We predict that purple morphs are larger than green morphs. What do we mean by that? We probably don't mean that every purple plant is bigger than every green plant. Instead, we need to know if purple plants are *generally* bigger than green ones. This is a statement about 'central tendency'—we want to evaluate whether purple plants are larger than green plants, *on average*. The population parameters of interest are, therefore, the *mean* dry weights of each morph.

**Gather samples:** The next step is to gather appropriate samples. Since this is a made-up example, we'll cut to the chase. We have already seen the samples we're going to use. When we read in the 'MORPH_DATA.CSV' in the previous chapter, we found a numeric variable called `Weight`. This contains our dry weight biomass information. The categorical `Colour` variable analysed in the last chapter tells us which kind of colour morph each observation corresponds to.

Now we're ready to tackle the remaining steps in the 'learning from data' process: estimating the population parameter(s), estimating uncertainty, and answering the question. We work through each of these steps in detail next.

> Once again, the best way to understand what follows is by working through the example, though you can also make good progress through careful reading. In any case, the code again assumes that 'MORPH_DATA.CSV'

has been read into a tibble called `morph_data`.

### 6.1.2 Examine the data

The first step is to calculate point estimates of the mean dry weight of each morph. These are our 'best guess' of the population means. It can also be useful to know something about the sample size and variability of dry weight in each sample. We can summarise variability using the standard deviation (`sd`) of each sample. Here is how to do this with **dplyr**:

```
# using morph data...
morph_data %>%
  # ...group the data by colour morph category
  group_by(Colour) %>%
  # ... calculate the mean, sd and sample size of weight in each category
  summarise(mean = mean(Weight),
            sd = sd(Weight),
            samp_size = n())
```

```
## # A tibble: 2 x 4
##   Colour  mean    sd samp_size
##   <chr>  <dbl> <dbl>     <int>
## 1 Green   708.  150.       173
## 2 Purple  767.  156.        77
```

This shows the mean dry weight of the purple morph is indeed greater than that of the green morph. Maybe our hypothesis is true. The standard deviation estimates indicate that the dry weight of purple morphs is also a bit more variable than the green morphs.

These numbers are point estimates taken from limited samples. If we sampled the populations again, sampling variation would lead to different estimates. We're not yet in a position to conclude that purple morphs are bigger than green morphs because we don't know if the difference is just down to 'luck'.

We should also visualise the sample distributions of colour morph weights. We could do this in a variety of ways but since we only have two samples we may as well use a fairly information-rich summary. Histograms are a good choice when we have a reasonable amount of data:

```
ggplot(morph_data, aes(x = Weight)) +
  geom_histogram(binwidth = 50) +
  facet_wrap(~Colour, ncol = 1) # <- separate histogram for each colour morph!
```

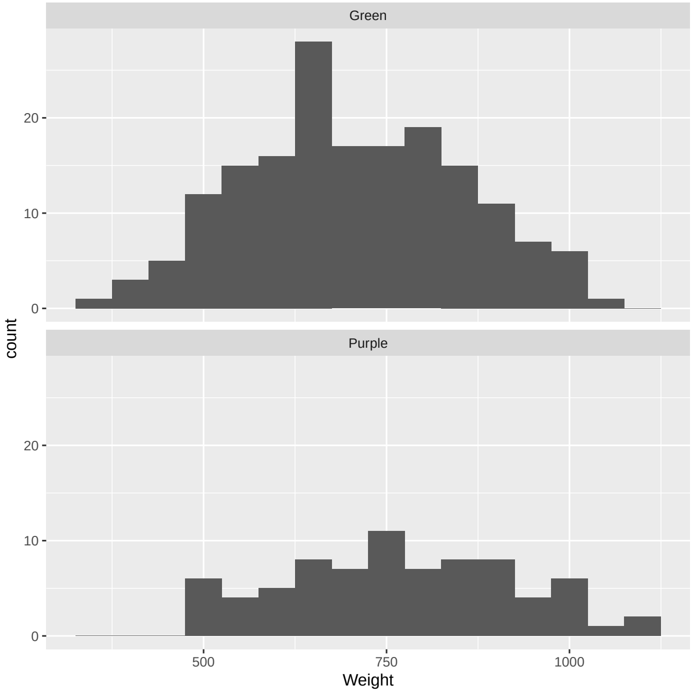


Figure 6.1: Size distributions of purple and green morph samples

What does this plot tell us? We're interested in the degree of similarity between the two samples. It looks like purple morph individuals tend to have higher dry weights than green morphs. We already knew this, of course. What the plot confirms is that the difference is probably not due to a few extreme cases ('outliers').

It looks like the histograms confirm there might be a general difference in the size of the two morphs. However, there is also a lot of overlap between the two dry weight distributions. Maybe the difference between the sample means is simply a consequence of sampling variation? We need to construct a statistical test to address this question.

## 6.2 Constructing a test

We will develop a frequentist test to evaluate whether the two population means are likely to be different. Regardless of the precise details, this kind of problem involves a four-step process:

1. First, we *assume* there is no difference between the population means. That is, we hypothesise that all the data are sampled from a pair of populations with the same population mean. To put it another way, we pretend there is really only one population. We know about this trick. It's a statement of the **null hypothesis**.
2. Next, we use information in the real samples to work out what would happen if we were to repeatedly take many new samples in this hypothetical situation of 'no difference'. We summarise this by calculating the null distribution of a suitable **test statistic** (<– this is something new).
3. We then ask, "if there were no difference between the two groups, what is the probability that we would observe a difference that is the same as, or more extreme than, the one we observed in the true sample?" We know about this number too. It's a ***p*-value**.
4. If the observed difference is sufficiently improbable, we conclude that we have found a **statistically significant** result. A statistically significant result is one that is inconsistent with the hypothesis of no difference.

This chain of logic is very similar to that used to construct the statistical test in the previous chapter. The main elaboration here is the introduction of the test statistic. When dealing with comparisons involving more than one parameter we need to work with a single numeric summary of 'the effect of interest', however defined. Its purpose is to summarise the effect as a single number so that we can construct a statistical test. That's why it is called the 'test statistic'.

There are many different kinds of test statistic and many different ways to realise the above process. We're going to use something called a **permutation test** here. We're not using this because we want to learn how to apply it. Instead, just as with the bootstrap, the permutation test offers a simple way to see how frequentist tests work without getting lost in mathematical detail.

### 6.2.1 Permutation tests

A hypothesis of 'no difference' between the mean dry weights of purple and green morphs implies the following: the labels 'purple' and 'green' are meaningless because the two morphs are effectively sampled from the same population. These labels don't carry any real information, meaning they may as well have been randomly assigned to each individual.

This suggests we can evaluate the statistical significance of the observed difference as follows:

1. Make a copy of the original purple and green dry weights sample, but do so by randomly assigning the labels 'purple' and 'green' to the new copy of the data. Do this in such a way that the original sample sizes are preserved. The process of assigning random labels is called **permutation**.

(We have to preserve the original sample sizes because we want to mimic the sampling process that we used, i.e. we want to hold everything constant apart from the labelling of individuals.)

2. Repeat the permutation scheme many times until we have a large number of artificial samples; 1000-10000 randomly permuted samples may be sufficient.

3. For each permuted sample, calculate whatever test statistic captures the relevant information. In our example, this is the *difference* between the mean dry weight of purple and green morphs in each permuted sample.

(It doesn't matter which way round we calculate the difference.)

4. Compare the observed test statistic from the real sample to the distribution of sample statistics from the randomly permuted samples.

This scheme is called a permutation test because it involves random permutation of the group labels. Why is this useful? *Each unique random permutation yields an observation from the null distribution of the difference among sample means, under the assumption that this difference is zero in the population.* We can use this to assess whether or not the observed difference is consistent with the hypothesis of no difference by looking at where it lies relative to this null distribution.

### 6.2.2 Carrying out a permutation test

It's easy to implement a permutation test in R. We won't show the code this time because it uses quite a few tricks that won't be needed again. It is worth having a quick look at the permuted samples, though. The first 40 values are:

*Sample 1-*

```
##     Green    Purple    Purple     Green     Green    Purple     Green    Purple
##  714.3592  693.4924  556.2063  653.6619  672.5207  661.0097  445.1001  481.5068
##     Green     Green     Green     Green    Purple    Purple     Green     Green
##  647.6679  858.4820  567.4104  597.1629  718.7132  539.8551  753.0170  807.7700
##    Purple    Purple     Green     Green     Green     Green     Green     Green
## 1085.7036  926.4972  617.1209  632.5897  859.7013  815.4634  666.7693  573.5907
##     Green     Green    Purple     Green    Purple    Purple     Green     Green
##  694.2877  836.6883  665.8489  617.7895  590.5936  775.8980  686.6790  813.4272
##     Green     Green     Green    Purple     Green     Green     Green    Purple
##  506.3904  566.9971  629.5894  878.2477  823.1128  542.7877  507.7345  786.2809
```

*Sample 2-*

```
##    Purple     Green     Green     Green     Green     Green    Purple    Purple
##  714.3592  693.4924  556.2063  653.6619  672.5207  661.0097  445.1001  481.5068
##    Purple     Green     Green    Purple    Purple     Green    Purple     Green
##  647.6679  858.4820  567.4104  597.1629  718.7132  539.8551  753.0170  807.7700
##    Purple     Green     Green     Green    Purple    Purple     Green    Purple
## 1085.7036  926.4972  617.1209  632.5897  859.7013  815.4634  666.7693  573.5907
##    Purple     Green     Green    Purple     Green     Green     Green     Green
##  694.2877  836.6883  665.8489  617.7895  590.5936  775.8980  686.6790  813.4272
##    Purple     Green     Green     Green    Purple     Green     Green    Purple
##  506.3904  566.9971  629.5894  878.2477  823.1128  542.7877  507.7345  786.2809
```

The data from each permutation are stored as numeric vectors, where each element of the vector is named according to the morph type it corresponds to—these are 'the labels' referred to above. The set of numbers does not vary among permuted samples. The *only* difference between them is the labelling which has been randomly assigned in each permuted sample.

The difference between the mean dry weights in the first permutation is 4.0700017, while in the second sample, the difference is -19.9656492. What matters for our test is the distribution of these differences over the complete set of permutations. This is the approximation to the sampling distribution of the difference between means under the null hypothesis—i.e. it's our null distribution.

We can visualise this null distribution by making a histogram that summarises the 2500 mean differences from the permuted samples:

Notice that the distribution is centred at zero. This makes perfect sense—if we

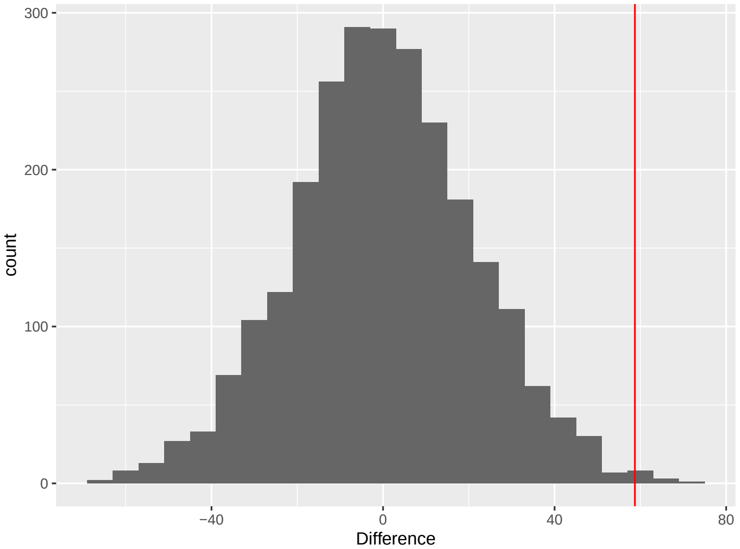


Figure 6.2: Difference between means of permuted samples

take a set of numbers and randomly allocate them to groups, on average, we expect the difference between the means of the groups to be zero.

The red line shows the estimated difference between the mean purple and green morph dry weights *in the real sample*. This is the test statistic. The key thing to notice here is the location of the test statistic within the null distribution.

In this example, the estimated difference lies at the end of the right 'tail' of the null distribution. What does that tell us? It suggests the observed difference (the red line) is unlikely to arise through pure sampling variation if the two population means were identical.

So… it looks like the data are inconsistent with the null hypothesis of no difference. What we need is a number to quantify this inconsistency—the all-important $p$-value.

## Calculating the $p$-value

We use the null distribution to quantify the probability of ending up with the observed difference under the null hypothesis. Only 10 out of the 2500 permutations ended up being equal to, or 'more extreme' (more positive) than, the observed difference. This means the probability of finding a difference in the means equal to, or more positive than, the observed difference is about, $p = 10/2500 = 0.004$. This is the $p$-value for our significance test.

Let's run through the interpretation of that $p$-value. Here's the general chain of logic again… The $p$-value is the probability of obtaining a test statistic equal to, or 'more extreme' than, the estimated value, assuming the null hypothesis is true. The null hypothesis is one of no effect, so a small $p$-value can be interpreted as evidence for the effect (e.g. a difference between means) being present. That's a weird way to seek evidence for the presence of an effect, but that's how frequentist statistics works.

How low does the $p$-value actually have to be before we decide we have 'enough evidence'? Less than 0.01? Less than 0.001? Any threshold is somewhat arbitrary to be honest, but in biology, a significance threshold of $p < 0.05$ is conventionally used. If we find $p < 0.05$, then we conclude that we found a 'statistically significant' effect at the 5% level.

Here's how this logic applies to our example… The permutation test assumed there was no difference between the purple and green morphs, so the low $p$-value indicates that the estimated difference between the mean dry weight of purple and green morphs was unlikely to have occurred by chance *if* there is really no difference at the population level. We interpret the low $p$-value as evidence for the existence of a difference in mean dry weight among the populations of purple and green morphs. Since $p = 0.004$, we say we found a statistically significant difference at the 5% level.

Here's how we might summarise our analysis in in a written report:

> The mean dry weight biomass of purple plants (77) was significantly greater than that of green plants (173) (one-tailed permutation test, p<0.05).

We report the sample sizes used, the type of test employed, and the significance threshold we passed (not the raw $p$-value).

### Directional tests

The test we just did is a 'one-tailed' test. It's called a 'one-tailed' because we only looked at one end of the null distribution. This kind of test is appropriate for evaluating directional predictions (e.g. purple > green). If, instead of testing whether purple plants were larger than green plants, we just want to know if they were different in some way (i.e. in either direction), we should use a 'two-tailed' test. These work by looking at both ends of the null distribution. We won't do this here though—the one- vs two-tailed distinction is discussed in the one-tailed vs. two-tailed tests supplementary chapter.

## 6.3 What have we learned?

Our goal was not really to learn how permutation tests work. Just as with bootstrapping in the previous chapter—we just used it to demonstrate the logic of how frequentist statistics work. In this instance, we wanted to see how to evaluate a difference between two groups. The basic ideas are no different from those introduced in the previous chapter…

1. define what constitutes an 'effect' (e.g. a difference between means), then assume that there is 'no effect'—i.e. define the **null hypothesis**,
2. select an appropriate **test statistic** that can distinguish between the presence of an 'effect' and 'no effect',

(In practice, each kind of statistical test uses a standard test statistic. We don't have to pick these ourselves.)

3. construct the corresponding **null distribution** of the test statistic by working out what would happen if we were to take frequent samples in the 'no effect' situation,
4. and finally, use the null distribution and the observed test statistic to calculate a ___*p*-value___that quantifies how frequently the observed difference, or a more extreme difference, would be observed under the hypothesis of no effect.

We only actually introduced one new idea in this chapter. When evaluating differences among populations, we need to work with a single number that can distinguish between 'effect' and 'no effect'. This is called the test statistic. Sometimes this can be expressed in terms of familiar quantities like means (we just used a difference between means above). However, this isn't always the case. For example, we can use something called an $F$-ratio to evaluate the statistical significance of differences among more than two means. We'll get to these kinds of tests later. For now, our task is to explore simple 'parametric tests'. These kinds of tests allow us to ask questions without resorting to things bootstrapping or permutation tests.

# Chapter 7

# Hypotheses and *p*-values

## 7.1 A few words about the null hypothesis

When using frequentist statistics, we are always asking what would happen if we continually sampled from a population *where the effect we are interested in is not present.* This idea of an hypothetical 'no effect' situation is so important that it has a special name; it is called **the null hypothesis**. Every kind of statistical test (in this book at least) works by first specifying a particular null hypothesis. It is hard to understand a statistical test's result if we don't know its null hypothesis.

### 7.1.1 Hypotheses and null hypotheses

When discussing the scientific process, we said that an hypothesis is a statement of a proposed process or mechanism responsible for an observed pattern or effect. We have also seen that in statistics, we encounter 'hypothesis' used differently. In particular, we often see **null hypotheses** (often written in statistics books as $H_0$).

The null hypothesis is a statement of what we would expect to see if there is no effect of the factor we are looking at (e.g., plant morphology) on the variable that we measure (e.g., dry weight biomass). So in the second plant morph example, our null hypothesis was *There is no difference in mean biomass of purple and green plants.* All frequentist statistical tests work by specifying a null hypothesis and then evaluating the observed data to see if they deviate from the null hypothesis in a way that is inconsistent with sampling variation. This may seem like a rather odd approach, but this is how frequentist tests work.

It is important to be aware of what a null hypothesis is, and what it is used for, so that we can interpret the results of statistical tests. However, in a

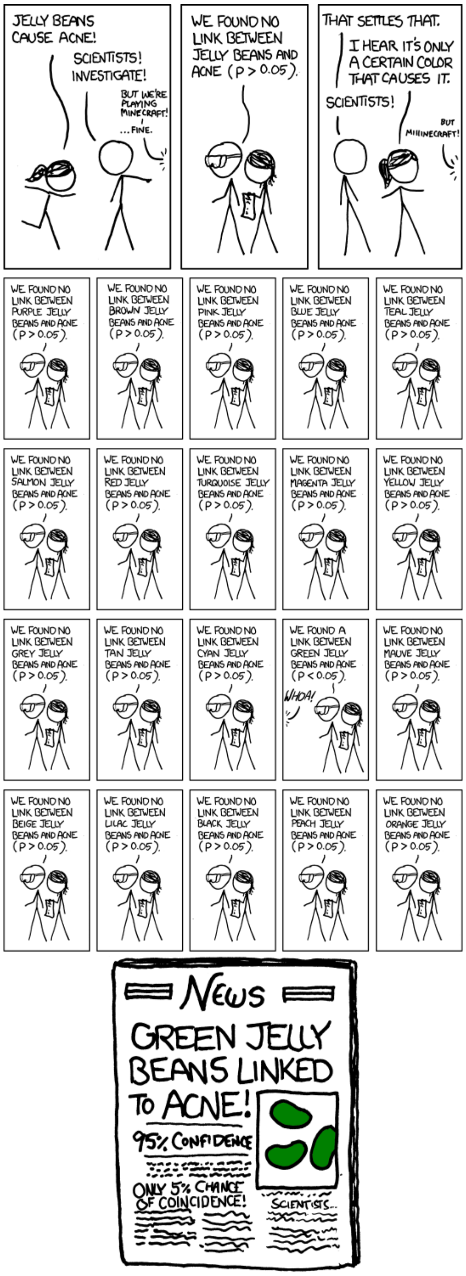

JELLY BEANS CAUSE ACNE!
SCIENTISTS! INVESTIGATE!
BUT WE'RE PLAYING MINECRAFT!
... FINE.
WE FOUND NO LINK BETWEEN JELLY BEANS AND ACNE (P > 0.05).
THAT SETTLES THAT.
I HEAR IT'S ONLY A CERTAIN COLOR THAT CAUSES IT.
SCIENTISTS!
BUT MIIIINECRAFT!
WE FOUND NO LINK BETWEEN PURPLE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BROWN JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN PINK JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BLUE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN TEAL JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN SALMON JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN RED JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN TURQUOISE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN MAGENTA JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN YELLOW JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN GREY JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN TAN JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN CYAN JELLY BEANS AND ACNE (P > 0.05).
WE FOUND A LINK BETWEEN GREEN JELLY BEANS AND ACNE (P < 0.05).
WHOA!
WE FOUND NO LINK BETWEEN MAUVE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BEIGE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN LILAC JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BLACK JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN PEACH JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN ORANGE JELLY BEANS AND ACNE (P > 0.05).
NEWS
GREEN JELLY BEANS LINKED TO ACNE!
95% CONFIDENCE
ONLY 5% CHANCE OF COINCIDENCE!
SCIENTISTS...

Figure 7.1: Significant: [xkcd.com](https://xkcd.com/882)

general discussion of an analysis, we normally refer to the effect we are actually interested in. This is called the *test hypothesis* or the *alternative hypothesis* (often denoted $H_1$ in statistics books). The alternative hypothesis is essentially a statement of the effect we are expecting (or hoping!) to see, e.g., purple and green plants differ in their mean size. It is a statement of whatever is implied if the null hypothesis is not true.

Having sorted out the different types of hypotheses in play, we can use a particular frequentist technique to evaluate the observed result against that expected if the null hypothesis was true. The test gives us a probability ($p$-value) telling us how likely it is that we would have got the result we observe, or a more extreme result, if the null hypothesis was really true. If the value is sufficiently small we judge it unlikely that we would have seen this result if the null hypothesis was true. Consequently, we say we *reject the null hypothesis* (i.e. reject the notion that there is no difference). This is not the same as 'proving' the alternative hypothesis is true. We can't prove anything by collecting data or carrying out an experiment.

If the $p$-value is large, then it is quite likely that we could have ended up with the observed result if the null hypothesis was true, i.e. it is probably due to sampling variation. In this case, we cannot reject the null hypothesis. Note that we say that we "*do not reject the null hypothesis*". This is not the same as accepting that the null hypothesis is true, paradoxical though this may seem. One possible reason for this is that if we only have a small sample, there may be an effect of the factor we are looking at, but we can't detect it because we don't have enough data.

**A *p*-value is not an evidence summary**

A null hypothesis significance test gives us a probability (the $p$-value) of how likely it is that we would have got the result we observe, or a more extreme result, if the null hypothesis was true. The 'result' is a property of the data and the test describes how likely the data are under a null hypothesis. A $p$-value is **not** a measure of evidence in the conventional sense or a statement about the likelihood that any particular hypothesis is true or not.

### 7.1.2 Report the null hypothesis?

It's important to realise that we **do not report null hypotheses when communicating science**. The null hypothesis is nothing more a device for setting up a frequentist statistical test. It is not an interesting or useful hypothesis in its own right. Again, for the avoidance of doubt… there's absolutely no need to state the null hypothesis of each and every test we use in a report or a presentation. Focus on substantive biological hypotheses instead.

## 7.2 Interpreting and reporting *p*-values

It is important to understand the meaning of the probabilities generated by frequentist tests. We have already said a $p$-value is the proportion of occasions on which we would expect to see a result at least as extreme as the one you actually observed if the null hypothesis (of no effect) was true. Conventionally, we accept a result as statistically significant if $p < 0.05$ (also expressed as 5%). This threshold is called the **significance level** of a test. We've said it before, but it is worth repeating: there is nothing special about the $p < 0.05$ significance level! It is just a widely used convention.

> ⚠
>
> We will always use the $p = 0.05$ threshold in this book. Try to remember this—we won't constantly remind you of it. Why use that significance level? There's no particularly special reason. We're simply following a somewhat arbitrary convention that applies to simple analyses of small datasets.

A probability of 0.05 is a chance of 1 in 20. This means that if there really was no effect of the factor we are investigating, we would expect to get a result significant at $p$=0.05 about 5 times in 100 samples. To envisage it more easily, it is slightly less than the chance of tossing a coin 4 times and getting 4 heads in a row ($p$=0.0625). It's not all that rare, really. This puts a 'significant' result into context. Would we launch a new drug on the market on the evidence of the strength of four heads coming up in a row when a coin is tossed? Probably not!

### 7.2.1 Careful with those *p*-values

This is a good time to issue an important warning about $p$-values. Frequentist $p$-values are counter-intuitive quantities that are easily (and often) misinterpreted. Whole books have been written about the problems associated with them. We don't have time to really cover the issues here, but here are a few key observations:

- Scientists tend to use $p$=0.05 to define 'significance', but $p$=0.055 is really no different from $p$=0.045. It would be irrational to reject an idea completely just on the basis of a result of $p$=0.055, while at the same time being prepared to invest large amounts of time and money implementing policies based on a result of $p$=0.045.

- The exact value of $p$ is affected by the size of the true effect being studied, the amount of data being analysed, and how appropriate a statistical model is for those data. It's very easy to arrive at a tiny $p$-value, when an effect is weak or even absent, by using a statistical model that is inappropriate for the data in hand.

- A significance test works out how likely our data are under a null hypothesis. The resulting $p$-value is not a measure of evidence or a statement about the likelihood a hypothesis is true or not. That's a shame, because those latter kinds of probabilities are the sort of thing most of us have in mind when we think about evaluating hypotheses. That doesn't change the fact that a $p$-value is not the thing we want it to be!
- A consequence of the previous point is that the relationship between the 'strength of an effect' and its associated $p$-value is complicated. It is simply not correct to equate the size of a $p$-value with the weight of evidence for the effect being present, nor is correct to interpret a $p$-value a statement about how big the effect is.

The take-home message: $p$-values are hard to interpret and only represent one possible (non-intuitive) source of information when answering scientific questions. They are not the 'final word' on truth.

### 7.2.2 Presenting $p$-values

R will typically display $p$-values from a statistical significance test to six decimal places (e.g. $p = 0.003672$). However, when we write about them, the results from tests are usually presented as one of the following four categories:

- $p > 0.05$, for results which are not statistically significant (sometimes also written as 'NS'),
- $p < 0.05$, for results where $0.01 < p < 0.05$,
- $p < 0.01$, for results where $0.001 < p < 0.01$,
- $p < 0.001$ for results where $p < 0.001$,

This presentation stems from the fact that statistical tests often had to be calculated by hand in the days before everyone had access to a computer. The significance of the result was difficult to calculate directly, so it would have been looked up in a special table. We still use this style because the value of $p$ does not have a simple interpretation in terms of weight of evidence or effect sizes. Knowing which category a $p$-value falls into provides sufficient information to roughly judge 'how significant' the result is.

**Significance thresholds vs. $p$-values**

The significance level is used to determine whether or not a result is deemed to be 'statistically significant'. We use $p < 0.05$ in this book, and we will generally adopt the above categories to report the results of a test. Don't confuse the category used to report the $p$-value with the actual significance level an investigator is using. Just because someone writes '$p$ $< 0.01$' when they report the results of a test, it does not mean that they were working at the 1% significance level ($p = 0.01$).

It's usually sufficient to use the four categories above when writing about the significance of a statistical test, though occasionally, giving the actual probability can be appropriate. For example, it can be informative to know that a test yielded $p = 0.06$ rather than simply quoting it just as $p > 0.05$ or NS. This is because $p$ is so close to the significance threshold. While not wholly convincing, it is still suggestive of the possibility that an effect is present.

**The asterisks convention**

It is common to see ranges of probabilities coded with asterisks in tables and figures:

- $*$ for $p = 0.05...0.01$,
- $**$ for $p = 0.01...0.001$,
- $***$ for $p < 0.001$.

This convention is sometimes used because it offers a compact way to represent the results of a significance test. It's fine to adopt it in tables and figures. However, people do not use the asterisks convention in the text of a report.

## 7.3 Biological vs. statistical significance

A final, but vital, point: do not confuse statistical significance with biological significance. A result may be statistically highly significant (say $p < 0.001$) but biologically trivial. To give a real example, in a study of the factors determining the distribution of freshwater invertebrates in a river, the pH of water was measured in the open water and in the middle of the beds of submerged vegetation. There was a statistically significant difference in pH ($p < 0.01$) but the mean pH values were 7.1 in the open water and 6.9 in the weeds. This is a very small effect, and almost certainly of no importance at all to the invertebrates.

The significance of a result depends on a combination of three things (1) the size of the true effect in the population, (2) the variability of the data, and (3) the sample size. Even a tiny effect can be significant if the sample size is very large. Do not automatically equate a significant result with a large biological effect. Plot the data, inspect the estimates, and consider the biological implications of the difference. The statistical results provide some guidance in separating genuine differences from random variation, but they can't tell us whether the difference is biologically interesting or important—that's the scientist's job!

# Part III

# Simple Statistics

# Chapter 8

# Parametric statistics

## 8.1 Introduction

The majority of statistical tools used in this book share one important feature—they are underpinned by a mathematical model of some kind. Because a mathematical model lurks in the background, this particular flavour of statistics called **parametric statistics**. In this context, the word 'parametric' refers to the fact that a mathematical model's behaviour depends on one or more quantities known as 'parameters'.

We are **not** going to study the mathematical details of these models. However, it is important to understand the **assumptions** underlying a statistical model. Assumptions are the aspects of a system that we accept as true, or at least nearly true. If our assumptions are not reasonable, we can't be sure that the results of our analysis (e.g. a statistical test) will be reliable. We always need to evaluate the assumptions of an analysis to determine whether or not we trust it.

Ultimately, we want to understand, in approximate terms, how a model and its assumptions lead to a particular statistical test. We have already explored aspects of this process in the last few chapters, such as sampling variation, null distributions, and *p*-values. By thinking about models and their assumptions we can connect these abstract ideas to the practical side of 'doing statistics'. Our goal in this chapter is to introduce a few more concepts required to make these connections.

## 8.2 Mathematical models

A mathematical model is a description of a system using the language and concepts of mathematics. That 'system' could be more or less anything scientists study—the movement of planets in a solar system, the carbon cycle of our planet, changes in abundance and distribution of a species, the development of an organ, or the spread of a plasmid conferring antibiotic resistance.

In the frequentist world, a statistical model is a type of mathematical model that describes how samples of data are generated from a hypothetical population. We're only going to consider only a small subset of the huge array of statistical models people routinely use. Conceptually, the parametric models we use in this book describe data in terms of a **systematic component** and a **random component**:

$$\text{Observed Data} = \text{Systematic Component} + \text{Random Component}$$

What are these two components?

- The systematic component of a model describes the structure, or the relationships, in the data. When people refer to 'the model', this is the bit they usually care about.
- The random component captures the leftover "noise" in the data. This is essentially the part of the data that the systematic component of the model fails to describe.

This is best understood by example. In what follows, we're going to label the individual values in the sample $y_i$. The $i$ in this label indexes the individual values; it takes values 1, 2, 3, 4, … and so on. We can think of the collection of the $y_i$'s as the variable we're interested in.

The simplest kind of model we might consider is one that describes a single variable. A model for these data can be written: $y_i = a + \epsilon_i$. In this model:

- The systematic component is given by $a$. This is usually thought of as the population mean.
- The random component is given by $\epsilon_i$, which describes how the individual values deviate from the mean.

A more complicated model might consider the relationship between the values of $y_i$ and another variable. We'll call this second variable $x_i$. A model for these data could be written as: $y_i = a + b \times x_i + \epsilon_i$. In this model:

- The systematic component is the $a + b \times x_i$ bit. This is just the equation of a straight line, with intercept $a$ and slope $b$.
- The random component is given by the $\epsilon_i$. Now, this term describes how the individual values deviate from the line.

These two descriptions have not yet completely specified our example models. We need to make one more assumption to complete them. The values of the $\epsilon_i$ differ from one observation to the next because they describe the noise in the system. For this reason, $\epsilon_i$ is thought of as a statistical variable, which means we also need to specify its *distribution*.

What is missing from our description so far is, therefore, a statement about the distribution of the $\epsilon_i$. We need to formally set out what kinds of values it is allowed to take and how likely those different values are. The distributional assumption is almost always the same in this book: we assume that the $\epsilon_i$ are drawn from a **normal distribution**.

## 8.3 The normal distribution

Most people have encountered the normal distribution at one point or another, even if they didn't realise it at the time. Here is a histogram of 100000 values drawn from a normal distribution:

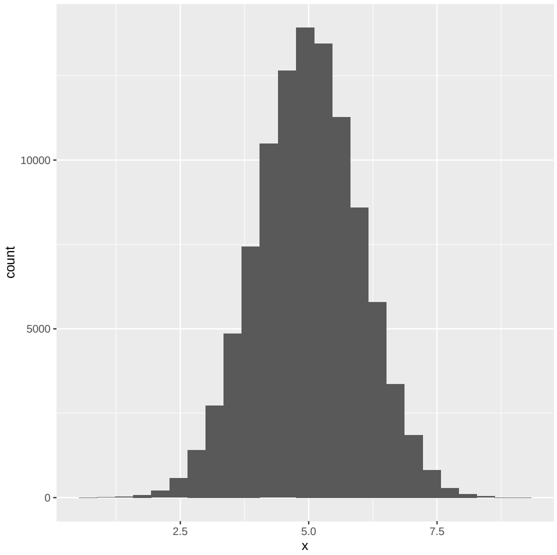


Figure 8.1: Distribution of a large sample of normally distributed variable

Does that look familiar? The normal distribution is sometimes called the 'Gaussian distribution', or more colloquially, the 'bell-shaped curve'. We don't have time in this book to study this distribution in any detail. Instead, we'll simply list some key facts about the normal distribution that relate to the statistical models we'll be using later on:

1. The normal distribution is appropriate for numeric variables measured on

an interval or ratio scale. Strictly speaking, the variable should also be continuous, though a normal distribution can provide a decent approximation for some kinds of discrete numeric data.

2. The normal distribution is completely described by its mean (its central tendency) and its standard deviation (its dispersion). If we know these two quantities for a particular normal distribution, we know everything there is to know about that distribution.
3. If a variable is normally distributed, then about 95% of its values will fall inside an interval that is 4 standard deviations wide: the upper bound is equal to the $\text{Mean} + 2 \times \text{S.D.}$; the lower bound is equal to $\text{Mean} - 2 \times \text{S.D.}$.
4. When we add or subtract two normally distributed variables to create a new variable, the resulting variable will also be normally distributed. Similarly, if we multiply a normally distributed by a number to create a new variable, the resulting variable will still be normally distributed.

The mathematical properties of the normal distribution are very well understood. This has made it possible for mathematicians to determine how the sampling distribution of means and variances behaves when the underlying variables are normally distributed. This knowledge underpins many of the statistical tests we use in this book.

### 8.3.1 Standard error of the mean

Let's consider a simple example. We want to estimate the standard error associated with a sample mean. If we're happy to assume that the sample was drawn from a normal distribution, then there's no need to resort to computationally expensive techniques like bootstrapping to work this out.

There is a well-known formula for calculating the standard error once we assume normality. If $s^2$ is the variance of the sample, and $n$ is the sample size, the standard error is given by:

$$\text{Standard error of the mean} = \sqrt{\frac{s^2}{n}}$$

That's it, if we know the variance and the size of a sample, it's easy to estimate the standard error of its mean *if we're happy to assume the sample came from a normally distributed variable*[1].

In fact, as a result of rule #4 above, we can calculate the standard error of *any quantity* that involves adding or subtracting the means of samples drawn from normal distributions.

[1] In fact, the equation for the standard error of a mean applies when we have a big sample, even if that sample did not come from a normally distributed variable. This result is a consequence of that 'the central limit theorem' mentioned in this chapter.

### 8.3.2 The *t* distribution

The normal distribution is usually the first distribution people learn about in a statics course. The reasons for this are:

- it crops up a lot as a consequence of something called 'the central limit theorem', and
- many other important distributions are related to the normal distribution in some way.

We're not going to worry about the central limit theorem here. However, we do need to consider one more distribution. One of the most important of those 'other distributions' is Student's *t*-distribution[2].

This distribution arises whenever:

- we take a sample from a normally distributed variable,
- estimate the population mean from the sample,
- and then divide the mean by its standard error.

The sampling distribution of this new quantity has a particular form. It follows a Student's *t*-distribution.

Student's *t*-distribution arises all the time in relation to means. For example, what happens if we take samples from a pair of normal distributions, calculate the difference between their estimated means, and then divide this difference by its standard error? The sampling distribution of the scaled difference between means also follows a Student's *t*-distribution.

Because it involves rescaling a mean by its standard error, the form of a *t*-distribution only depends on one thing: the sample size. This may not sound like an important result, but it really is, because it allows us to construct simple statistical tests to evaluate differences between means. We'll be relying on this result in the next two chapters as we learn about so-called '*t*-tests'.

[2] Why is it called Student's *t*? The *t*-distribution was discovered by W.G. Gosset, a statistician employed by the Guinness Brewery. He published his statistical work under the pseudonym of 'Student', because Guinness would have claimed ownership of his work if he had used his real name.

# Chapter 9

# One sample *t*-tests

## 9.1 When do we use a one-sample *t*-test?

The one-sample *t*-test is a very simple statistical test. It is used when we have a sample of a numeric variable and we want to compare its population mean to a particular value. The one-sample *t*-test evaluates whether the population mean is likely to be different from this value. The expected value could be any value we are interested in. Here are a couple of specific examples:

- We have developed a theoretical model of foraging behaviour that predicts an animal should leave a food patch after 10 minutes. Suppose we have data on the actual time spent by 25 animals observed foraging in the patch. In that case, we could test whether the mean foraging time is 'significantly different' from the prediction using a one-sample *t*-test.
- We are monitoring sea pollution and have collected a series of water samples from a beach. We wish to test whether the mean density of faecal coliforms—bacteria indicative of sewage discharge—can be regarded as greater than the legislated limit. A one-sample *t*-test will test whether the mean value for the beach as a whole exceeds this limit.

Let's see if we can get a sense of how these tests would work.

## 9.2 How does the one-sample *t*-test work?

Imagine we have taken a sample of some variable, and we want to evaluate whether its mean is different from some number (the 'expected value').

Here's an example of what these data might look like if we had used a sample size of 50:

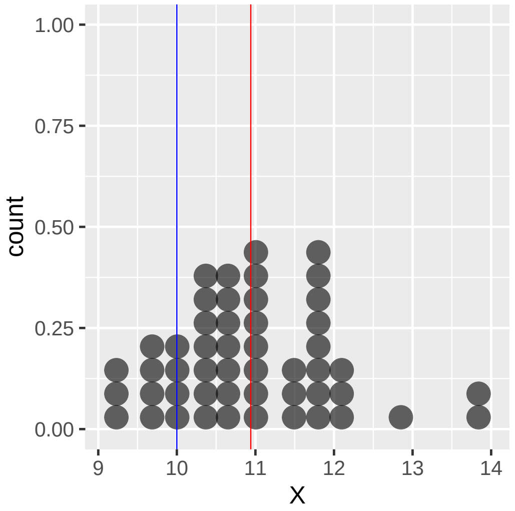


Figure 9.1: Example of data used in a one-sample t-test

We're calling the variable 'X' in this example. It needs a label, and 'X' is as good as any other. The red line shows the sample mean. This is a bit less than 11. The blue line shows the expected value. This is 10, so this example could correspond to the foraging study mentioned above.

The observed sample mean is about one unit larger than the expected value. How do we decide whether the population mean is really different from the expected value? Perhaps the difference between the observed and expected value is due to sampling variation. Here's how a frequentist tackles this kind of question:

1. Set up an appropriate null hypothesis, i.e. an hypothesis of 'no effect' or 'no difference'. The null hypothesis in this type of question is that *the population mean is equal to the expected value*.

2. Work out what the sampling distribution of a sample mean looks like under the null hypothesis. This is the null distribution. Because we're now using a parametric approach, we will assume this has a particular form.

3. Finally, we use the null distribution to assess how likely the observed result is under the null hypothesis. This is the *p*-value calculation that we use to summarise our test.

This chain of reasoning is no different from that developed in the bootstrapping example from earlier. We're just going to make an extra assumption this time to allow us to use a one-sample *t*-test. This extra assumption is that *the variable ('X') is normally distributed in the population.* When we make this normality assumption, the whole process of carrying out the statistical test is actually very simple because the null distribution will have a known mathematical form—it ends up being a *t*-distribution.

We can use this knowledge to construct the test of statistical significance. Instead of using the whole sample, as we did with the bootstrap, now we only need three simple pieces of information to construct the test: the sample size, the sample variance, and the sample mean. The one-sample *t*-test is then carried out as follows:

**Step 1.** Calculate the sample mean. This happens to be our 'best guess' of the unknown population mean. However, its role in the one-sample *t*-test is to allow us to construct a test statistic in the next step.

**Step 2.** Estimate the standard error of *the sample mean*. This gives us an idea of how much sampling variation we expect to observe. The standard error doesn't depend on the true value of the mean, so the standard error of the sample mean is also the standard error of any mean under any particular null hypothesis.

This second step boils down to applying a simple formula involving the sample size and the standard deviation of the sample:

$$\text{Standard Error of the Mean} = \sqrt{\frac{s^2}{n}}$$

…where $s^2$ is the square of the standard deviation (the sample variance) and $n$ is for the sample size. This is the formula introduced in the parametric statistics chapter. The standard error of the mean gets smaller as the sample size grows or the sample variance shrinks.

**Step 3.** Calculate a 'test statistic' from the sample mean and standard error. We calculate this by dividing the sample mean (step 1) by its estimated standard error (step 2):

$$\text{t} = \frac{\text{Sample Mean}}{\text{Standard Error of the Mean}}$$

If our normality assumption is reasonable, this test-statistic follows a *t*-distribution. This is guaranteed by the normality assumption. So this particular test statistic is also a *t*-statistic. That's why we label it *t*. This knowledge leads to the final step…

**Step 4.** Compare the *t*-statistic to the theoretical predictions of the *t*-distribution to assess the statistical significance of the difference between observed and expected value. We calculate the probability that we would have observed a difference with a magnitude as large as, or larger than, the observed difference, if the null hypothesis were true. That's the *p*-value for the test.

We could step through the actual calculations involved in these steps in detail, using R to help us, but there's no need to do this. We can let R handle everything for us. But first, we should review the assumptions of the one-sample $t$-test.

### 9.2.1 Assumptions of the one-sample $t$-test

Several assumptions need to be met in order for a one-sample $t$-test to be valid. Some of these are more important than others. We'll start with the most important and work down the list in reverse order of importance:

1. **Independence.** In rough terms, independence means each observation in the sample does not 'depend on' the others. We'll discuss this more carefully when we consider the principles of experimental design. The key thing to know now is why this assumption matters: if the data are not independent, the $p$-values generated by the one-sample $t$-test will be unreliable.

(In fact, the $p$-values will be too small when the non-independence assumption is broken. That means we risk the false conclusion that a difference is statistically significant when in reality, it is not)

2. **Measurement scale.** The variable being analysed should be measured on an interval or ratio scale, i.e. it should be a numeric variable of some kind. Applying a one-sample $t$-test to a variable that is not measured on one of these scales doesn't make much sense.

3. **Normality.** The one-sample $t$-test will only produce completely reliable $p$-values when the variable is normally distributed in the population. However, this assumption is less important than many people think. The $t$-test is robust to mild departures from normality when the sample size is small, and when the sample size is large, the normality assumption hardly matters at all.

We don't have the time to explain why the normality assumption is not too important for large samples, but we can at least state the reason: it is a consequence of that central limit theorem we mentioned in the last chapter.

### 9.2.2 Evaluating the assumptions

The first two assumptions—independence and measurement scale—are aspects of experimental design. We can only evaluate these by thinking carefully about how the data were gathered and what was measured. It's too late to do anything about these after we have collected our data.

What about that 3$^{\text{rd}}$ assumption—normality? One way to evaluate this is by visualising the sample distribution. For small samples, if the sample distribution looks approximately normal then it's probably fine to use the $t$-test. For large samples, we don't even need to worry about a moderate departure from normality[1].

## 9.3 Carrying out a one-sample *t*-test in R

> As usual, a good way to understand what follows is by working through the example. Remember, the code again assumes that 'MORPH_DATA.CSV' has been read into a tibble called `morph_data`.

We'll use the plant morph example again to learn how to carry out a one-sample *t*-test in R. Remember, the data were 'collected' to 1) compare the frequency of purple morphs to a prediction and 2) compare the mean dry weight of purple and green morphs. Neither of these questions can be tackled with a one-sample *t*-test.

Instead, let's pretend that we unearthed a report from 30 years ago that found the mean size of purple morphs to be 710 grams. We want to evaluate whether the mean size of purple plants in the contemporary population is different from this expectation, because we think they may have adapted to local conditions.

We only need the purple morph data for this example, so we should first `filter` the data to get hold of only the purple plants:

```r
# get just the purple morphs
purple_morphs <- filter(morph_data, Colour == "Purple")
```

The `purple_morphs` data set has two columns: `Weight` contains the dry weight biomass of purple plants, and `Colour` indicates which sample (plant morph) an observation belongs to. We don't need the `Colour` column anymore because we've just removed all the green plants, but there's no harm in leaving it in.

### 9.3.1 Visualising the data and checking the assumptions

We start by calculating a few summary statistics and visualising the sample distribution of purple morph dry weights. That's right… always check the data before carrying out a statistical analysis! We already looked over these data in the Statistical comparisons chapter. However, we'll proceed as though this is the first time we've seen them to demonstrate a complete workflow.

[1] It's hard to say what constitutes a 'large' sample. If we're lucky enough to be working with hundreds of observations it's fine to ignore mild departures from normality. If we're working with 10s of observations we probably need to pay a bit more attention to the normality assumption in a *t*-test.

Here is the **dplyr** code to produce the descriptive statistics:

```
purple_morphs %>%
  summarise(mean = mean(Weight),
            sd = sd(Weight),
            samp_size = n())
```

```
## # A tibble: 1 x 3
##    mean    sd samp_size
##   <dbl> <dbl>     <int>
## 1  767.  156.        77
```

We have 77 purple plants in the sample. Not bad but we should keep an eye on the normality assumption. Let's check this by making a histogram with the sample of purple plant dry weights:

```
ggplot(purple_morphs, aes(x = Weight)) +
  geom_histogram(binwidth = 50)
```

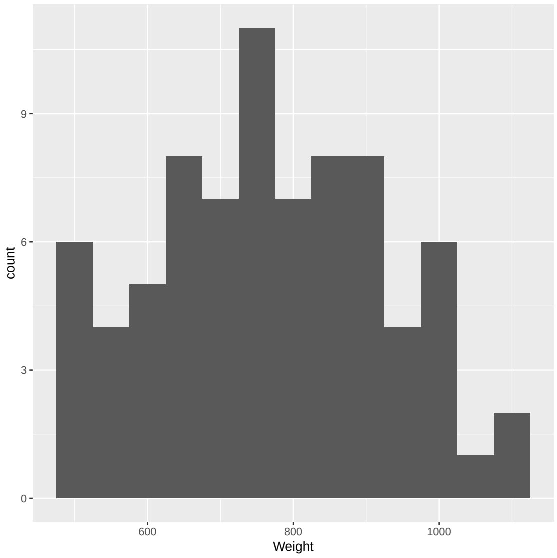


Figure 9.2: Size distributions of purple morph dry weight sample

These is nothing too 'non-normal' about this sample distribution—it's roughly bell-shaped—so it seems reasonable to assume it came from normally distributed population.

### 9.3.2 Carrying out the test

It's easy to carry out a one-sample *t*-test in R. We use a function called `t.test` (no surprises there). Remember, `Weight` contains the dry weight biomass of purple plants. Here's the R code to carry out a one-sample *t*-test:

```
t.test(purple_morphs$Weight, mu = 710)
```

We have suppressed the output because we initially want to focus on how to use the `t.test` function. We have to assign two arguments to control what the function does:

1. The first argument (`purple_morphs$Weight`) is simply a numeric vector containing the sample values. Sadly we can't give `t.test` a data frame when doing a one-sample test. Instead, we have to pull out the column we're interested in using the `$` operator.

2. The second argument (called `mu`) sets the expected value to which we want to compare the sample mean. So `mu = 710` tells the function to compare the sample mean to a value of 710. This can be any value we like, and the correct value depends on the question we're asking.

That's it for setting up the test. Here is the R code for the test again, this time with the output included:

```
t.test(purple_morphs$Weight, mu = 710)
```

```
##
##  One Sample t-test
##
## data:  purple_morphs$Weight
## t = 3.1811, df = 76, p-value = 0.002125
## alternative hypothesis: true mean is not equal to 710
## 95 percent confidence interval:
##  731.1490 801.9787
## sample estimates:
## mean of x
##  766.5638
```

The first line tells us what kind of *t*-test we used. This says: `One Sample t-test`. OK, now we know that we used the one-sample *t*-test (there are other kinds). The next line reminds us about the data. This says: `data: purple_morphs$Weight`, which is R-speak for 'we compared the mean of the `Weight` variable to an expected value. Which value? This is given later.

The third line of text is the most important. This says: `t = 3.1811, df = 76, p-value = 0.002125`. The first part of this, `t = 3.1811`, is the test statistic, i.e. the value of the *t*-statistic. The second part, `df = 76`, summarise the 'degrees of freedom'. This is required to work out the *p*-value. It also tells us something about how much 'power' our statistical test has (see the box below). The third part, `p-value = 0.002125`, is the all-important *p*-value.

That *p*-value indicates a statistically significant difference between the mean dry weight biomass and the expected value of 710 g ($p$ is less than 0.05). Because the *p*-value is less than 0.01 but greater than 0.001, we report this as '$p < 0.01$'. Read through the Presenting *p*-values section again if this logic is confusing.

The fourth line of text (`alternative hypothesis: true mean is not equal to 710`) tells us the alternative to the null hypothesis ($H_1$). More importantly, this reminds us which expected value was used to formulate the null hypothesis (mean = 710).

The next two lines show us the '95% confidence interval' for the difference between the means. We don't need this information now, but we can think of this interval as a rough summary of the likely values of the 'true' mean[2].

The last few lines summarise the sample mean. This might be useful if we had not already calculated.

**A bit more about degrees of freedom**

Degrees of freedom (abbreviated d.f. or df) are closely related to the idea of sample size. The greater the degrees of freedom associated with a test, the more likely it is to detect an effect if it's present. To calculate the degrees of freedom, we start with the sample size, and then reduce this number by one for every quantity (e.g. a mean) we had to calculate to construct the test.

Calculating degrees of freedom for a one-sample *t*-test is easy. The degrees of freedom are just n-1, where n is the number of observations in the sample. We lose one degree of freedom because we have to calculate one sample mean to construct the test.

### 9.3.3 Summarising the result

Having obtained the result, we now need to write our conclusion. We are testing a scientific hypothesis, so we must always return to the original question to write the conclusion. In this case, the appropriate conclusion is:

> The mean dry weight biomass of purple plants is significantly different from the expectation of 710 grams ($t = 3.18$, d.f. $= 76$, $p <$

[2] In reality, a frequentist confidence interval is more complicated than that but this book probably isn't the right place to get into that discussion. Take our word for it though—confidence intervals are odd things.

> 0.01).

This is a concise and unambiguous statement in response to our initial question. The statement indicates the result of the statistical test **and** which value was used in the comparison. It is sometimes appropriate to give the values of the sample mean in the conclusion:

> The mean dry weight biomass of purple plants (767 grams) is significantly different from the expectation of 710 grams ($t = 3.18$, d.f. = 76, $p < 0.01$).

Notice that we include details of the test in the conclusion. However, keep in mind that when writing scientific reports, the reporting of any statistical test should be a conclusion like the one above. **Simply writing $t = 3.18$ or $p < 0.01$ is not an adequate.**

There are a number of common questions that arise when presenting $t$-test results:

1. **What do I do if $t$ is negative?** Don't worry. A $t$-statistic can come out negative or positive, it simply depends on which order the two samples are entered into the analysis. Since the absolute value of $t$ determines the $p$-value, when presenting the results, just ignore the minus sign and give $t$ as a positive number.

2. **How many significant figures for $t$?** The $t$-statistic is conventionally given to 3 significant figures. This is because, in terms of the $p$-value generated, there is almost no difference between, say, $t = 3.1811$ and $t = 3.18$.

3. **Upper or lower case** The $t$ statistic should always be written as lower case when writing it in a report (as in the conclusions above). Similarly, d.f. and $p$ are always best as lower case. Some statistics we encounter later are written in upper case, but even with these, d.f. and $p$ should be lower case.

4. **How should I present $p$?** There are various conventions in use for presenting $p$-values. We discussed these in the Hypotheses and $p$-values chapter. Learn them! It's not possible to understand scientific papers or prepare reports properly without knowing these conventions.

**p = 0.00? It's impossible! p = 1e-16? What's that?**

Some statistics packages will sometimes give a probability of $p = 0.00$. This does not mean the probability was really zero. A probability of zero would mean an outcome was impossible, even though it happened! When a computer package reports $p = 0.00$ it just means that the probability was 'very small' and ended up being rounded down to 0.

R typically uses a slightly different convention for presenting small probabilities. A very small probability is given as `p-value < 2.2e-16`. What does 2.2e-16 mean? This is R-speak for scientific notation, i.e. $2.2e^{-16}$ is equivalent to $2.2 \times 10^{-16}$.

It is fine to report a result like this as $p < 0.001$; definitely do not write something like $p =$ 2.2e-16.

# Chapter 10

# Two-sample *t*-test

## 10.1 When do we use a two-sample *t*-test?

The two-sample *t*-test is a parametric version of the permutation procedure we studied in the statistical comparisons chapter. The test can be used to compare the means of a numeric variable sampled from two independent populations. A two-sample t-test aims to evaluate whether or not a mean is different in the two populations. Here are two examples:

- We're studying how the dietary habits of Scandinavian eagle-owls vary among seasons. We suspect that the dietary value of a prey item is different in the winter and summer. To evaluate this prediction, we measure the size of Norway rat skulls in the pellets of eagle-owls in summer and winter, and then compare the mean size of rat skulls in each season using a two-sample *t*-test.
- We're interested in the costs of anti-predator behaviour in *Daphnia* spp. We conducted an experiment where we added predator kairomones—chemicals that signal the presence of a predator—to jars containing individual *Daphnia*. There is a second control group where no kairomone was added. The change in body size of individuals was measured after one week. We could use a two-sample *t*-test to compare the mean growth rate in the control and treatment groups.

## 10.2 How does the two-sample *t*-test work?

Imagine we have taken a sample of a variable from two populations, labelled 'A' and 'B'. We'll call the variable 'X' again. Here's an example of how these kinds of data might look if we had a sample of 50 items from each population:

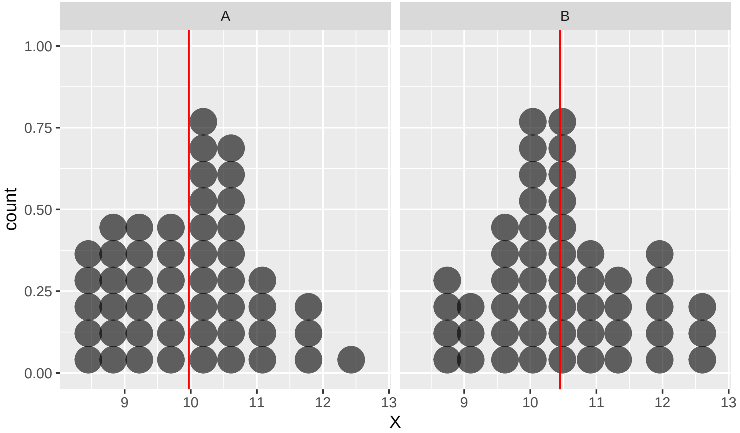


Figure 10.1: Example of data used in a two-sample t-test

The first thing to notice is the two distributions overlap quite a lot. However, this observation isn't necessarily all that significant. Why? Because we're not interested in the raw values of 'X' in the two samples. It's the *difference between the means* that our test is going to focus on.

The red lines show the mean of each sample. Sample B has a larger mean than sample A. The question is, how do we decide whether this difference is 'real', or purely a result of sampling variation?

Using a frequentist approach, we tackle this question by first setting up the appropriate null hypothesis. The null hypothesis here is one of no difference between the population means (they are equal). We then have to work out what the null distribution looks like. This is the sampling distribution of the difference between sample means under the null hypothesis. Once we have the null distribution worked out, we can calculate a $p$-value.

How is this different from the permutation test? The logic is virtually identical. The one key difference is that we have to make an extra assumption to use the two-sample $t$-test. We assume the variable is normally distributed in each population. If this assumption is valid, then the null distribution will have a known form—a $t$-distribution.

We only need a few pieces of information to carry out a two-sample $t$-test. These are basically the same quantities needed to construct the one-sample $t$-test, except now there are two samples involved. We need the sample sizes of groups A and B, the sample variances and the estimated difference between the sample means of X in A and B.

How does a two-sample $t$-test work in practice? It is carried out as follows:

**Step 1.** Calculate the two sample means, then calculate the difference between these estimates. This estimate is our 'best guess' of the true difference between means. As with the one-sample test, its role in the two-sample *t*-test is as a test statistic.

**Step 2.** Estimate the standard error of *the difference between the sample means* under the null hypothesis of no difference. This gives us an idea of how much sampling variation we expect to observe in the estimated difference if there were actually no difference between the means.

There are a number of different options for estimating this standard error. Each one makes a different assumption about the variability of the two populations. Whatever choice we make, the calculation always boils down to a simple formula involving sample sizes and variances. The standard error gets smaller when the sample sizes grow, or when the sample variances shrink. That's the important point.

**Step 3.** We can calculate the test statistic once we have estimated the difference between sample means and its standard error. This is a type of *t*-statistic, which we calculate by dividing the difference between sample means (from step 1) by the estimated standard error of the difference (from step 2):

$$\text{t} = \frac{\text{Difference Between Sample Means}}{\text{Standard Error of the Difference}}$$

This *t*-statistic is guaranteed to follow a *t*-distribution if the normality assumption is met. This knowledge leads to the final step…

**Step 4.** Compare the test statistic to the theoretical predictions of the *t*-distribution to assess the statistical significance of the observed difference. That is, we calculate the probability that we would have observed a difference between means with a magnitude as large as, or larger than, the observed difference, if the null hypothesis were true. That's the *p*-value for the test.

We will not step through the various calculations involved in these steps. The formula for the standard two-sample *t*-test and its variants are summarised on the *t*-test Wikipedia page. There's no need to learn these because we can get R to handle the calculations for us.

Before we do that, we need to review the assumptions of the two-sample *t*-test.

### 10.2.1 Assumptions of the two-sample *t*-test

Several assumptions need to be met for a two-sample *t*-test to be valid. These are essentially the same as those for the one-sample version. Starting with the most important and working down in decreasing order of importance, these are:

1. **Independence.** Remember what we said in our discussion of the one-sample *t*-test? If the data are not independent, the *p*-values generated

by the test will be too small, and even mild non-independence can be a serious problem. The same is true of the two-sample *t*-test.

2. **Measurement scale.** The variable we are working with should be measured on an interval or ratio scale, which means it will be numeric. It makes little sense to apply a two-sample *t*-test to a categorical variable of some kind.
3. **Normality.** The two-sample *t*-test will produce reliable *p*-values if the variable is normally distributed in each population. However, the two-sample *t*-test is fairly robust to mild departures from normality and this assumption matters little when the sample sizes are large.

### 10.2.2 Evaluating the assumptions

How do we evaluate the first two assumptions? As with the one-sample test, these are aspects of experimental design—we can only evaluate them by thinking about the data we've collected.

We can evaluate the normality assumption by plotting the sample distribution *of each group* using, for example, a pair of histograms or dot plots. We must examine the distribution in each group, not the distribution of the combined sample. If both samples looks approximately normal then we can proceed with the test. If we think we can see a departure from normality, we should consider our sample sizes before proceeding. Remember that the test is robust to mild departures when sample sizes are large (e.g. >100 observations in each group).

### 10.2.3 What about the *equal variance* assumption?

It is sometimes said that a two-sample *t*-test requires the variability ('variance') of each sample to be the same, or at least quite similar. This would be true if we used the original version of Student's two-sample *t*-test. However, R doesn't use this version of the test by default. R uses the "Welch" two-sample *t*-test. This version of the test does not rely on the equal variance assumption. As long as we stick with this version of the *t*-test, the equal variance assumption isn't something we need to worry about.

**Should we use Student's original two-sample *t*-test?**

The original 'equal-variance' two-sample *t*-test is a little more powerful than the Welch version. That is, it is more likely to detect a difference in means (if present). However, the increase in statistical power is really quite small if the sample sizes of each group are similar and the original test is only correct when the population variances are identical. Since we can never prove the ‘equal variance’ assumption—we can only ever reject it—it is generally safer to just use the Welch two-sample $t$-test.

One last warning. Student’s two-sample $t$-test assumes the variances of *the populations* are identical. It is the population variances, not the sample variances, that matter. There are methods for comparing variances, and people sometimes suggest using these to select ‘the right’ $t$-test. This is bad advice. For reasons just outlined, there’s little advantage to using Student’s version of the test if the variances really are the same. What’s more, the process of picking the test based on the results of another statistical test affects the reliability of the resulting $p$-values.

## 10.3 Carrying out a two-sample *t*-test in R

We’re going to use the plant morph example one last time. And yes, the code below assumes ‘MORPH_DATA.CSV’ has been read into a tibble called `morph_data`.

We’ll use the plant morph size example again to learn the workflow for two-sample $t$-tests in R. We’ll use the test to evaluate whether or not the mean dry weight of purple plants is different from that of green plants. The background to *why* this comparison is worth doing was covered in the statistical comparisons chapter. We considered this comparison using a permutation test in that chapter. It’s much simpler to use a two-sample $t$-test, as we are about to find out…

### 10.3.1 Visualising the data and checking the assumptions

We start by calculating a few descriptive statistics and visualising the sample distributions of the green and purple morph dry weights. We already did this in the statistical comparisons chapter, but here’s the **dplyr** code for the descriptive statistics again:

```
morph_data %>%
  # group the data by plant morph
  group_by(Colour) %>%
  # calculate the mean, standard deviation and sample size
  summarise(mean = mean(Weight),
            sd = sd(Weight),
            samp_size = n())
```

```
## # A tibble: 2 x 4
##   Colour  mean    sd samp_size
##   <chr>  <dbl> <dbl>     <int>
## 1 Green   708.  150.       173
## 2 Purple  767.  156.        77
```

As we already know, the purple plants tend to be bigger than green plants. The sample sizes are 173 (green plants) and 77 (purple plants). These are good-sized samples, which means we probably don't need to be too obsessed with the normality assumption. However, we still need to visualise the sample distributions to be sure they aren't too odd or contain obvious outliers:

```
ggplot(morph_data, aes(x = Weight)) +
  geom_histogram(binwidth = 50) +
  facet_wrap(~Colour, ncol = 1)
```

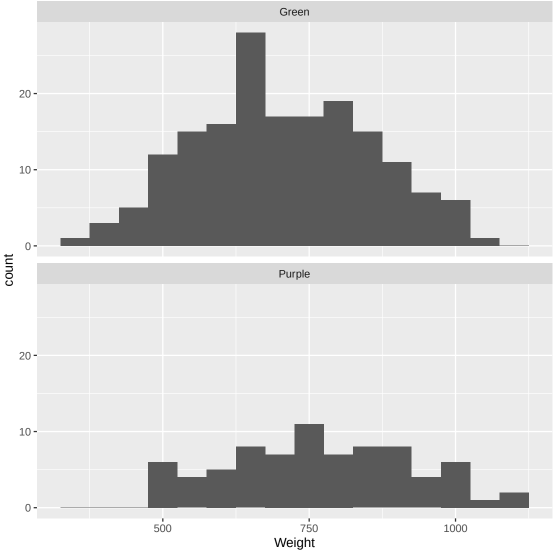


Figure 10.2: Size distributions of purple and green morph samples

There is nothing too 'non-normal' about the two samples in this example, so it's reasonable to assume they both came from normally distributed populations.

### 10.3.2 Carrying out the test

The function we need to carry out a two-sample *t*-test in R is the `t.test` function, i.e. the same functions used for the one-sample test.

Remember, `morph_data` has two columns: `Weight` contains the dry weight biomass of each plant, and `Colour` is an index variable that indicates which sample (plant morph) an observation belongs to. Here's the code to carry out the two-sample *t*-test:

```
t.test(Weight ~ Colour,  morph_data)
```

We suppressed the output, for now, to focus on how to use the function. We have to assign two arguments:

1. The first argument is something called a **formula**. We know this because it includes a 'tilde' symbol: `~`. The variable name on the left of the `~` must be the variable whose mean we want to compare (`Weight`). The variable on the right must be the indicator variable that says which group each observation belongs to (`Colour`).

2. The second argument is the name of the data frame that contains the variables listed in the formula.

Let's take a look at the output:

```
t.test(Weight ~ Colour,  morph_data)
```

```
## 
##  Welch Two Sample t-test
## 
## data:  Weight by Colour
## t = -2.7808, df = 140.69, p-value = 0.006165
## alternative hypothesis: true difference in means between group Green and group Purple is not e
## 95 percent confidence interval:
##  -100.46812  -16.97476
## sample estimates:
##  mean in group Green mean in group Purple 
##             707.8424             766.5638
```

The first line reminds us what kind of *t*-test we used. This says: `Welch two-sample t-test`, so we know that we used the Welch version of the test that accounts for the possibility of unequal variance. The next line reminds us about the data. This says: `data: Weight by Colour`, which is R-speak for 'we compared the means of the `Weight` variable, where the group membership is defined by the values of the `Colour` variable'.

The third line of text is the most important. This says: `t = -2.7808, d.f. = 140.69, p-value = 0.006165`. The first part, `t = -2.7808`, is the test statistic (i.e. the value of the *t*-statistic). The second part, `df = 140.69`, summarise the 'degrees of freedom' (see the box below). The third part, `p-value = 0.006165`, is the all-important p-value. This says there is a statistically significant difference in the mean dry weight biomass of the two colour morphs, because $p$<0.05. Because the $p$-value is less than 0.01 but greater than 0.001, we would report this as '$p < 0.01$'.

The fourth line of text (`alternative hypothesis: true difference in means is not equal to 0`) reminds us of the alternative to the null hypothesis ($H_1$). This isn't hugely important or interesting.

The next two lines show us the '95% confidence interval' for the difference between the means. Just as with the one-sample *t*-test we can think of this interval as a rough summary of the likely values of the true difference (again, a confidence interval is more complicated than that in reality).

The last few lines summarise the sample means of each group. This could be useful if we did not bother to calculate these already.

**A bit more about degrees of freedom**

In the original version of the two-sample *t*-test (the one that assumes equal variances), the degrees of freedom of the test are given by ($n_A$-1) + ($n_B$-1), where $n_A$ is the number of observations in sample A, and $n_B$ the number of observations in sample B. The plant morph data included 77 purple plants and 173 green plants, so if we had used the original version of the test we would have (77-1) + (173-1) = 248 d.f.

The Welch version of the *t*-test reduces the degrees of freedom by using a formula that takes into account the difference in variance in the two samples. The greater the difference in the two sample sizes, the smaller the number of degrees of freedom becomes:

- When the sample sizes are similar, the adjusted d.f. will be close to that in the original version of the two-sample *t*-test.
- When the sample sizes are very different, the adjusted d.f. will be close to the sample size of the smallest sample.

Notice that the 'unequal sample size accounting' results in degrees of freedom that are not whole numbers.

We don't need to know all of this to use the test. Here's the important point: whatever flavour of *t*-test we're using, a test with high degrees of freedom is more powerful than one with low degrees of freedom, i.e. the higher the degrees of freedom, the more likely we are to detect an effect if it is present. This is why degrees of freedom matter.

### 10.3.3 Summarising the result

Having obtained the result, we need to report it. We should go back to the original question to do this. In our example, the appropriate summary is:

> Mean dry weight biomass of purple and green plants differs significantly (Welch's t = 2.78, d.f. = 140.7, $p < 0.01$), with purple plants being the larger.

This is a concise and unambiguous statement in response to our initial question. The statement indicates the result of the statistical test **and** which of the mean values is the larger (readers usually want to know this). Alternatively, we might prefer to give the values of the two means:

> The mean dry weight biomass of purple plants (767 grams) is significantly greater than that of green plants (708 grams) (Welch's $t =$ 2.78, d.f. = 140.7, $p < 0.01$)

When writing scientific reports, the result of any statistical test should be a statement like the one above—simply writing t = 2.78 or p < 0.01 is not an adequate conclusion!

# Part IV

# Correlation and Regression

# Chapter 11

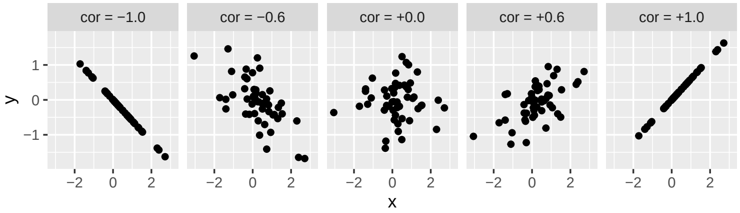
# Correlation tests

We're going to consider 'correlation' in this chapter. A correlation is a statistical measure of **association** between two variables. An association is any relationship between the variables that makes them dependent in some way: knowing the value of one variable gives you information about the possible values of the other.

The terms 'association' and 'correlation' are often used interchangeably but strictly speaking, correlation has a narrower definition. Correlation analysis quantifies the degree to which an association tends to a certain pattern via a measure called a **correlation coefficient**. For example, the correlation coefficient studied below—Pearson's correlation coefficient—measures the degree to which two variables tend toward a straight line relationship.

There are different methods for quantifying correlation, but these all share a number of properties:

1. If there is no relationship between the variables, the correlation coefficient will be zero. The closer to 0 the value, the weaker the relationship. A perfect correlation will be either -1 or +1, depending on the direction. This is illustrated in the figure below:



2. The value of a correlation coefficient indicates the direction and strength of the association, but it says nothing about the steepness of the relationship.

A correlation coefficient is just a number, so it can't tell us exactly how one variable depends on the other.

3. Correlation coefficients do not describe 'directional' or 'casual' relationships. We can't use correlations to make predictions about one variable based on knowledge of another or make statements about the effect of one variable on the other.
4. A correlation coefficient doesn't tell us whether an association is likely to be 'real' or not. We have to use a statistical significance test to evaluate whether a correlation may be different from zero. Like any statistical test, this requires certain assumptions about the variables to be met.

We're going to make sense of all this by studying one particular correlation coefficient in this chapter: **Pearson's product-moment correlation coefficient** ($r$). Various measures of association exist, so why focus on this one? Well… once you know how to work with one type of correlation in R, it isn't hard to use another. Pearson's product-moment correlation coefficient is the most well-known, which means it is as good a place as any to learn about correlation analysis.

## 11.1 Pearson's product-moment correlation

What do we need to know about Pearson's product-moment correlation? Let's start with the naming conventions. People often use "Pearson's correlation coefficient" or "Pearson's correlation" as a convenient shorthand because writing "Pearson's product-moment correlation coefficient" all the time soon becomes tedious. If we want to be concise, we can use the standard mathematical symbol to denote Pearson's correlation coefficient—lower case '$r$'.

The one thing we absolutely have to know about Pearson's correlation coefficient is that it is **a measure of linear association between numeric variables**. This means Pearson's correlation is appropriate when numeric variables follow a 'straight-line' relationship. That doesn't mean they have to be perfectly related, by the way. It simply means there shouldn't be any 'curviness' to their pattern of association[1].

Finally, calculating Pearson's correlation coefficient serves to *estimate* the strength of an association. An estimate can't tell us whether that association is likely to be 'real' or not. We need a statistical test to tackle that question. There is a standard parametric test associated with Pearson's correlation coefficient. Unfortunately, this does not have its own name. We will call it "Pearson's correlation test" to distinguish the test from the coefficient. Just keep in mind these are not 'official' names.

[1] If non-linear associations *are* apparent it's generally better to use a different correlation coefficient (we'll consider one alternative later in the book).

### 11.1.1 Pearson's correlation test

The logic underpinning Pearson's correlation test is the same as we've seen in previous tests:

1. Define a null hypothesis.
2. Calculate an appropriate test statistic.
3. Work out the null distribution of that statistic.
4. Use this to calculate a *p*-value from the observed coefficient.

We won't work through the details other than to note a few important aspects:

- When working with Pearson's correlation coefficient, the 'no effect' null hypothesis corresponds to one of zero (linear) association between the two variables ($r = 0$).
- The test statistic associated with $r$ turns out to be a *t*-statistic. This has nothing to do with comparing means—*t*-statistics pop up all the time in frequentist statistics.

Like any parametric technique, Pearson's correlation test makes several assumptions. These need to be met in order for the statistical test to be reliable. The assumptions are:

1. Both variables are measured on an interval or ratio scale.
2. The two variables are normally distributed (in the population).
3. The relationship between the variables is linear.

The first two requirements should not need any further explanation at this point—we've seen them before in the context of the one- and two-sample *t*-tests. The third one stems from Pearson's correlation coefficient being a measure of linear association.

Only the linearity assumption needs to be met for Pearson's correlation coefficient ($r$) to be a valid measure of association. As long as the relationship between two variables is linear, $r$ produces a sensible measure of association. However, the first two assumptions need to be met for the associated statistical test to be appropriate.

That's enough background and abstract concepts. Let's see how to perform correlation analysis in R using Pearson's correlation coefficient.

## 11.2 Pearson's product-moment correlation coefficient in R

> We will use a new data set to demonstrate correlation analysis. The data live in a file called 'ALZHEIMERS.CSV'. The code below assumes those data have been read into a tibble called `alzheimers`. Set that up if you plan to work along.

The plant morph example is not suitable for correlation analysis. We need a new example to review the workflow for correlation tests in R. The example we're going use is about the association a plasma biomarker with cognitive impairment in patients with Alzheimer's disease.

In Alzheimer's disease (AD) patients, brain degeneration is associated with the accumulation of tau protein tangles in the brain. Studies have shown that total tau (t-Tau) levels are higher in the plasma of AD patients than in healthy subjects. A group of biomedical researchers are interested in using plasma t-Tau to assess the extent of AD progression. They have designed a sensitive assay to measure plasma t-Tau. The team recruits 40 AD patients to a pilot study designed to evaluate the new assay. Each patient is undergoes a mini-mental state examination (MMSE) to assess the degree of cognitive impairment (defined as score $< 26$). Their t-Tau levels are then measured using the new assay.

### 11.2.1 Visualising the data and checking the assumptions

We can start by printing some information about the `alzheimers` data:

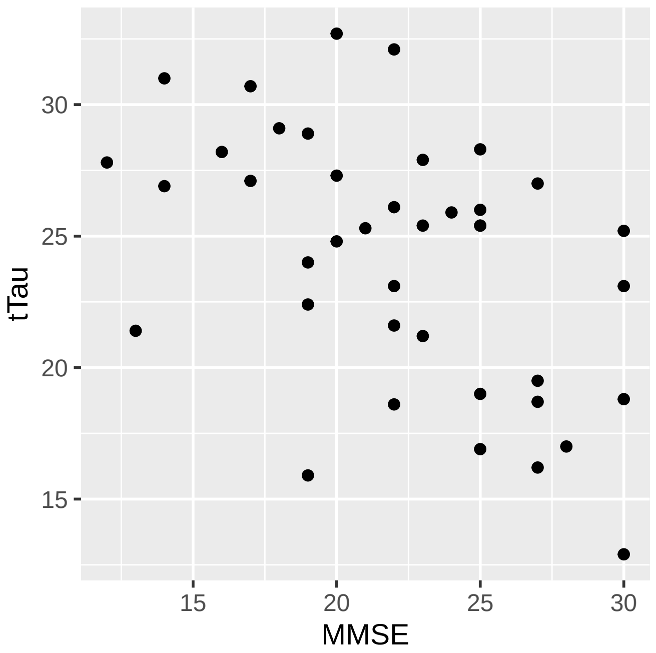
```
glimpse(alzheimers)
```

```
## Rows: 40
## Columns: 2
## $ MMSE <dbl> 25, 20, 22, 14, 25, 22, 28, 23, 20, 16, 19, 30, 21, 12, 30, 19, 2~
## $ tTau <dbl> 19.0, 24.8, 23.1, 26.9, 26.0, 21.6, 17.0, 27.9, 32.7, 28.2, 15.9,~
```

There are 40 observations (rows) and two variables (columns) in this data set. The two variables, `MMSE` and `tTau`, contain patient's MMSE scores and t-Tau protein concentrations, respectively.

Always explore the data thoroughly before carrying out a statistical analysis. To begin, we can visualise the form of the association with a scatter plot:

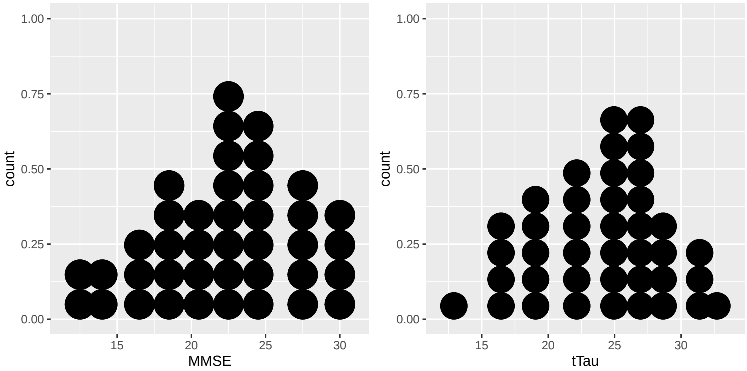
```
ggplot(alzheimers, aes(x = MMSE, y = tTau)) +
  geom_point()
```



There appears to be a negative association between cognitive performance and the assayed t-Tau levels. The relationship seems to follow a 'straight line' pattern, meaning Pearson's correlation may be a reasonable measure of association for these data (assumptions #3).

Before confirming the association with a Pearson correlation test, we should determine if it is appropriate to carry out the planned test. We are working with numeric variables measured in ratio scale—that's assumption #1 dealt with. What about their distributions (assumptions #2)? Here's a quick visual summary:

```
ggplot(alzheimers, aes(x = MMSE)) +
  geom_dotplot(binwidth = 2)
ggplot(alzheimers, aes(x = tTau)) +
  geom_dotplot(binwidth = 2)
```



It is impossible to be 100% sure with such a small data set, but these plots suggest there is no obvious problem with the normality assumption—both distributions are roughly ‘bell-shaped’.

That’s all three assumptions met: 1) the variables are on a ratio scale, 2) the normality assumption is met, and 3) the abundance relationship is linear. It looks like the statistical test should give reliable results.

### 11.2.2 Doing the test

Let’s proceed with the correlation analysis. We use the `cor.test` function to do this in R:

```
cor.test(~ MMSE + tTau, method = "pearson", data = alzheimers)
```

We have suppressed the output here to focus on how the function works:

- We use the R formula syntax to determine which pair of variables are analysed. The `cor.test` function expects the two variable names to appear to the right of the `~`, separated by a `+` symbol, with nothing on the left.
- We use `method = "pearson"` to control which type of correlation coefficient was calculated. The default method is Pearson’s correlation, but it never hurts to be explicit, so we wrote `method = "pearson"` anyway.
- When we use the formula syntax, as we are doing here, we have to tell the function where to find the variables. That’s the `data = alzheimers` part.

Notice that `cor.test` uses a different convention from `t.test` to specify the formula. `t.test` places one variable on the left and the other variable on the right-hand side of the `~` (e.g. `MMSE ~ tTau`), whereas `cor.test` requires both two variables to appear to the right of the `~`, separated by a `+` symbol. This convention exists to emphasise the absence of ‘directionality’ in correlations—neither variable has a special status.

The output from `cor.test` is, however, very similar to that produced by the `t.test` function. Here it is:

```
##
##  Pearson's product-moment correlation
##
## data:  MMSE and tTau
## t = -3.4685, df = 38, p-value = 0.001317
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.6956181 -0.2111049
## sample estimates:
##        cor
## -0.4903671
```

We won't step through most of this as its meaning should be clear. The `t = -3.4685, df = 38, p-value = 0.001317` line is the one we really care about:

- The first part says that the test statistic associated with the test is a *t*-statistic, where `t = -3.4685`. Remember, this has nothing to do with 'comparing means'.
- Next, we see the degrees of freedom for the test. Can you see where this comes from? It is $n-2$, where $n$ is the sample size. R uses the degrees of freedom and the *t*-statistic determine the *p*-value.
- The *p*-value is generated under the null hypothesis of zero correlation ($r = 0$). Since $p < 0.05$, we conclude that there is a statistically significant negative correlation between t-Tau levels and MMSE.

What is the correlation between t-Tau levels and MMSE? That's given at the bottom of the test output: $-0.49$. As expected from the scatter plot, there is a moderate negative association between the two measurements.

### 11.2.3 Reporting the result

When using Pearson's method, we should report the value of the correlation coefficient, the sample size, and the *p*-value[2]. Here's how to report the results of this analysis:

> There is a negative correlation between t-Tau and MMSE scores among patients (r=-0.49, n=38, p < 0.01).

Notice that we stated the direction of the association. It is important to do that because both the direction and strength of the association are relevant to the clinical context. As hoped, elevated plasma t-Tau levels determined by the new assay are associated with reduced cognitive performance.

[2] Some people like to report the the *t*-statistic alongside the correlation coefficient, the degrees of freedom, and the *p*-value. There's no need to do this though because the *t*-statistic is calculated directly from the correlation coefficient and the degrees of freedom.

## 11.3 Next steps

Our correlation analysis only characterises the strength and direction of the association between t-Tau and MMSE scores. We would need to use a different kind of analysis to say anything about how MMSE scores change with t-Tau concentration. That's the focus of our next topic: simple linear regression.

# Chapter 12

# Relationships and regression

## 12.1 Introduction

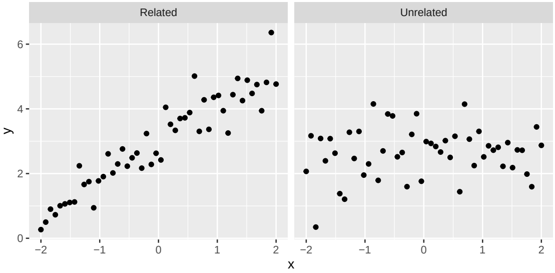
Much of biology is concerned with relationships between numeric variables. For example:

- We sample fish and measure their length and weight because we want to understand how weight changes with length.
- We survey grassland plots and measure soil pH and species diversity to understand how species diversity depends on soil pH.
- We manipulate temperature and measure fitness in insects because we want to describe their thermal tolerance.

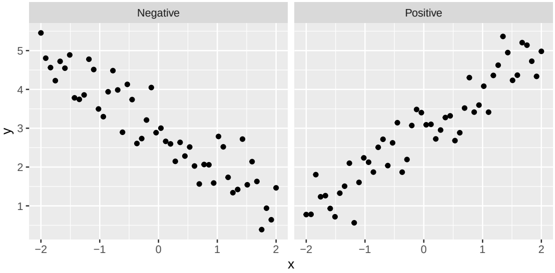
In the previous chapter, we learnt about one technique for analysing associations between numeric variables (correlation). A correlation coefficient quantifies the strength and direction of an association between two variables: it will be close to zero if there is no association between the variables, whereas a strong association is implied if the coefficient is near -1 or +1. A correlation coefficient tells us nothing about the form of a relationship. Nor does it allow us to predict the value of one variable from the value of a second variable.

In contrast to correlation, a regression analysis allows us to make precise statements about how one numeric variable depends on the values of another. Graphically, we can evaluate such dependencies using a scatter plot. We may be interested in knowing:

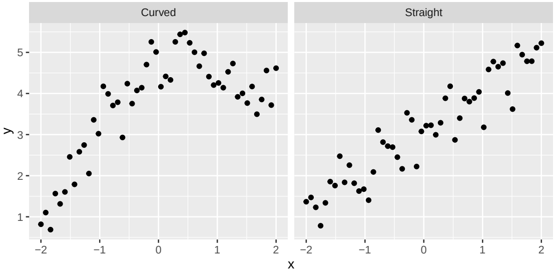
1. Are the variables related or not? There's not much point in studying a relationship that isn't there:



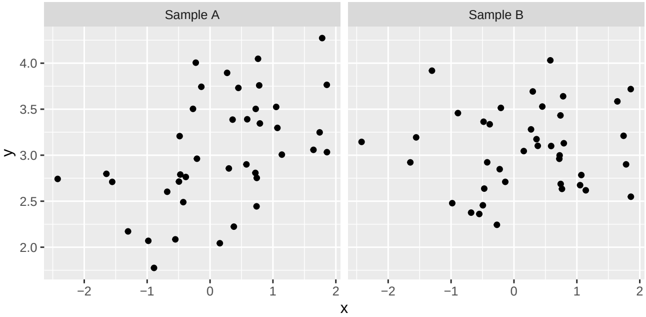
2. Is the relationship positive or negative? Sometimes we can answer a scientific question just by knowing the direction of a relationship:



3. Is the relationship a straight line or a curve? It is important to know the form of a relationship if we want to make predictions:



Although sometimes it may be obvious that there is a relationship between two variables from a plot of one against the other, at other times it may not. Take a look at the following:



We might not be very confident in judging which, if either, of these plots provides evidence of a positive relationship between the two variables. Maybe the pattern we perceive can just be explained by sampling variation, or perhaps it can't. We need a procedure—a statistical test—to evaluate how likely the relationship could have arisen as a result of sampling variation. In addition to judging the statistical significance of a relationship, we may also be interested in describing the relationship mathematically – i.e. finding the equation of the best fitting line through the data.

A linear regression analysis allows us to do all this.

## 12.2 What does linear regression do?

**Simple linear regression** allows us to *predict* how one variable (the **response variable**) *responds* or *depends* on to another (the **predictor variable**), assuming a straight-line relationship.

- What does the word 'simple' mean here? Simple linear regression is a regression model which only accounts for one predictor variable. If more than one predictor variable is considered, the correct term to describe the resulting model is 'multiple regression'. Multiple regression is a very useful tool, but we're only going to study simple regression in this book.
- What does the word 'linear' mean here? In statistics, the word linear is used in two slightly different but closely related ways. When discussing simple linear regression, the term linear is often understood to mean that the relationship follows a straight line. That's all. The more technical definition concerns the relationship between the parameters of a statistical model. We don't need to worry about that one here.

Writing 'simple linear regression' all the time becomes tedious, so we'll often write 'linear regression' or 'regression'. Just keep in mind that we're always referring to simple linear regression in this book. These regression models account for a straight-line relationship between two numeric variables, i.e. they describe how the response variable changes in response to the values of the predictor variable. It is conventional to label the response variable as '$y$' and the predictor variable as '$x$'. When we present such data graphically, the response variable always goes on the $y$-axis and the predictor variable on the $x$-axis. Try not to forget this convention!

**'response / predictor' or 'dependent / independent'?**

Another way to describe linear regression is that it finds the straight-line relationship which best describes the dependence of one variable (the **dependent variable**) on the other (the **independent variable**). The dependent vs. independent and response vs. predictor conventions for variables in a regression are equivalent. They only differ in the words they use to describe the variables involved.

To avoid confusion, we will stick with response / predictor naming convention in this book.

How do we decide how to select which is to be used as the response variable and which as the predictor variable? The decision is straightforward in an experimental setting: the manipulated variable is the predictor variable, and the measured outcome is the response variable. Consider the thermal tolerance example from earlier. The temperature was manipulated in that experiment, so it must be designated the predictor variable. Moreover, *a priori* (before conducting the experiment), we can reasonably suppose that temperature changes may cause changes in enzyme activity, but the reverse seems pretty unlikely.

Things are not so clear cut when working with data from an observational study. Indeed, the 'response vs predictor' naming convention can lead to confusion because the term 'response variable' tends to make people think in terms of causal relationships, i.e. that variation in the the predictor somehow *causes* changes in the response variable. Sometimes that is true, such as in the experimental setting described above. However, deciding to call one variable a response and the other a predictor should not be taken to automatically imply a casual relationship between them.

Finally, it matters which way round we designate the response and predictor variable in a regression analysis. Suppose we have two variables A and B. In that case, the relationship we find from a regression will not be the same for A against B as for B against A. Choosing which variable to designate as the response boils down to working out which of the variables needs to be explained (response) in terms of the other (predictor).

## 12.3 How does simple linear regression work?

### 12.3.1 Finding the best fit line

If we draw a straight line through a set of points on a graph then, unless they form a perfect straight line, some points will lie close to the line and others further away. The vertical distances between the line and each point (i.e. measured parallel to the $y$-axis) have a special name. They are called the *residuals*. Here's a visual example:

In this plot the blue points are the data and the vertical lines represent the residuals. The residuals represent the 'left over' variation after the line has been fitted through the data. They indicate how well the line fits the data. If all the

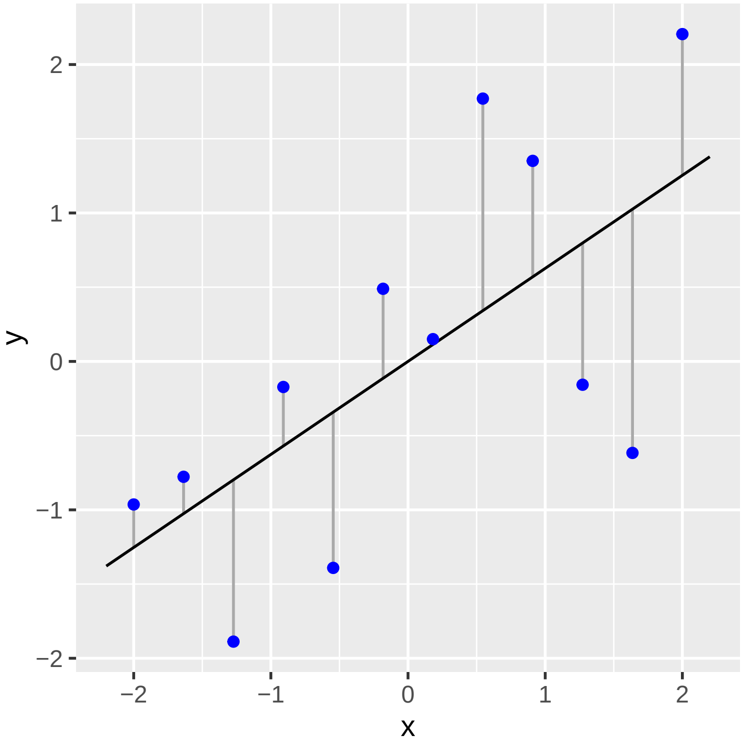
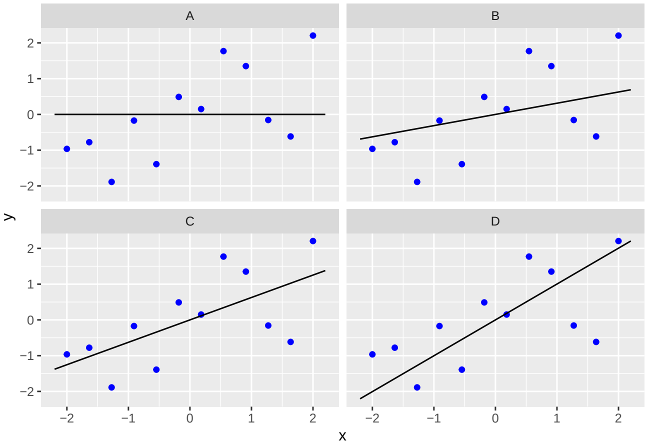


Figure 12.1: Example of data (blue points) used in a simple regression. A fitted line and the associated residuals (vertical lines) are also shown

points lay close to the line, the variability of the residuals would be low relative to the overall variation in the response variable, $y$. When the observations are more scattered around the line, the variability of the residuals would be large relative to the variation in the response variable, $y$.

Regression works by finding the line which minimises the size of the residuals in some sense. We'll explain exactly how in a moment. The following illustration indicates the principle of this process:

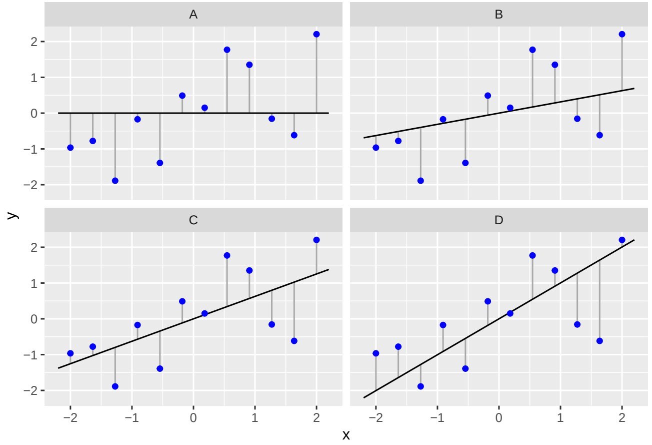


The data are identical in all four graphs, but in the top left-hand graph a horizontal line (i.e. no effect of $x$ on $y$) has been fitted, while on the remaining three graphs sloping lines of different magnitude have been fitted.

**Which line is best?**

One of the four lines is the 'line of best' fit from a regression analysis. Spend a few moments looking at the four figures. Which line seems to fit the data best? Why do you think this line is 'best'?

Let's visualise the data, the candidate lines and the residuals:



We said that regression works by finding the intercept and slope that minimises the vertical distances between the line and each observation in some way[1]. In fact, it minimises something called the 'sum of squares' of these distances: we calculate a sum of squares for a particular set of observations and a fitted line by squaring the residual distances and adding all of these up. This quantity is called the **residual sum of squares**. The line with the *lowest* residual sum of squares is the best line because it 'explains' the most variation in the response variable.

You should be able to see that, for the horizontal line ('A'), the residual sum of squares is larger than any of the other three plots with the sloping lines. This suggests that the sloping lines fit the data better. Which one is best among the three we've plotted? To get at this, we need to calculate the residual sum of squares for each line:

```
##   Line    Residual Sum of Squares
## 1    A                   17.55067
## 2    B                   11.97966
## 3    C                   10.12265
## 4    D                   12.79674
```

So it looks like the line in panel C is the best fitting line among the candidates. In fact, it is the best fit line among all possible candidates. Did you manage to guess this by looking at the lines and the raw data? If not, think about why you got the answer wrong. Did you consider the vertical distances or the perpendicular distances?

[1] Notice that it is the vertical distance that matters, not the perpendicular distance from the line.

**Know your residuals**

It is important to understand what a residual represents. Why? Because they pop up all the time when working with statistical models (not only regression, in fact). It is hard to understand what R is telling you about a model without knowing about these residual things.

## 12.4 What do you get out of a regression?

A regression analysis involves two activities:

- **Interpretation.** When we 'fit' a regression model to data we estimate the coefficients of a best-fit straight line through the data. This is the equation that best describes how the $y$ (response) variable *responds to* the $x$ (predictor) variable. To put it in slightly more technical terms, it describes the $y$ variable as a function of the $x$ variable. This model may be used to understand how the variables are related or make predictions.

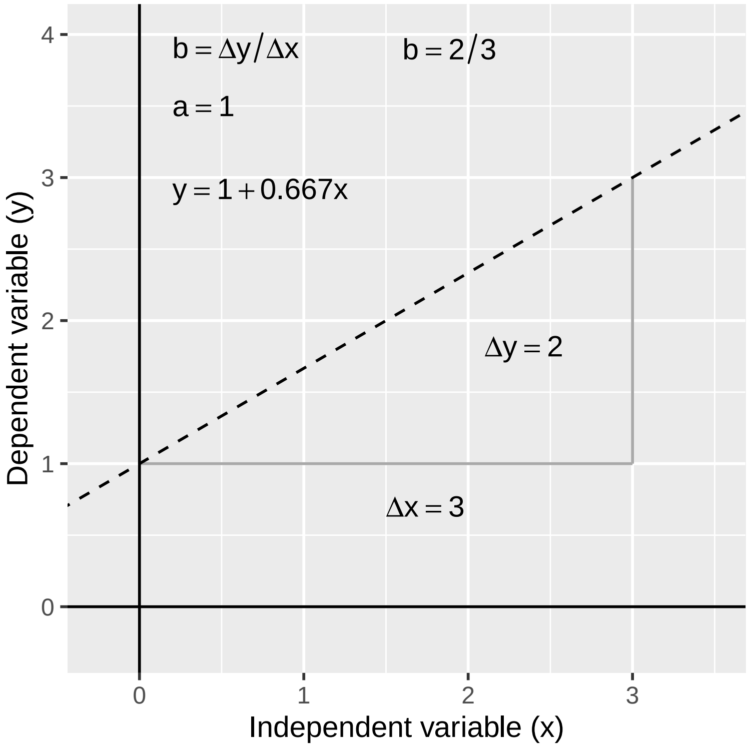
- **Inference.** It is not enough to just estimate the regression equation. Before we can use it, we need to determine whether there is a statistically significant relationship between the $x$ and $y$ variables. That is, the analysis will tell us whether an apparent association is likely to be real or just a chance outcome resulting from sampling variation.

### 12.4.1 Interpreting a regression

What is the form of the relationship? The equation for a straight-line relationship is $y = a + b \times x$, where

- $y$ is the response variable,
- $x$ is the predictor variable,
- $a$ is the intercept (i.e. where the line crosses the $y$ axis), and
- $b$ is the slope of the line.

The $a$ and the $b$ are referred to as the *coefficients* (or *parameters*) of the line. The slope of the line is often the coefficient we care about most. It tells us the amount by which $y$ changes for a change of one unit in $x$. If the value of $b$ is positive (i.e. a plus sign in the above equation) this means the line slopes upwards to the right. A negative slope ($y = a - bx$) means the line slopes downwards to the right. The diagram below shows the derivation of an equation for a straight line.



Having the equation for a relationship allows us to predict the value of the $y$ variable for any value of $x$. For example, in the thermal tolerance example, we want an equation that will allow us to work out how fitness changes with temperature. Such predictions can be made by hand (see below) or using R (details later).

In the above diagram, the regression equation is: $y = 1 + 0.66x$. So to find the value of $y$ at $x = 2$ we use: $y = 1 + (0.667 \times 2) = 2.32$. Obviously, by finding $y$ values for 2 (or preferably 3) different $x$ values from the equation, the actual line can easily be plotted on a graph manually if required—plot the values and join the dots! It's much easier to use R to do this kind of thing though.

**Regression involves a statistical model**

A simple linear regression is underpinned by a statistical model. If you skim back through the parametric statistics chapter, you will see that the equation $y = a + b \times x$ represents the 'systematic component' of the regression model. This bit describes the component of variation in $y$ that is explained by the model for the dependence of $y$ on $x$. The residuals correspond to the 'random component' of the model. These represent the component of variation in the $y$ variable that our regression model fails to describe.

### 12.4.2 Evaluating hypotheses ('inference')

More than one kind of significance test can be carried out with a simple linear regression. We're going to focus on the most common test: an $F$ test of whether the slope coefficient is significantly different from 0. This addresses the important question, "Is there a relationship?"

How do we do this? We play exactly the same kind of gambit we used to develop the earlier significance tests:

1. We start with a null hypothesis of 'no effect'. This corresponds to the hypothesis that the slope of the regression is zero.

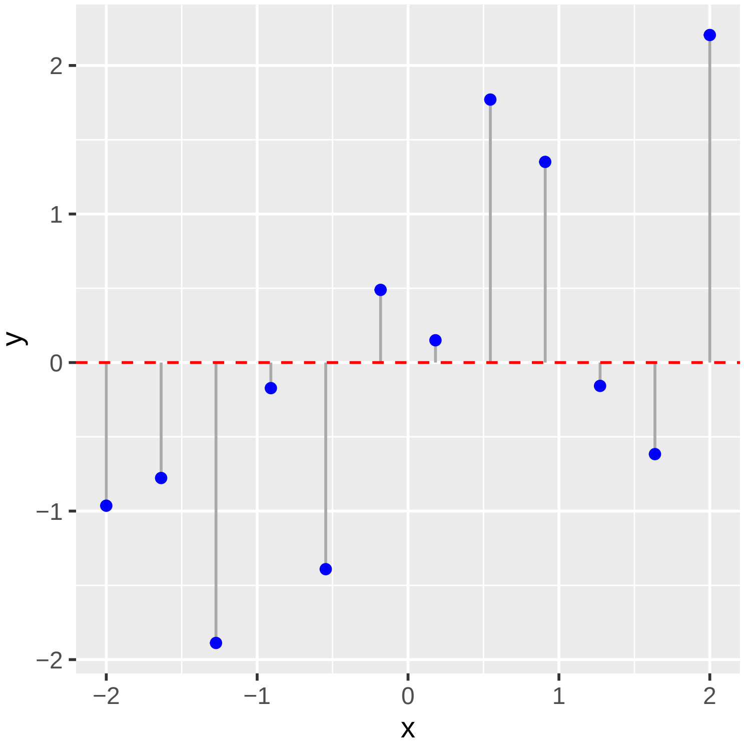
2. We then work out what the distribution of some kind of test statistic should look like under the null hypothesis. The test statistic in this case is called the $F$-ratio.

3. We then calculate a $p$-value by asking how likely it is that we would see the observed test statistic, or a more extreme value, if the null hypothesis were really true.

It's not so critical that someone understands the mechanics of an $F$-test to use it. However, knowing a bit about where is comes from does help to demystify the output produced by R. To that end, let's step through the calculations involved in the $F$ test using the example data shown in the above four-panel plot.

#### Total variation

We first need to calculate something called the **total sum of squares**. The figure below shows the raw data (blue points) and the grand mean (i.e. the sample mean).

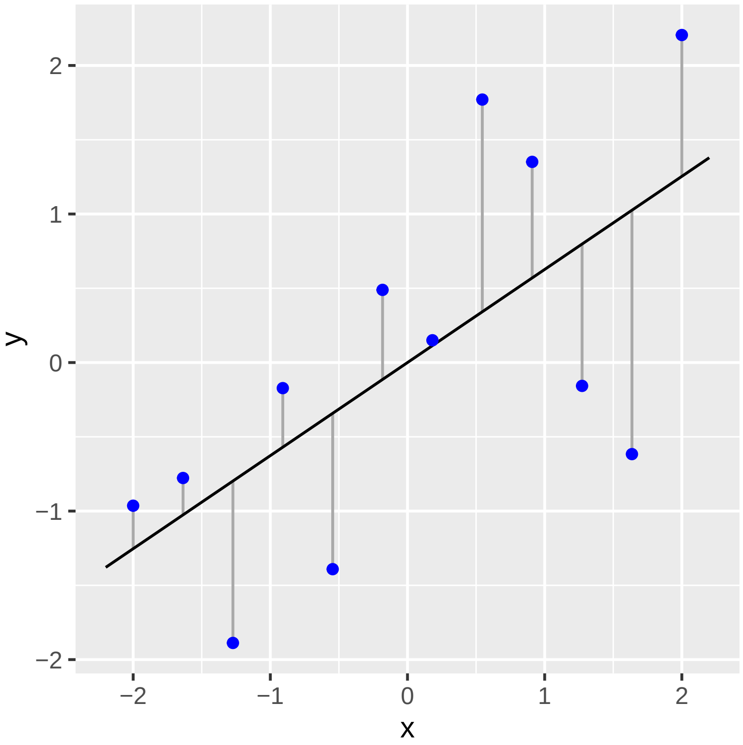


The vertical lines show the distance between each observation and the grand mean. These vertical lines are just the residuals from a model where the slope of the line is set to zero. What we need to do is quantify the variability of these residuals. We can't just add them up, because by definition, they have to sum to zero, i.e. they are calculated relative to the grand mean.

Instead, we calculate the total sum of squares by taking each residual in turn, squaring it, and then adding up all the squared values. We call this the total sum of squares because it is a measure of the total variability in the response variable, $y$. This number is 17.55 for the data in the figure above.

### Residual variation

Next we need to calculate the **residual sum of squares**. We have already seen how this calculation works because it is used in the calculation of the best fit line—the best fit line is the one that minimises the residual sum of squares. Let's plot this line along with the associated residuals of this line again:



The vertical lines show the distance between each observation and the best fit line. We need to quantify the variability of these residuals. Again, we can't just add up the deviations because they have to sum to zero as a result of how the best fit line is found. Instead we calculate the residual sum of squares by taking each residual in turn, squaring it, and then adding up all the squared values. This number is 10.12 for the figure above. We call this the residual, or error, sum of squares because it is a measure of the variation in $y$ that is 'left over' after accounting for the influence of the predictor variable $x$.

**Explained variation**

Once the total sum of squares and the residual sum of squares are known, we can calculate the quantity we really want: the **explained sum of squares**. This is a measure of the variation in $y$ that is explained by the influence of the predictor variable $x$. We calculate this by subtracting the residual sum of squares from the total sum of squares. This makes intuitive sense: if we subtract the variation in $y$ we can't explain (residual) from all the variation in $y$ (total), we end up with the amount 'explained' by the regression. This number is 7.43 for the example.

### Degrees of freedom, mean squares and *F* tests

The problem with sums of squares is that they are a function of sample size. The more data we have, the larger our sum of squares will get. The solution to this problem is to convert them into a measure of variability that doesn't scale with sample size. We need to calculate **degrees of freedom** (written as df, or d.f.) to do this. We came across the concept of degrees of freedom when we studied the *t*-test. The idea is closely related to sample size. It is difficult to give a precise definition, but roughly speaking, the degrees of freedom of a statistic is a measures of how much information it is based on (bigger is better).

Each of the measures of variability we just calculated for the simple linear regression has a degrees of freedom associated with it. We need the explained and error degrees of freedom:

- Explained d.f. = 1
- Error d.f. = (Number of observations - 2)

Don't worry if those seem a little cryptic. We don't need to carry out degrees of freedom calculations by hand because R will do them for us. We'll think about degrees of freedom a bit more when we start to learn about ANOVA models.

The reason degrees of freedom matter is because we can use them to standardise the sum of squares to account for sample size. The calculations are very simple. We take each sum of squares and divide it by its associated degrees of freedom. The resulting quantity is called a **mean square** (it's the mean of squared deviations):

$$\text{Mean Square} = \frac{\text{Sum of Squares}}{\text{Degrees of Freedom}}$$

A mean square is actually an estimate of variance. Remember the variance? It is one of the standard measures of a distribution's dispersion, or spread.

Now for the important bit. The two mean squares can be compared by calculating the ratio between them, which is designated by the letter *F*:

$$F = \text{Variance Ratio} = \frac{\text{Explained Mean Square}}{\text{Residual Mean Square}}$$

This is called the *F* ratio, or sometimes, the variance ratio. If the explained variation is large compared to the residual variation, then the *F* ratio will be large. Conversely, if the explained variation is relatively small the *F* ratio will be small. We can see where this is heading…

The *F* ratio is a type of test statistic—if the value of *F* is sufficiently large, we judge it to be statistically significant. For this judgement to be valid, we have to make one key assumption about the population from which the data has been sampled: we assume the residuals are drawn from a normal distribution.

If this assumption is correct, it can be shown that the distribution of the $F$ ratio under the null hypothesis (the 'null distribution') has a particular form: it follows a theoretical distribution called an $F$-distribution. And yes, that's why the variance ratio is called '$F$'.

All this means we can assess statistical significance of the slope coefficient by comparing the $F$ ratio calculated from a sample to this theoretical distribution. This procedure is called an $F$ test. The $F$ ratio is 7.34 in our example. This is quite high, which indicates that the slope is likely to be significantly different from 0.

We need more than an $F$ ratio to calculate a $p$-value. We also need to consider the degrees of freedom of the test. Furthermore, because this test involves an $F$ ratio, **there are two different degrees of freedom to consider**: the explained and residual df's. Try to remember that fact—$F$ ratio tests involve a pair of degrees of freedom.

We could calculate a $p$-value now by messing around with tables of $F$ ratios and degrees of freedom. However, it is much easier to let R do this for us when working with a regression model. That's what we're going to do in the next chapter, so for now, we'll leave significance tests alone.

## 12.5 Correlation or regression?

Before we finish up, it is worth pausing to review the difference between regression and correlation analyses. Whilst regression and correlation are both concerned with associations between numeric variables, they are different techniques and each is appropriate under distinct circumstances. This is a frequent source of confusion. Which technique is required for a particular analysis depends on

- the way the data were collected and
- the goal of the analysis.

There are two broad questions to consider—

#### Where do the data come from?

Think about how the data have been collected. If they are from a study where one of the variables has been experimentally manipulated, then choosing the best analysis is easy. We should use a regression analysis, in which the predictor variable is the experimentally manipulated variable and the response variable is the measured outcome. The fitted line from a regression analysis describes how the outcome variable *depends on* the manipulated variable—it describes the causal relationship between them.

It is generally inappropriate to use correlation to analyse data from an experimental setting. A correlation analysis examines association but does not imply the dependence of one variable on another. Since there is no distinction of response or predictor variables, it doesn't matter which way round we do a correlation. In fact, the phrase 'which way round' doesn't even make sense in the context of a correlation.

If the data are from a so-called 'observational study' where someone took measurements, but nothing was actually manipulated experimentally, then either method may be appropriate.

### What is the goal of the analysis?

Think about what question is being asked. A correlation coefficient only quantifies the strength and direction of an association between variables—it tells us nothing about the form of a relationship. Nor does it allow us to make predictions about the value of one variable from the values of a second. A regression does allow this because it involves fitting a line through the data—i.e. regression involves **a model** for the relationship.

This means that if the goal is to understand the form of a relationship between two variables, or to use a fitted model to make predictions, we have to use regression. If we just want to know whether two variables are associated or not, the direction of the association, and whether the association is strong or weak, then a correlation analysis is sufficient. It is better to use a correlation analysis when the extra information produced by a regression is not needed, because the former will be simpler and potentially more robust.

# Chapter 13

# Simple regression in R

Our goal in this chapter is to learn how to work with regression models in R. We'll start with the example problem and the data, and then work through model fitting, significance testing, and finally, presenting the results.

## 13.1 Introduction

A plant physiologist studying the process of germination in the broad bean (*Vicia faba*) is interested in the relationship between the activity of the enzyme amylase and the temperature at which the germinating beans are kept. As part of this work, she carries out an experiment to find the relationship between glucose release (from the breakdown of starch by amylase) and temperature (over the range 2 - 20C). The data obtained from such an experiment are given below.

| Temperature ($C$) | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|
| Glucose ($\mu g\ mg^{-1}$ dry weight) | 1.0 | 0.5 | 2.5 | 1.5 | 3.2 | 4.3 | 2.5 | 3.5 | 2.8 | 5.6 |

We want to work out whether there a statistically significant relationship between temperature and glucose release (and hence, presumably, amylase activity). That's a job for linear regression.

⚡

We will be using this new data set to demonstrate how to conduct simple regression in R. The data live in the 'GLUCOSE.CSV' file. The code below assumes those data have been read into a tibble called `vicia_germ`. Set that up if you plan to work along.

### 13.1.1 First steps

We start by using `glimpse` to see what resides within `vicia_germ`:

```
glimpse(vicia_germ)
```

```
## Rows: 10
## Columns: 2
## $ Temperature <dbl> 2, 4, 6, 8, 10, 12, 14, 16, 18, 20
## $ Glucose     <dbl> 1.0, 0.5, 2.5, 1.5, 3.2, 4.3, 2.5, 3.5, 2.8, 5.6
```

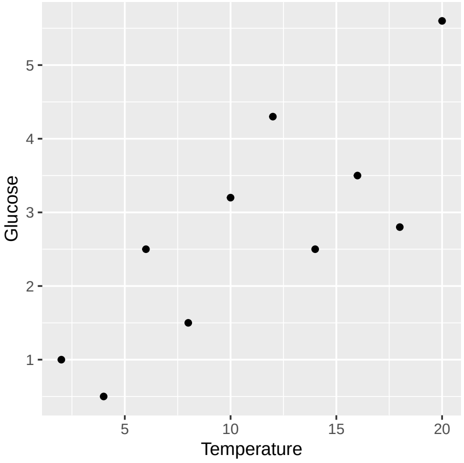
There are just two numeric variables. The first column (`Temperature`) contains the information about the experimental temperature treatments, and the second column (`Glucose`) contain the glucose measurements.

Notice that we refer to the different temperatures as 'experimental treatments'. This is because these data are from an experiment where temperature was controlled by the investigator. We'll discuss this terminology in more detail in the Principles of Experimental Design chapter.

### 13.1.2 Visualising the data

We should visualise the data next so that we understand it more. A simple scatter plot will do:

```
ggplot(vicia_germ, aes(x = Temperature, y = Glucose)) +
  geom_point()
```



Remember, `Glucose` is the response variable and `Temperature` is the predictor variable, so they belong on the $y$ and $x$ axes, respectively.

**Variables and axes**

Be careful when you produce a scatter plot to summarise data in a regression analysis. You need to make sure the two variables are plotted the right way around with respect to the $x$ and $y$ axes: place the response variable on the $y$ axis and the predictor on the $x$ axis. Nothing says, "I don't know what I'm doing," quite like mixing up the axes.

Because linear regression involves fitting a straight line through our data, it only makes sense to fit this model if the relationship between the two variables is linear. Plotting our data lets us see whether or not there appears to be a linear relationship. The scatter plot we produced above suggests that, in this case, the relationship between $x$ and $y$ is linear.

**Assumptions**

A linear relationship between the predictor and response variables is not the only assumption that we have to make when we fit a linear regression. We'll come back to the other assumptions and how to check whether they have been met in the Assumptions and Diagnostics chapters, respectively. For now you should just be aware that there are assumptions that you should check when working with your own data.

## 13.2 Model fitting and significance tests

Carrying out a regression analysis in R is a two step process.

The first step involves a process known as *fitting the model* (or just *model fitting*). In effect, this is the step where R calculates the best fit line, along with a large amount of additional information needed to generate the results in step two. We call this step model fitting because, well, we end up fitting the straight line model to the data.

How do we fit a linear regression model in R? We will do it using the `lm` function. The letters 'lm' in this function name stand for 'linear model'. We won't say much more at this point other than point out that a linear regression is a particular case of a general linear model. R doesn't have a special regression function.

Here is how we fit a linear regression in R using the enzyme data:

```
vicia_model <- lm(Glucose ~ Temperature, data = vicia_germ)
```

This should look quite familiar. We have to assign two arguments:

1. The first argument is a **formula**. We know this because it includes a 'tilde' symbol: `~`. The variable name on the left of the `~` should be the response variable. The variable name on the right should be the predictor variable. These are `Glucose` and `Temperature`, respectively. Make sure you get these the right way round when carrying out regression.

2. The second argument is the name of the data frame or tibble object that contains the two variables listed in the formula (`vicia_germ`).

### How does R knows we want to carry out a regression?

How does R know we want to use regression? After all, we didn't specify this anywhere. The answer is that R looks at what type of variable `Temperature` is. It is numeric, and so R automatically carries out a regression. If it had been a factor or a character vector (representing a categorical variable), R would have carried out a different kind of analysis, called a one-way Analysis of Variance (ANOVA). Many of the models we examine in this book are very similar and can be fitted using the `lm` function. The only thing that really distinguishes them is the type of variables that appear to the right of the `~` in a formula: if they are categorical variables, we end up fitting an ANOVA model, while numeric variables lead to a regression.

The key message is that you have to keep a close eye on the type of variables you are modelling to understand what kind of model R will fit.

Notice that we did not print the results to the console. Instead, we assigned the result a name (`vicia_model`). This now refers to a fitted **model object**. What happens if we print a regression model object to the console?

```
print(vicia_model)
```

```
##
## Call:
## lm(formula = Glucose ~ Temperature, data = vicia_germ)
##
## Coefficients:
## (Intercept)  Temperature
##      0.5200       0.2018
```

This prints a quick summary of the model we fitted and some information about the 'coefficients' of the model. The coefficients are the intercept and slope of the fitted line: the intercept is always labelled `(Intercept)` and the slope is labelled with the name of the predictor variable (`Temperature` in this case). We'll come back to these coefficients once we have looked at how to compute *p*-values.

The second step of a regression analysis involves using the fitted model to assess statistical significance. We usually want to determine whether the slope is significantly different from zero. That is, we want to know if the relationship between the $x$ and $y$ variables is likely to be real or just the result of sampling variation. Carrying out the required $F$ test is actually very easy. The test relies on a function called `anova`. To use this function, all we have to do is pass it one argument—the name of the fitted regression model object:

```
anova(vicia_model)
```

```
## Analysis of Variance Table
##
## Response: Glucose
##             Df  Sum Sq Mean Sq F value   Pr(>F)
## Temperature  1 13.4411 13.4411  14.032 0.005657 **
## Residuals    8  7.6629  0.9579
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Let's step through the output to see what it means. The first line informs us that we are looking at an Analysis of Variance Table—a set of statistical results derived from a general tool called Analysis of Variance. The second line reminds us what response variable we analysed (`Glucose`). Those parts are simple to describe at least, though the Analysis of Variance reference may seem a little cryptic. In a nutshell, every time we carry out an $F$-test we are performing some kind of Analysis of Variance because the test boils down to a ratio of two variances.

The important part of the output is the table at the end:

```
##             Df  Sum Sq Mean Sq F value   Pr(>F)
## Temperature  1 13.4411 13.4411  14.032 0.005657 **
## Residuals    8  7.6629  0.9579
```

This summarises the different parts of the $F$-test calculations: `Df` – degrees of freedom, `Sum Sq` – the sum of squares, `Mean Sq` – the mean square, `F value` – the $F$-statistic, `Pr(>F)` – the p-value. These were touched on in the last chapter.

The $F$-statistic (variance ratio) is the key term. When working with a regression model, this quantifies how much variability in the data is explained when we include the best fit slope term in the model. Larger values indicate a stronger relationship between $x$ and $y$. The $p$-value gives the probability that the relationship could have arisen through sampling variation if there were no real association. As always, a $p$-value of less than 0.05 is taken as evidence that the relationship is real, i.e. the result is statistically significant.

We should also note down the two degrees of freedom given in the table because these will be needed when we report the results.

### 13.2.1 Extracting a little more information

There is a second function, called `summary`, that can be used to extract a little more information from the fitted regression model:

```
summary(vicia_model)
```

```
##
## Call:
## lm(formula = Glucose ~ Temperature, data = vicia_germ)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.35273 -0.77909 -0.08636  0.74227  1.35818
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.52000    0.66858   0.778  0.45909
## Temperature  0.20182    0.05388   3.746  0.00566 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9787 on 8 degrees of freedom
## Multiple R-squared:  0.6369, Adjusted R-squared:  0.5915
## F-statistic: 14.03 on 1 and 8 DF,  p-value: 0.005657
```

This is easiest to understand if we step through the constituent parts of the output. The first couple of lines just remind us about the model we fitted

```
## Call:
## lm(formula = Glucose ~ Temperature, data = vicia_germ)
```

The next couple of lines aren't really all that useful—they summarise some properties of the residuals–so we'll ignore these.

The next few lines comprise a table that summarises some useful information about the coefficients of the model (the intercept and slope):

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.52000    0.66858   0.778  0.45909
## Temperature  0.20182    0.05388   3.746  0.00566 **
```

The `Estimate` column shows us the estimated intercept and slope of the regression. We saw these earlier when we printed the fitted model object to the console.

Staying with this table, the next three columns (`Std. Error`, `t value` and `Pr(>|t|)`) show us the standard error associated with each coefficient, the corresponding $t$-statistics, and the $p$-values. Remember standard errors? These measure the uncertainty of the sampling distributions associated with various estimates from a sample. We discussed standard errors in the context of sample means. One can calculate a standard error for many different kinds of quantities, including the intercept and slope of a regression model.

We can use the standard errors to evaluate the significance of the coefficients via $t$-statistics. In this case, the $p$-values associated with these $t$-statistics indicate that the intercept is not significantly different from zero ($p$>0.05), but that the slope is significantly different from zero ($p$<0.01).

Notice that the $p$-value associated with the slope coefficient is the same as the one we found when we used the `anova` function. This is not a coincidence—`anova` and `summary` test the same thing when working with simple linear regression models. **This is not generally true for other models involving the `lm` function.**

The only other part of the output from summary that is of interest now is the line containing the `Multiple R-squared` value:

```
## Multiple R-squared:  0.6369, Adjusted R-squared:  0.5915
```

This shows the $R$-squared ($R^2$) of our model. It tells you what proportion (sometimes expressed as a percentage) of the variation in the data is accounted for by the fitted line. If $R^2 = 1$ the line passes through all the points on the graph (all the variation is accounted for) and if $R^2 \approx 0\%$ the line explains little or none of the variation in the data. The $R^2$ value here is 0.64. This is very respectable but still indicates that there are other sources of variation which remain unexplained by the line (e.g. differences between beans, inaccuracies in the assay technique)[1].

## 13.3 Presenting results

From the preceding analysis we can conclude:

> There is a significant positive relationship between the incubation temperature (°C) and glucose released ($\mu gmg^{-1}$ dry weight) in germinating bean seeds ($y = 0.52 + 0.20x$, F=14, d.f.=1,8, $p$<0.01).

Don't forget to quote both degrees of freedom in the result. These are obtained from the ANOVA table produced by `anova` and should be given as the slope degrees of freedom first (which is always 1), followed by the error degrees of freedom.

If the results are being presented only in the text, it is usually appropriate to specify the regression equation and the significance of the relationship. This allows the reader to see in which direction and how steep the relationship is and perhaps use the equation in further calculations. It may also be useful to give the units of measurement—though these should already be stated in the Methods. Often, however, we will want to present the results as a figure, showing the original data and the fitted regression line. In this case, most of the statistical detail can go in the figure legend instead.

### 13.3.1 Plotting the fitted line and the data

We already know how to make a scatter plot. The only new trick we need to learn is how to add the fitted line. Remember the output from the summary table—this gave us the intercept and slope of the best fit line. We could extract these (there is a function called `coef` that does this), and using our knowledge of the equation of a straight line, use them to then calculate a series of points on the fitted line. However, there is an easier way to do this using the `predict` function.

This next part on predictions is usually confusing at first. Don't worry! You may have to come back to it a few times before it sinks in. At first reading, try to focus on the logic of the calculations without worrying too much about the details.

[1] The `Adjusted R-squared:` value can be ignored in this analysis—it is used when doing a form of regression called *multiple regression*, in which there is more than one $x$ variable.

To use `predict`, we have to let R know the values of the predictor variable for which we want predictions. In the bean example, the temperature was varied from 2-20 °C, so it makes sense to predict glucose concentrations over this range. Therefore the first step in making predictions is to generate a sequence of values from 2 to 20, placing these inside a data frame:

```
pred_data <- data.frame(Temperature = seq(2, 20, length.out = 25))
```

This uses `seq` to make a sequence of 25 evenly spaced numbers from 2 to 20. Notice that we gave the sequence the exact same name as the predictor variable in the regression (`Temperature`). **This is important**: the name of the numeric sequence we plan to make predictions from has to match the name of the predictor variable in the fitted model object.

Once we have set up a data frame to predict from (`pred_data`) we are ready to use the `predict` function:

```
predict(vicia_model, pred_data)
```

```
##         1         2         3         4         5         6         7         8 
## 0.9236364 1.0750000 1.2263637 1.3777273 1.5290909 1.6804546 1.8318182 1.9831818 
##         9        10        11        12        13        14        15        16 
## 2.1345455 2.2859091 2.4372727 2.5886364 2.7400000 2.8913636 3.0427273 3.1940909 
##        17        18        19        20        21        22        23        24 
## 3.3454545 3.4968182 3.6481818 3.7995454 3.9509091 4.1022727 4.2536363 4.4050000 
##        25 
## 4.5563636
```

This take two arguments: the first is the name of the model object (`vicia_model`); the second is the data frame (`pred_data`) containing the values of the predictor variable at which we want to make predictions. The predict function generated the predicted values in a numeric vector and printed these to the console.

To be useful, we need to capture these somehow, and because we want to use **ggplot2**, these need to be kept inside a data frame. We can use `mutate` to do this:

```
pred_data <- mutate(pred_data, Glucose = predict(vicia_model, pred_data))
```

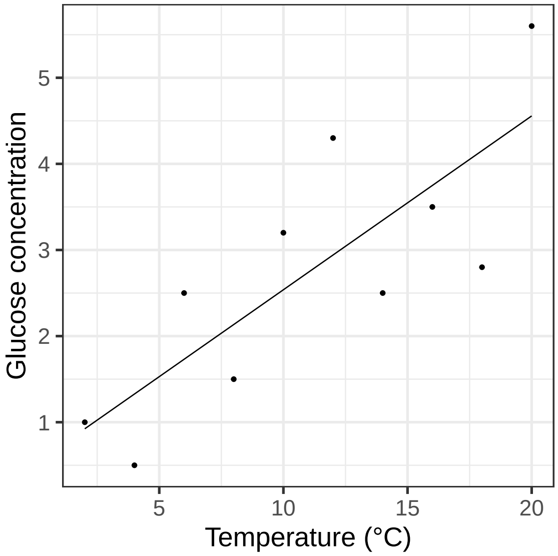
Look at the first 10 rows of the resulting data frame:

```
head(pred_data, 10)
```

```
##    Temperature   Glucose
## 1         2.00 0.9236364
## 2         2.75 1.0750000
## 3          3.50 1.2263637
## 4          4.25 1.3777273
## 5          5.00 1.5290909
## 6          5.75 1.6804546
## 7          6.50 1.8318182
## 8          7.25 1.9831818
## 9          8.00 2.1345455
## 10         8.75 2.2859091
```

The pred_data is set out much like the data frame containing the experimental data. It has two columns, called Glucose and Temperature, but instead of data, it contains predictions from the model. Plotting these predictions along with the data is now easy:

```
ggplot(pred_data, aes(x = Temperature, y = Glucose)) +
  geom_line() + geom_point(data = vicia_germ) +
  xlab("Temperature (°C)") + ylab("Glucose concentration") +
  theme_bw(base_size = 22)
```



Notice that we have to make **ggplot2** use the vicia_germ data (i.e. the raw data) when adding the points. We also threw in a little theming to make the plot look nicer.

Here is a quick summary of what we just did:

- Using seq and data.frame, we made a data frame with one column containing the values of the predictor variable we want predictions at.
- We then used the predict function to generate these predictions, adding them to the prediction data with mutate.

- Finally, we used **ggplot2** to plot the predicted values of the response variable against the predictor variable, remembering to include the data.

## 13.4 What about causation?

No discussion of regression would be complete without a little sermon on the fact that just because you observe a (significant) relationship between two variables, this does not necessarily mean that the two variables are causally linked. Suppose we find a negative relationship between the density of oligochaete worms (the response variable) and the density of trout (the predictor variable) in a sample of different streams. In that case, this need not indicate that the trout reduce the numbers of oligochaetes. Oligochaete numbers are often very high in slow-flowing, silty streams where they live in the sediments, whereas trout prefer faster flowing, well-oxygenated, stony streams. A negative correlation could occur simply for that reason. There are many situations in biology where a relationship between two variables can occur, not because there is a causal link between them but because each is related to a third variable (e.g. habitat).

This difficulty must always be considered when interpreting relationships between variables in data collected from non-experimental situations. However, it is often said that regression analysis can never be used to infer a causal link because of this problem. This is incorrect. What is important is how the data are generated, not the statistical model used to analyse them.

Imagine ten plants were randomly assigned to be grown under ten different light intensities, with all other conditions held constant. In those circumstances it would be entirely proper to analyse the effect of light level on plant height by a regression of plant height ($y$) against light level ($x$) and, if a significant positive straight-line relationship was found, to conclude that increased light level caused increased plant height. But the fact that we are experimentally producing an effect in plants randomly allocated to each light level is what gives us the confidence to draw a conclusion about causality. Light might not be the direct causal agent—maybe another factor (e.g. temperature) is varying along with light and causing the effect—but it must be at least indirectly related to plant growth because it was experimentally manipulated.

# Part V

# One-way ANOVA

# Chapter 14

# Introduction to one-way ANOVA

## 14.1 Introduction

A two-sample *t*-test evaluates whether the mean of a numeric variable changes among two groups. The obvious question is, what happens if we need to compare differences among the means of such a variable measured in more than two groups? It may be tempting to test every pair of means using a two-sample *t*-test. However, this procedure is statistically flawed. In this chapter, we will introduce a new method to assess the statistical significance of differences among several means simultaneously. This is called **Analysis of Variance** (abbreviated to ANOVA).

ANOVA is one of those statistical terms that unfortunately has two slightly different usages:

1. In its most general sense, ANOVA refers to a methodology for evaluating statistical significance. It often pops up when working with a statistical model known as the 'general linear model'. Simple linear regression is one special case of the general linear model[1].
2. In its more narrow sense, the term ANOVA is used to describe a particular type of statistical model. When used like this, ANOVA refers to a class of model used to compare means among two or more groups. The ANOVA-as-a-model is the focus of this chapter.

ANOVA models underpin the analysis of many different kinds of experimental data—they are one of the main 'work horses' of data analysis. As with many

[1] In fact, whenever we see an $F$-ratio in a statistical test it means we're carrying out an Analysis of Variance of some kind.

statistical models, we can use ANOVA without really understanding the details of how it works. However, when interpreting the results of statistical tests associated with ANOVA, it is helpful to have a basic conceptual understanding of how it works. The goal of this chapter is to provide this basic understanding.

We'll do this by exploring the simplest type of ANOVA model: **One-way Analysis of Variance.** One-way Analysis of Variance allows us to *predict* how the mean value of a numeric variable (the **response variable**) *responds* or *depends* on the values of a categorical variable (the **predictor variable**). This statement is just another way of way of saying that the ANOVA model compares means among groups defined by the categorical variable.

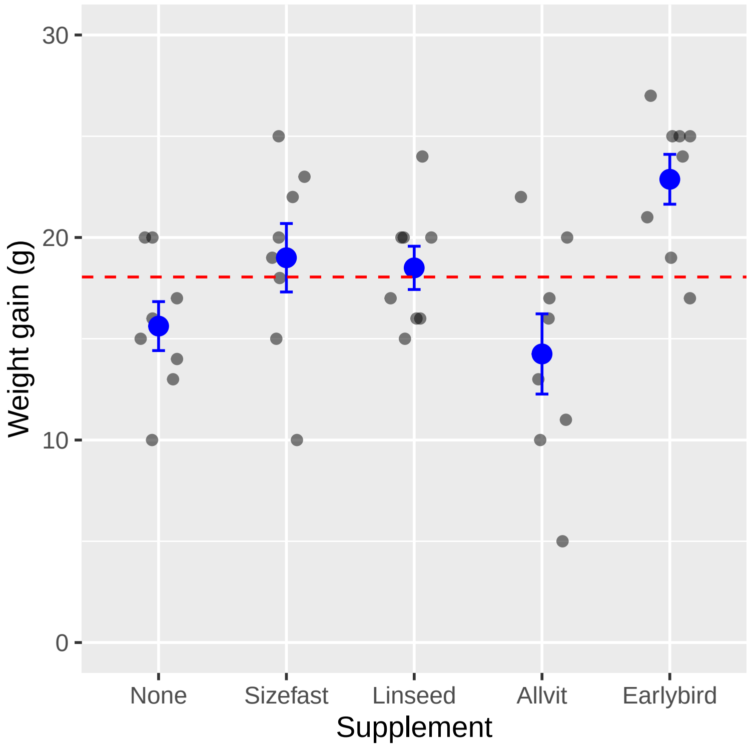
It is no coincidence that this sounds a lot like our description of simple regression. ANOVA and regression are both special cases of the general linear model. This observation explains why:

1. regression and ANOVA can be described with a common language (e.g. response vs predictor variables), and
2. the workflows for developing ANOVA and regression analyses are actually very similar in terms of what we have to do in R.

## 14.2 Why do we need ANOVA models?

The corncrake, *Crex crex*, underwent severe declines in the UK thought to be due to changes in farming practices. Captive breeding and reintroduction programmes were introduced to try to help supplement the wild population. The scientists in charge of the breeding programme wanted to determine the success of 4 different supplements (the predictor variable) for increasing initial growth (the response variable) in the captive-bred hatchlings. They conducted an experiment in which groups of 8 hatchlings were fed with different supplements. A fifth group of 8 hatchlings served as the control group—they were given the base diet with no supplements. At the end of the experiment, they measured how much weight each hatchling had gained over the week.

We can plot the weight gain of 8 hatchlings on each of the supplements (this is the raw data), along with the means of each supplement group, the standard error of the mean, and the sample mean of all the data (the 'global' mean):



The grey points are the raw data, the means and standard error of each group are in blue, and the overall sample mean is shown by the dashed red line. We can see that there seem to be differences *among* the means: hatchlings in each of the different groups often deviate quite a lot from the global mean (the dashed line). This would seem likely to be an effect of the supplements they are on. At the same time, there is still a lot of variation *within* each group: not all of the hatchlings on the same supplements have the same weight gain.

Perhaps all of this could be explained away as sampling variation—that is, the supplements make no difference at all to weight gain and all we're observing are chance variations. We need a statistical test (or tests) to decide whether these differences are 'real'.

It might be tempting to use $t$-tests to compare each mean value with every other mean. This would involve ten $t$-tests on all possible comparisons of the 5 different supplements. Remember, if there is no effect of supplement, then each time we do a $t$-test, there is a chance that we will get a false significant result. Using the conventional $p = 0.05$ significance level, there is a 1 in 20 chance of getting such 'false positives'. However, doing several such tests increases the overall risk of finding at least one false positive. In fact, doing ten $t$-tests gives about a 40% chance of at least one test giving a false positive, even though the individual tests are conducted with $p = 0.05$.

That doesn't sound like a very good way to do science. We need a reliable way to determine whether several means are different or not without increasing the chance of getting a spurious result. That's the job of Analysis of Variance (ANOVA). Just as a two-sample *t*-test compares means between two groups, ANOVA compares means among two *or more* groups. Its name raises an obvious question: if the job of an ANOVA model is to compares means, why is it called Analysis of *Variance*? Let's find out…

## 14.3 How does ANOVA work?

The key to understanding ANOVA is to realise that it works by quantifying and comparing different sources of variation in the data. We start with the total variation in the response variable—the variation among all the units in the study—and then partition this into two sources:

1. Variation attributed to the different groups in the data (e.g. the supplement groups). This is called the 'between-group' variation and is equivalent to the 'explained variation' described in the regression chapter. It quantifies the variation that is 'explained' by the different means. This is the source of variation we usually care about. If this is very low, it means group membership doesn't explain much variation.

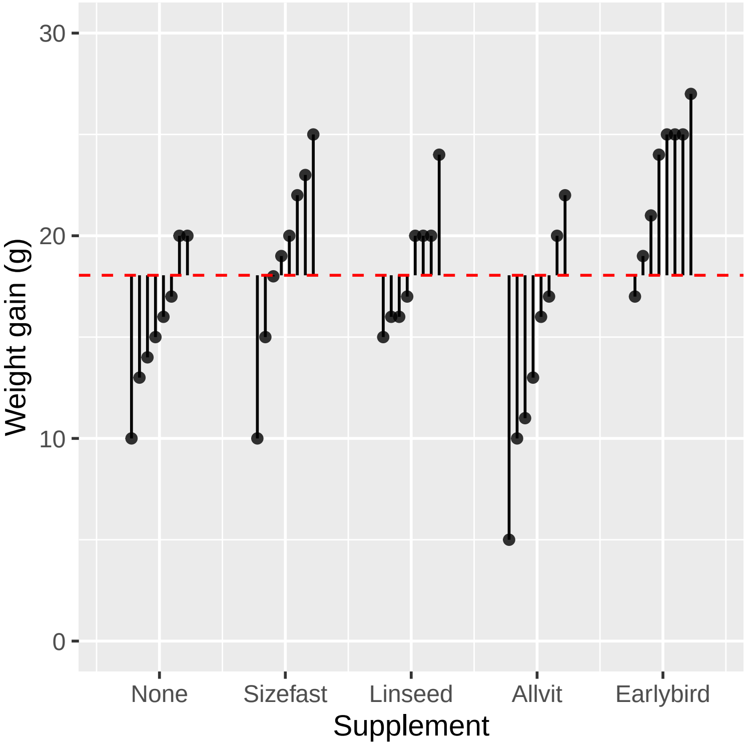
2. Variation due to other, unmeasured sources. This second source of variation is usually referred to as the 'within-group' variation because it applies to experimental units within each group. This quantifies the variation due to everything else that isn't accounted for by the treatments. Within-group variation is also called the 'error variation'. We'll mostly use this latter term because it is a bit more general.

ANOVA does compare means, but it does this by looking at changes in variation. That might seem odd, but it works. If the amount of variation among treatments is sufficiently large compared to the within-group variation, this suggests that the treatments are probably having an effect. This means that to understand ANOVA, we have to keep three sources of variation in mind: the total variation, the between-group variation, and the error variation.

We'll get a sense of how this works by carrying on with the corncrake example. We'll look at how to quantify the different sources of variation, and then move on to evaluate statistical significance using these quantities. The thing to keep in mind is that the logic of these calculations is no different from that used to carry out a regression analysis. The only real difference is that we fit means to different groups when working with an ANOVA model instead of fitting a line through the data. Regression and ANOVA both aim to partition (i.e. split up) the variation in a numeric variable into the parts we can explain and an unexplained leftover part.

### Total variation

The figure below shows the weight gain of each hatchling in the study and the grand mean (i.e. we have not plotted the group-specific means).



The vertical lines show the distance between each observation and the grand mean—we have ordered the data within each group to make the plot a little tidier. A positive deviation occurs when a point is above the line, and a negative deviation corresponds to a case where the point is below the line. We're not interested in the direction of these deviations. We need to quantify the variability of the deviations. This is a feature of their magnitude (i.e. the length of the lines).

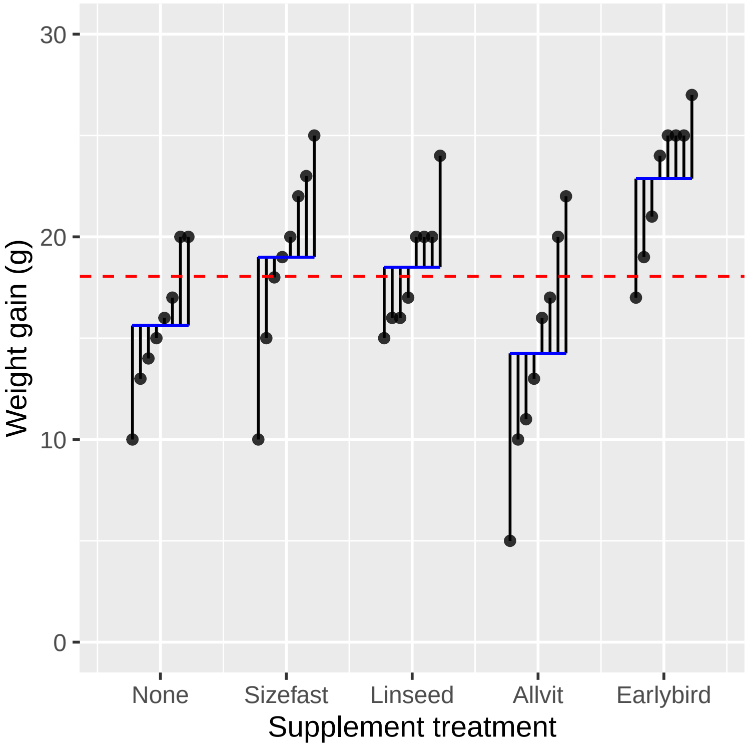
What measure of variability should we use? We can't add up the deviations because they add to zero. Instead, we apply the same idea introduced in the Relationships and regression chapter: the measure of variability we need is based on the 'sum of squares' (abbreviated SS) of the deviations. A sum of squares is calculated by taking each deviation in turn, squaring it, and adding up the squared values. Here are the numeric values of the deviations shown graphically above:

```
##  [1]  -5.05   1.95  -1.05  -2.05  -3.05  -4.05  -8.05   1.95  -3.05  -0.05
## [11]  -8.05   3.95   4.95   0.95   6.95   1.95   1.95   5.95  -1.05  -2.05
## [21]   1.95   1.95  -3.05  -2.05   3.95 -13.05  -8.05  -2.05  -7.05  -1.05
## [31]   1.95  -5.05  -1.05   6.95   5.95   6.95   6.95   0.95   8.95   2.95
```

The sum of squares of these numbers is 967.9. This is called the total sum of squares, because this measure of variability completely ignores the information about treatment groups. It measures the total variability in the response variable, calculated relative to the grand mean.

### Residual variation

The next component of variability we need relates to the within-group variation. Let's replot the original figure showing the weight gain of each hatchling (points), the mean of each supplement group (horizontal blue lines), and the grand mean:



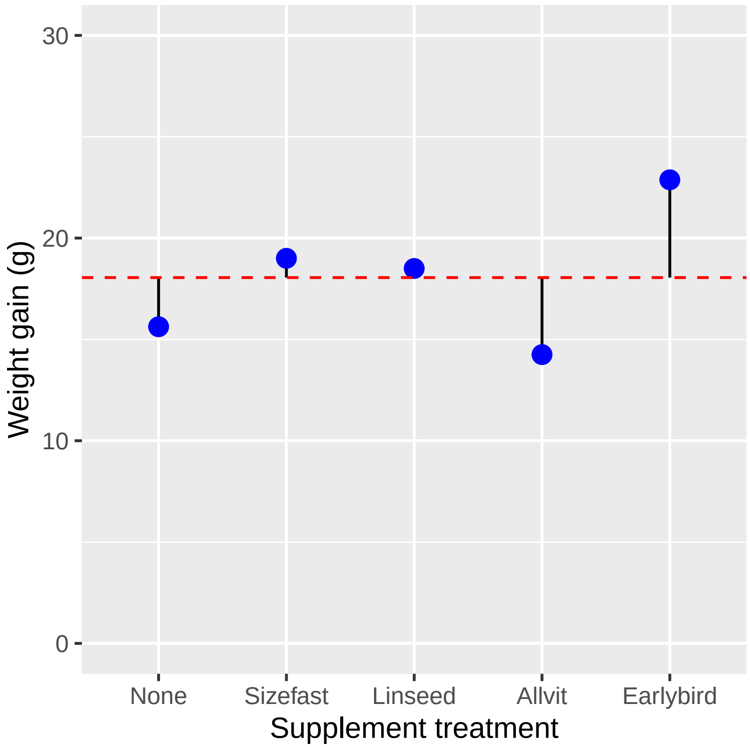
The vertical lines show something new this time. They display the distance between each observation and the group-specific means. They summarise the variation among hatchlings *within* treatment groups. Here are the numeric values of these deviations:

```
##  [1] -9.250 -5.625 -9.000 -4.250 -3.250 -2.625 -1.250 -1.625 -0.625 -4.000
## [11] -3.500  0.375 -2.500 -2.500  1.750  1.375 -1.500  2.750 -5.875 -1.000
## [21]  0.000 -3.875  4.375  4.375  1.000  1.500  1.500  1.500  5.750 -1.875
## [31]  3.000  7.750  4.000  5.500  1.125  6.000  2.125  2.125  2.125  4.125
```

These values are a type of residual: they quantify the 'left over' variation after accounting for differences due to treatment groups. Once again, we can summarise this variability as a single number by calculating the associated sum of squares: we take each deviation in turn, square it, and then add up the squared values. The sum of squares of these numbers is (610.25). This is called the residual sum of squares[2]. It is a measure of the variability that may be attributed to differences among individuals after controlling for the effect of different groups.

### Between-group variation

The last component of variability we need relates to the between group variation. We'll replot the figure one more time, but this time we'll show just the group-specific means (blue points), the overall grand mean (dashed red line), and the deviations of each group mean from the grand mean:



Now the vertical lines show the distance between each group-specific mean and the grand mean. We have five different treatment groups, so there are only five

[2] You will sometimes see something called error sum of squares, or possibly, the within-group sum of squares. These are just different names for the residual sum of squares.

lines. These lines show the variation due to differences among treatment groups. Here are the numeric values of these deviations:

```
## [1] -2.425  0.950  0.450 -3.800  4.825
```

These values quantify the variation that can be attributed to differences among treatments. Once again, we can summarise this variability as a single number by calculating the associated sum of squares—this number is called the treatment sum of squares. This is the same as the 'explained sum of squares' discussed in the context of regression. It is a measure of the variability attributed to differences among treatments.

This is 44.71 in the corncrake example. Notice that this is much smaller than the total sum of squares and the residual sum of squares. This isn't all that surprising as it is based on five numbers, whereas the other two measures of variability are based on all the observations.

### 14.3.1 Degrees of freedom

The raw sums of squares in ANOVA are a function of sample size and the number of groups. To be useful, we need to convert them into measures of variability that don't scale with the size of a data set. We use **degrees of freedom** (written as df, or d.f.) to do this. We came across the concept of degrees of freedom when we studied regression: the degrees of freedom associated with a sum of squares quantifies how much 'information' it is based on. Each of the three sums of squares we just calculated has a different degrees of freedom calculation associated with it:

- Total d.f. = (Number of observations - 1)
- Treatment d.f. = (Number of treatment groups - 1)
- Error d.f. = (Number of observations - Number of treatment groups)

The way to think about these is as follows. We start out with a degrees of freedom equal to the total number of deviations associated with a sum of squares. We then 'lose' one degree of freedom for every mean we have to calculate to work out the deviations. Here is how this works in the corncrake example:

- Total d.f. = 40 - 1 = 39 — The total sum of squares was calculated using all 40 observations in the data, and the deviations were calculated relative to 1 mean (the grand mean).

- Treatment d.f. = 5 - 1 = 4 — The treatment sum of squares was calculated using the 5 treatment group means, and the deviations were calculated relative to 1 mean (the grand mean).

- Error d.f. = 40 - 5 = 35 — The error sum of squares was calculated using all 40 observations in the data, and the deviations were calculated relative to 5 means (the treatment group means).

Don't worry too much if that seems confusing. We generally don't have to carry out degrees of freedom calculations by hand because R will do them for us. We have reviewed them because knowing where they come from can help understand the output of an ANOVA significance test.

### 14.3.2 Mean squares, variance ratios, and F-tests

Once we know how to calculate the degrees of freedom, we can use them to standardise each of the sums of squares. The calculations are very simple. We take each sum of squares and divide it by its associated degrees of freedom. The resulting quantity is called a **mean square** (abbreviated as MS):

$$\text{Mean square} = \frac{\text{Sum of squares}}{\text{Degrees of freedom}}$$

We stated what a mean square represents when discussing regression: it is an estimate of a variance. The mean squares from an ANOVA quantify the variability of the whole sample (total MS), the variability explained by treatment group (treatment MS), and the unexplained residual variation (residual MS).

ANOVA quantifies the strength of the treatment effect by comparing the treatment mean square to the residual mean square. When the treatment MS is large relative to the residual MS, this suggests that the treatments are more likely to be having an effect. In practise, they are compared by calculating the ratio between them (designated by the letter $F$):

$$F = \text{Variance ratio} = \frac{\text{Variance due to treatments}}{\text{Error variance}}$$

This is the equivalent to the $F$-ratio mentioned in the context of regression. When the variation among treatment means (treatment MS) is large compared to the variation due to other factors (residual MS), the $F$-ratio will be large. If the variation among treatment means is small relative to the residual variation, the $F$-ratio will be small.

How do we decide when the $F$-ratio is large enough? That is, how do we judge a result to be statistically significant? We play out the usual 'gambit':

1. We assume that there is no difference between the population means of each treatment group. That is, we hypothesise that the data in each group were sampled from a single population with one overall global mean.

2. Next, we use information in the sample to help us work out what would happen if we were to repeatedly take samples in this hypothetical situation. The 'information' in this case corresponds to the mean squares.

3. We then ask, 'if there is no difference between the groups, what is the probability that we would observe a variance ratio that is the same as, or more extreme than, the one we observed in the real sample?'

4. If the observed variance ratio is sufficiently improbable, then we conclude that we have found a 'statistically significant' result, i.e. one that is inconsistent with the hypothesis of no difference.

In order to work through these calculations, we make one key assumption about the population from which the data in each treatment group has been sampled. We assume that the residuals are normally distributed. Once we make this assumption the distribution of the $F$-ratio under the null hypothesis (the 'null distribution') has a particular form: it follows an $F$ distribution. This means we assess the statistical significance of differences between means by comparing the $F$-ratio calculated from a sample of data to the theoretical $F$ distribution.

This procedure is "methodology for evaluating statistical significance" we alluded to at the start of this chapter. The resulting $F$-test is no different from the significance testing methodology we outlined for regression models. The important point is that this procedure uses only one comparison—the treatment variation and the error variation—rather than the ten individual $t$-tests that would have been required to compare all the pairs.

One thing to be aware of is that the $F$-tests are typically global tests of significance: they indicate whether groups are different but don't tell us anything about which groups might be driving a significant result. That kind of question needs another test. We will examine one such test in the multiple comparison chapter later.

## 14.4 Different kinds of ANOVA model

There are many different flavours of ANOVA model. The one we've just been learning about is called a one-way ANOVA. It's called one-way ANOVA because it involves only one factor: supplement type (this includes the control). If we had considered two factors—e.g. supplement type and amount of food—we would have to use something called a two-way ANOVA. A design with three factors is called a three-way ANOVA, and… you get the idea.

There are many other ANOVA models, each of which is used to analyse a specific type of experimental design. We will only consider two different types of ANOVA in this book: one-way ANOVA and ANOVA for one-way, blocked design experiments.

## 14.5 Some common questions about ANOVA

To finish off with, three common questions that often arise:

### 14.5.1 Can ANOVA only be applied to experimental data?

We have been discussing ANOVA in the context of a designed experiment (i.e. we talked about treatments and control groups). Although ANOVA was developed to analyse experimental data, which is where it is most powerful, it can be used in an observational setting. As long as we're careful about how we sample different groups, we can use ANOVA to analyse their differences. The main difference between ANOVA for experimental and observational studies arises in the interpretation of the results. If the data aren't experimental, we can't say anything concrete about the causal nature of observed among-group differences.

### 14.5.2 Do we need equal replication?

One of the frequent problems with biological data is we often don't have equal replication. Even if we started with equal replication in our design, all sorts of things conspire to upset even the best-designed experiments. For example, plants and animals have a habit of dying before we have gathered all our data: a pot may get dropped or a culture contaminated. Does this matter? Not really. One-way ANOVA does not require equal replication and will work just fine where sample sizes differ between treatments.

### 14.5.3 Can ANOVA be done with only two treatments?

Although the $t$-test provides a convenient way of testing means from two treatments, nothing stops you from doing an ANOVA on two treatments. A $t$-test (assuming equal variances) and ANOVA on the same data should give the same $p$-value (in fact, the $F$-statistic from the ANOVA will be the square of the $t$-value from the $t$-test). However, one advantage to the $t$-test is that you can do the version of the test that allows for unequal variances—something a standard ANOVA does not do. There is a version of Welch's test for one-way ANOVA, but we won't study it in this book (look at the `oneway.test` function if you are interested).

# Chapter 15

# One-way ANOVA in R

Our goal in this chapter is to learn how to work with one-way ANOVA models in R. As we did with regression, we'll do this by working through an example. We'll start with the problem and the data, then work through model fitting, significance testing, and finally, present the results.

## 15.1 Introduction

We will use the corncrake example from the last chapter to demonstrate how to conduct a one-way ANOVA in R. Recall that these data show the results of an experiment comparing the weight gain of eight corncrake hatchlings on each of four supplements and one control diet.

> The data live in the 'CORN_CRAKE.CSV' file. The code below assumes those data have been read into a tibble called `corn_crake`. Set that up if you plan to work along.

### 15.1.1 First steps

What lives inside the `corn_crake` data set? There are no surprises:

```
glimpse(corn_crake)
```

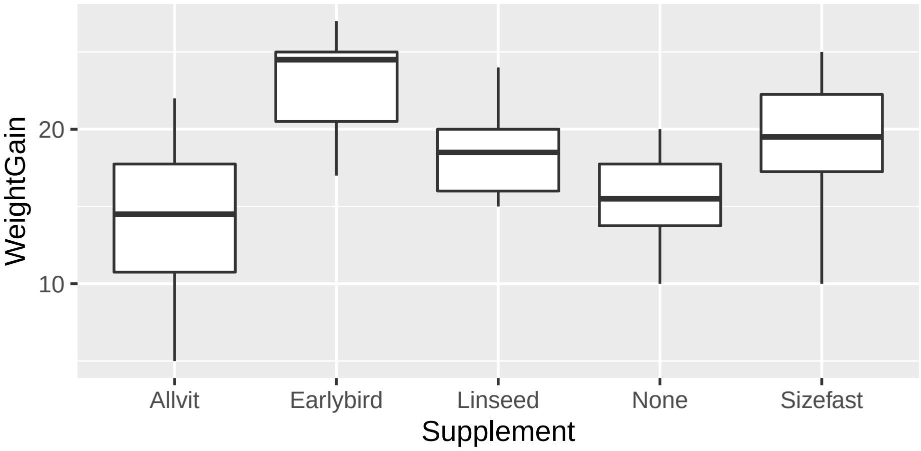
```
## Rows: 40
## Columns: 2
## $ Supplement <chr> "None", "None", "None", "None", "None", "None", "None", "No~
## $ WeightGain <dbl> 13, 20, 17, 16, 15, 14, 10, 20, 15, 18, 10, 22, 23, 19, 25,~
```

There are 40 observations in the data set and 2 variables (columns), called `Supplement` and `WeightGain`.

### 15.1.2 Visualising the data

We should plot our data before we even consider fitting a model. One way to do this is by creating a boxplot:

```
ggplot(corn_crake, aes(x = Supplement, y = WeightGain)) +
  geom_boxplot()
```



At this point, we just want to get an idea of what our data look like. We don't need to worry about customising the plot to make it look nice.

**Examine the plot**

Have a look at the plot and think about what it means biologically. Which supplement seems to have had the biggest effect? Do all of the supplements increase growth relative to the control?

## 15.2 Model fitting and significance tests

Carrying out ANOVA in R is quite simple, but as with regression, there is more than one step.

The first involves a process known as *fitting the model* (or just *model fitting*). This is the step where R calculates the relevant means and the additional information needed to generate the results in step two. We call this step model fitting because an ANOVA is a type of model for our data: it is a model that allows the mean of a variable to vary among groups.

How do we fit an ANOVA model in R? We use the `lm` function again. Remember what the letters 'lm' stand for? They stand for '(general) linear model'. An ANOVA model is just another special case of that general linear model. Here is how we fit a one-way ANOVA in R, using the corncrake data:

```
corncrake_model <- lm(WeightGain ~ Supplement, data = corn_crake)
```

By now, this kind of thing is probably starting to look familiar. We have to assign two arguments:

1. The first argument is a formula. We know this because it includes a 'tilde' symbol: `~`. The variable name on the left of the `~` must be the numeric response variable whose means we want to compare among groups. The variable on the right should be the indicator (or predictor) variable that says which group each observation belongs to. These are `WeightGain` and `Supplement`, respectively.

2. The second argument is the name of the data frame that contains the two variables listed in the formula.

### How does R know which model to use?

How does R know we want to use an ANOVA model? After all, we didn't specify this anywhere. The answer is that R looks at what type of vector `Supplement` is. It is a categorical character vector, and so R carries out an ANOVA. It would do the same if `Supplement` had been a factor. However, if the values of `Supplement` had been stored as numbers then R would not have fitted an ANOVA. Instead, it would have assumed we meant to fit a regression model, which is seldom appropriate. **This is why we never store categorical variables as numbers in R.**

Avoid using numbers to encode the values of a categorical variable if you want to avoid making mistakes when fitting models!

Notice that we did not print the results to the console. Instead, we assigned the result a name —`corncrake_model` now refers to a **model object**. What happens if we print this to the console?

```
corncrake_model
```

```
##
```

```
## Call:
## lm(formula = WeightGain ~ Supplement, data = corn_crake)
## 
## Coefficients:
##         (Intercept)  SupplementEarlybird    SupplementLinseed  
##              14.250                8.625                4.250  
##      SupplementNone   SupplementSizefast  
##               1.375                4.750
```

Not a great deal. Printing a fitted model object to the console is not very useful when working with ANOVA. We just see a summary of the model we fitted and some information about the model coefficients. Yes, an ANOVA model has coefficients, just like a regression does.

**Assumptions**

The one-way ANOVA model has a series of assumptions about how the data are generated. This point in the workflow would be a good place to evaluate whether or not these may have been violated. However, we're going to tackle this later in the Assumptions and Diagnostics chapters. For now, it is enough to be aware that there are assumptions associated with one-way ANOVA and that you should evaluate these when working with your own data.

### 15.2.1 Interpreting the results

We really want a *p*-value to help us determine whether there is statistical support for a difference among the group means. That is, we need to calculate things like degrees of freedom, sums of squares, mean squares, and the *F*-ratio. This is step 2.

We use the `anova` function to do this:

```
anova(corncrake_model)
```

```
## Analysis of Variance Table
## 
## Response: WeightGain
##            Df Sum Sq Mean Sq F value   Pr(>F)   
## Supplement  4 357.65  89.413  5.1281 0.002331 **
## Residuals  35 610.25  17.436                    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notice that all we did was pass the `anova` function one argument: the name of the fitted model object. Let's step through the output to see what it means. The first line just informs us that we are looking at an ANOVA table, i.e. a table of statistical results from an analysis of variance. The second line just reminds us of what variable we analysed.

The important information is in the table that follows:

```
##            Df Sum Sq Mean Sq F value   Pr(>F)
## Supplement  4 357.65  89.413  5.1281 0.002331 **
## Residuals  35 610.25  17.436
```

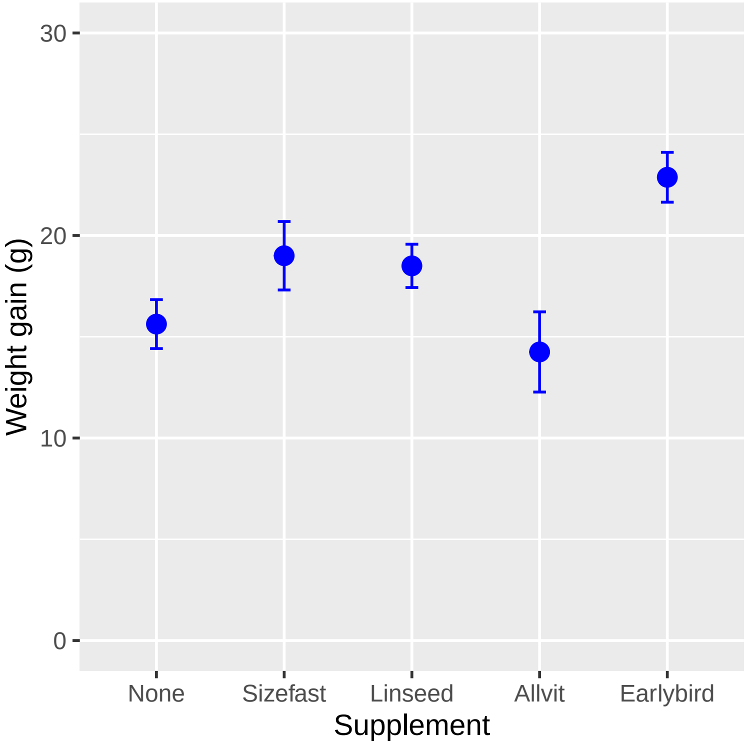
This is an Analysis of Variance Table. It summarises the parts of the ANOVA calculation: `Df` – degrees of freedom, `Sum Sq` – the sum of squares, `Mean Sq` – the mean square, `F value` – the $F$-ratio (i.e. variance ratio), `Pr(>F)` – the $p$-value.

The $F$-ratio (variance ratio) is the key term. This is the test statistic. It measures how large and consistent the differences between the means of the five different treatments are. Larger values indicate clearer differences between means, in just the same way that large values of Student's $t$ indicate clearer differences between means in the two sample situation.

The $p$-value gives the probability that the observed differences between the means, or a more extreme difference, could have arisen through sampling variation under the null hypothesis. What is the null hypothesis: it is one of no effect of treatment, i.e. the null hypothesis is that all the means are the same. As always, the $p$-value of 0.05 is used as the significance threshold, and we take $p < 0.05$ as evidence that at least one of the treatments is having an effect. For the corncrake data, the value is 5.1, and the probability ($p$) of getting an $F$-ratio this large is given by R as 0.0023, i.e. less than 0.05. This provides good evidence that there are differences in weight gain between at least some of the treatments.

So far so good. The test that we have just carried out is called the **global test of significance**. It goes by this name because it doesn't tell us anything about which means are different. The analyses suggest that there is an effect of supplement on weight gain. Still, some uncertainty remains because we have only established that there are differences among at least some supplements. A global test doesn't say which supplements are better or worse. This could be very important. If the significant result is generated by all supplements being equally effective (hence differing from the control but not from each other) we would draw very different conclusions than if the result was a consequence of one supplement being very effective and all the others being useless. Our result could even be produced by the supplements all being less effective than the control!

So having got a significant result in the ANOVA, we should always look at the means of the treatments to understand where the differences actually lie. We did this in the previous chapter but here is the figure again anyway:



What looking at the means tells us is that the effect of the supplements is generally to increase weight gain (with one exception, ‘Allvit’) relative to the control group (‘None’), and that it looks as though ‘Earlybird’ is the most effective, followed by ‘Sizefast’ and ‘Linseed’.

Often inspection of the means in this way will tell us all we need to know and no further work will be required. However, sometimes it is desirable to have a more rigorous way of testing where the significant differences between treatments occur. A number of tests exist as ‘add ons’ to ANOVA which enable you to do this. These are called **post hoc multiple comparison tests** (sometimes just ‘multiple comparison tests’). We’ll see how to conduct these later.

## 15.3 Presenting results

As with all tests, it will be necessary to summarise the result in a written form. With an ANOVA on several treatments, we always need to at least summarise the result of the global test of significance, for example:

> There was a significant effect of supplement on the weight gain of the corncrake hatchlings (ANOVA: F=5.1; d.f.= 4,35; p<0.01).

There are several things to notice here:

- The degrees of freedom are always quoted as part of the result, and…there are *two values for the degrees of freedom* to report in ANOVA because it involves $F$-ratios. These are obtained from the ANOVA table and should be given as the treatment degrees of freedom first, followed by the error degrees of freedom. Order matters. Don't mix it up.
- The degrees of freedom are important because, like a $t$-statistic, the significance of an $F$-ratio depends on the degrees of freedom, and giving them helps the reader to judge the result you are presenting. A large value may not be very significant if the sample size is small, a smaller may be highly significant if the sample sizes are large.
- The $F$-ratio rarely needs to be quoted to more than one decimal place.

When it comes to presenting the results in a report, it helps to present the means, as the statement above cannot entirely capture the results. We could use a table to do this, but tables are ugly and difficult to interpret. A good figure is much better.

Box and whiskers plots and multi-panel dot plots or histograms are exploratory data analysis tools. We use them at the beginning of an analysis to understand the data, but we don't tend to present them in project reports or scientific papers. Since ANOVA is designed to compare means, a minimal plot needs to show the point estimates of each group-specific mean, along with a measure of their uncertainty. We often use the standard error of the means to summarise this uncertainty.

In order to be able to plot these quantities we first have to calculate them. We can do this using `dplyr`. Here's a reminder of the equation for the standard error of a mean:

$$SE = \frac{\text{Standard deviation of the sample}}{\sqrt{\text{Sample size}}} = \frac{SD}{\sqrt{n}}$$

So, the required `dplyr` code is:

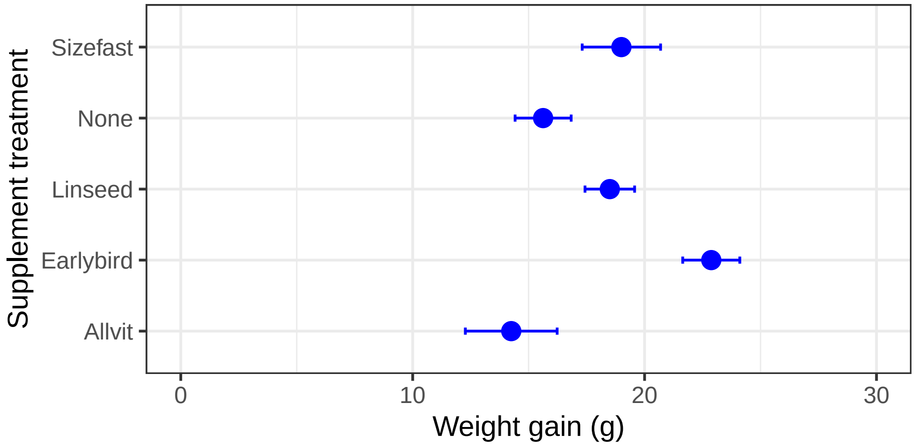
```
# get the mean and the SE for each supplement
corncrake_stats <-
  corn_crake %>%
  group_by(Supplement) %>%
  summarise(Mean = mean(WeightGain), SE = sd(WeightGain)/sqrt(n()))
# print to the console
corncrake_stats
```

```
## # A tibble: 5 x 3
##   Supplement  Mean    SE
##   <chr>      <dbl> <dbl>
## 1 Allvit      14.2  1.98
## 2 Earlybird   22.9  1.23
## 3 Linseed     18.5  1.07
## 4 None        15.6  1.21
## 5 Sizefast    19    1.69
```

Notice that we used the `n` function to get the sample size. The rest of this R code should be quite familiar by now. We gave the data frame containing the group-specific means and standard errors the name `corncrake_stats`.

We have a couple of different options for making a good summary figure. The first plots a point for each mean and places error bars around this to show ±1 SE. In order to do this using `ggplot2` we have to add *two layers*—the first specifies the points (the means) and the second specifies the error bar (the SE). Here is how to do this:

```
ggplot(data = corncrake_stats,
       aes(x = Supplement, y = Mean, ymin = Mean - SE, ymax = Mean + SE)) +
  # this adds the means
  geom_point(colour = "blue", size = 3) +
  # this adds the error bars
  geom_errorbar(width = 0.1, colour = "blue") +
  # controlling the appearance
  scale_y_continuous(limits = c(0, 30)) +
  # use sensible labels
  xlab("Supplement treatment") + ylab("Weight gain (g)") +
  # flip x and y axes
  coord_flip() +
  # use a more professional theme
  theme_bw()
```



First, notice that we set the `data` argument in `ggplot` to be the data frame containing the summary statistics (not the original raw data). Second, we set up four aesthetic mappings: x, y, ymin and ymax. Third, we added one layer using geom_point. This adds the individual points based on the x and y mappings. Fourth, we added a second layer using geom_errorbar. This adds the error bars based on the x, ymin and ymax mappings. Finally we adjusted the y limits and the labels (this last step is optional).

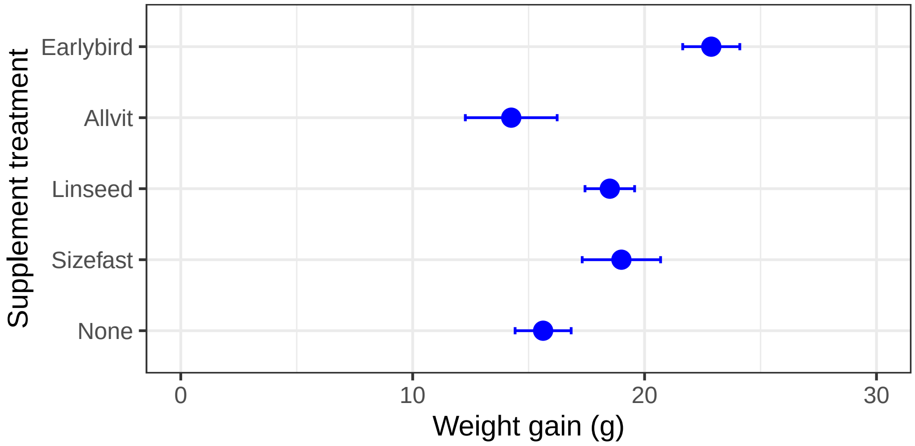
### Factors

Now is good time to take a short detour to remind ourselves about factors. In the world of experimental design, the word 'factor' is used to describe a controlled variable where the levels—i.e. its different values or categories—are set by the experimenter. R includes a special type of vector for representing factors. This is called, rather sensibly, a factor. One can get quite a long way in R-land without using factors. However, we inevitably end up using them because so much of R's plotting and statistical modelling machinery rely on factors. We are going to need to them to help us reorder the groups in our plot.

Take a close look at that last figure. Is there anything wrong with it? The control group in this study is the no diet group ('None'). Conventionally, we display the control groups first. R hasn't done this because the different categories of Supplement are presented in alphabetical order by default. One way to change that order if by converting Supplement to a factor and define the order we want as we do so. Here is how to do this using dplyr and a function called factor:

```
corncrake_stats <-
  corncrake_stats %>%
  mutate(Supplement = factor(Supplement,
                             levels = c("None", "Sizefast", "Linseed", "Allvit", "Earlybird")))
```

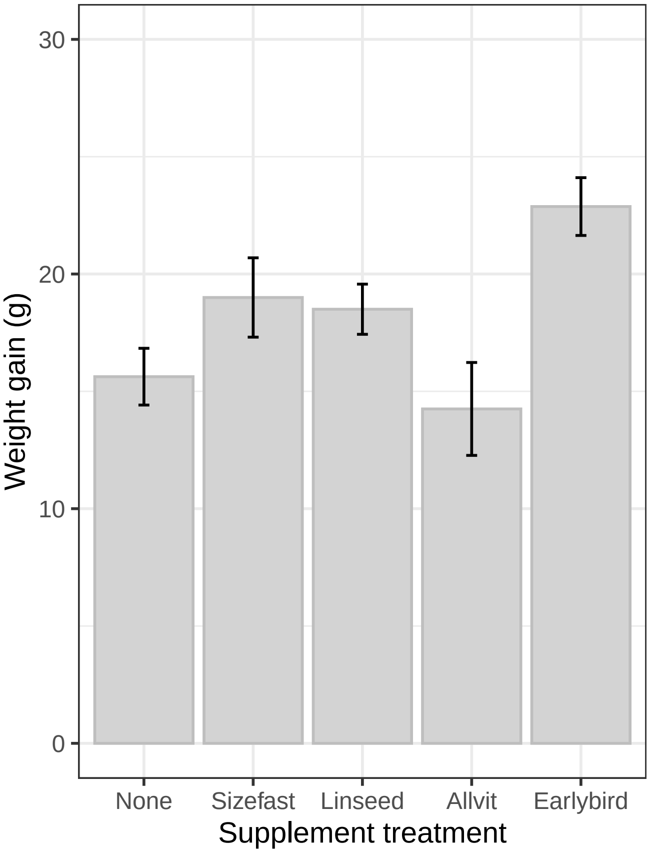
We use mutate to update Supplement, using the factor function to redefine the levels of Supplement and overwrite the original. Now, when we rerun the ggplot2 code we end up with a figure like this:



The treatments are presented in the order specified with the `levels` argument. Problem solved!

A bar plot is another popular visualisation for summarising the results of an ANOVA. We only have to change one thing about the last chunk of `ggplot2` code to make a bar plot. Instead of using `geom_point`, we use `geom_col` (we'll drop the `coord_flip` bit too):

```
ggplot(data = corncrake_stats,
       aes(x = Supplement, y = Mean, ymin = Mean - SE, ymax = Mean + SE)) +
  # this adds the means
  geom_col(fill = "lightgrey", colour = "grey") +
  # this adds the error bars
  geom_errorbar(width = 0.1, colour = "black") +
  # controlling the appearance
  scale_y_continuous(limits = c(0, 30)) +
  # use sensible labels
  xlab("Supplement treatment") + ylab("Weight gain (g)") +
  # use a more professional theme
  theme_bw()
```


30
20
10
0
Weight gain (g)
None
Sizefast
Linseed
Allvit
Earlybird
Supplement treatment

# Chapter 16

# Multiple comparison tests

In the One-way ANOVA in R chapter, we learned how to examine the global hypothesis of no difference between means. That test does not evaluate *which* means might be driving a significant result. For example, in the corncrake example, we found evidence of a significant effect of dietary supplement on the mean hatchling growth rate. However, our analysis did not tell us which supplements are better or worse and did not make any statements about which supplements differ significantly from each other. The purpose of this chapter is to examine one method for assessing where the differences lie.

## 16.1 Post hoc multiple comparisons tests

The general method we will use is called a **post hoc multiple comparisons test**. The phrase 'post hoc' refers to the fact that these tests are conducted without any particular prior comparisons in mind. The words 'multiple comparisons' refer to the fact that they consider many different pairwise comparisons. There are quite a few multiple comparison tests—Scheffé's test, the Student-Newman-Keuls test, Duncan's new multiple range test, Dunnett's test, … (the list goes on and on). Each one applies to particular circumstances, and none is universally accepted as 'the best'. We are going to work with the most widely used test: the **Tukey multiple comparison test**. This test is also known as Tukey's Honestly Significant Difference (Tukey HSD) test[1].

People tend to favour Tukey's HSD test because it is 'conservative': the test has a low false-positive rate compared to the alternatives. A false positive occurs when a test turns up a statistically significant result for an effect that is not

[1] N.B. Try to avoid a common mistake: it is *Tukey*, after its originator, the statistician Prof. John Tukey, not Turkey, a large domesticated bird which has made no useful contributions to statistical theory or practice.

there. A low false-positive rate is a good thing. It means that if we find a significant difference, we can be more confident it is 'real'. The cost of using the Tukey HSD test is that it isn't as powerful as alternatives: the test turns up a lot of false negatives. A false negative occurs when a test fails to produce a statistically significant result for an effect when it is present. A test with a high false-negative rate tends to miss effects.

One line of thinking says post hoc multiple comparisons tests of any kind should never be undertaken. We shouldn't carry out an experiment without a prior prediction of what will happen—we should know which comparisons need to be made and should only undertake those particular comparisons rather than making every possible comparison. Nonetheless, post hoc multiple comparisons tests are easy to apply and widely used, so there is value in knowing how to use them. The Tukey HSD test at least tends to guard against picking up non-existent effects.

## 16.2 Tukey's HSD test in R

We will use the corncrake example to illustrate how to use R to carry out and interpret Tukey's HSD test.

The code below assumes CORN_CRAKE.CSV has been read into a tibble called `corn_crake`. It also assumes a one-way ANOVA model for the effect of diet supplement on hatchling growth rate has already been fitted and given the name `corncrake_model`:

```
corncrake_model <- lm(WeightGain ~ Supplement, data = corn_crake)
```

In the One-way ANOVA in R chapter, we saw how to use `anova` on a model object to evaluate whether means differ, for example:

```
anova(corncrake_model)
```

```
## Analysis of Variance Table
##
## Response: WeightGain
##            Df Sum Sq Mean Sq F value   Pr(>F)
## Supplement  4 357.65  89.413  5.1281 0.002331 **
## Residuals  35 610.25  17.436
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

That gives us the result of the global significance test. Unfortunately, we need different functions to carry out the Tukey HSD test. First, we convert the linear model object into a different kind of model object using the `aov` function:

```
corncrake_aov <- aov(corncrake_model)
```

We don't need to understand too much about what this is doing. In a nutshell, `aov` does some extra calculations needed to perform a Tukey HSD test. We gave the new object a name (`corncrake_aov`) because we need to use it in the next step.

It's easy to perform a Tukey HSD test once we have the 'aov' version of our model. There are a few different options. Here is how to do this using the `TukeyHSD` function:

```
TukeyHSD(corncrake_aov, ordered = TRUE)
```

Pay attention! We applied the `TukeyHSD` function to the 'aov' object, *not* the original `lm` object. We have suppressed the output for now. Before we review it we need to get an idea of what it is going to show us.

The `ordered = TRUE` tells `TukeyHSD` that we want to order the treatment means from smallest to largest and then apply every pairwise comparison, starting with the smallest mean ('Allvit') and working up through the order. Here are the means ordered from smallest to largest, working left to right:

| Supplement | Allvit | None | Linseed | Sizefast | Earlybird |
|---|---|---|---|---|---|
| Mean | 14.3 | 15.6 | 18.5 | 19.0 | 22.9 |

So the `TukeyHSD` with `ordered = TRUE` will first compare 'Allvit' to 'None', then 'Allvit' to 'Linseed', then 'Allvit' to 'Sizefast', then 'Allvit' to 'Earlybird', then 'None' to 'Linseed', then 'None' to 'Sizefast', … and so on, until we get to 'Sizefast' vs. 'Earlybird'. Let's look at the output:

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##     factor levels have been ordered
##
## Fit: aov(formula = corncrake_model)
##
## $Supplement
##                     diff        lwr       upr     p adj
## None-Allvit        1.375 -4.627565  7.377565 0.9638453
## Linseed-Allvit     4.250 -1.752565 10.252565 0.2707790
## Sizefast-Allvit    4.750 -1.252565 10.752565 0.1771593
## Earlybird-Allvit   8.625  2.622435 14.627565 0.0018764
## Linseed-None       2.875 -3.127565  8.877565 0.6459410
## Sizefast-None      3.375 -2.627565  9.377565 0.4971994
## Earlybird-None     7.250  1.247435 13.252565 0.0113786
## Sizefast-Linseed   0.500 -5.502565  6.502565 0.9992352
## Earlybird-Linseed  4.375 -1.627565 10.377565 0.2447264
## Earlybird-Sizefast 3.875 -2.127565  9.877565 0.3592201
```

This table look confusing at first glance. It enables you to look up every pair of treatments, and see whether they are significantly different from each other. Lets see how this works... The first four lines compare the Allvit treatment ('Allvit') with each of the other treatments in turn:

```
##                     diff       lwr       upr     p adj
## None-Allvit        1.375 -4.627565  7.377565 0.9638453
## Linseed-Allvit     4.250 -1.752565 10.252565 0.2707790
## Sizefast-Allvit    4.750 -1.252565 10.752565 0.1771593
## Earlybird-Allvit   8.625  2.622435 14.627565 0.0018764
```

So to look up the difference between the control treatment and the 'Allvit' treatment, we read the first results row in the table. This says the means differ by 1.375, the confidence interval associated with this difference is [-4.63, 7.38], and that the comparison has a $p$-value of 0.96. So in this case we would conclude that there was a no significant difference between the control treatment and the 'Allvit' treatment. We could look up any comparison of the 'Allvit' treatment with a different treatment in the next three lines of this portion of the table.

This basic logic extends to the rest of the table. If we want to know whether the 'Earlybird' treatment is different from the control, we look up the 'Earlybird-None' row:

```
##                     diff       lwr       upr     p adj
## Earlybird-None     7.250  1.247435 13.252565 0.0113786
```

It looks like the means of the 'Earlybird' and 'None' levels are significantly different at the $p < 0.05$ level.

Now we know how to look up any set of comparisons, we need to see whether the difference is significant. The next question is: How should we summarise such a table?

### 16.2.1 How to summarise multiple-comparison results

Summarising the results of multiple comparison tests can be a tricky business. The first rule is: don't present the results like the `TukeyHSD` function does! A clear summary of the results will help us to interpret them correctly and makes it easier to explain them to others. How should we do this? Let's list the means in order from smallest to largest again:

| Supplement | Allvit | None | Linseed | Sizefast | Earlybird |
|---|---|---|---|---|---|
| Mean | 14.3 | 15.6 | 18.5 | 19.0 | 22.9 |

Now, using the table of Tukey results, we can perform a sequence of pairwise comparisons between the supplements starting with the smallest pair… 'Allvit' and 'None'. The appropriate test is in the first table:

```
##       diff        lwr        upr      p adj
##  1.3750000 -4.6275646  7.3775646  0.9638453
```

The last column gives the *p*-value, which in this case is certainly not significant (it is much greater than 0.05), so we conclude there is no difference between the 'Allvit' and 'None' treatments. So now we continue with 'Allvit', but compare it to the next larger mean ('Linseed'). In this case the values are:

```
##      diff       lwr       upr     p adj
##  4.250000 -1.752565 10.252565  0.270779
```

The last column gives the *p*-value, which again is not significant, so we conclude there is no difference between the 'Allvit' and 'Linseed' treatments. So now we continue with 'Allvit', but compare it to the next larger mean ('Sizefast'). In this case, the values are:

```
##       diff        lwr        upr      p adj
##  4.7500000 -1.2525646 10.7525646  0.1771593
```

Once again, this difference is not significant, so we conclude there is no difference between the 'Allvit' and 'Sizefast' treatments either. So again, we continue with 'Allvit', which we compare to the next larger mean ('Earlybird').

```
##       diff        lwr        upr      p adj
##  8.6250000  2.6224354 14.6275646  0.0018764
```

This time the *p*-value is clearly less than 0.05, so we conclude that this pair of treatments are significantly different. We record this by marking 'Allvit', 'None', 'Linseed' and 'Sizefast' to indicate that they don't differ from each other. We'll use the letter 'b' to do this.

| Supplement | Allvit | None | Linseed | Sizefast | Earlybird |
|---|---|---|---|---|---|
| Mean | 14.3 | 15.6 | 18.5 | 19.0 | 22.9 |
|  | b | b | b | b |  |

The 'b' defines a group of treatment means—'Allvit', 'None', 'Linseed' and 'Sizefast'—which are not significantly different from one another. It doesn't matter which letter we use by the way (the reason for using 'b' here will become apparent in a moment).

The means are ordered from smallest to largest, which means we can forget about ‘None’, ‘Linseed’ and ‘Sizefast’ treatments for a moment—if they are not significantly different from ‘Allvit’ they can’t be significantly different from one another.

We move on to ‘Earlybird’, but now, we *work back down* the treatments to see if we can define another overlapping group of means that are not significantly different from one another. When we do this, we find that ‘Earlybird’ is not significantly different from ‘Linseed’ and ‘Sizefast’, but that it is significantly different from ‘None’. This forms a second ‘not significantly different’ group. We will denote this with a new letter (‘a’) in our table:

| Supplement | Allvit | None | Linseed | Sizefast | Earlybird |
|---|---|---|---|---|---|
| Mean | 14.3 | 15.6 | 18.5 | 19.0 | 22.9 |
| | b | b | b | b | |
| | | | a | a | a |

If there were additional treatments with a mean that was greater than ‘Earlybird’, we would have to carry on this process, *working back up* from ‘Earlybird’. Thankfully, there are no more treatments, so we are finished.

This leaves us with a concise and complete summary of where the differences between treatments are, which greatly simplifies the task of interpreting the results. Treatment labels that share a letter represent sets of means that are not different from each other. Treatments that are not linked by one or more shared letters are significantly different.

**Significant ANOVA but no differences in a Tukey test?**

ANOVA and the Tukey HSD test are different tests, with different goals. Because of this, it is possible to end up with a significant result from ANOVA, indicating at least one difference between means, but fail to get any differences detected by the Tukey test. This situation can arise when there are many group means being compared and the ANOVA result is marginal (close to $p = 0.05$).

When this happens, there isn’t much we can do except make the best interpretation we can from inspecting the data, and be suitably cautious in the conclusions we draw. It might be tempting to run a new *post hoc* analysis using a different kind of test. Don’t do this. It is a terrible strategy for doing statistics because this kind of practise is guaranteed to increase the chances of finding a false positive.

### 16.2.2 Doing it the easy way...

The results table we produced is concise and complete but no reasonable person would say it was easy to arrive at. Fortunately, someone has written an R function to do this for us. It isn't part of 'base R' though, so we have to install a package to use it. The package we need is called `agricolae`:

```
install.packages("agricolae")
```

Once this has been installed and then loaded, we can use its `HSD.test` function to find the 'not significantly different' groups via a Tukey HSD test:

```
HSD.test(corncrake_aov, "Supplement", console=TRUE)
```

```
##
## Study: corncrake_aov ~ "Supplement"
##
## HSD Test for WeightGain
##
## Mean Square Error:  17.43571
##
## Supplement,  means
##
##           WeightGain      std r Min Max
## Allvit        14.250 5.599745 8   5  22
## Earlybird     22.875 3.482097 8  17  27
## Linseed       18.500 3.023716 8  15  24
## None          15.625 3.420004 8  10  20
## Sizefast      19.000 4.780914 8  10  25
##
## Alpha: 0.05 ; DF Error: 35
## Critical Value of Studentized Range: 4.065949
##
## Minimun Significant Difference: 6.002565
##
## Treatments with the same letter are not significantly different.
##
##           WeightGain groups
## Earlybird     22.875      a
## Sizefast      19.000     ab
## Linseed       18.500     ab
## None          15.625      b
## Allvit        14.250      b
```

The `console = TRUE` argument tells the function to print the results for us. That's a lot of output, but we can ignore most of it. The part that matters most is the table at the very end. This shows the group identities as letters, the treatment names, and the treatment means. If we compare that table with the one we just made, we can see they convey the same information. The package labels each group with a letter. For example, we can see that 'Linseed' and 'SizeFast' are both members of the 'a' and 'b' group.

So, there is no need to build these Tukey HSD tables by hand. Use the `HSD.test` function in the `agricolae` package instead. Why did we do it the long way first? Well, the usual reasoning applies: it helps us understand how the 'letters notation' works.

## 16.3 Summarising and presenting the results of a Tukey test

As with any statistical test it will usually be necessary to summarise the Tukey HSD test in a written form. This gets quite complicated when both the global significance test and the multiple comparisons tests need to be presented. In most cases, it is best to summarise the ANOVA results and concentrate on those comparisons that relate to the original hypothesis we were interested in. Then, refer to a table or figure for the additional detail. For example…

> There was a significant effect of supplement on the weight gain of hatchlings (ANOVA: F=5.1; d.f.= 4,35; p<0.01) (Figure 1). The only supplement that led to a significantly higher rate of weight gain than the control group was Earlybird (Tukey multiple comparisons test, $p < 0.05$).

When it comes to presenting the results in a report, we really need some way of presenting the means, and the results of the multiple comparison test, as the statement above cannot entirely capture the form of the results. The information can often be conveniently incorporated into a table or figure, using more or less the same format as the output from the `HSD.test` function in the `agricolae` package.

An example table might be:

| **Supplement** | Mean weight gain (g) |
|---|---|
| Earlybird | $22.9^{a}$ |
| Sizefast | $19.0^{ab}$ |
| Linseed | $18.5^{ab}$ |
| None | $15.6^{b}$ |
| Allvit | $14.3^{b}$ |

However we present it, we need to provide some explanation to set out:

- what test we did,
- what the letter codes mean, and
- the critical threshold we used to judge significance.

In this case the information could be presented in a table legend:

> Table 1: Mean weight gain of hatchlings in the five supplement treatments. Means followed by the same letter did not differ significantly (Tukey test, p>0.05).

Letter coding can also be used effectively in a figure. Again, we must ensure all the relevant information is given in the figure legend.

# Part VI

# Experimental Design

# Chapter 17

# Principles of experimental design

## 17.1 Introduction

The data we use to test hypotheses may be generated by recording information from natural systems ('observational studies') or by carrying out some sort of experiment in which the system under study is manipulated in some way ('experimental studies'). There is often considerable scope for deliberately arranging the system to generate data in the best way to test a particular effect when conducting experiments. For this reason, we tend to use the term 'design' primarily in the context of experiments. However, collecting data in both situations requires thought and planning, and many considerations of what is termed experimental design apply equally to observational and experimental studies[1].

The underlying principle of experimental design is: *to extract data from a system so that variation in the data can be unambiguously attributed to the particular process we are investigating.*

To do this, we need to know how to maximise the **statistical power** of an experiment or data collection protocol. Statistical power is the likelihood that a study will detect an effect when there really is an effect present. In statistics, the word 'effect' is an umbrella term for anything measurable we care about, like differences between groups or associations between variables. Statistical

[1] It is worth noting that in reports *experiment* and *observation* should always be distinguished. If we have carried out observations on a natural system of any sort, but where there has been no experimental manipulation of any aspect of the system, that is not an experiment. It would be inappropriate to write in a report: "This experiment consisted of measuring mean stomatal density from thirty trees growing at a range of altitudes." Instead, we might write: "We conducted an observational study measuring mean stomatal density from thirty trees growing at a range of altitudes."

power is influenced by: (1) the size of the effect and (2) the size of the sample used to detect it. Bigger effects are easier to detect than smaller effects, while large samples present greater test sensitivity than small samples. A second consideration is that the less variable the data, the smaller the effects we can detect.

Given these facts, there are obviously two things to do when designing an experiment:

1. Use the maximum feasible sample sizes.
2. Take steps to minimise the variability in the data[2].

Exactly what combination of these is appropriate will depend on the subject area. In a physiological experiment using complex apparatus and monitoring equipment, the scope for replication may be very limited. Here maximum effort should be put into experimentally controlling extraneous sources of variation. With the subject material, this may mean using animals of the same age, reared under the same conditions, of the same stock; it may involve using clones of plant material. It will involve running the experiment under controlled conditions of light and temperature and using measurement methods that are as *precise* as possible. On the other hand, an ecologist studying an organism in the field may have relatively little scope for experimental control of either the material studied or the environmental conditions and may be forced to make relatively crude measurements. In this case, the best approach is to control what can be controlled and then try and maximise the sample size.

## 17.2 Jargon busting

Before we delve further into experimental design concepts, we need to introduce a little bit of statistical jargon. We'll define the terms and then run through an example to better understand them:

- An **experimental unit** is the physical entity assigned to a treatment (see next definition). Examples of possible experimental units are individual clones or organisms.

- A **treatment** is any kind of manipulation applied to experimental units. A group of experimental units that all receive the same treatment is called a **treatment group**.

- Most experiments include one or more complementary groups, called **control groups**. The experimental units in a control group receive either no treatment or some kind of standard treatment.

[2] This variability could be due to all kinds of things: the organisms/material being used; of the experimental conditions; and of the methods of measurement.

- An experimental **factor** is a collection of related treatments and controls, and the different treatments/controls are called the **levels** of that factor.

Here's an example. Suppose we wanted to compare cattle weight gain on four different dietary supplements to determine which is the most effective. We conducted an experiment in which groups of eight cows are given a particular supplement for one month. A fifth group serves as the control group—they do not receive any supplements. At the end of the experiment, we measure how much weight each cow has gained over the month. In this example, individual cows are the experimental units, dietary supplements are the treatments, and the 'no supplement' group is the control group. Together, the four 'supplement type' and the 'no diet' control constitute the five levels of the 'dietary supplement' factor.

Finally, a word of warning—it is common to lump control groups and treatment groups together and just call them 'treatments'. This is fine, but be aware of the distinction between the two.

## 17.3 Replication

We cannot do statistics without understanding the idea of *replication*—the process of assigning several experimental units to the same treatment or combination of treatments. Why does replication matter? Replication affects the power of a statistical test—by increasingly the replication in a study, we increase the sample size available to detect specific effects. Replication is particularly important in biology because the material we work with is often inherently variable and hard to make precise measurements on.

The basic idea is simple: increased replication = more statistical power. However, we have to be very careful about how we replicate…

### 17.3.1 Independence and pseudoreplication

Most statistical tests assume that the data are independent. Independence means that the value of a measurement from one object is not affected by the values of other objects. Common sources of non-independence in biology include:

- genetics - e.g. if a set of mice are taken from the litter of a single female, they are more likely to be similar to each other than mice taken from the litters of several different females.
- geography - e.g. samples from sites close together will experience similar microclimate, have similar soil type etc.
- sampling within biological 'units' - e.g. leaves on a tree will be more similar to each other than to leaves from other trees.

- experimental arrangements in the lab - e.g. plants are grown together in a pot or fish kept in one aquarium will all be affected by the conditions in that pot/aquarium.

Non-independence occurs at many levels in biological data, and in statistical testing, the common consequence of non-independence is *pseudoreplication*. Pseudoreplication is an artificial increase in the sample size caused by using non-independent data. It may be easiest to see what this means by example.

Imagine we are interested in whether plants of a particular species produce flowers with different numbers of petals when grown in two different soil types. We have three plants in each soil type, and each plant produces 4 flowers. If we count the petals in a single flower from each plant, and then test the difference using a *t*-test we get the following result:

| **Soil type** | | **Num. Petals** | | **Mean** | |
|---|---|---|---|---|---|
| | (Plant 1) | (Plant 2) | (Plant 3) | | |
| **Soil type A** | 3 | 4 | 5 | 4 | |
| | (Plant 1) | (Plant 2) | (Plant 3) | | |
| **Soil type B** | 4 | 5 | 6 | 5 | |
| | | | | | **p = 0.29** |

The difference is not significant. Now imagine that instead of sampling a single flower from each plant, we counted the petals of all four flowers on each plant and (incorrectly) use all the values in the analysis (giving an apparent sample size of 12 in each treatment):

| **Soil type** | | **Num. Petals** | | **Mean** | |
|---|---|---|---|---|---|
| | (Plant 1) | (Plant 2) | (Plant 3) | | |
| **Soil type A** | 3, 2, 3, 4 | 4, 4, 3, 5 | 3, 6, 7, 4 | 4 | |
| | (Plant 1) | (Plant 2) | (Plant 3) | | |
| **Soil type B** | 4, 5, 4, 3 | 5, 7, 3, 5 | 6, 5, 7, 6 | 5 | |
| | | | | | **p = 0.009** |

The same difference in the means now appears to be highly significant! The problem here is that the flowers within each plant are not independent - there is variation among plants in petal numbers, but within the plant (perhaps for genetic reasons), the number of petals produced is similar. Because of this non-independence, the apparent significance in the final result is spurious. There are only three independent entities in each soil type treatment—the plants—so the first of the two tests here is correct. The second is *pseudoreplicated*.

To illustrate the effect in a still more obvious way, consider if we were interested in the heights of plants in the two soil types but we only had one plant in Soil A and one in Soil B. If we measure the plants and find they differ somewhat in height, we cannot tell whether this is due to the soil, or just because no two plants are identical. With one plant in each soil, we cannot carry out a statistical test to compare the heights. Now, if it was suggested that we measure the height of each plant 20 times and then used those numbers to do a statistical test to compare the plant heights in the two soils, we would realise that this was an entirely pointless exercise.

There is no more information about the effect of soil type in the two sets of 20 measurements than in the single measurement (except we now know how variable our measuring technique is). And why stop at 20? Why not just keep remeasuring until we have enough numbers to get a significant difference?!

The pitfall of pseudoreplication may seem obvious. However, it can creep into biological studies in quite subtle ways and occurs in a significant number of published studies. One very common problem occurs in ecological studies where different habitats, or experimental plots, are being compared. Say we are looking at zooplankton abundance in two lakes, one with fish and one without. We would normally take a number of samples from each lake and could obviously compare the zooplankton numbers between these two sets of samples. It would be tempting to attribute any differences we observe to the effect of fish. However, this would not be correct.

We have measured the difference in zooplankton between the two lakes (and this is quite a valid thing to do), but the lakes may differ in any number of ways, not just the presence of fish, so it is not correct to interpret our result in relation to the effect of fish. To do this, we would need data on zooplankton abundance in several lakes with fish and several without. In other words, for testing the effect of fish, our replicates should be *whole lakes* with and without the relevant factor (fish), not samples from within a single lake.

Surely it is still better to take lots of samples from each site than just one; it must give a more accurate picture? This is true. Taking several measurements or samples from each object guards against the possibility of the results being influenced by a single, possibly unusual, sample. It would be much more reliable to have twenty zooplankton samples from a lake than just one. This is important, but it is not the same as having measurements from more objects (lakes)—true replication—which increases the power of the statistical test to detect differences among experimental units with respect to the particular factor (e.g. fish / no fish) we are interested in.

In summary, when carrying out an investigation the key question to ask is: What is the *biological unit of replication* relevant to the effect we trying to test? As this implies, the appropriate unit of replication may vary depending on what we are investigating. If we want to test for a difference in the plankton density between two lakes, then taking 10 samples from each lake and comparing them would be the correct approach. But if, as above, we wanted to assess the effect of fish on plankton density, it would be inappropriate—the correct unit of replication in this case is the whole lake and we would therefore want to sample several lakes with and without fish.

## 17.4 Controls

We are told repeatedly, probably starting at primary school, that every experiment must have a control—a reference treatment against which the other treatments can be compared. The idea does sometimes generate confusion since some experiments do not require a control, whereas others may require more than one control. What's more, it can be difficult to pin down what to control for.

In some cases, the appropriate control is obvious. In a toxicity test, we are interested in the mortality due to the toxicant. We want the control to tell us the background mortality rate without toxicants. However, if we measure the movement rates of slugs on surfaces of differing moisture content, no control is required—indeed, none possible. Slugs encounter many different moisture conditions in their daily lives, and there isn't a 'control' moisture level.

More tricky is the situation where the objects we are investigating are affected not just by the treatment we are administering but also by other effects of applying that treatment. The use of control treatments can sometimes address this, but these are now not simply the 'natural' situation. They may have to be specifically designed to mimic certain aspects of the experiment, not others. These sorts of controls are discussed in more detail below.

## 17.5 Confounded and noisy experiments

Unwanted variation comes in two forms:

- The first is *confounding variation.* This occurs when one or more other sources of variation work in parallel to the factor we are investigating and make it hard, or impossible, to attribute any effects we see to a single cause unambiguously. Confounding variation is particularly problematic in observational studies because, by definition, we don't manipulate the factors we're interested in.
- The second is *noise.* This describes variation that is unrelated to the factor we are investigating. Noise adds variability to the results so that it is harder to see and detect statistically any effect of that factor. As noted above, much of experimental design is about improving our ability to account for noise in a statistical analysis.

We will consider these together because some of the techniques for dealing with them apply to both.

### 17.5.1 Confounding

The potential for confounding effects may sometimes be easy to recognise. Suppose we measure growth rates in plants growing at sites of differing altitudes. In that case, several factors all change systematically with altitude (temperature, ultraviolet radiation, precipitation, wind speed, etc.), and it may be hard to use such data to examine the effects of any one of these factors alone. The important thing to remember is that observing a relationship between two variables does not necessarily indicate a causal link. A negative relationship between plant growth and increased precipitation up a mountain may be determined by one or more of the other factors that vary with altitude.

Confounding doesn't just occur in observational studies. Confounding occurs when the administration of a treatment itself generates other unwanted effects—this is called procedural confounding. An example might be in the administration of nutrients to plants. Changing the supply of nitrogen may be done by supplying different levels of a nitrate (NO3) salt (e.g. Mg(NO3)2 or Ca(NO3)2), but how can we be sure that the effects we see are a consequence of nitrogen addition, rather than effects of the magnesium or calcium cations?

### 17.5.2 Noise

Noise in the data can be generated by the same processes that generate confounding. The difference is that noise is generated even when the confounding factors don't align with the treatments. So, going back to measuring growth rates in plants, if we were looking at growth rates of different subspecies of plant on a mountain, then we might find that we can get five samples from each different subspecies, but the samples are scattered across very different altitudes on the mountain. This will add variation to the estimates of growth rate—this is unwanted noise. On the other hand, if the subspecies grow predominantly at different altitudes, the variation due to altitude is confounded with the variation due to subspecies.

## 17.6 Dealing with confounding effects and noise

Confounding effects occur often in biological work and noise of some sort is always present. Techniques for dealing with such effects include: randomisation, blocking, experimental control, and additional treatments.

### 17.6.1 Randomisation

Randomisation is fundamental to experimental design. Although we can identify specific confounding factors and explicitly counter using experimental techniques, we can never anticipate all such factors. Randomisation provides an 'insurance' against the unpredictable confounding effects encountered in experiments. The basic principle is that each experimental unit should be selected or allocated to a particular treatment, 'at random'. This may involve selecting which patients to give a drug and which a placebo at random, or it may include setting out experimental plots at random locations in a field. The important thing is that those who get a particular treatment are randomly selected from all the possible patients or plots.

Randomisation guards against a variety of possible biases and confounding effects, including the inadvertent biases that might be introduced simply in the process of setting up an experiment. For example, if in a toxicological experiment the chemical treatment is set up first and then the control, it may be that the animals caught most easily from the stock tank (the largest? the weakest?) will all end up in the chemical treatment and the remainder in the control, with consequent bias in the death rates observed in the subsequent experiment.

Randomisation is a critical method for guarding against confounding effects. It is the best insurance we have against unwittingly getting another factor working parallel to a treatment. It does not, of course, do anything to reduce noise in the data. In fact, if randomisation removes confounding effectively, it can appear to increase that variation—but it is a necessary cost to pay for being able to interpret treatment effects correctly.

**What does 'at random' mean?**

The *random* bit of the word randomisation has a specific meaning: objects chosen 'at random' are chosen *independently with equal probabilities*. How do we achieve this in practice? First, we need a set of random numbers. For example, if we need to assign 10 experimental units to treatments, we might start with a set of random integers: 4, 3, 5, 8, 7, 1, 10, 9, 6, 2 (attaining such a set is easy with R — e.g. `sample(1:10)`).

Exactly how these numbers are used in setting up the experiment will depend on what is practical. For example, in the toxicological experiment, we might place animals in each of the test containers to be used for the experiment, number each container and then use the first half of the set of random numbers to randomly select half the containers to be the test and use the remainder as the controls.

### 17.6.2 Blocking

Another way of tackling potential confounding effects, and the general heterogeneity of biological material leading to noise, is to organise experimental material into ‘blocks’. This technique, called **blocking**, is arguably the most important experimental design concept after replication. It works as follows:

1. Group the objects being studied into blocks such that variation among objects within blocks is small; variation between blocks may be larger.
2. Each treatment should occur at least once within each block[3].

For example, in an experiment in which mice are reared on three different diets (I, II, III), we might expect the responses of mice from within a particular litter to be fairly similar to each other, but they might be rather different to the responses of mice from different litters. If we have five litters of mice (A … E) it would be sensible to select three mice from each litter (at random) to be allocated to each treatment.

| | | | | | |
|---|---|---|---|---|---|
| I | $A_1$ | $B_1$ | $C_1$ | $D_1$ | $E_1$ |
| II | $A_2$ | $B_2$ | $C_2$ | $D_2$ | $E_2$ |
| II | $A_3$ | $B_3$ | $C_3$ | $D_3$ | $E_3$ |

Here, $A_1$ denotes the first randomly chosen animal from litter $A$, $A_2$ denotes the second randomly chosen animal from litter $A$, and $A_3$ denotes the third randomly chosen animal from litter $A$. Blocking the design like this is guaranteed to increase the power of the experiment to detect effects of the treatments when there are differences between litters.

In the case of only two treatments (e.g. if we just had diets I and II), this type of blocking is simply the pairing of treatments we have encountered in the paired-sample t-test. Blocked designs with more than two blocks are typically analysed using Analysis of Variance (ANOVA). We will learn how to apply ANOVA to a blocked experimental design this in later chapters.

Note that randomisation is important here also. Mice were selected at random from each litter to be allocated to each treatment, and litters are essentially ‘random’ in the sense that they are not deliberately chosen to be different in any particular way. We just anticipate that they are likely to be different in some ways.

Blocking crops up in all sorts of experimental (and non-experimental) study designs. Some examples are given below.

- If plants in an experiment on soil water levels are being grown in pots

[3] Actually, there are special types of experimental design that use blocking, but where each treatment does not appear in every block. These are much more advanced than anything we will cover in this book.

on greenhouse benches, there may be differences in light or temperature at differing distances from the glass. Treatments could be blocked along the gradient—at each position on the bench we have one pot from each treatment. This way, every treatment is represented at each position along the gradient.

- If a field experiment involving several treatments is set up in an environment known to have some spatial variation (e.g., different parts of a field, sections of a river, etc.) setting up one replicate of each treatment in blocks at different locations ensures that no one treatment ends up confounded by some environmental difference, and helps remove noise due to environmental effects in the final analysis.

- An immunity response is being tested using insects kept in a parallel set of laboratory cultures. There are insufficient insects from a single culture to run the whole experiment, so we could set up one replicate of each treatment using insects from each culture. The cultures would be the blocks. We are not interested in the differences between cultures but we want to be able to control and remove variation due to differences between them.

- If the process of collecting and analysing samples from an experiment is very time consuming, we might block the experiment in time. Set up one replicate of each treatment on each of a sequence of days, and then collect the samples after a particular time, again over the same sequence of days. Each replicate has then been run for the same length of time (we would randomize the order in which treatments were sampled each day), and we could then include ‘days’ as a block within the analysis to control for any unknown differences resulting from the different setup, or sample days.

It’s worth saying again: blocking is one of the most important experimental design concepts. Many experimental settings lend themselves to some kind of blocking scheme. If there is a way to block an experiment, we should do it. Why? Because a study is more likely to detect an effect if it uses a blocked design compared to the equivalent non-blocked version.

### 17.6.3 Experimental control

Obviously, some unwanted variation in data will arise if there is poor measurement or careless implementation of the treatments. In every study we do, we should look at the ‘protocol’ issues and see if they can be improved. This means considering the precision of the measurements we are making in relation to the size of effects we are interested in. There would be no point in timing measurement intervals over which seedling growth was determined to the millisecond, but it would be good to measure seedling height using a standard approach and to the nearest millimetre, rather than centimetre.

The second form of experimental control is where we can use experimental manipulation of some sort to control for factors that might vary among replicates or treatments. At its simplest, this involves controlling the other conditions (e.g. temperature) so that all treatments experience identical conditions. It may not always be necessary for the conditions to be *constant*—it may be sufficient that whatever variation occurs is the same for all treatments.

More complex problems arise where the unwanted variation is produced as a by-product of the treatment we administer (procedural confounding again). Suppose we were interested in the effect of leaf litter decomposition on the microbial communities in soils. In that case, we might have an experimental treatment that involves varying the amount of leaf litter placed on the soil surface in test plots. The problem is that this will vary not just the amount of decomposing material entering the soil, but also the physical presence of the leaf litter layer will affect the microclimate at the soil surface (e.g. the dryness in the surface of the soil). So we might create some sort of artificial litter that can be mixed in with the real litter, but which does not decompose so that each plot has a constant volume of 'litter' on the surface, but different amounts of decomposing material entering the soil.

Other situations in which this type of experimental 'adjustment' can be used include experiments in which different nutrient solutions have to be adjusted so that they have the same pH or where different temperature treatments have to have humidity adjusted to ensure that it remains constant. In general, this type of approach can be very useful, but it depends on the necessary adjustment being known and sometimes requires continuous monitoring to keep the adjustments correct.

### 17.6.4 Additional treatments: 'designing in' unwanted variation

Often we are faced by situations in which the unwanted variation — in particular confounding effects — cannot be removed by manipulating the treatments themselves, but has to be tackled by creating additional treatments whose function is to measure the extent of the unwanted variation, and then allow us to remove it statistically, from the data after the experiment is done. In other words, instead of just designing the experiment with the factor we are interested in, we 'design in' the sources of unwanted variation.

#### 17.6.4.1 Transplants and cross-factoring

Imagine we had an investigation that involved looking at the effects of air pollution on the ability of trees to defend themselves chemically against attack by leaf-mining insects. The obvious thing to do would be to look at trees along a gradient of air pollution and monitor leaf damage by the insects. We might find that the insects attack the trees more in polluted areas. However, the problem here is that the trees growing in areas of high air pollution might be attacked more because they are stressed and less able to invest resources in defending themselves (as hypothesised), or because the insects own natural enemies are less abundant in areas of high air pollution, leading to reduced suppression. One way of escaping this confounding effect would be to take tree saplings from polluted and unpolluted areas and do reciprocal transplants—moving trees from polluted areas into clean areas, and vice versa. This then enables us to separate out to a large extent the effect of tree quality from the effect of insect abundance as we can compare trees that have grown with and without air pollution in both polluted and unpolluted areas.

It is also possible that by careful choice of location, or other elements of design, we can include the unwanted variation as an additional factor in the design without necessarily physically manipulating the subjects, but by sampling material systematically with regard to both the thing we are interested in and the additional unwanted factor(s), so that we can *cross-factor* the two. For example, suppose we were interested in how habitat use determines gut parasite load in dogfish. In that case, we might sample dogfish from different habitats and record the sex, age, or size of the fish. It would then be possible to separate the effects of sex or age from those of where the fish were living. If we didn't do this, then both factors would probably contribute unwanted variation or confounding effects (e.g. male and female dogfish have somewhat different habitat preferences).

#### 17.6.4.2 Procedural controls

Confounding effects are not only a problem along natural gradients. The experimental procedures can also introduce them. For example, a marine biologist investigating the effect of crab predation on the density of bivalve molluscs in an estuarine ecosystem might have cages on the mudflats from which crabs are removed and in which any change in bivalve settlement and survival can be monitored. The obvious control for this would be equivalent plots on the adjacent mudflats with normal crab numbers. Suppose the experiment just compares the bivalve density in cages with reduced crab numbers to their density in the adjacent mudflat. In that case, any effects observed could be attributable to crab density, environmental changes brought about by the cages, or disturbance due to the repeated netting to remove crabs. Several additional controls might be useful here. In addition to the proper treatment, bivalve density could be monitored in:

- a 'no cage / no disturbance control'—open mudflat adjacent to the experiment (so no cage effects, no added disturbance).
- a 'cage control'—crabs at normal density but with a cage (usually done as cage with openings to allow crabs to enter and leave).

- a ‘disturbance control’—crabs at normal densities, but subject to the same disturbance as the reduced density treatments (cages netted to remove crabs, but all crabs returned to the cages)

The latter two could be combined if it wasn’t important to separate disturbance and cage effects.

The additional treatments in this sort of situation are effectively additional controls—in fact, they are termed *procedural controls*—but they are not simply the natural ‘background’ conditions. A classic example of this type of control is the use of placebo treatments in medical trials. For example, if we are investigating the effect of a drug, there may be a confounding effect due to psychological, behavioural or even physiological changes in patients resulting simply from being treated, rather than any active compound in the drug. Therefore, it is common to give the drug to one group of patients and a ‘placebo’ to another group. The placebo is a secondary manipulation designed to equalise the effect of simply ‘being treated’ (i.e. equivalent treatment process, but with no active component in the substance administered).

## 17.7 Ethics and practicality

Although experimental design is often fairly straightforward in principle, the ideal design to test an hypothesis may turn out to be impractical, unaffordable or unethical. All experiments are constrained by practicality and finance. Ethical considerations also constrain a rather smaller but important set. Ethical factors constrain experiments in every biological discipline. However, nowhere is the issue more pronounced than in medicine.

Drug testing presents the classic difficulty. Effective testing of a drug’s efficacy depends on comparing patients receiving the drug with closely equivalent patients not doing so, or receiving some alternative treatment. Since one of the treatments will likely be better than another, by definition, at least one group of people have an available and better treatment withheld from them (e.g., Aspinal and Goodman 1995). Thus, as soon as the experimental evidence indicates which treatment is best, it is very hard to justify withholding it from all patients, even if the experimenter feels that further work is necessary.

Good experimental design and appropriate analysis cannot remove ethical, practical or financial problems. Still, they can help ensure that the maximum useful information is returned where time and money are invested in a problem.

# Chapter 18

# Paired-sample t-test

In the previous chapter, we learned how blocking is a valuable technique for dealing with confounding effects. However, we didn't discuss how we would analyse such data. If we've used a block design, we can't just throw our data at something like a *t*-test or a standard one-way ANOVA. Remember, one of the assumptions of these tests is the independence of the experimental units, yet a blocked design creates a situation where some units are not independent. In this chapter and the next, we'll consider versions of the *t*-test and ANOVA that can be used to analyse such non-independent data.

## 18.1 When do we use a paired-sample *t*-test?

There are situations in which data may naturally form pairs of non-independent observations: the first value in sample A is linked to the first value in sample B, the second value in sample A is linked with the second value in sample B, and so on. This is known as a *paired-sample* data. A common example of paired-sample data is when we have a group of organisms, and we record some measurements from each organism before and after an experimental treatment. For example, if we were studying heart rate in relation to position (sitting vs standing), we might measure the heart rate of several people in both positions. In this case, the heart rate of a particular person when sitting is paired with the heart rate of the same person when standing.

An experimental design that sets out to produce paired data is known as a *paired-sample design.* More often than not, paired-sample data arise as a consequence of such deliberate decisions. Why? In biology, we often have the problem that there is a great deal of variation between the units we're studying (individual organisms, tissue samples, etc). There may be so much among-unit variation that the effect of any difference among the situations we're interested in is obscured. Using a paired-sample design gives us a way to control for the among-unit variation and increase the power of our statistical tests. However, we should not use a two-sample *t*-test when our data have this kind of structure. Let's find out why.

## 18.2 Why do we use a paired-sample design?

Consider the following. A drug company wishes to test two drugs for their effectiveness in treating a rare illness in which glycolipids are poorly metabolised. An effective drug is one that lowers glycolipid concentrations in patients. The company can only find 8 patients willing to cooperate in the early trials of the two drugs. What's more, the 8 patients vary in their age, sex, body weight, severity of symptoms and other health problems.

One way to conduct an experiment that evaluates the effect of the new drug is to randomly assign the 8 patients to one or other drug and monitor their performance. However, this kind of design is very unlikely to detect a statistically significant difference between the treatments. This is because it provides very little replication, yet can we expect considerable variability from one person to another in glycolipid levels before any treatment is applied. This variability would lead to a large standard error in the difference between means.

A solution to this problem is to treat each patient *with both drugs in turn* and record the glycolipid concentrations in the blood after taking each drug. One arrangement would be for four patients to start with drug A and four with drug B, and then after a suitable break from the treatments, they could be swapped over onto the other drug. This would give us eight replicate observations on the effectiveness of each drug, and we can determine, for each patient, which drug is more effective.[1]

The experimental design, and one hypothetical outcome, is represented in the diagram below…

Each patient is represented by a unique number (1-8). The order of the drugs in the plot does not matter—it doesn't mean that Drug A was tested before Drug B just because Drug A appears first. Notice that there is a lot of variability in these data, both in the glycolipid levels of each patient and in the amount by which the drugs differ in their effects (e.g. the drugs have roughly equal effects for patient 5, while drug B appears to be more effective for patient 2). What can also be inferred from this pattern is that although the glycolipid levels vary a good deal between patients, Drug B seems to reduce glycolipid levels more than Drug A.

[1] This kind of experimental design is called a cross-over study. It can be problematic if, for example, "carry-over" effects occur, e.g., the effect of one drug is altered when the other drug has previously been administered. We won't worry about these problems here though.

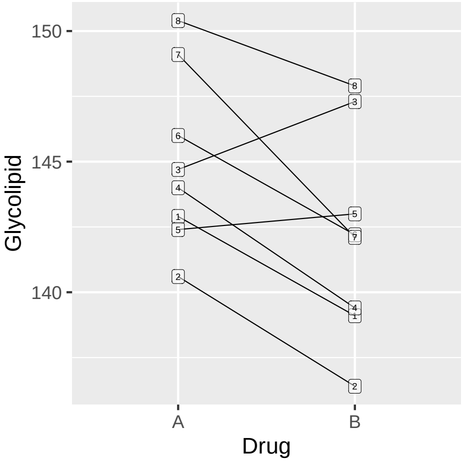


Figure 18.1: Data from glycolipid study, showing paired design. Each patient is denoted by a unique number.

The advantage to using a paired-sample design in this case is clear if we look at the results we might have obtained on the same patients, but where they have been divided into two groups of four, giving one group Drug A and one group Drug B:

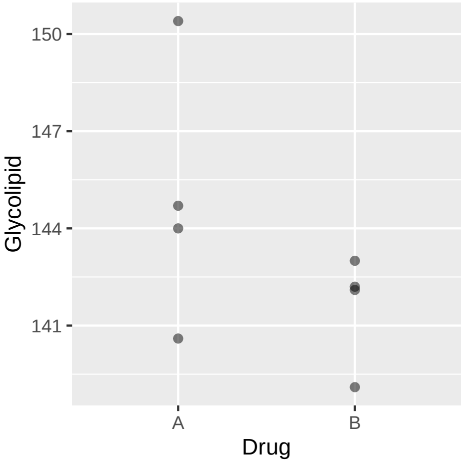


Figure 18.2: Data from glycolipid study, ignoring paired design.

The patients and their glycolipid levels are identical to those in the previous diagram, but now, patients 2, 3, 4 and 8 (selected at random) were given Drug A, while only patients 1, 5, 6, and 7 were given Drug B. The means of the two groups are different, with Drug B performing better, but the associated standard error are also large relative to this difference. A two-sample *t*-test would undoubtedly fail to identify a significant difference between the two drugs.

So, it would be quite possible to end up with two groups where there was no apparent difference in the mean glycolipid levels between the two drug treatments even though Drug B seems to be more effective in the majority of patients. The pairing allows us to factor out (i.e. remove) the variation among individuals and concentrate on the differences between the two treatments. This works by focussing on the differences occurring *within* individuals. The result is a much more sensitive evaluation of the effect we're interested in.

The next question is, how do we go about analysing paired data in a way that properly accounts for the structure in the data?

## 18.3 How do you carry out a *t*-test on paired samples?

It should be clear why a paired-sample design might be useful, but how do we construct the right test? The 'trick' is to work directly with the differences between pairs of values. In the case of the glycolipid levels illustrated in the first diagram, we noted that there was a greater decrease of glycolipids in 75% of patients using Drug B compared with Drug A.

If we calculate the actual differences (i.e. subtracted the value for Drug A from the value for Drug B) for each patient, we might see something like: -3.9, -4.1, 2.5, -4.7, 0.5, -3.7, -6.9, -2.4. Notice that there are only two positive values in this sample of differences, one of which is close to 0. The mean difference is -2.8, i.e. on average, glycolipid levels are lower with Drug B. Another way of stating this observation is that *within subjects* (patients), the mean difference between drug B and drug A is negative. A paired-sample design focusses on the within-subject (or more generally, within-item) change.

On the other hand, if the two drugs had had similar effects, what would we expect to see? We would expect no consistent difference in glycolipid levels between the Drug A and Drug B treatments. Glycolipid levels are unlikely to remain exactly the same over time. Still, there shouldn't be any pattern to these changes with respect to the drug treatment: some patients will show increases, some decreases, and some no change at all. The mean of the differences, in this case, should be somewhere around zero (though sampling variation will ensure it isn't exactly equal to zero).

So, to carry out a *t*-test on paired-sample data, we have to: 1) find the mean of the difference of all the pairs and 2) evaluate whether this is significantly different from zero. We already know how to do this! **This is just an application of the one-sample *t*-test**, where the expected value (i.e. the null hypothesis) is 0. The thing to realise is that although we started out with two sets of values, what matters is the sample of differences between pairs and the population we're interested in a 'population of differences'.

When used to analyse paired data in this way, the test is referred to as a paired-sample *t*-test. This is not wrong, but it is important to remember that a paired-sample *t*-test is just a one-sample *t*-test applied to the sample of differences between pairs of associated observations. A paired-sample *t*-test isn't a new kind of test. Instead, it is a one-sample *t*-test applied to *a particular kind of situation*.

### 18.3.1 Assumptions of the paired-sample *t*-test

The assumptions of a paired-sample *t*-test are no different from the one-sample *t*-test. After all, they boil down to the same test! We just have to be aware of the target sample. The key point to keep in mind is that the sample of differences is important, not the original data. There is no requirement for the original data to be drawn from a normal distribution because the normality assumption applies to the differences. This is very useful because even where the original data seem to be drawn from a non-normal distribution, the differences between pairs can often be acceptably normal. The differences need to be measured on an interval or ratio scale, but this is guaranteed if the original data are on one of these scales.

## 18.4 Carrying out a paired-sample *t*-test in R

R offers the option of a paired-sample *t*-test to save us the effort of calculating differences. It does this for us and then carries out a one-sample test on those differences. We'll look at how to do it the 'old fashioned' way first—calculating the differences ourselves and running a one-sample test—before using the short-cut method provided by R. We'll do this using the glycolipid drug example.

> The data live in the 'GLYCOLIPID.CSV' file. The code below assumes those data have been read into a tibble called `glycolipid`. Set that up if you plan to work along.

We start by looking at the data:

```
glimpse(glycolipid)
```

```
## Rows: 16
## Columns: 4
## $ Patient    <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 1, 2, 3, 4, 5, 6, 7, 8
## $ Sex        <chr> "Male", "Female", "Male", "Female", "Female", "Male", "Fema~
## $ Drug       <chr> "A", "A", "A", "A", "A", "A", "A", "A", "B", "B", "B", "B",~
## $ Glycolipid <dbl> 142.9, 140.6, 144.7, 144.0, 142.4, 146.0, 149.1, 150.4, 139~
```

There are four variables in this data set: `Patient` indexes the patient identity, `Sex` is the sex of the patient (we don't need this), `Drug` denotes the drug treatment, and `Glycolipid` is the glycolipid level.

Next, we need to calculate the differences between each pair. We can do this with the `dplyr` functions `group_by` and `summarise`:

```
glycolipid_diffs <-
  glycolipid %>%
  group_by(Patient) %>%
  summarise(Difference = diff(Glycolipid))
```

What we did was group the data by the values of `Patient`, and then used a function called `diff` to calculate the difference between the two Glycolipid concentrations *within each patient*. We stored the result of this calculation in a new data frame called `glycolipid_diffs`. This is the data we'll use to carry out the paired-sample *t*-test:

```
glycolipid_diffs
```

```
## # A tibble: 8 x 2
##   Patient Difference
##     <dbl>      <dbl>
## 1       1     -3.80
## 2       2     -4.20
## 3       3      2.60
## 4       4     -4.60
## 5       5      0.600
## 6       6     -3.80
## 7       7     -7
## 8       8     -2.5
```

We should check that the differences could plausibly have been drawn from a normal distribution, though keep in mind that the normality assumption is hard to assess with only 8 observations:

```
ggplot(glycolipid_diffs, aes(x = Difference)) +
  geom_dotplot() + theme_grey(base_size = 22)
```

```
## Bin width defaults to 1/30 of the range of the data. Pick better value with `binwid
```

Hmm… there really isn't a lot of data to work with here but there's nothing in

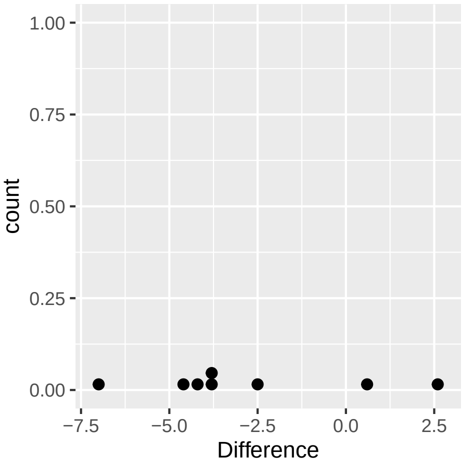


Figure 18.3: Within-individual differences from glycolipid study

that plot that screams 'non-normal', which means the assumptions have been met as best they can be. Let's proceed to carry out a one-sample *t*-test on the calculated differences. This is easy to do in R:

```
t.test(glycolipid_diffs$Difference)
```

```
##
##  One Sample t-test
##
## data:  glycolipid_diffs$Difference
## t = -2.6209, df = 7, p-value = 0.03436
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -5.397549 -0.277451
## sample estimates:
## mean of x
##   -2.8375
```

We don't have to set the `data` argument to carry out a one-sample *t*-test on the differences. We just passed along the numeric vector of differences extracted from `glycolipid_diffs` (using the `$` operator). What happened to the `mu` argument used to set up the null hypothesis? Remember, the null hypothesis is that the population mean is zero. R assumes that this is 0 if we don't supply it, so no need to set it here.

The output is quite familiar… The first line reminds us what kind of test we did, and the second line reminds us what data we used to carry out the test. The third line is the important one: `t = -2.6209, df = 7, *p*-value =`

`0.03436`. This gives the $t$-statistic, the degrees of freedom, and the all-important $p$-value associated with the test. The $p$-value tells us that the mean within-individual difference is significant at the $p < 0.05$ level.

We need to express these results in a clear sentence incorporating the relevant statistical information:

> Individual patients had significantly lower serum glycolipid concentrations when treated with Drug B than when treated with Drug A ($t = 2.62$, d.f. $= 7$, $p < 0.05$).

There are a couple of things to point out in interpreting the result of such a test:

1. The sample of differences was used in the test, not the sample of paired observations. This means the degrees of freedom for a paired-sample $t$ test are one less than the number of differences (= number of pairs); not one, or two, less than the total number of observations.

2. Since we used a paired-sample design our conclusion stresses the fact that the use of the Drug B results in a lower glycolipid level in individual patients. It doesn't say that the use of Drug B resulted in lower glycolipid concentrations for everyone given Drug B than for anyone given Drug A.

### 18.4.1 Using the `paired = TRUE` argument

R has a built-in procedure for doing paired-sample $t$-tests in one step. Now that we've done it the hard way, let's try carrying out the test using that built-in procedure. This looks very similar to a two-sample $t$-test, except that we have to set the `paired` argument of the `t.test` function to `TRUE`:

```
t.test(Glycolipid ~ Drug, data = glycolipid, paired = TRUE)
```

```
##
##  Paired t-test
##
## data:  Glycolipid by Drug
## t = 2.6209, df = 7, p-value = 0.03436
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  0.277451 5.397549
## sample estimates:
## mean difference
##          2.8375
```

R takes care of the differencing for us, so now we can work with the original `glycolipid` data rather than the `glycolipid_diffs` data frame constructed above. We won't step through the output because it should make sense by this point.

**Order matter!**

Be careful when using the built in procedure for doing paired-sample *t*-tests. The only information R uses to associate pairs of observations is their order in each group. The first observation in the 'A' group is paired with the first observation in the 'B' group, the second observation in the 'A' group is paired with the second observation in the 'B' group, and so on. If the items/individuals aren't ordered the same way in each group, the test will be wrong and we'll end up with a meaningless *p*-value!

It's nice that R makes it easy to do a paired-sample *t*-test in one step. Just don't forget that this version of the *t*-test is just a one-sample test on paired differences.

# Chapter 19

# ANOVA for randomised block designs

*Block what you can; randomize what you cannot.*

George Box

## 19.1 Randomised Complete Block Designs

We have only considered one type of experimental ANOVA design up until now: the **Completely Randomised Design** (CRD). The defining feature of a CRD is that treatments are assigned completely at random to experimental units. This is the simplest type of experimental design. Randomisation is a good thing because it prevents systematic biases from creeping in. A CRD approach is often 'good enough' in many situations. However, it isn't necessarily the most powerful design—if at all possible, we should use **blocking** to account for **nuisance factors**. A nuisance factor affects the response—it creates variation—but is of no interest to the experimenter. The variability induced by nuisance factors should be accounted for at the design stage of an experiment if possible. Let's consider a hypothetical experiment to remind ourselves how blocking works.

Imagine we're evaluating the effect of three different fertiliser application rates on wheat yields (t/ha). We suspect that our experimental fields' soil type and management histories are quite different, leading to significant 'field effects' on yield. We don't care about these field effects—they are a nuisance—but we'd like to account for them. There are two factors in play in this setting: the first is the set of experimental treatments (fertiliser application rate); the second is the source of nuisance variation (field). Fertiliser application rate is the 'treatment factor', and field is the 'blocking factor'.

Here is one way to block the experiment. The essence of the design is that a set of fields are chosen, which may differ in various unknown conditions (soil water, aspect, disturbance, etc.), and within each field, three plots are set up. Each plot receives one of the three fertiliser rate treatments at random. If we chose to work with eight fields, the resulting data might look like this:

| | **Fertilizer** | Control | Absent | High |
|---|---|---|---|---|
| **Block** | Field 1 | 9.3 | 8.7 | 10.0 |
| | Field 2 | 8.7 | 7.1 | 9.1 |
| | Field 3 | 9.3 | 8.2 | 10.4 |
| | Field 4 | 9.5 | 8.9 | 10.0 |
| | Field 5 | 9.9 | 9.1 | 10.8 |
| | Field 6 | 8.9 | 8.0 | 9.0 |
| | Field 7 | 8.3 | 6.2 | 8.9 |
| | Field 8 | 9.1 | 7.0 | 8.1 |

Each treatment level is represented in each field (block), but only once. The experiment is 'blocked by field'. Now consider these two questions:

- Why is this design useful? Blocking allows us to partition out the environmental variation due to different field conditions. For example, the three treatments in field 5 produced a relatively high yield, while the yields in field 7 are consistently below average within each treatment. This among field variation is real, and if we hadn't blocked the experiment, it would appear as a 'noise' component of our analysis. But since we blocked the experiment, and every treatment is present in every block, we can 'remove' the block variation from the noise. Less noise means more statistical power.

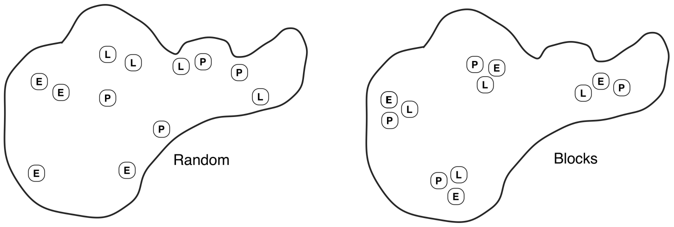
- Why is each treatment level represented only once within blocks? This design gives us the best chance of generalising our results. If we are interested in the overall effect of fertiliser, we should prefer to put our effort into including a range of possible environmental conditions. If we only did the experiment in one field, the results might turn out to be rather unusual. We are not interested in the environmental variation as such. We want to know for a range of conditions, whatever they might be, whether there are consistent differences between fertiliser application rates.

There are many different ways to introduce blocking into an experiment. The most commonly used design—and the one that is easiest to analyse—is called a **Randomised Complete Block Design**. The defining feature of this design is that *each block sees each treatment exactly once*. The fertiliser study is an example of a Randomised Complete Block Design (RCBD). The obvious question is: How do we analyse an RCBD? We'll explore that in moment. First, a small detour….

## 19.2 Analysing an RCBD experiment

Let's consider a new example to drive home how an RCBD works. We want to assess whether there is a difference in the impact that the predatory larvae of three damselfly species (*Enallagma*, *Lestes* and *Pyrrhosoma*) have on the abundance of midge larvae in a pond. We plan to conduct an experiment in which small (1 m$^2$) nylon mesh cages are set up in the pond. All damselfly larvae will be removed from the cages and each cage will then be stocked with 20 individuals of one of the species. After 3 weeks, we will sample the cages and count the density of midge larvae in each. We have 12 cages altogether, so four replicates of each of the three species can be established.

On the face of it, this looks like a one-way design with each species as a treatment. The only problem to resolve is how to distribute the enclosures in the pond. The pond is unlikely to be uniform in depth, substrate, temperature, shade, etc… Some of the variation will be obvious, some will not. We have two options: 1) use a CRD and distribute the cages at random, or 2) adopt an RCBD by grouping the cages into clusters of three, placing each cluster at a randomly chosen location, and assigning the three treatments to cages at random within each cluster. These are illustrated below (left = CRD, right = RCBD):



What are the consequences of the two alternatives? If the cages are distributed at random (CRD), they will cover a wide range of variation in these factors. These sources of variation will almost certainly cause the density of midge larvae to vary around the pond unpredictably, increasing the noise in the data. If we group sets of treatments into clusters, we are creating 'spatial blocks'. There may be considerable differences between blocks, but these won't obscure differences between the treatments because all three treatments are present in every block.

## 19.3 Carrying out the analysis with R

We'll now illustrate how to analyse data from an RCBD experiment using the damselfly predation example.

The data live in the 'DAMSELS.CSV' file. The code below assumes those data have been read into a tibble called `damsels`. Set that up if you plan to work along.

The density of midge larvae in each enclosure after 3 weeks are in the `Midge` variable (number m$^{-2}$); codes for species in the `Species` variable (levels: *Enallagma*, *Lestes* and *Pyrrhosoma*), and the block identities (A, B, C, D) in the third column:

```
glimpse(damsels)
```

```
## Rows: 12
## Columns: 3
## $ Midge   <dbl> 304, 464, 320, 578, 509, 458, 680, 740, 630, 356, 390, 350
## $ Species <chr> "Enallagma", "Lestes", "Pyrrhosoma", "Enallagma", "Lestes", "P~
## $ Block   <chr> "A", "A", "A", "B", "B", "B", "C", "C", "C", "D", "D", "D"
```

The process of analysing an RCBD experiment is essentially the same as any other type of ANOVA. First, we fit the model using the `lm` function, and then we use `anova` to calculate *F*-statistics, degrees of freedom, and *p*-values:

```
damsels.model <- lm(Midge ~ Species + Block, data = damsels)
anova(damsels.model)
```

We suppressed the output for now. Notice that we have put two factors on the right-hand side of the `~` symbol. This tells R that we want to fit a model that accounts for the main effects of `Species` and `Block`. We have to put a `+` between terms to delineate them (this is not optional).

### Careful with your + and *

Notice that we used the 'plus' symbol (`+`) to include the blocking factor in the model: `Species + Block`. R also allows us to use the 'times' symbol (`*`) in a model formula, e.g. `Species * Block`. However, this formulation is not appropriate for including block effects. It is used to set up models for more complicated designs that involve something called an interaction term. We're not going to cover those in this book, so for now, make a mental note of the fact that blocking factors must be included via the `+` symbol.

And by the way… the `+` has nothing to do with addition when working a model specification in R.

Here are the results of the global significance tests using the correct ANOVA model for our randomised block experiment:

```
anova(damsels.model)
```

```
## Analysis of Variance Table
##
## Response: Midge
##           Df Sum Sq Mean Sq F value    Pr(>F)
## Species    2  14904    7452  3.0053 0.1246687
## Block      3 208425   69475 28.0182 0.0006306 ***
## Residuals  6  14878    2480
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What does all this mean? We interpret each line of the ANOVA table in exactly the same way as we do for a one-way ANOVA. The first part tells us what kind of output we are looking at:

```
## Analysis of Variance Table
##
## Response: Midge
```

This reminds us that we are looking at an ANOVA table where our response variable was called `Midge`. The table contains the key information:

```
##           Df Sum Sq Mean Sq F value    Pr(>F)
## Species    2  14904    7452  3.0053 0.1246687
## Block      3 208425   69475 28.0182 0.0006306 ***
## Residuals  6  14878    2480
```

This ANOVA table is similar to the ones we have already seen, except that we now have to consider two lines—one for each term in the model. The first is for the main effect of Species and the second for the main effect of Block.

The $F$-ratio is the test statistic for each term. These provide a measure of how large and consistent the effects associated with each term are. Each $F$-ratio has a pair of degrees of freedom associated with it: one belonging to the term itself, the other due to the error (residual). Together, the $F$-ratio and its degrees of freedom determine the $p$-value.

The $p$-value gives the probability that the differences between the set of means for each term in the model, or a more extreme difference, could have arisen through sampling variation under the null hypothesis of no difference. We take $p < 0.05$ as evidence that at least one of the treatments is having an effect. Here, there is a significant effect of block ($p < 0.001$), which says that the density of midge larvae varies across the lake. It looks like blocking was a good idea—there is a lot of spatial (nuisance) variation in midge larvae density. However, what we care about is the damselfly species effect. This term is not significant ($p >$

0.05), so we conclude that there is no difference in the impact of the predatory larvae of three damselfly species.

It is instructive to see what happens if we analyse the damselfly data as though they are from a one-way design. We do this by including only the experimental treatment term (Species) in the model:

```
damsels.oneway <- lm(Midge ~ Species, data = damsels)
anova(damsels.oneway)
```

```
## Analysis of Variance Table
##
## Response: Midge
##           Df Sum Sq Mean Sq F value Pr(>F)
## Species    2  14904  7452.1  0.3003 0.7477
## Residuals  9 223303 24811.4
```

Look at the degrees of freedom and the sums of squares of the residual (error). How do these compare to the previous model that accounted for the block effect? The degrees of freedom is higher. In principle, this is a good thing because it means we have more power to detect a significant difference among the treatment means. However, the error sum of squares is also much higher when we ignore the block effect. We accounted for much less noise by ignoring the block effect. As a result, the error mean square is a lot lower, and so the *F*-statistic associated with the treatment effect is also much lower. The take-home message is that designing a blocked experiment and properly accounting for the blocked structure will result in a more powerful analysis.

## 19.4 Are there disadvantages to randomised block designs?

The short answer is no, not really. There are instances when a randomised block design might appear to be disadvantageous at first glance, but these don't stand up to criticism:

1. What if we want to know if the effect of treatment varies across blocks? For example, we might be interested in whether or not damselfly effects on the midge densities are consistent in all habitat areas (e.g. some species may forage more effectively in muddy areas, others where there are more leaves). If this is the question we are trying to answer, we should have designed a different experiment. For example, a design in which we consider treatment combinations of different midge species and habitat characteristics might be appropriate. Fundamentally, the goal of blocking is to account for *uncontrolled* variation. Designing a blocked experiment and

then lamenting the fact that we can't fully evaluate differences among blocks is a good example of trying to "have our cake and eat it too".

2. If the blocking term does not account for some variation, the analysis may be slightly less powerful than just using a one-way ANOVA. This is because there are fewer error degrees of freedom associated with the blocked analysis (we lose a few to the block effects). This argument only works if the block effect accounts for minimal variation. We can never know before we start an experiment whether or not blocking is needed, but we do know that biological systems are inherently noisy, with many sources of uncontrolled variation coming into play. In most experimental settings, we can be fairly sure that blocking will 'work'. If we choose not to block an experiment, there is no way to account for uncontrolled variation, and we will almost certainly lose statistical power as a result.

The advice contained in the quote at the beginning of this chapter is probably the best experimental design advice ever dished out: "Block what you can; randomize what you cannot."

## 19.5 Multiple blocking factors

It is common to find ourselves in a situation whereby we need to account for more than one blocking factor. The simplest option is to combine the nuisance factors into a single factor. However, this isn't always possible or even desirable. Consider an instance where there is a single factor of primary interest (the treatment factor) and two nuisance factors. For example, imagine that we want to test the efficacy of three drugs (A, B and C) to alleviate the symptoms of a disease. Three patients are available for a trial, and each will be available for three weeks. Testing a single drug requires a week, meaning an experimental unit is a 'patient-week'. The obvious question is, how should we randomise treatments across 'patient-weeks'?

We have to design an experiment like this with great care, or there is a risk that we will not be able to statistically separate the treatment effects (drug) and block effects (week & patient). The most appropriate design for this kind of experiment has the following structure:

| Week | Patient | Drug |
|---|---|---|
| 1 | 1 | A |
| 1 | 2 | B |
| 1 | 3 | C |
| 2 | 1 | C |
| 2 | 2 | A |
| 2 | 3 | B |
| 3 | 1 | B |

| Week | Patient | Drug |
|---|---|---|
| 3 | 2 | C |
| 3 | 3 | A |

This kind of experimental design is called a Latin square design. It gets its name from the fact that if we organise the treatments into the rows and columns of a grid according to week and patient number, we arrive at something like this:

```
A B C
C A B
B C A
```

Each letter appears once in each column and row. This is called a Latin square arrangement. Latin square designs (and their more exotic friends, e.g. 'Hyper-Graeco-Latin square designs!') have a very useful property: they allow us to unambiguously separate treatment and block effects. The reasoning behind this conclusion is quite technical so we won't try to explain it. We just want to demonstrate that it is perfectly possible to block an experiment by more than one factor, though this needs to be done with care.

# Part VII

# Checking and Fixing Models

# Chapter 20

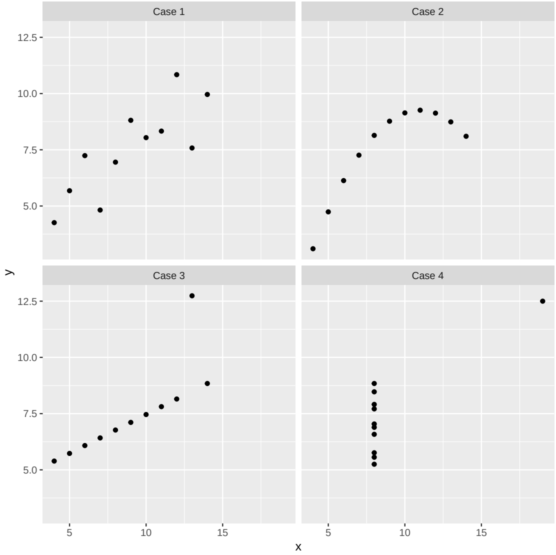
# Assumptions and diagnostics

We often have a statistical analysis in mind when we collect some data. Though it can be tempting to jump straight into that analysis without first examining the data, this is seldom a good idea. We began each investigation in the last few chapters by inspecting the data set to understand what it was telling us. The initial inspection also provides an opportunity to evaluate whether or not our planned analysis assumptions are likely to be violated.

Assumptions are important. We learnt in previous chapters that regression and ANOVA are special cases of the general linear model. This is a parametric model, which makes several assumptions about the data that we need to be aware of. If our data do not meet those assumptions, at least approximately, then we cannot rely on the results given by any associated tests. This chapter aims to review those assumptions for simple regression and one-way ANOVA models.

## 20.1 Understanding data

We've been using 'well-behaved' data sets in this book so far, which tends to give the impression that visual inspections of the data are not all that necessary. Here's an example of why it matters. Imagine we are interested in quantifying the relationship between two variables called $x$ and $y$. We might be tempted to carry out a linear regression analysis without first inspecting these data to get straight to 'the answer': the coefficients of the linear regression model. This could be very misleading. Take a look at these four scatter plots:



These four artificial data sets were constructed by the statistician Francis Anscombe. The means and variances of $x$ and $y$ are nearly identical in all four data sets. What's more, the intercepts and slopes of the best fit regression lines are almost identical—they are 3.0 and 0.5, respectively. The nature of the relationship between $x$ and $y$ is quite obviously different among the four cases:

1. "Case 1" shows two linearly related, normally distributed variables. This is the kind of data we often hope for in a statistical analysis.

2. "Case 2" shows two variables that are not normally distributed, but there is a perfect non-linear relationship between the two.

3. "Case 3" shows an example the variables are perfectly linearly associated for all but one observation which ruins the perfect relationship.

4. "Case 4" shows an example where a single outlier generates an apparent relationship where the two variables are otherwise unrelated.

Each plot tells a different story about the relationship between $x$ and $y$, yet the linear regression model says the relationship is the same in each case. These are obviously somewhat pathological examples, but they illustrate the issues that can and do arise with real data. There is a real risk we will apply an inappropriate analysis if we fail to detect such problems.

Every statistical model makes certain assumptions about the data[1]. Even if a dataset doesn't exhibit the pronounced problems seen in the Anscombe examples, we still need to assess whether the assumptions of the statistical model we want to use are likely to be valid. For example, when working with a lin-

[1] Even so-called 'non-parametric' models have underpinning assumptions; these are just not as restrictive as their parametric counterparts.

ear regression model, we started with a scatter plot of the response variable vs the predictor variable. This allowed us to assess whether the two variables are linearly related. However, as we noted at the time, linearity is not the only assumption we need to consider when carrying out a linear regression. In the rest of this chapter, we'll go through the remaining regression assumptions and then consider the assumptions for a one-way ANOVA.

In the later regression diagnostics chapters, we'll move on to how to check whether these assumptions are valid with your data.

## 20.2 Assumptions of regression

Let's consider each of the assumptions of regression in their approximate order of importance:

1. **Independence.** The residuals must be independent. Another way of stating this assumption is that the value of each residual does not depend on the value of any others. This can be difficult to check. If the data are from a carefully designed experiment, everything should be OK. If the data are observational, then we need to be a lot more careful. This assumption matters because when the residuals are not independent, any $p$-values we generate will be unreliable.

2. **Measurement scale.** The response ($y$) and predictor ($x$) variables are measured on an interval or ratio scale. It doesn't really make sense to use categorical data in a regression[2]. This one is easy to assess.

3. **Linearity.** The relationship between the predictor $x$ variable and the response $y$ variable is linear. There is little point in fitting a straight line to data that don't form a straight-line relationship. There may also be circumstances in which it is theoretically unlikely for a relationship to be linear, e.g. a linear relationship will not well describe the length and weight of an animal because weight is roughly a cubic function of length.

4. **Constant variance.** The variance of the residuals is constant. This assumption essentially means the variability of the residuals is not related to the value of the predictor $x$ variable. It is violated if the magnitudes of the residuals increase or decrease markedly as $x$ gets larger.

5. **Normality.** The residuals are drawn from a normal distribution. This essentially means that for a particular value of $x$ we expect a range of responses in $y$ which follow a normal distribution. Note that the distribution of the residuals is assumed to be normal, not the raw $y$ values. This

[2] It sometimes makes sense to use a regression analysis when the predictor variable is an ordinal categorical variable. It depends what you want to do with the resulting model. However, some people disagree with this approach, so it's best to avoid doing it unless you're confident you can justify it.

means we can generally only test this assumption *after* the model has been fitted. Note that regression is reasonably robust to small departures from normality, but larger departures can start to matter.

6. **Measurement error.** The values of the predictor $x$ variable are determined with negligible measurement error[3]. It is usually impossible to obtain the $x$ values with absolutely no measurement error, but the error $x$ in should at least be much smaller than that in the $y$ values. For example, in a thermal tolerance experiment, the temperature values set by the experimenter will often have low error, so it is fine to use simple regression.

Assumptions 1 (independence), 2 (measurement scale) and 6 (measurement error) are features of experimental design and data collection protocols. The only way to evaluate these assumptions is by thinking about where the data come from rather than examining them somehow. If it seems likely that one of these assumptions is violated, there's not much we can do about it after the data have been collected. This is why it's so important to consider how we will collect and analyse a data set before going ahead with the work. There's no point expending vast amounts of time and effort to collect data that we can't then usefully analyse.

What about the remaining assumptions: linearity, constant variance and normality? These can be checked using a set of visual tools called 'regression diagnostics'. Regression diagnostics use properties of the fitted model to understand how well it fits the data and thereby evaluate the model assumptions. This means they can only be applied after we have fitted a model to the data.

## 20.3 Regression diagnostics

The best way to learn about regression diagnostics by working through an example. Our goal is to learn how to check the linearity, constant variance and normality assumptions by creating a specific diagnostic plot for each one.

These data can be found in the 'DIAGNOSTICS_EG.CSV' file. The code below assumes they have been read into a tibble called `example`. Set that up if you plan to work along.

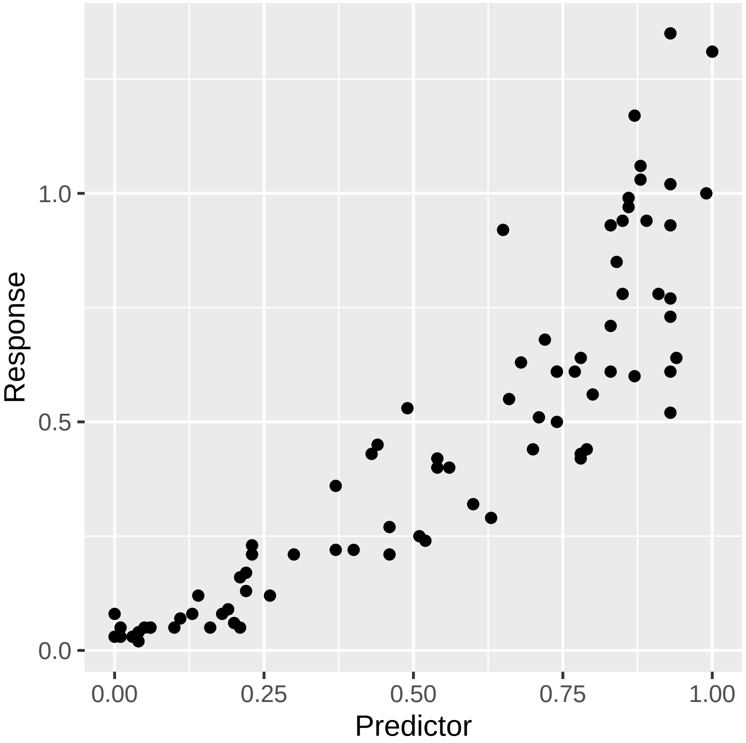
The data set contains two numeric variables: `Predictor` and `Response`:

```
glimpse(example)
```

[3] It is the measurement error, not the sampling error, that matters. This means it is fine to use regression when the $x$ variable represent a sample from a population.

```
## Rows: 80
## Columns: 2
## $ Predictor <dbl> 0.00, 0.00, 0.01, 0.01, 0.03, 0.04, 0.04, 0.05, 0.06, 0.06, ~
## $ Response  <dbl> 0.08, 0.03, 0.05, 0.03, 0.03, 0.04, 0.02, 0.05, 0.05, 0.05, ~
```

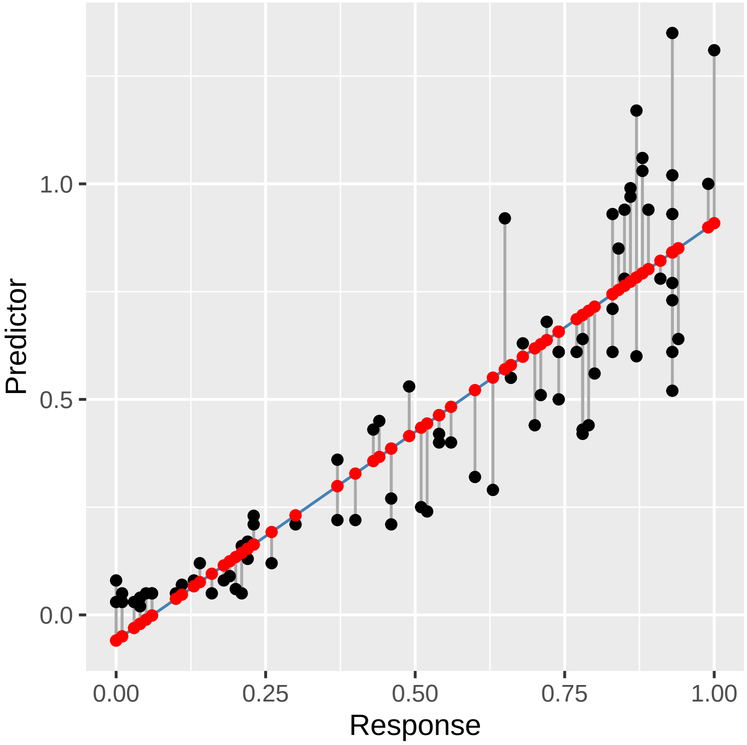
The precise details of where the data come from are not important in this example. All we need to know is that the goal is to construct a regression model to understand how `Predictor` depends on `Response`. We can show their relationship using a scatter plot:



It is worth spending a bit of time staring at that plot. Does it look like these data satisfy the linearity assumption? What about the constant variance assumption? Can we say anything about the normality of the residuals from a plot like this?

### 20.3.1 Fitted values

To understand regression diagnostics we have to know what a **fitted value** is. The phrase 'fitted value' is simply another expression for 'predicted value'. Look at the plot below:



This shows the raw data (black points), the line of best fit (blue line), the residuals (the vertical grey lines), and the fitted values (red points). The line of best fit was created using the workflow we introduced in the regression chapter, using the following fitted model:

```
example_mod <- lm(Response ~ Predictor, data = example)
```

How do we find the fitted values? Draw a vertical line from each observation to the line of best fit. The values of the response variable at the point where these meet the line of best fit are the 'fitted values'. The fitted values are just predictions from the statistical model generated for each value of the predictor variable.

In reality, R does this for us. If we ever need to find the fitted values of a model we can extract these with the `fitted` function:

```
fitted(example_mod)
```

```
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## -0.1 -0.1  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.1  0.1  0.1  0.1
##   17   18   19   20   21   22   23   24   25   26   27   28   29   30   31   32
##  0.1  0.1  0.1  0.1  0.2  0.2  0.2  0.2  0.2  0.2  0.3  0.3  0.3  0.4  0.4  0.4
##   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48
##  0.4  0.4  0.4  0.4  0.5  0.5  0.5  0.5  0.6  0.6  0.6  0.6  0.6  0.6  0.6  0.7
##   49   50   51   52   53   54   55   56   57   58   59   60   61   62   63   64
##  0.7  0.7  0.7  0.7  0.7  0.7  0.7  0.7  0.7  0.7  0.7  0.8  0.8  0.8  0.8  0.8
##   65   66   67   68   69   70   71   72   73   74   75   76   77   78   79   80
##  0.8  0.8  0.8  0.8  0.8  0.8  0.8  0.8  0.8  0.8  0.8  0.8  0.8  0.9  0.9  0.9
```

### 20.3.2 Checking the linearity assumption

The linearity assumption states that the general relationship between the response and predictor variable should look like a straight line. We can evaluate this assumption by constructing a **residuals vs fitted values** plot.

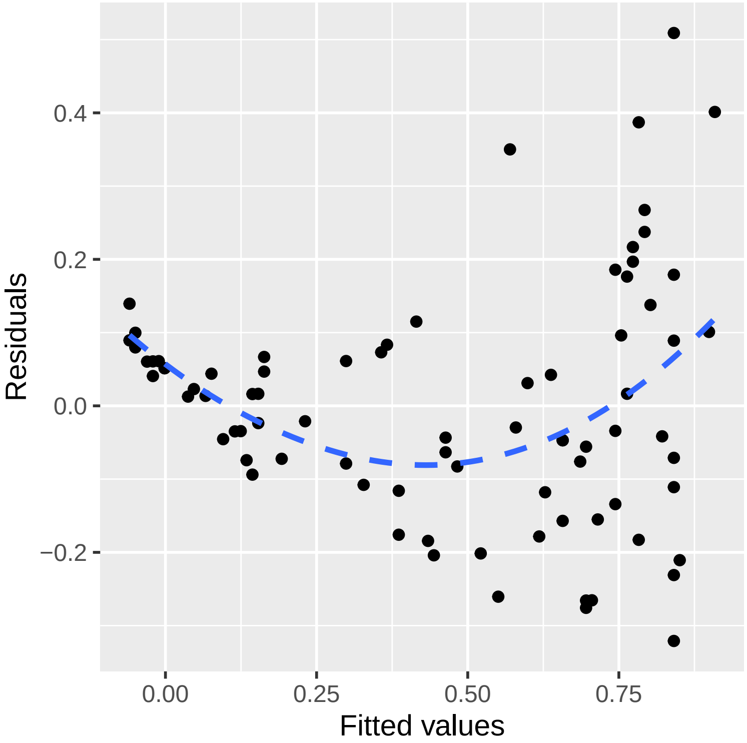
We can make this in two steps. First, use the `fitted` and `resid` functions to construct a data frame containing the fitted values and residuals from the model (called `plt_data`):

```
plt_data <-
  data.frame(Fitted = fitted(example_mod),
             Resids =  resid(example_mod))
```

Once we have made this data frame, we use `ggplot2` to plot the residuals against the fitted values:

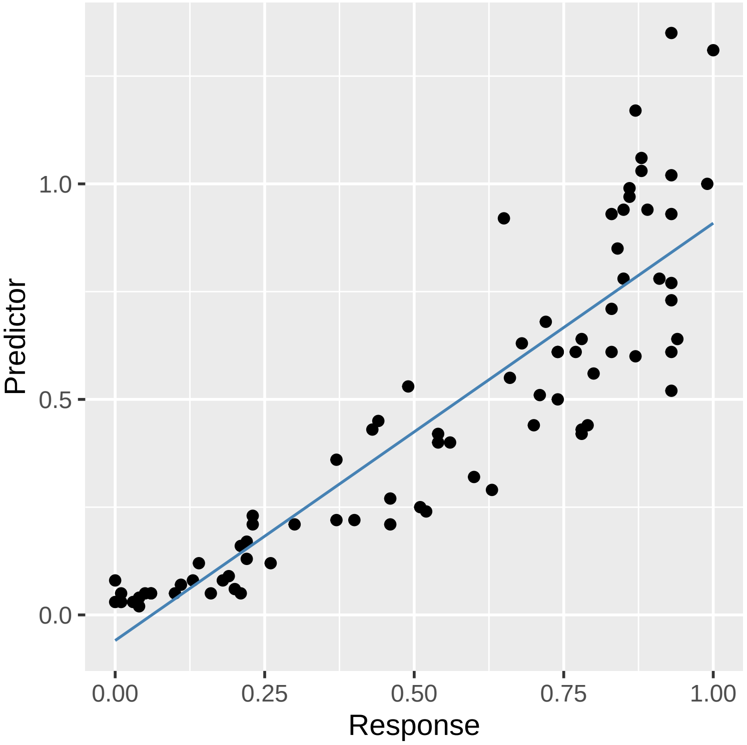
```
ggplot(plt_data, aes(x = Fitted, y = Resids)) +
  geom_point() +
  geom_smooth(span = 2, se = FALSE, linetype = 2) +
  xlab("Fitted values") + ylab("Residuals")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



For a well-behaved model, there should be no relationship between the residuals and fitted values. In general terms, a pattern in this plot indicates that some aspect of the relationship between the predictor and response variables is not captured by the model we used. We added that blue trend line using `geom_smooth` to help reveal the pattern. This shows that the residuals tend to be positive at the largest and smallest fitted values and generally negative in the middle of the range. This u-shaped pattern is indicative of a potential problem with our model.

We can see where this pattern comes from when we look at the raw data and the fitted model:



There is some curvature in the relationship between the two variables, yet we fitted a straight line through the data. The U-shape in the residuals vs fitted values plot arises because there is an ‘accelerating’ (or ‘convex’) relationship between the response and predictor variables that we failed to capture with the model. In other words, the pattern in the diagnostic plot is telling us that the true relationship is non-linear.

What other kinds of patterns might we see in a residuals vs fitted values plot? Many things are possible, but a hump-shape is another very common pattern. This corresponds to the case where the residuals tend to be negative at the largest and smallest fitted values and positive in the middle of the range. A hump shape in the residuals vs fitted values plot will arise when there is a ‘decelerating’ (or ‘concave’) relationship between the response and predictor variables that we failed to capture with the model.

This may seem like a lot of extra work to evaluate an aspect of the model that we can assess by plotting the data. This is true when we are working with a simple linear regression model. However, it is much harder to evaluate the linearity assumption when working with more complicated models that include multiple predictor variables[4]. In that situation, a residuals vs fitted values plot

[4] This is the situation we face with multiple regression. A multiple regression is a type of regression with more than one predictor variable—we don’t study them in this book, but they are often used in biology.

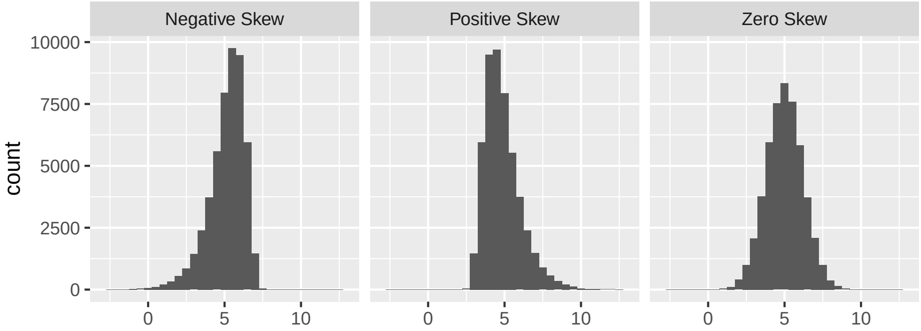
gives us a powerful way to evaluate whether or not the assumption of a linear relationship is reasonable.

That's enough about residuals vs fitted values plots. Let's move on to the normality evaluation.

### 20.3.3 Checking the normality assumption

How do we evaluate the normality assumption of linear regression? We need to assess whether or not the residuals are drawn from a normal distribution. One way to do this is by extracting the residuals from a model and plotting their distribution with a histogram of dot plot. However, there is a more powerful graphical tool: the **normal probability plot**.

A normal probability allows us to identify departures from normality. If we know what we are looking for, we can use it to identify many different kinds of problems. We will focus on the most common type of assessment: determining whether the distribution of the residuals is excessively **skewed**. A skewed distribution is one that is not symmetric. For example, the first distribution below is skewed to the left ('negative skew'), the second is skewed to the right ('positive skew'), and the third is symmetric ('zero skew'):



The skewness in the first two distributions is easy to spot because they contain a lot of data and the asymmetry is quite pronounced. A normal probability plot allows us to pick up potential problems when we are not so lucky.

The methodology underlying the construction of a normal probability plot is quite technical, so don't worry if the next segment is confusing—interpreting a normal probability plot is much easier than making one. We start by extracting the residuals from the fitted model into a numeric vector using the `resids` function, and then standardise these by dividing them by their standard deviation:

```r
# 1. extract the residuals
mod_resids <- resid(example_mod)
```

```
# 2. standardise them
mod_resids <- mod_resids / sd(mod_resids)
```

The standardisation step is not essential, but dividing the residuals by their standard deviation ensures that the standard deviation of the new residuals is equal to 1. Standardising the residuals this way makes it a little easier to compare more than one normal probability plot. We call these new residuals the ‘standardised residuals’.

The next step is to find the rank order of each residual. To do this, we sort the data from lowest to highest and find the position of each case in the sequence (its ‘rank’). The function `order` can do this for us:

```
resid_order <- order(mod_resids)
resid_order
```

```
##  [1] 74 53 54 55 41 75 78 36 40 35 66 45 32 49 56 57 46 33 76 29 19 39 28 51 18
## [26] 25 71 37 52 48 50 15 38 70 16 17 59 43 22 26 11 13 20 21 62 12 44  7 47 14
## [51] 24  9 10  5  6  8 27 23 30  4 31 77  2 60  3 79 34 69  1 61 72 58 64 63 67
## [76] 68 42 65 80 73
```

This tells us that the first residual is the 74th largest, the second is the 53rd largest, the third is the 54th largest, and so on.

The last step is the tricky one. Once we have established the rank order of the residuals, we ask the following question: if the residuals were drawn from a normal distribution, what is their most likely value based on their rank? We can’t explain how to do this without delving into the mathematics of distributions, so this will have to be a ‘trust us’ situation.

As usual, R can do the hard calculations for us. We don’t even need the ranks—we only calculated those to help explain what happens when we build a normal probability plot. The function we need is called `qqnorm`:

```
all_resids <- qqnorm(mod_resids, plot.it = FALSE)
all_resids <- as.data.frame(all_resids)
```

The `qqnorm` doesn’t produce a data frame by default, so we converted the result using a function called `as.data.frame`.

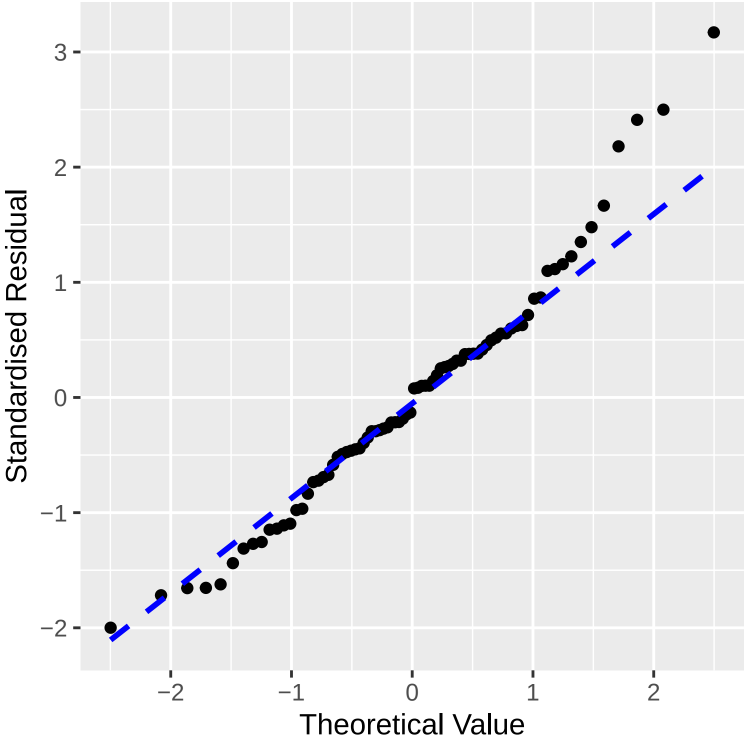
The `all_resids` object is now a data frame with two variables: `x` contains the theoretical values of normally distributed residuals based on the rank orders of the residuals from the model, and `y` contains the standardised residuals. Here are the first 10 values:

```
head(all_resids, 10)
```

```
##              x         y
## 1  1.0636219 0.8684423
## 2  0.7764218 0.5570255
## 3  0.8641600 0.6212898
## 4  0.6549499 0.4967230
## 5  0.4364642 0.3761182
## 6  0.4711971 0.3780992
## 7  0.2372021 0.2535324
## 8  0.5065081 0.3800801
## 9  0.3685006 0.3197777
## 10 0.4022501 0.3197777
```

Finally, we can plot these against one another to make a normal probability plot:

```
ggplot(all_resids, aes(x = x, y = y)) +
  geom_point() +
  geom_qq_line(aes(sample = y), colour = "blue", size = 1, linetype = 2) +
  xlab("Theoretical Value") + ylab("Standardised Residual")
```

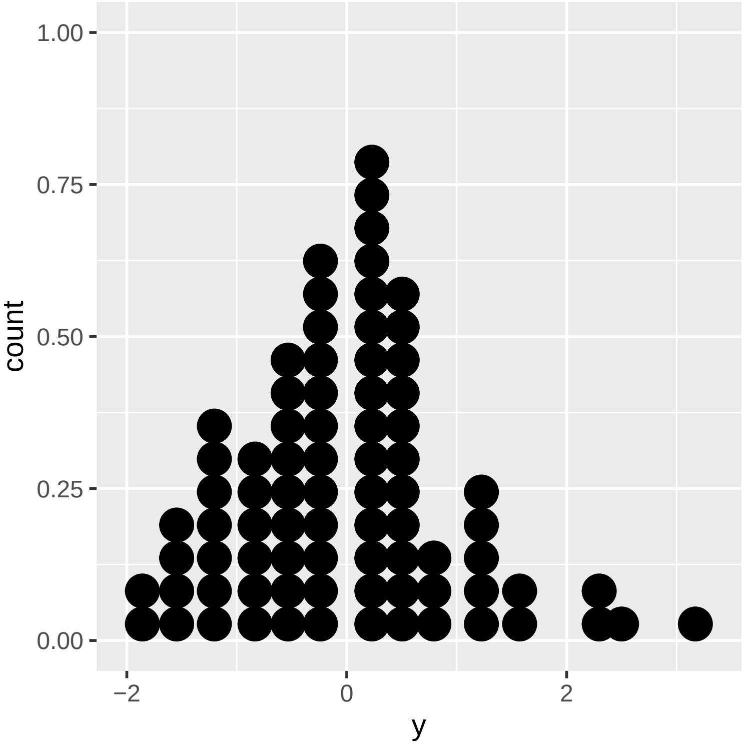


At the beginning of this section, we said that it's not essential to understand how a normal probability plot is constructed. It's only important to know how to interpret one. The critical feature to look out for is the curvature of the relationship between the standardised residuals and their theoretical values. If the residuals really were drawn from a normal distribution, they should generally match the theoretical values, and the points should lie on a straight line.

What we are looking for are systematic deviations from the expected relationship. We used `geom_qq_line` to help us make this assessment. The details aren't too important, but in a nutshell, `geom_qq_line` adds a line to summarise the expected relationship based on the middle 50% of residuals. Deviations from that line relationship at either end of the normal probability plot suggest a departure from normality.

That is what we see above. The largest positive residuals on the right-hand side of the plot lie above the line. What's more, they start to move above the line and then get further away from it as we continue rightward. This is telling us something about the distribution of the residuals. It suggests their distribution is skewed to the right ('positive skew'). To see this, we can visualise their distribution as a dot plot:

```
ggplot(all_resids, aes(x = y)) +
  geom_dotplot(binwidth = 0.3)
```



This shows the asymmetry in the distribution of the residuals—the left 'tail' is bunched up toward the middle of the distribution, while the right tail stretches away from it.

Normal probability plots are useful because they better reveal potential distributional problems than histograms or dot plots of raw residuals. This is particularly true when we have a small data set. Keep in mind that the relationship will never be perfect due to sampling variation—it is very common to see the points to 'wobble' around the line. This isn't something to worry about. If there are no systematic departures from the line at either end of the plot, that is sufficient to conclude that the residuals are probably normally distributed.

That's enough discussion of normal probability plots. Let's move on to the constant variance evaluation.

### 20.3.4 Checking the constant variance assumption

How do we evaluate the constant variance assumption of a linear regression? That is, how do we assess whether or not the variability of the residuals is constant or not? This assumption can be evaluated by producing something called a **scale-location plot**. We construct this by plotting residuals against the fitted values, but instead of plotting raw residuals, we transform them first using the following 'recipe':

1. Standardise the residuals by dividing them by their standard deviation. Remember, this ensures the new residuals have a standard deviation of 1.

2. Find the absolute value of the residuals produced in step 1. If they are negative, make them positive; otherwise, leave them alone.

3. Take the square root of the residuals produced in step 2.

These calculations are simple enough in R. We'll demonstrate them using the partridge data set again:

```r
# extract the residuals
sqrt_abs_resids <- resid(example_mod)
# step 1. standardise them
sqrt_abs_resids <- sqrt_abs_resids / sd(sqrt_abs_resids)
# step 2. find their absolute value
sqrt_abs_resids <- abs(sqrt_abs_resids)
# step 3. square root these
sqrt_abs_resids <- sqrt(sqrt_abs_resids)
```

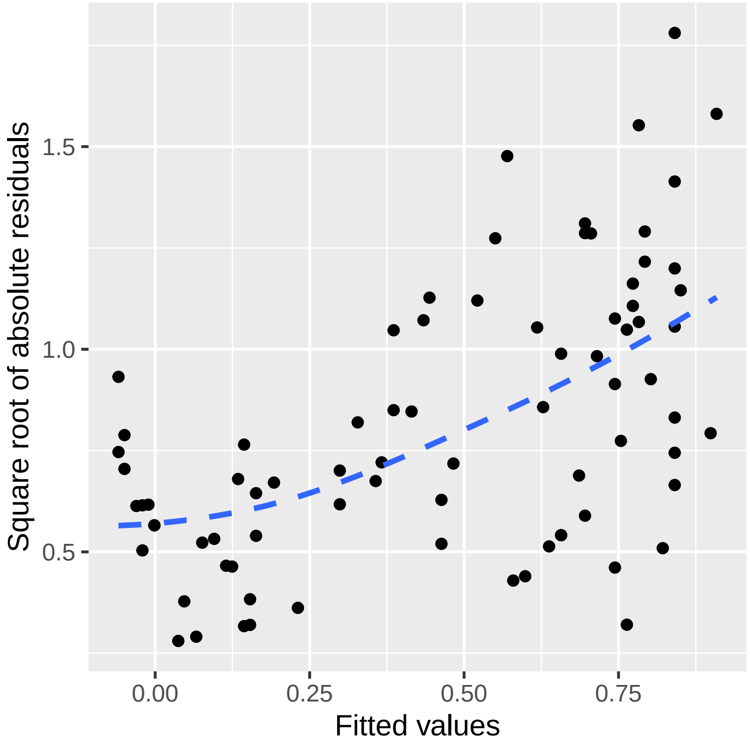
Now we use the `fitted` function to extract the fitted values from the model and place these in a data frame with the transformed residuals:

```
plt_data <-
  data.frame(Fitted = fitted(example_mod), Resids = sqrt_abs_resids)
```

We called the data frame `plt_data`. Once we have made this data frame, we use `ggplot2` to plot the transformed residuals against the fitted values:

```
ggplot(plt_data, aes(x = Fitted, y = Resids)) +
  geom_point() +
  geom_smooth(span = 2, se = FALSE, linetype = 2) +
  xlab("Fitted values") + ylab("Square root of absolute residuals")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



This is a scale-location plot. Why is this useful? We want to assess whether or not the sizes of these new residuals increase or decrease as the fitted values get larger. If they do not and the relationship is essentially flat, we can conclude that the variability in the residuals is constant. Otherwise, we have to conclude that the constant variance assumption is likely violated.

The pattern in the above plot is reasonably clear-cut. There is a positive trend, meaning the absolute size of the residuals tends to get larger as the fitted values increase. That is the same as saying the amount of variation in the residuals is positively related to the fitted values, i.e. the variance of the residuals is *not* constant.

### 20.3.5 Summary of the diagnostic checks

We have now seen how to construct three very useful plots for assessing assumptions and diagnosing problems with a model (a.k.a. regression diagnostics):

1. The **residuals vs fitted values plot** was used to assess whether the relationship between the two variables follows a straight line. We found that a straight-line model is a poor choice because there is an upward-curving relationship.
2. The **normal probability plot** was used to assess whether the residuals follow a normal distribution. We found some evidence that their distribution is asymmetric and right-skewed, meaning the normality assumption may not be justified.
3. The **scale-location plot** was used to assess whether the variability of the residuals is constant. We found evidence that their variance increases with the fitted values.

Taken together, these three diagnostics plots suggest that our simple linear regression model is not a good fit for the data in our example. Why might this matter? A poorly fitting model can lead to unreliable inference about the world. For example:

- using the simple regression model would generate biased predictions about the likely values of response variable, and
- the non-constant variance is a problem because it will affect the validity of any $p$-values associated with a model.

This is why it is crucial to evaluate the assumptions of a model and seek solutions when problems are identified. For example, if one or more assumptions are not met, applying a mathematical transformation to the response variable might help. We will come back to this idea in a later chapter. Finally, remember that interpreting diagnostic plots is as much art as science. We're often forced to work with limited samples, and diagnostic plots will always show mild deviations from the ideal relationship.

## 20.4 Assumptions of one-way ANOVA

The assumptions of one-way ANOVA are very similar to those of simple regression. This is not surprising once you know they are both special cases of the general linear model. However, there are some differences. For example, the idea of a linearity assumption does not make sense with one-way ANOVA because the predictor variable is not numeric. Let's step through each of the assumptions:

1. **Independence.** If the experimental units of the data are not independent, then the $p$-values generated by an $F$-test in an ANOVA will not be reliable. This one is important. Even mild non-independence can be a serious problem.

2. **Measurement scale.** The response variable ($y$) should be measured on an interval or ratio scale.

3. **Constant variance.** The validity of statistical tests associated with ANOVA depends on an assumption of equal variance in the treatment groups. This assumption essentially means the variability of the residuals is not related to the group means. It is violated if the magnitudes of the residuals increase or decrease markedly as the group means get larger. If the data do not support this assumption, at least approximately, the $p$-values we generate may be unreliable.

4. **Normality.** The validity of statistical tests associated with ANOVA also depends on an assumption that the residuals are drawn from a normal distribution. As with regression, ANOVA is reasonably robust to small departures from normality, but larger departures can start to matter. However, unlike with the $t$-test, having a large sample size does not make this assumption less important.

How do we assess these assumptions? By using regression diagnostic plots! The term 'regression diagnostic' is a bit of a misnomer. A more accurate term might be 'linear model diagnostic' because regression diagnostic plots can be used with any model fitted by the `lm` function. We'll see this in the next chapter, where we'll also introduce a *much* easier way to make the key diagnostic plots.

# Chapter 21

# Regression diagnostics in R

We learnt how to interpret some practical regression diagnostic plots in the last chapter. It was quite a lot of work to make each one of these. Fortunately, there is no need to do all that work each time we fit a new model. R has a built-in facility to make these diagnostic plots for us using only a model object. This chapter will walk through how to use this workflow with the regression and one-way ANOVA models from earlier chapters.

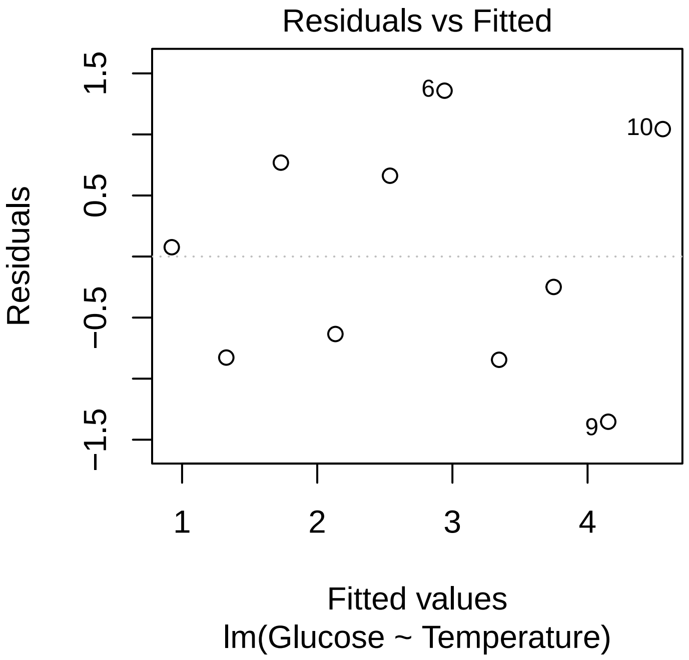
## 21.1 Diagnostics for regression

We'll use the glucose release example from the earlier Simple regression chapter to step through the easy way to make diagnostic plots.

> We assume the data from 'GLUCOSE.CSV' have been read into a tibble called `vicia_germ` and the associated regression model has been fitted as before:
>
> ```
> vicia_model <- lm(Glucose ~ Temperature, data = vicia_germ)
> ```
>
> Create `vicia_germ` and `vicia_model` if you plan to work along.

The built in facility for diagnostic plots in R works via a function called `plot`. It is easy to use. For example, to produce a residuals vs fitted values plot, we use:

```
plot(vicia_model, which = 1, add.smooth = FALSE)
```

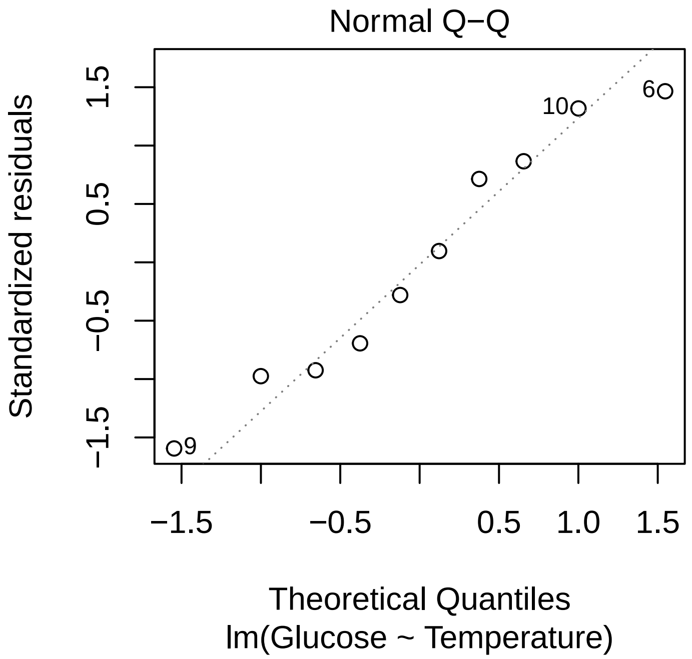


The first argument is the name of fitted model object. The second argument controls the output of the `plot` function: `which = 1` argument tells it to produce a residuals vs fitted values plot. The `add.smooth = FALSE` is telling the function not to add a line called a 'loess smooth'. This line is supposed to help us pick out patterns, but it tends to overfit the relationship and causes people to see problems that aren't there. For this reason, it's often better to suppress the line.

Remember that this plot allows us to evaluate the linearity assumption: we're looking for any evidence of a systematic trend (e.g. a hump or U-shaped curve) here. Because there is no obvious pattern, we can accept the linearity assumption and move on to the normality assumption.

We use the `plot` function to plot a normal probability diagnostic by setting `which = 2`:

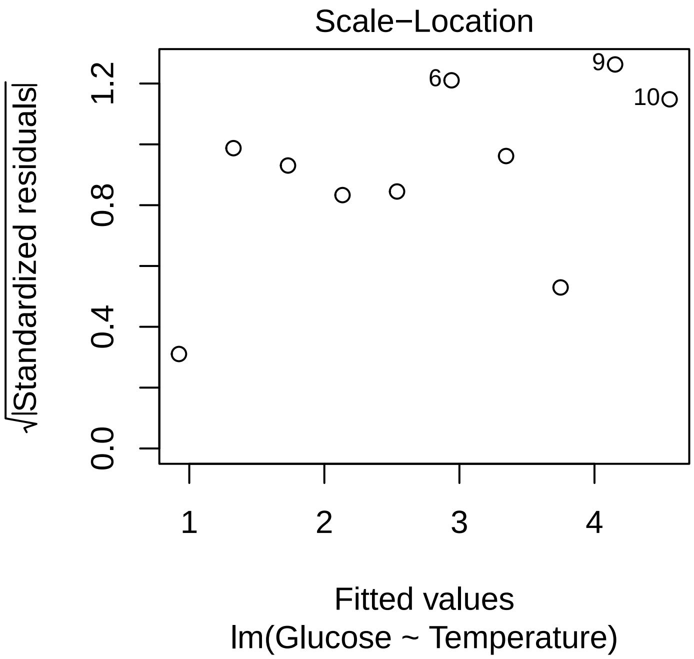
```
plot(vicia_model, which = 2)
```



This produces essentially the same kind of normal probability plot we made in the previous chapter, with one small difference. Rather than drawing a 1:1 line, the 'plot' function shows us a line of best fit. This allows us to pick out the curvature a little more easily. There's nothing here to lead us to worry about this assumption—the points are close to the line with no systematic trend away from it.

Finally, we can produce a scale-location diagnostic plot using the `plot` function to assess the constant variance assumption. Here we use `which = 3` and again we suppress the line (using `add.smooth = FALSE`) to avoid finding spurious patterns:

```
plot(vicia_model, add.smooth = FALSE, which = 3)
```



We're looking for any sign of the size of the transformed residuals either increasing or decreasing as the fitted values get larger. This plot looks good—as the relationship here looks pretty flat, we can conclude that the variability in the residuals is constant.

**Don't panic if your diagnostics aren't perfect!**

The good news about regression is that it is quite a robust technique. It will often give reasonable answers even when the assumptions are not perfectly fulfilled. We should be aware of the assumptions but not become too obsessed with them. If the violations are modest, it is often fine to proceed. Of course, we have to know what constitutes a 'modest' violation. There are no hard and fast rules—the ability to make that judgement comes with experience.

## 21.2 Diagnostics for one-way ANOVA

We pointed out in the previous chapter that the term 'regression diagnostic' is misleading because an equivalent set of diagnostic plots can be produced for any general linear model. This includes one-way ANOVA models. We'll use the corncrake density example from the One-way ANOVA chapter to demonstrate this.

We assume the data from 'CORN_CRAKE.CSV' have been read into a tibble called `corn_crake` and the associated one-way ANOVA model has been fitted as before:

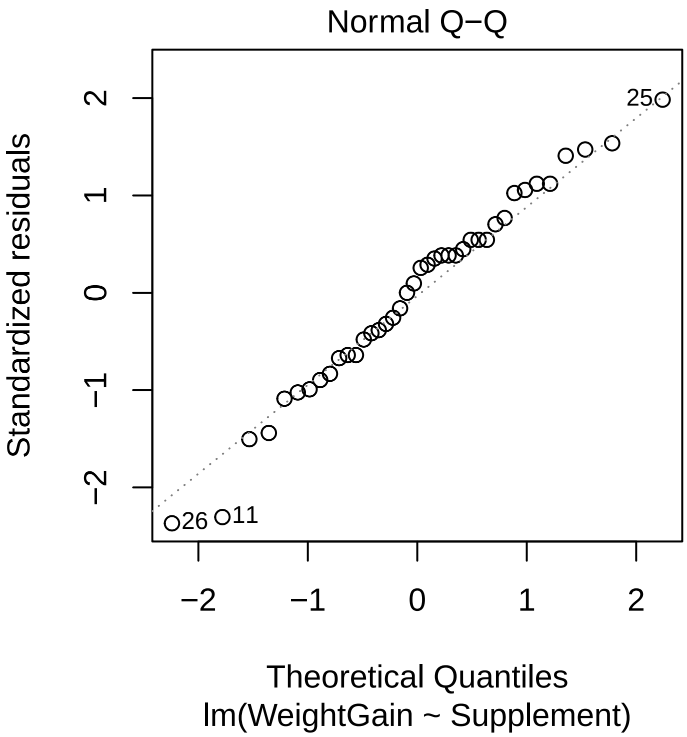
```
corncrake_model <- lm(WeightGain ~ Supplement, data = corn_crake)
```

Create `corn_crake` and `corncrake_model` if you plan to work along.

The first diagnostic plot we produced for the regression model above is the residuals vs fitted values plot. In a regression, this is used to evaluate the linearity assumption. What does it do in a one-way ANOVA? Not much of use to be honest. There isn't much point making a residuals vs fitted values plot for a one-way ANOVA. Why? Because the residuals will never show a trend with respect to the 'fitted values', which are just the group-specific means. That's one thing less to worry about.

The normal probability plot is used to identify departures from normality. This plot allows us to check whether the residuals—the deviation of each point from its group's mean—are likely to have been drawn from a normal distribution. Here's the normal probability plot for our example:

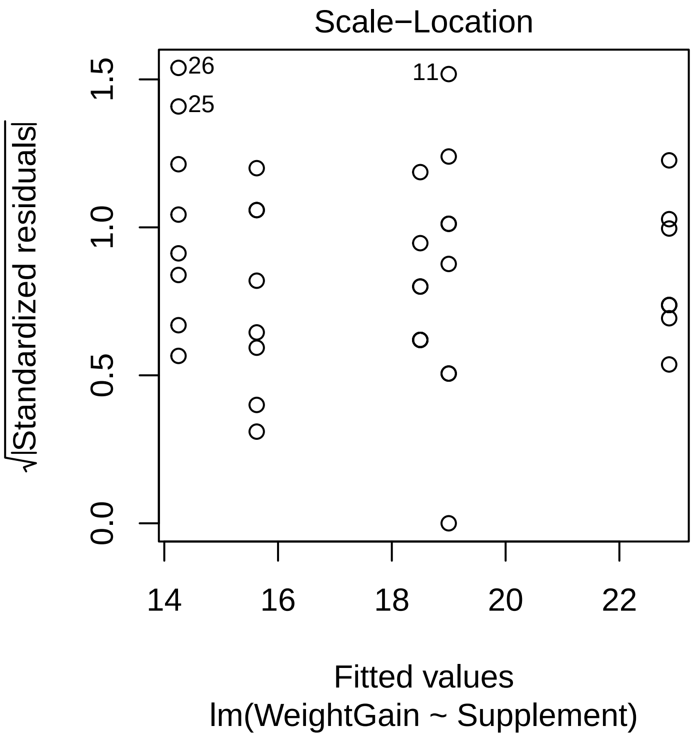
```
plot(corncrake_model, which = 2)
```



This looks very good. The points don't deviate from the line systematically, except for a couple at the lower end (that's not unusual), so it looks like the normality assumption is satisfied.

The scale-location plot allows us to evaluate the constant variance assumption. This allows us to see whether or not the variability of the residuals is roughly constant within each group. Here's the scale-location plot for the corncrake example:

```r
plot(corncrake_model, which = 3, add.smooth = FALSE)
```



The warning sign we're looking for here is a systematic pattern. We want to see if the magnitude of the residuals tends to increase or decrease with the fitted values. If such a pattern is apparent, then it suggests that variance changes with the mean. There is no such pattern in the above plot, so the constant variance assumption is satisfied.

## 21.3 Can we test equality of variance?

Looking back over the scale-location plot, it seems like three of the groups exhibit similar variability, while the remaining two are different. They aren't wildly different, though, so it is reasonable to assume the differences are due to sampling variation. Unfortunately, some people are uncomfortable using this sort of visual reasoning. They want to see a *p*-value to back up the decision to proceed.

We think this is a bad idea. Here is the reason why. Formal tests of equality of variance are not very powerful. This means that to detect a difference, we either need a lot of data, or the differences need to be so large that they would be easy to spot using a graphical approach. The test is a crutch that we don't need.

That said, some people will demand to see a statistical test evaluating the equality of variance assumption. Several such tests exist but the most widely used is the Bartlett test:

```
bartlett.test(WeightGain ~ Supplement, data = corn_crake)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  WeightGain by Supplement
## Bartlett's K-squared = 3.6444, df = 4, p-value = 0.4563
```

This looks just like the *t*-test specification. We use a ‘formula’ (`WeightGain ~ Supplement`) to specify the response variable (`WeightGain`) and the grouping variable (`Supplement`), and use the `data` argument to tell the `bartlett.test` function where to look for these variables. The null hypothesis of a Bartlett test is that the *variances are equal*, so a non-significant *p*-value (>0.05) indicates that the data are consistent with the equal variance assumption. That’s what we find here.

# Chapter 22

# Data transformations

## 22.1 Data that violate ANOVA assumptions

Up until now, the data we've examined have conformed, at least roughly, to the assumptions of the statistical models we've been studying. This is all very handy but perhaps a little unrealistic. The real world being the messy place it is, biological data often don't conform to the distributional assumptions of *t*-tests and ANOVA:

- The residuals may not be normally distributed.
- Variances may be unequal among groups.

The kinds of problems with the **distributional assumptions** can also arise when working in a regression (i.e. non-normal residuals, non-constant variance). Furthermore, we might run into additional issues if there is some non-linearity in the relationship between the response and predictor variables.

Most biological data are unlikely to conform perfectly to all the assumptions. Fortunately, tools like *t*-tests, ANOVAs and regressions are somewhat robust—they perform reasonably well with data that deviate to some extent from the tests' assumptions. However, when residuals are very far from normality or variances change a lot across groups, steps may need to be taken to deal with the problem. This chapter introduces one way to tackle the analysis of data that don't fit the assumptions: **transformation**. We will mostly focus on ANOVA / *t*-test setting, but keep in mind that the ideas are equally applicable to regression analysis.

## 22.2 Transformation: ANOVAs and *t*-tests

### 22.2.1 The data—foraging in ants

Red wood ants, *Formica rufa*, forage for food (mainly insects and 'honeydew' produced by aphids) both on the ground and in the canopies of trees. Rowan, oak and sycamore support very different communities of insect herbivores (including aphids), and it would be interesting to know whether the foraging efficiency of ant colonies is affected by the type of trees available to them. As part of an investigation of the foraging of *Formica rufa*, observations were made of the prey being carried by ants down trunks of rowan, oak and sycamore trees. The total biomass of prey being transported was measured over a 30 minute sampling period and the data were expressed as the biomass (dry weight in mg) of prey divided by the total number of ants leaving the tree to give the rate of food collection per ant per half hour. Observations were made on 28 rowan, 26 sycamore, and 27 oak trees.

> The data for this example are in the file ANTS1.CSV, which we assume has been read into a tibble called `ants`. Set that up if you wish to work throughth example.

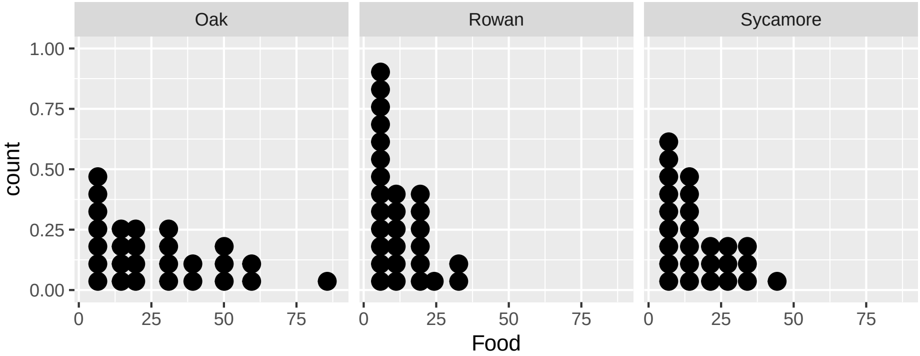
The `Tree` variable contains the tree identities and the `Food` variable contains the food collection rates:

```
glimpse(ants)
```

```
## Rows: 81
## Columns: 2
## $ Food <dbl> 11.9, 33.3, 4.6, 5.5, 6.2, 11.0, 24.3, 20.7, 5.7, 12.6, 10.2, 4.7~
## $ Tree <chr> "Rowan", "Rowan", "Rowan", "Rowan", "Rowan", "Rowan", "Rowan", "R~
```

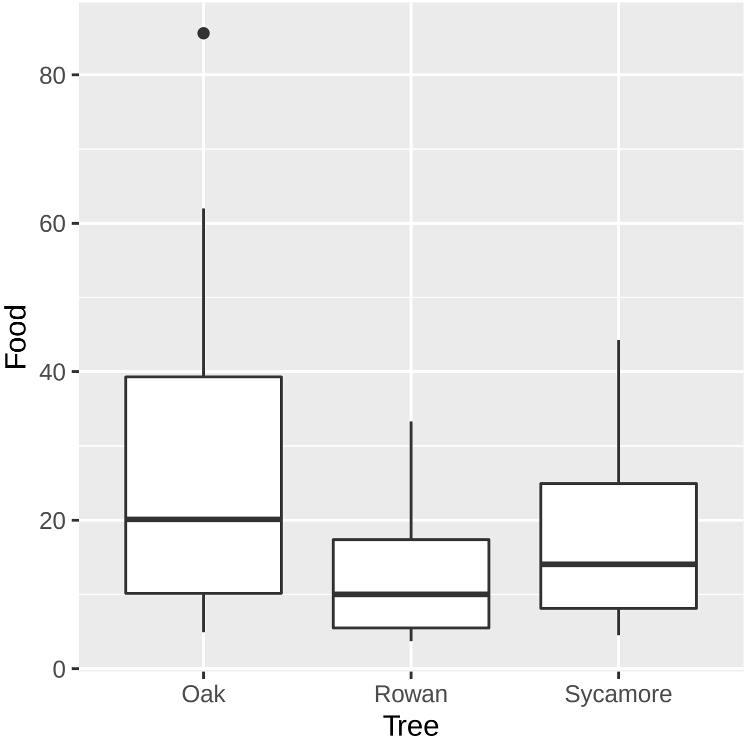
Let's visualise the data. We could make a dot plot...

```
ggplot(ants, aes(x = Food)) +
  geom_dotplot(binwidth = 6) + facet_wrap(~ Tree)
```



...or we could construct a box and whiskers plot:

```
ggplot(ants, aes(x = Tree, y = Food)) +
  geom_boxplot()
```



It doesn't matter which plot we use. They tell the same story. The food collection rate is generally highest in Oaks and lowest in Rowans (Sycamores are in between). Notice too that the sample distribution of food collection rate is right-skewed. The test we are most likely to use with these data is an ANOVA, i.e. we want to assess whether the mean food collection rates are different among the three tree species. There is already an indication that an ANOVA with the raw food values may be problematic.

### 22.2.2 Fit the model and checking the assumptions

This chapter is about fixing models when the assumptions are not satisfied. What assumptions do we need to check? The test we are most likely to want to use with these data is an ANOVA, so the following assumptions must be evaluated:

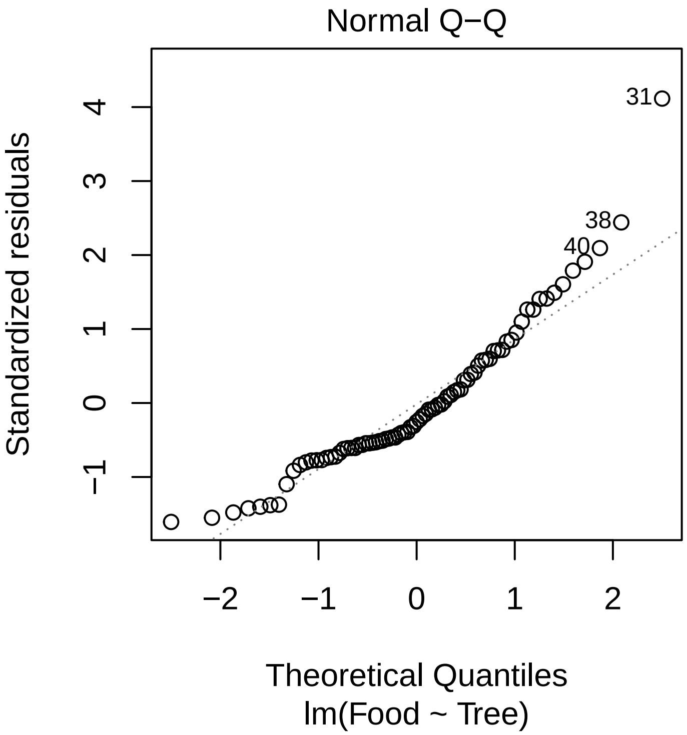
1. **Independence.** The experimental units of the data must be independent.
2. **Measurement scale.** The response variable is measured on an interval or ratio scale.
3. **Normality.** The residuals are normally distributed in each level of the grouping factor.
4. **Equal variance.** The variance in each level of the grouping factor is the same.

We'll have to assume the first assumption is satisfied and the food collection rate (second assumption) is obviously measured on a ratio scale. The distributional assumptions (normality and equality of variance) are the ones we can address with a transformation. Let's fit the ANOVA model and produce regression diagnostics to evaluate these. Remember that we make these kinds of plots *after* we have fitted a statistical model:

```r
ant_mod <- lm(Food ~ Tree, data = ants)
```

We need to produce a 'normal probability plot' to assess the normality assumption:

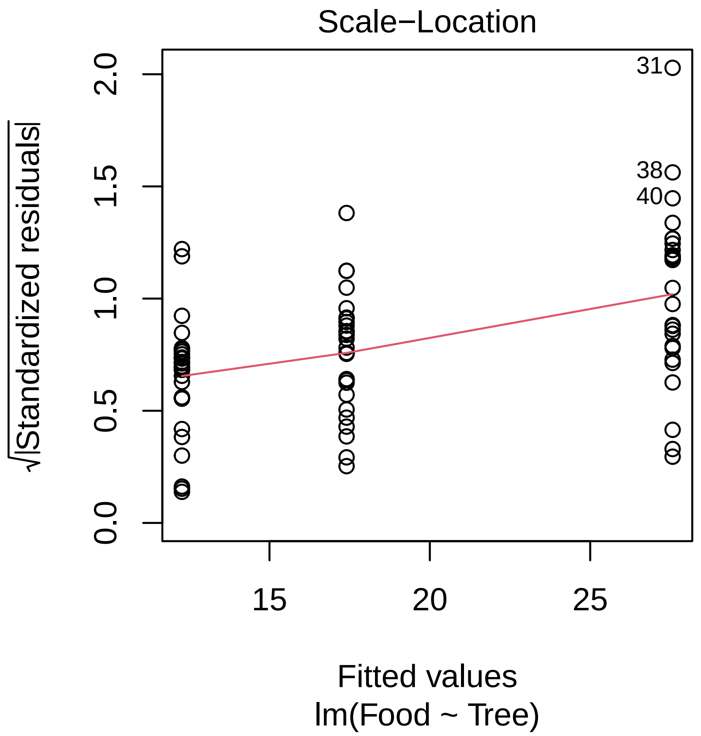
```r
plot(ant_mod, which = 2)
```



This plot exhibits the accelerating curvature that is indicative of right-skewed residuals. This probably isn't just sampling variation because there is a systematic departure from the dashed line everywhere along it. So it looks like there is a problem. This sort of pattern is quite common in biological data, especially when it involves counts. Clearly, we might be a bit worried about using an ANOVA with these data since it assumes the residuals to be at least approximately normally distributed.

Are the variances significantly different? Look at the box plots above. The data from the three samples seem to have rather different scatter. The sample from the rowan has less variation than that from the sycamore, and the sycamore has less variation than the oak. Does the scale-location plot tell the same story?

```
plot(ant_mod, which = 3)
```



This shows that the variance increases with the fitted values—it looks like there is also a problem with the constant variance assumption. Again, this pattern is very common in biological data.

In the light of these evaluations, we have three options …

1. To carry on and carry out an ANOVA anyway and hope the violation of the assumptions won't matter too much.
2. To try and transform the response variable to make it fit the assumptions better, then carry out an ANOVA.
3. To use a different sort of test which doesn't require the data to conform to these assumptions. Such tests are known as nonparametric tests.

We will consider the first two options below, and return to the third in the non-parametric tests chapter.

## 22.3 Carrying on anyway

Carrying on doesn't sound like a very good idea if we've already decided that the assumptions are suspect, but don't dismiss it straight away. Mild departures from the assumptions often do not make a huge difference to the results of

ANOVA (i.e. the $p$-values). At the very least, it can be instructive to continue an analysis without 'fixing' the apparent problems so that we can get a sense of whether they matter or not.

We already fitted the ANOVA model to allow us to make the diagnostic plots. All we have to do is pass the model object to the `anova` function to get the $F$-ratio and $p$-value for the tree effects:

```
anova(ant_mod)
```

```
## Analysis of Variance Table
##
## Response: Food
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## Tree       2  3317.1 1658.53  8.0305 0.0006741 ***
## Residuals 78 16109.2  206.53
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on these results, it looks like there is a highly significant difference in food collection rates across the three tree species. However, we know the data are problematic, so the question is, does this result stand up when we deal with these problems?

### *t*-tests are robust

The 'carry on anyway' strategy can sometimes be justified if we only need to compare the sample means of two groups because then we can use a two-sample $t$-test rather than an ANOVA. By default, R uses a version of the $t$-test that allows for unequal sample variances. This at least deals with one potential problem. The $t$-test is also fairly robust to violations of the normality assumption when the sample sizes are small, and when the sample sizes are large, the normality assumption matters even less.

The ability to do a $t$-test that doesn't require equal variances is undoubtedly useful. A word of warning, though: some people advise carrying out a statistical test of equal variance, and if the variances are not significantly different, using the version of a two-sample $t$-test that assumes equal variances. This is not good advice. Following this procedure leads to less reliable $p$-values. The reason for this is somewhat technical, but trust us, this procedure is not good statistical practice.

## 22.4 Transforming the data

One approach to dealing with difficult response variable is to apply a mathematical function to it to make the transformed data fits the model assumptions better: a process called *data transformation*. This may sound a bit dubious, but it is a perfectly valid procedure that will often allow us to use the statistical model we want to, even if the data don't initially fit the assumptions. The critical thing to keep in mind is that the transformation should be applied to the *response variable*.

### 22.4.1 The logarithmic transformation

Let's try a simple transformation on the food collection rate variable in the ant data set. Instead of using the original numbers we will convert them to their logarithms. We can use common logs (logs to the base 10, written $\log_{10}$) or natural logs (logs to the base, written $\log_e$ or ln). It doesn't matter: they have the same effect on the data in terms of making it meet the assumptions of ANOVA (or not).

Applying a log transform is quick and easy in R—there are built-in functions to take common logs and natural logs, called `log10` and `log`, respectively. We'll use `mutate` to add a new variable, which is the common log of `Food`:

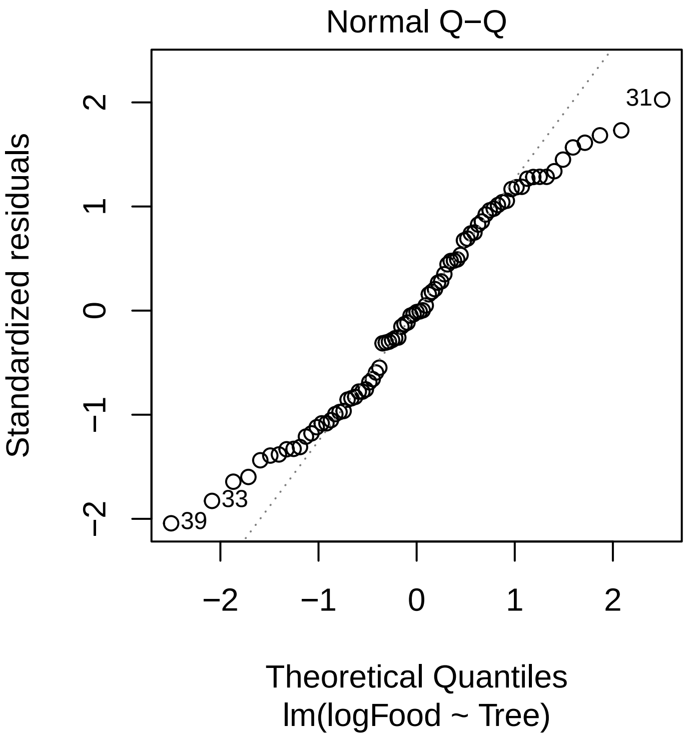
```
ants <- mutate(ants, logFood = log10(Food))
```

We stored the transformed variable in a new column called `logFood`.

```
ant_mod_log <- lm(logFood ~ Tree, data = ants)
```

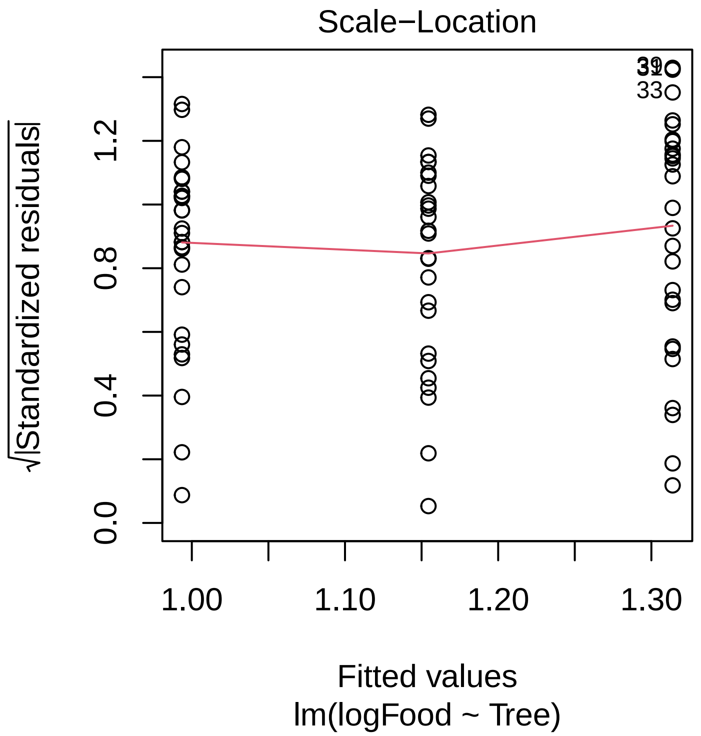
We need to produce a 'normal probability plot' to assess the normality assumption:

```
plot(ant_mod_log, which = 2)
```



The accelerating curvature (indicative of right-skewed residuals) has gone. The new normal probability plot is a little better than before. Now perhaps 60% of the cases are on the dashed line. It's hardly perfect, though—the tails of the distribution are not where we'd like them to be. What about the variances?

```
plot(ant_mod_log, which = 3)
```



The scale location-plot indicates that the constant variance assumption is now OK, i.e. the variance no longer increases with the fitted values. It looks like the log transformation seems to have improved things quite a lot, but the diagnostics are still not perfect.

The assumptions are closer to being satisfied. Let's carry out ANOVA again using the model with the transformed food variable to see how the results change:

```
anova(ant_mod_log)
```

```
## Analysis of Variance Table
##
## Response: logFood
##           Df Sum Sq Mean Sq F value   Pr(>F)
## Tree       2 1.4106 0.70530  7.2867 0.001255 **
## Residuals 78 7.5498 0.09679
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What has happened? We still see evidence for a significant effect of tree ($p$<0.01) with the transformed data, but the $p$-value is somewhat bigger than when we used the original data. This illustrates why it is important to evaluate assumptions and deal with them when they are violated—the output of a statistical test is affected. It does not matter too much here, but in other settings, we can end up with misleading or downright spurious results if we ignore problems with the assumptions of a statistical model.

**Values of 0**

One thing to be aware of is that **we cannot take the log of zero**. If our response variable contain zeros we have add a small value to it before taking the logs. We usually add 1 to the variable, i.e. the transformation is $log(x+1)$. It is worth trying this—it's easy to do and can work well—but don't be surprised if this transformation results in a model with poor diagnostics. For example, the $log(x+1)$ transformation doesn't work well when there are many zeros in the response variable.

### 22.4.2 Presenting results from analyses of transformed data

Having compared the transformed means, how should we present the results in a report? There are three alternatives. We'll illustrate them using the log-transformation, though they also apply to other kinds of transformations.

1. We could present the transformed means (having stated what the transformation was, e.g. $log_e(x+1)$). The disadvantage to this is that the numbers themselves convey little information about the values on the original scale. This isn't always a problem. For example, effects given on a log scale act in a 'multiplicative' manner, so a model with log-transformed response variable can still be interpreted if we know what we're doing.

2. We could back-transform the means of the log-transformed data by taking the antilogs: $10^x$ (for logs to the base 10) and $e^x$ (for natural logs)[1]. However, when we back-transform data, we need to be aware of two things: (1) The back-transformed mean will not be the same as a mean calculated from the original data; (2) We have to be careful when we back-transform standard errors. Suppose we want to display the back-transformed means on a bar plot, with some indication of the variability of the data. In that case, we must calculate the standard errors and then back transform the upper and lower limits, which will not then be symmetrical about the mean.

3. We could also present the means calculated from the original data but state clearly that the statistical analysis was carried out on transformed data. This is often the simplest way to proceed.

[1] N.B. if we used $\log(x+1)$ we must remember to subtract the 1 again after taking the antilog.

⚠ **Diagnostic plots vs *p*-values?**

In the case study above, applying a log-transformation altered the outcome of the statistical test. Does that mean *p*-values tell us something about model adequacy? Not really. Transforming the response variable will not always make the difference between a significant and not significant result. In fact, *p*-values don't tell us much at all about how good a model is for the data in hand. It is entirely possible for a really bad model to produce small p-values.

We have to use diagnostic plots to judge the success or not of a transformation. Do not use *p*-values to make that assessment!

## 22.5 Types of transformations

Taking logarithms is only one of many possible transformations. Each one is appropriate for solving different problems in the data. The following summarises four commonly used transformations and the sort of situations where they are useful.

### 22.5.1 Logarithms

Log transformation, as we've just seen, affects the response variable in two ways:

1. A log transformation stretches out a distribution's left-hand side (smaller values) and squashes in the right-hand side (larger values). This is useful where the residuals' distribution has a long tail to the right, as in the ant example above.

2. The 'squashing' effect of a log transformation is more pronounced at higher values. This means a log transformation may also deal with another common problem in biological data (also seen in the ant data)—samples with larger means having larger variances.

If we are carrying out an ANOVA and the scale-location plot exhibits a positive relationship—i.e. groups with larger means have larger variances—then a log transformation could be appropriate[2].

[2] Log transformations have a variety of other uses in statistics. One is in transforming a *predictor* variable when looking at the form of relationships between two variables.

### 22.5.2 Square roots

Taking the square root of the response variable is often appropriate where the data are whole number counts (the log transformation may also work here). This typically occurs where your data are counts of organisms (e.g. algal cells in fields of view under a microscope). The corresponding back-transformation is obviously $x^2$.

In R, the square root can be taken using the `sqrt` function. It works like this using `mutate`:

```
mydata <- mutate(mydata, sqrt_x = sqrt(x))
```

By the way, there is no `square` function. Taking squares is done using the `^` operator (e.g. if the variable is called `x`, use `x^2`).

### 22.5.3 Arcsine square root

The arcsine square root transformation is generally used where the response variable is in the form of percentages or proportions. Such data tend to get 'bunched up' at there limits—0 (0%) and 1 (100%)—which leads to non-normal distributions. A correction for this effect is to take the inverse sine of the square roots of the original variable, i.e. $\arcsin\sqrt{x}$.

In R the transformation can be achieved by combining the `sqrt` and `asin` functions inside `mutate`. For example, if we need to transform a proportion stored in the `x` variable use something like:

```
mydata <- mutate(mydata, assqrt_x = asin(sqrt(x)))
```

…where `mydata` is the name of hypothetical data frame containing the data. Just remember to apply `sqrt` and `asin` in the correct order. We used nested functions here, which are applied from the inside to the outside.

**Converting percentages to proportions**

The arcsine square root transformation is often discussed in the context of percentages, However, the transformation only works on numbers that lie between 0 and 1, yet percentages range between 0 and 100. This is why we have to convert percentages to proportions (e.g. 100% = 1, 20% = 0.2, 2% = 0.02, etc.) before applying the arcsine square root transformation.

### 22.5.4 Squaring

A problem the above transformations don't deal with is when a response variable has a negative skew (i.e. a long tail to the left). This problem can sometimes be dealt with, or at least reduced, by squaring the data values.

In R, the transformation can be achieved by combining the `^` operator inside `mutate`. For example, we might use something like…

```r
mydata <- mutate(mydata, sqr_x = x^2)
```

…where `mydata` is again the name of hypothetical data frame containing the data.

### 22.5.5 Situations which cannot be dealt with by transformations

There are some situations where no transformation will get round the fact that the data are problematic. Two in particular are worth noting:

- **Dissimilar distribution shapes**: If two or more subsets of the data exhibit very different problems, then a transformation that corrects one subset ends up distorting the other(s). For example, this occurs when the response variable is strongly right-skewed in one treatment group but strongly left-skewed in another.
- **Samples with many exactly equal values**: When the response variable takes a limited set of values (e.g. small integers), there will be many identical values. Non-normality then results from groups of identical values forming particular 'peaks'. This cannot be corrected by transformation since equal values will still be equal after transformation!

**Why is it OK to transform a response variable?**

Changing the response variable by transformation alters the results of a statistical test. Isn't this a bit dodgy? The key thing to realise is that the scales on which we measure things are, to some extent, arbitrary. Transforming data to a different scale replaces one arbitrary scale with another. The transformations we have discussed don't alter the ordering of data points relative to each other—they only alter the gaps between them (that is the critical point). In some cases, this rescaling can make the data more amenable to study, analysis or interpretation.

In fact, we often use data transformations, perhaps without realising it, in many situations other than doing statistical tests. For example, when we look at a set of pH readings, we are already working with data on a log scale because pH units (0-14) are the negative logarithms of the hydrogen ion concentration in the water. Similarly, measurements of noise in decibels (dB), and seismic disturbance on the Richter scale, are logarithmic scales.

One final comment… Obviously, we do have to apply the same transformation to every observation. For example, we would never log transform the observations in one group and leave the others alone—that really would be cheating!

## 22.6 What about other kinds of models?

We have focussed on ANOVA here for the simple reason that the assumptions are a bit easier to evaluate compared to regression. However, exactly the same ideas apply when working with other kinds of models that `lm` can fit. The workflow is the same in every case:

1. Always check the diagnostic plots after we fit a regression or ANOVA (and do this before worrying about *p*-values).
2. If there is evidence for a problem with the assumptions, try transforming the response variable.
3. Refit the model using the transformed variable and generate new diagnostic plots to see if the new model is any better.

Keep in mind that this is often an iterative process. We might have to go through several rounds of transforming and model checking before we arrive at a good model.

## 22.7 Final thoughts

Evaluating assumptions and picking transformations is as much 'art' as science. It takes time and experience to learn how to do it. R makes it easy to try out different options, so don't be afraid to do this. Frequently, with biological data, no straightforward transformation improves the data, or in correcting one problem, we generate another. Fortunately, in many cases where the assumptions of a test are not reasonably well fulfilled, there is an alternative approach— which we will discuss in the non-parametric tests chapter.

# Part VIII

# Non-linear Regression

# Chapter 23

# Linear vs non-linear models

## 23.1 Introduction

We have seen how to use simple linear regression to characterise the dependence of one numeric variable on another. The most obvious assumption of that model is that the dependence between the two variables follows a straight line. What should we do when the relationship of interest is not so simple and instead follows a non-linear curve of some kind?

One option is to transform one or both variables to enforce a straight-line relationship. Sometimes this works well, but on other occasions, it can prove tricky to identify a transformation that works. In any case, for some kinds of problems, we prefer to work with variables on the scale we measured them because this makes it easier to interpret the relationship.

This is where **non-linear regression** is useful. Non-linear relationships arise all the time in biology. Our goal in this chapter is to set out what we mean by the term 'non-linear' when dealing with regression models. To do this, we will establish the link between two models we have already studied: ANOVA and linear regression. Once that linkage is clear, it is relatively straightforward to define linear and non-linear models. The next chapter will cover the practical details of working with non-linear regression models in R.

## 23.2 The relationship between regression and ANOVA

Though not apparent at first, ANOVA models can be viewed as a kind of regression analysis. We will look at the corncrake growth data again to understand this idea. This was the study where scientists wanted to determine how four different diet supplements affected the initial growth of captive-bred corncrake hatchlings.

Imagine that we have read these data into a data frame called `corn_crake`. The weight gain measurements are stored in the `WeightGain` variable, and the food supplement treatments are encoded in the categorical `Supplement` variable:

```
glimpse(corn_crake)
```

```
## Rows: 40
## Columns: 2
## $ Supplement <chr> "None", "None", "None", "None", "None", "None", "None", "No~
## $ WeightGain <dbl> 13, 20, 17, 16, 15, 14, 10, 20, 15, 18, 10, 22, 23, 19, 25,~
```

We can repeat the one-way ANOVA global significance test with these data using the usual R code:

```
diet_model <- lm(WeightGain ~ Supplement, data = corn_crake)
anova(diet_model)
```

```
## Analysis of Variance Table
##
## Response: WeightGain
##            Df Sum Sq Mean Sq F value   Pr(>F)
## Supplement  4 357.65  89.413  5.1281 0.002331 **
## Residuals  35 610.25  17.436
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This shows a significant effect of supplement on the weight gain of the corncrake hatchlings ($F$=, d.f.=4,35; $p$<0.01).

Now let's look at how this same analysis can be carried out using a type of regression model with numeric predictors. There are four steps to this.

#### 23.2.0.1 Step 1

We need to construct a new representation of the data. Rather than including one categorical variable that labels the observations from each supplement (+control), we construct a series of binary (i.e. 0/1) *indicator variables*. We need one indicator variable for each supplement category. Each indicator variable takes the value '1' if an observation belongs to its corresponding supplement and '0' otherwise.

Here is one way to create these data using R:

```
corn_crake_reg <-
  mutate(corn_crake,
         Allvit    = ifelse(Supplement == "Allvit",    1, 0),
         Earlybird = ifelse(Supplement == "Earlybird", 1, 0),
         Linseed   = ifelse(Supplement == "Linseed",   1, 0),
         Sizefast  = ifelse(Supplement == "Sizefast",  1, 0),
         None      = ifelse(Supplement == "None",      1, 0)) %>%
  select(-Supplement)
```

This code creates a tibble called `corn_crake_reg` containing the original `WeightGain` variable and five new indicator variables called `Allvit`, `Earlybird`, `Linseed`, `Sizefast` and `None`:

```
print(corn_crake_reg, n = 40)
```

```
## # A tibble: 40 x 6
##    WeightGain Allvit Earlybird Linseed Sizefast  None
##         <dbl>  <dbl>     <dbl>   <dbl>    <dbl> <dbl>
##  1         13      0         0       0        0     1
##  2         20      0         0       0        0     1
##  3         17      0         0       0        0     1
##  4         16      0         0       0        0     1
##  5         15      0         0       0        0     1
##  6         14      0         0       0        0     1
##  7         10      0         0       0        0     1
##  8         20      0         0       0        0     1
##  9         15      0         0       0        1     0
## 10         18      0         0       0        1     0
## 11         10      0         0       0        1     0
## 12         22      0         0       0        1     0
## 13         23      0         0       0        1     0
## 14         19      0         0       0        1     0
## 15         25      0         0       0        1     0
## 16         20      0         0       0        1     0
## 17         20      0         0       1        0     0
## 18         24      0         0       1        0     0
## 19         17      0         0       1        0     0
## 20         16      0         0       1        0     0
## 21         20      0         0       1        0     0
## 22         20      0         0       1        0     0
## 23         15      0         0       1        0     0
## 24         16      0         0       1        0     0
## 25         22      1         0       0        0     0
## 26          5      1         0       0        0     0
## 27          10       1         0        0         0      0
## 28          16       1         0        0         0      0
## 29          11       1         0        0         0      0
## 30          17       1         0        0         0      0
## 31          20       1         0        0         0      0
## 32          13       1         0        0         0      0
## 33          17       0         1        0         0      0
## 34          25       0         1        0         0      0
## 35          24       0         1        0         0      0
## 36          25       0         1        0         0      0
## 37          25       0         1        0         0      0
## 38          19       0         1        0         0      0
## 39          27       0         1        0         0      0
## 40          21       0         1        0         0      0
```

Take a look at the last column (None). The first eight values are '1', reflecting the fact that the first eight rows of the data set are the control cases (i.e. no supplement). The remaining values of None are all '0' because they correspond to rows with data from chicks that received a supplement. The same logic applies to the other four indicator variables.

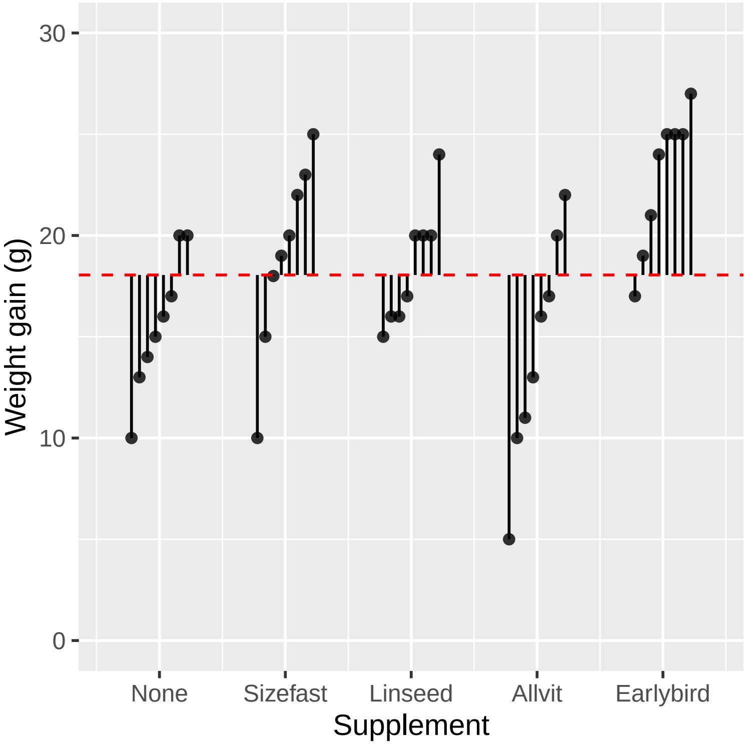
The information in the data is unchanged. All we have done is alter how the different supplement identities are represented.

#### 23.2.0.2 Step 2

The next step is to fit a minimal model to the data, called an 'intercept only' or 'null' model. We fit this using the lm function:

```
mod_null <- lm(WeightGain ~ 1, data = corn_crake_reg)
```

The formula WeightGain ~ 1 tells R to fit a model with exactly one coefficient—the overall grand mean of all the data. In graphical terms we can visualise this using one of the plots we made to explore how ANOVA works:



The intercept (i.e. the grand mean) is shown by the vertical position dashed red line. The vertical lines show the distance between each observation and the grand mean. Therefore, the vertical lines are residuals from the 'intercept only' model. These capture the total variation in the data.

#### 23.2.0.3 Step 3

In the third step, we use `lm` to fit a regression model using the indicator variable version of the data set. We have to include the indicator variables as separate terms when we do this:

```
mod_mreg <- lm(WeightGain ~ 0 + Allvit + Earlybird + Linseed + Sizefast + None, data = corn_crake
```

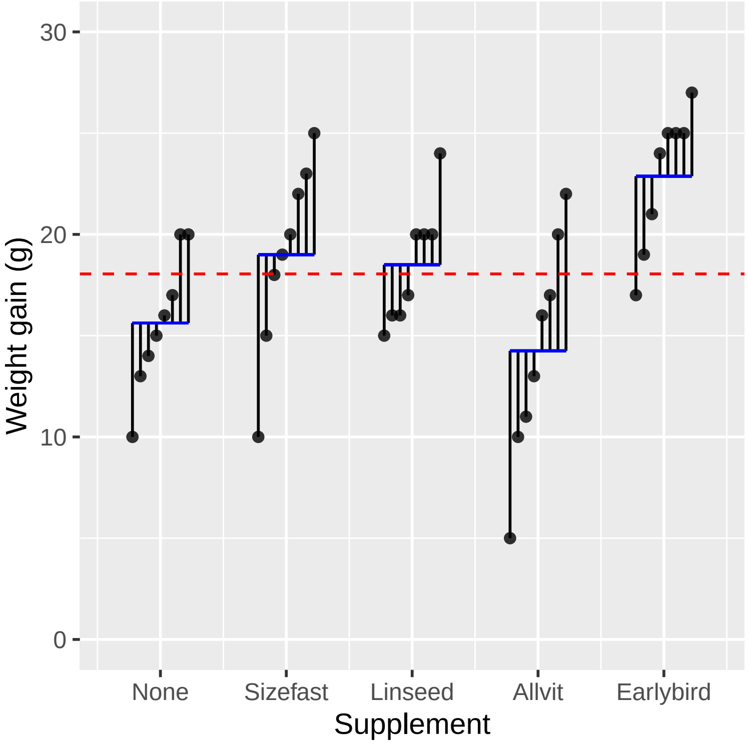
The unusual '0' in the formula is a trick to force R to fit the particular model we want. `lm` fits a regression because the indicator variables contain numeric values (i.e. 0 or 1). This model can be thought of as something called a **multiple regression**. It is a multiple regression because the model involves more than one predictor variable.

This model involves five regression coefficients (a.k.a. 'slopes'). Each of these describes the change in weight gain that occurs if we alter the corresponding indicator variable by one unit. Since the indicator variables only take values of '0' or '1', these coefficients describe the mean weight gain associated with each supplement. We can see those values by printing the model object:

```
print(mod_mreg)
```

```
##
## Call:
## lm(formula = WeightGain ~ 0 + Allvit + Earlybird + Linseed +
##     Sizefast + None, data = corn_crake_reg)
##
## Coefficients:
##    Allvit  Earlybird    Linseed   Sizefast       None
##     14.25      22.88      18.50      19.00      15.62
```

Those are the supplement group means. We can visualise this model as follows:



The coefficients of the regression model are denoted by the vertical positions of the horizontal blue lines, i.e. the mean weight gain associated with each supplement group. The variability of those means around the overall grand mean is the the *among* group variability. This describes the variation explained by the indicator variable model. The vertical lines capture the variation due to differences among individual chicks *within* each group after accounting for the mean supplement effect.

This figure was used earlier in the book to explain how ANOVA works. That is no coincidence. Fitting an ANOVA model really is nothing more than 'regression with indicator variables'.

#### 23.2.0.4 Step 4

The final step is to evaluate the significance of the supplement effect. We do this with the `anova` function. However, we have to use `anova` in a new way by passing it *two* models—the intercept-only model and the regression-with-indicators model:

```
anova(mod_null, mod_mreg)
```

```
## Analysis of Variance Table
##
## Model 1: WeightGain ~ 1
## Model 2: WeightGain ~ 0 + Allvit + Earlybird + Linseed + Sizefast + None
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     39 967.90
## 2     35 610.25  4    357.65 5.1281 0.002331 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

When passed two like this, `anova` uses the models to construct a statistical significance test. Here is the logic of that test:

- The null hypothesis is embodied by the simpler of the two models. This has to be a special case of the more complex model.
- The more complex model incorporates the additional effects we are interested in testing (e.g. differences between groups).
- `anova` uses the variability explained by the two models to construct an $F$-test of the significance of the target effect.

In this example, the simpler model (`mod_null`) only contains the grand mean. This is equivalent to assuming no effect of diet supplement on weight gain. The more complex model (`mod_mreg`) allows weight gain to vary according to supplement. Therefore, if the $F$-test produced by `anova` is significant, this is taken as evidence that supplement affects weight gain. That is what we find: the test reveals a significant effect of supplement on the weight gain of corncrake hatchlings (F=5.1; d.f.= 4,35; p<0.01).

Look at the $p$-value, the test statistic ($F$) and the degrees of freedom from the test. These are the same as those produced by the standard one-way

ANOVA global significance test. This is because the convoluted regression-with-indicators analysis and standard ANOVA analyses are functionally identical.

Linear regression and ANOVA are not so different after all. Whenever we work with an ANOVA in R, `lm` uses the formula we supply to make something called a 'design matrix' behind the scenes. The design matrix is a representation of the model with the series of 0/1 indicator variables required to estimate the associated group means (or, more usually, differences between means).

## 23.3 Linear models

It turns out that many of the statistical models we have seen so far are special cases of something called the General Linear Model. The only thing that distinguishes the different cases is the type of predictor variables they accommodate:

- **One-way ANOVA**: one categorical predictor
- **Multi-way ANOVA**: two or more categorical predictors
- **Simple Regression**: one numeric predictor
- **Multiple Regression**: two or more numeric predictors
- **ANCOVA**: mixture of categorical and numeric predictors

Every one of these is a special case of the General Linear Model. To understand the key properties of the General Linear Model, we need to express it in mathematical terms. Think back to the earlier chapter on parametric statistics. We introduced the idea of a statistical model as a device to describe how samples of data are generated from a hypothetical population. We also mentioned a particular model for the relationship between two numeric variables:

$$y_i = a + b \times x_i + \epsilon_i$$

That equation is the mathematical representation of a simple linear regression model, where $y_i$ is the response variable and $x_i$ is the predictor variable. This model has two parts:

- The systematic component ($a + b \times x_i$) describes the general relationship between the $y_i$ and $x_i$. This is the equation of a straight line with an intercept $a$ and slope $b$ parameter. Those two parameters are known as the coefficients of the model.
- The random component ($\epsilon_i$) describes how the individual $y_i$ values deviate from their expected values on the line. Those are the residuals. These residuals are assumed to follow a normal distribution in a regression model.

Fitting a regression model boils down to finding the values of $a$ and $b$ that minimise the variability associated with $\epsilon_i$. The normality assumption justifies using $F$ tests to carry out statistical significance tests with the model.

We can take that basic model description and expand it to represent a more complex model with multiple predictors:

$$y_i = a + b_1 \times x_{1i} + b_2 \times x_{2i} + b_3 \times x_{3i} + ... + \epsilon_i$$

Compared to the simpler model, the thing that has changed is the systematic part of the model:

- The $x_{1i}$, $x_{2i}$, $x_{3i}$ represent a set of predictor variables used to explain variation in the response variable, $y_i$.
- The $a$, $b_1$, $b_2$, $b_3$ are the model coefficients. These describe the relationship between the response and predictors.
- The +...+ is a generic placeholder for all other predictors. We wrote out the first three, but there could be many more.

The random component ($\epsilon_i$) is the same as before. That describes the deviations of the $y_i$ values from their expected values given by the systematic part of the model.

Now we arrive at the important part of this section. If we assume the $\epsilon_i$ come from a normal distribution, then that equation describes the General Linear Model. The various models listed above can all be written in the form of that equation. Numeric predictors enter into the model unchanged. Categorical predictors are included by expanding them into a series of 0/1 indicator variables.

The word 'general' in the name of the General Linear Model refers to the fact that the model encompasses various different special cases. The word 'linear' needs more unpacking. That relates to the mathematical form of the model. To be precise, it relates to systematic part of the model, which is called as the **linear predictor**.

A linear predictor is made up of multiple parts, where each part is the product of a parameter and an associated predictor or derived index variable. The value of the linear predictor is found by adding those parts together. That's exactly how the General Linear Model equation works:

$$y_i = a + b_1 \times x_{1i} + b_2 \times x_{2i} + b_3 \times x_{3i} + ... + \epsilon_i$$

Whenever the systematic part of a statistical model can be expressed in that form, the model is said to be 'linear in its parameters'. This linearity feature of the parameters allows mathematicians to use well-established, robust methods to identify best-fitting models and construct statistical significance tests. That's one reason why the General Linear Model is so widely used.

People often abbreviate the name to 'Linear Model' or 'LM' because saying General Linear Model all the time soon gets tedious. This is where the `lm` function gets its name.

## 23.4 Non-linear models

Now that we know what makes a linear model a linear model, we can finally address the actual topic of this chapter—non-linear regression models. A non-linear regression model is any parametric model for which the systematic component is **not** linear in its parameters. This is best understood by example.

There are many ways to express non-linear regression models, ranging from comparatively simple to very complex. At their simplest, they look like this:

$$y_i = f(x_i) + \epsilon_i$$

In this formulation, there is one predictor variable. The $x_i$ are the values of that predictor variable, the $y_i$ are the corresponding values of the response variable, $f$ is a non-linear function describing the systematic part of the model, and the $\epsilon_i$ are the residual deviations.

Here are a few concrete examples of what we might use for the function $f$:

*Power function*:

$$f(x_i) = a \times x_i^b$$

Raising the $x_i$ to the power of the parameter $b$ is what makes this a non-linear regression model.

*Exponential function*:

$$f(x_i) = a \times e^{b \times x_i}$$

The exponentiation of the $b \times x_i$ is what makes this a non-linear regression model.

*Michaelis–Menten equation for enzyme kinetics*:

$$f(x_i) = V_{max} \frac{x_i}{K_m + x_i}$$

The parameters of this model are the maximum reaction rate ($V_{max}$) and the Michaelis constant $K_m$. The non-linearity arises from the ratio of the terms involving the predictor variable.

What happens if we fix the values of parameters $b$ or $K_m$ in those examples, meaning they do not have to be estimated from data? Those three models could then be rewritten as:

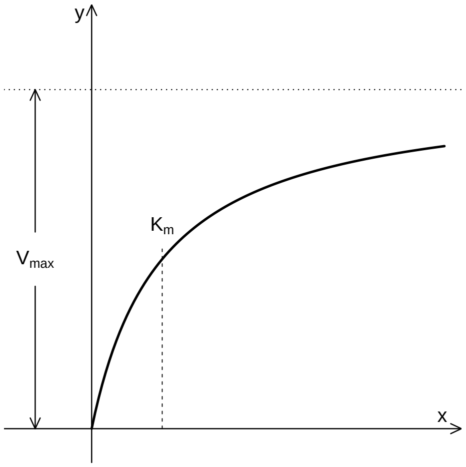
$$f(x_i) = a \times z_i$$

…where $z_i$ is the part of the equation involving $x_i$. For example, if $b$ set to 2 in the power function, then $z_i = x_i^2$. In that scenario, the function $f(x_i)$ is now linear in its parameters because we only need to estimate $a$.

This observation highlights an important point—it is the linearity of the *unknown parameters* that distinguishes linear and non-linear regression models.

## 23.5 Why do we need non-linear regression models?

Non-linear regression models are useful because they allow us to model relationships that involve **curves**. For example, the Michaelis–Menten equation looks like this:



The Michaelis–Menten equation describes the dependence of an enzyme's reaction rate ($y$) on its substrate concentration ($x$). This is a saturating function — reaction rate increases a lot with small increases in substrate at low concentrations, but the rate of increase tails off at higher concentrations. That kind of relationship obviously requires a non-linear model.

We can't cover every important aspect of non-linear regression modelling in a book like this. What we can do is develop a workflow for dealing with simple non-linear regression models in R and highlight some of the key concepts along the way. We will do that in the next chapter.

# Chapter 24

# Non-linear regression in R

Our goal in this chapter is to learn how to work with non-linear regression models in R. We'll start with the example problem and the data, then discuss model fitting, evaluating assumptions, significance testing, and finally, presenting the results.

## 24.1 Introduction

The enzyme-linked immunosorbent assay (ELISA) is a standard assay for detecting and quantifying soluble substances such as proteins, antibodies, and hormones. The direct form of ELISA works as follows:

- Antigens from a test sample are attached to a surface, and a matching antibody is applied to bind the antigen.
- The antibody is linked to a reporter enzyme, and any unbound antibodies are removed.
- The reporter enzyme's substrate is added, and the subsequent reaction produces a detectable signal if there was binding.

The activity of the reporter enzyme is measured using spectrophotometry, such that the concentration of the target substance is associated with a standardised **optical density** measurement.

A biochemist is developing direct ELISA for a new recombinant protein. To investigate how well it works, she ran the assay using a series of standardised protein concentrations (measured in $ng\ ml^{-1}$). The next step is to quantify how the optical density readings depend on protein concentration.

> We will be using this new data set to demonstrate how to conduct non-linear regression in R. The data live in the ‘ELISA.CSV’ file. The code below assumes those data have been read into a tibble called `ELISA.CSV`. Set that up if you plan to work along.

### 24.1.1 First steps

We start by using `glimpse` to see which variables reside within `elisa_assay`:

```
glimpse(elisa_assay)
```

```
## Rows: 9
## Columns: 2
## $ pconc   <dbl> 0.04882812, 0.19531250, 0.39062500, 0.78125000, 1.56250000, 3.~
## $ density <dbl> 0.017500, 0.122500, 0.210500, 0.375500, 0.611500, 1.010000, 1.~
```

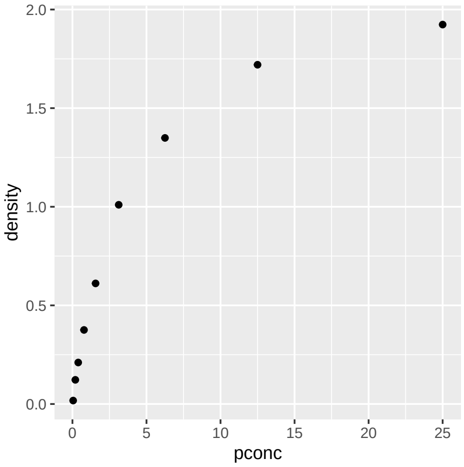
The first column (`pconc`) contains the information about about the protein concentration ($ng\ ml^{-1}$). The second column (`density`) contain the optimal density measurements registered by the assay. These are both numeric ratio variables.

Our goal is to predict optimal density from protein concentration, meaning `pconc` will be the response variable and `density` will be the predictor variable.

### 24.1.2 Visualising the data

We need to visualise the data to help us understand it. A scatter plot is a reasonable option here:

```
ggplot(elisa_assay, aes(x = pconc, y = density)) +
  geom_point()
```

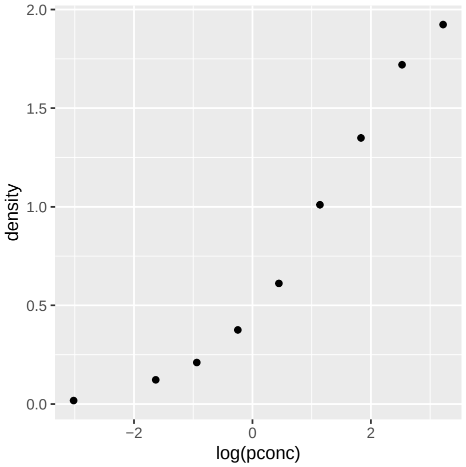


Since `density` will be the response and `pconc` will be the predictor, we place them on the $y$ and $x$ axes, respectively. The plot reveals the form of the relationship between the two variables. The plot suggests there is a tight, non-linear positive association between optical density and protein concentration. It is obvious we need a non-linear model as well — the relationship cannot be captured by a straight line.

The plot also contains a clue about how to analyse these data. Look carefully at the spacings between successive protein concentrations. These are not even. Instead, most increments are associated with a rough doubling of the lower value. When we see a sequence of values that arise by repeatedly multiplying by a factor, it is often a sign they should be analysed on a *logarithmic* scale.

With that in mind, let's make a new scatter plot using a natural log transformation of protein concentration:

```
ggplot(elisa_assay, aes(x = log(pconc), y = density)) +
  geom_point()
```



This shows that the relationship between optical density and log protein concentration is initially shallow, steepens toward the middle, and then becomes more shallow again at higher protein concentrations. In other words, the relationship follows an "S"-shaped sigmoid function.

There are a many different sigmoid functions that can describe this kind of relationship. We'll work with the most the commonly-used one: the **logistic function**.

## 24.2 Model fitting

Carrying out non-linear regression analysis in R is similar to simple linear regression. The first step involves fitting the model to find the best fit line and calculating various bits of additional information needed to use the model. However, since we are no longer in linear model land we cannot use `lm` to do this.

There are many different packages and functions for working with non-linear models in R. We will use the base R `nls` function because it is simple, flexible and readily available.

### How does `nls` work?

The letters in `nls` stand for 'non-linear least squares'. This is a reference to the method used to identify the best-fitting curve through the data. Non-linear least squares aims to find the parameter set that minimises the 'sum of squares' of the model residuals (i.e. the residual sum of squares). If that sounds familiar, it is because this is the same idea that underpins how general linear models like regression and one-way ANOVA work. The

> only difference is that non-linear modelling tools like **`nls`** has to use fancier algorithms than `lm` to cope with the non-linearity in the parameters.

Before we use **`nls`** to fit the logistic model to the ELISA data, we need to understand the model parameters. In mathematical terms, the logistic model can be written down like this:

$$y = \frac{\phi_1}{1 + e^{(\phi_2 - x)/\phi_3}}$$

The model has three parameters:

- $\phi_1$ is the value the function approaches as the predictor variable $x$ gets very large (i.e. the asymptote). Inspection of the scatter plot suggests this is some number greater than 2 in the ELISA example.
- $\phi_2$ is the value of $x$ at which the logistic function is exactly half its maximum value (i.e. the midpoint). This value is also the inflection point, i.e. where the rate of change is greatest. It looks like this is roughly 1 in the ELISA example.
- $\phi_3$ is a scale parameter that controls how rapidly the function transitions from low to high values as $x$ changes. It is the inverse of the rate of change of the function at the inflection point. This one is a little harder to guess from plot.

Now that we know a bit about how the logistic model works we can fit it to the ELISA data. There are a few different ways to do this with **`nls`**. The easiest looks like this:

```
elisa_mod1 <-
  nls(density ~ SSlogis(log(pconc), phi_1, phi_2, phi_3),
      data = elisa_assay)
```

We had to assign at least two arguments:

1. The first argument is a **formula** because it includes a 'tilde' symbol: `~`. The variable named to the left of the `~` is the response variable (`density`). The part to the right of the `~` defines the non-linear model. We will come back to how this works this in a moment. For now, notice that the the predictor variable (`pconc`) appears on the right-hand side of the formula. Just as with an `lm` model, it is essential to ensure the predictor and response variables are in the correct position when setting up a non-linear regression model.

2. The second `data` argument is just the name of the data frame or tibble object that contains the two variables listed in the formula (`elisa_assay`).

What is the SSlogis function doing in the formula? The 'logis' part of the name tells nls to fit a logistic function to the data. What about the 'SS' part? That stands for 'self-starting'. Non-linear models can be tricky to work with because the algorithms that search for the best fit curve are prone to failure in some circumstances. One such circumstance arises when the algorithm starts its search with a set of parameters that are too far away from their best fit values. The self-starting functions automatically identify good starting values so that nls users don't have to worry about that problem.

That leaves us with the arguments of SSlogis to worry about:

1. The first argument is the predictor variable. Our predictor variable is protein concentration (pconc). However, since we decided to model the dependence of optimal density on the *logarithm* of protein concentration, we applied the log function to pconc to ensure the model works on the correct scale.

2. The next three arguments are the names used to label the model's three parameters. We can use any name we like for these arguments. We chose phi_1, phi_2 and phi_3 to match the equation above, but we could have used something like Asym, xmid and scal to reflect the meaning of each parameter (those happen to be the labels used in the SSlogis help file).

Keep in mind that SSlogis uses the position of the three names in the argument list to match them with their roles in the logistic function. If we swapped phi_1 and phi_2 around in SSlogis, the labels in the model output would no longer match their role in the equation. That would be very confusing because $\phi_1$ would be labelled phi_2 and $\phi_2$ would be labelled phi_1 in the model output.

Now we have fitted the logistic model to the ELISA data and called the resulting model object elisa_mod1. What happens if we print this model object to the console?

```
print(elisa_mod1)
```

```
## Nonlinear regression model
##   model: density ~ SSlogis(log(pconc), phi_1, phi_2, phi_3)
##    data: elisa_assay
## phi_1 phi_2 phi_3
## 2.247 1.384 1.009
##  residual sum-of-squares: 0.002415
##
## Number of iterations to convergence: 0
## Achieved convergence tolerance: 1.019e-07
```

This prints out a summary of the fitted model, the values of the best-fit parameters, and a bit of technical information that we can usually ignore. Notice that the labelling of the parameters reflects the names we provided to SSlogis.

We often fit non-linear models because their parameters relate to a particular biological, chemical or physical process. When that is the case, it is important to think about what those numbers mean. For example:

- The asymptote of the fitted function ($\phi_1$) is about 2.2 units. This is the maximum optical density value produced by the assay when it is saturated with protein.
- The midpoint of the fitted function ($\phi_2$) is reached at around 1.4 $ng\ ml^{-1}$. This is the threshold protein concentration at which the optical density is half its maximum.
- The inverse of the rate of change at the midpoint value ($\phi_3$) is about 1.0 $ng\ ml^{-1}$. We can use this to understand the sensitivity of the ELISA. When this value is very small, it means there is a tight threshold in the response.

### 24.2.1 Custom functions

There are self-starting functions for many common non-linear relationships. If one is available for the model we want to use, it is a good idea to use it. But what do we do if there is no self-starting function for the relationship we want to model? In those circumstances, it is up to the user to define the curve.

This is relatively easy. Here is an example of how to set up and fit the logistic model without using its self-starting function:

```
elisa_mod2 <-
  nls(density ~ phi_1 / (1 + exp((phi_2 - log(pconc)) / phi_3)),
      data = elisa_assay,
      start = list(phi_1 = 2, phi_2 = 1, phi_3 = 1))
```

Now we have to set three arguments:

1. The first argument is a formula with the response variable (`density`) to the left of the `~`. The part to the right of the `~` defines the model we want to fit, but we have to explicitly define the curve in terms of the relevant R functions. This is easy as long as we understand how to express mathematical calculations in R.

2. As usual, the `data` argument is again the name of the data frame or tibble object containing the variables listed in the formula (`elisa_assay`).

3. The `start` argument provides some starting parameter values for `nls` to use when it begins searching for the best-fit curve. We have to provide a starting value for every parameter listed in the formula in something called a list. We do this by providing a series of comma-separated name-value pairs to the `list` function.

A user must provide starting values to the `nls` function when not using a self-starting function. However, we can't choose any numbers we like because `nls` will fail if they generate a curve that is too far away from the best fit. How do we choose reasonable starting values? By looking at the data. We did that earlier when we discussed how the logistic model works. There's no simple shortcut for choosing starting values. We have to understand how a model's parameters relate to the relationship in the data. The good news is for simple models like the logistic function, the initial guess does not need to be that close to the best fit parameters for `nls` to work.

**Which self-starting functions are available?**

Many standard non-linear curves have been implemented as self-start functions:

- `SSasymp` is a three-parameter asymptotic model that allows the curve to go through a non-zero intercept.
- `SSasympOff` is another version of the asymptotic model that uses a different mathematical parametrisation.
- `SSasympOrig` is the two-parameter version of the asymptotic model that forces the curve to go through the origin.
- `SSbiexp` is a four-parameter model that creates a non-linear curve by adding two exponential curves.
- `SSfpl` is a four-parameter model that generalises the logistic function to allow a non-zero lower bound.
- `SSgompertz` is the three-parameter Gomertz growth curve that provides an alternative choice for S-shaped relationships.
- `SSgompertz` is the four-parameter Weibull growth curve that provides another alternative to the logistic model.
- `SSmicmen` is the two-parameter Michaelis-Menten curve for modelling enzyme kinetics.

The easiest way to quickly learn about these is to navigate to their help page (e.g. `?SSmicmen`) and run all the example code at the end. That code creates a plot showing the shape of the curve along with some annotation revealing how the parameters control its shape.

## 24.3 Checking the assumptions

Like any parametric technique, non-linear regression makes assumptions about the nature of the data:

1. **Independence.** The residuals must be independent, such that each residual does not depend on any others.

2. **Measurement scale.** The response and predictor variables are both measured on an interval or ratio scale.
3. **Measurement error.** The values of the predictor variable are determined with negligible measurement error.
4. **Shape.** The general relationship between the predictor and response variable follows the assumed curve.
5. **Constant variance.** The variance of the residuals is not related to the value of the predictor variable.
6. **Normality.** The residuals are drawn from a normal distribution.

These are almost identical to the assumptions of simple linear regression. The only practical difference is that now we need to assess whether the assumed curve is a good fit to the data rather than worrying about whether the relationship follows a straight line.

The experimental design and data collection protocols affect the first three assumptions. We can only evaluate these by thinking about where the data come from. The remaining assumptions can be checked by using the fitted values and residuals from the fitted model to construct regression diagnostic plots.

The fitted values and residuals of a non-linear regression are defined in the usual way:

- The fitted values are the predicted values of the response for each value of the predictor variable from the best-fit model.
- The residuals are the deviations of each value of the response variable from their corresponding fitted values.

Base R has no simple facility for creating diagnostic plots with `nls` models. Fortunately, there *is* a package that makes these. This is called **nlstools**. Once installed, we need to load and attach **nlstools** to use it:

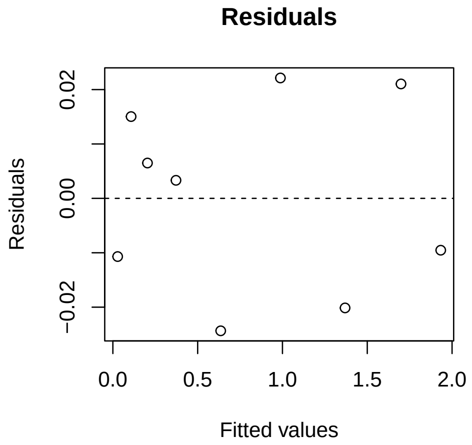
```
library(nlstools)
```

Creating a regression diagnostic plot with **nlstools** is then a two-step process. To begin, use the `nlsResiduals` function to extract the required information from the fitted model object:

```
elisa_mod_diag <- nlsResiduals(elisa_mod1)
```

We then plot the object produced by `nlsResiduals` to make different types of diagnostic plot. For example, we can look at the association between the residuals and fitted values from the model as follows:

```
plot(elisa_mod_diag, which = 1)
```



This plot allows us to evaluate whether the shape of the relationship follows the assumed curve. There should be no obvious relationship between the residuals and fitted values if the non-linear model is appropriate for the data. There is no obvious pattern in that plot, indicating that that logistic model is a good fit for the ELISA data.

Setting `which` to 3 creates a scale-location plot to assess the constant-variance assumption. Setting `which` to 6 creates a normal probability plot to assess the normality assumption. We won't make these here. They can be interpreted in the same way as the corresponding plots for a simple linear regression model. Refer to the 'Checking and Fixing Models' section of the book for a reminder of how to use regression diagnostic plots.

## 24.4 What about significance tests?

We know how to set up significance tests for linear models to assess whether a regression slope differs from zero or a set of means are different. Conceptually similar tests can be constructed for non-linear regression models. However, these kinds of analyses are more complex. There are two main reasons for this:

1. Comparing a non-linear model across treatments or some other set of groups requires us to investigate changes in the shape of the curve. These changes may involve differences in one or several model parameters. Deciding which comparisons to make and setting these up is not entirely straightforward.

2. Non-linear models often involve dependent data. For example, if we measured the body mass of individuals through time, each individual's measurements belong to their own curve. In this kind of situation, each curve becomes its own statistical unit and the independence assumption of the residuals no longer holds.

We can use nls to deal with the first of these challenges—though it requires more work than something like lm—but nls definitely can't deal with the second kind of challenge. Instead, advanced tools such as the **nlme** package are required to construct comparisons and deal with non-independence in the data. These are well beyond the scope of this book, so we'll not consider significance tests for comparing non-linear models here.

The one thing we can do with an nls model is extract the standard errors and significance tests for the model parameters. This can be useful when the parameters quantify something scientifically meaningful. For example, the scientist developing the new assay might want to know the protein concentration at which the optical density midpoint is reached. That is given by the $\phi_2$ parameter.

We use the summary function to extract these types of information from a fitted model:

```
summary(elisa_mod1)
```

```
##
## Formula: density ~ SSlogis(log(pconc), phi_1, phi_2, phi_3)
##
## Parameters:
##       Estimate Std. Error t value Pr(>|t|)
## phi_1  2.24676    0.05276   42.58 1.12e-08 ***
## phi_2  1.38398    0.06393   21.65 6.34e-07 ***
## phi_3  1.00871    0.03473   29.05 1.10e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02006 on 6 degrees of freedom
##
## Number of iterations to convergence: 0
## Achieved convergence tolerance: 1.019e-07
```

This looks a lot like the output we get when using summary with an lm object. That is to be expected—the summary function typically provides a table of coefficients/parameters when used with regression models in R. There are four columns in the table:

1. Estimate gives the estimated parameter values of the best-fit model. This is the same information produced by printing the model object.
2. Std. Error gives the standard errors of the parameters to summarise their uncertainty.
3. t value provides a test statistic (= estimate / standard error) under the null hypothesis that the parameter value is 0.
4. Pr(>|t|) gives the p-value associated with the null hypothesis that the parameter value is 0.

Do we need to care about every column? Probably not. Unless there is a sound scientific reason to assess whether a parameter is significantly different from 0, we usually only care about the estimates and their standard errors. The latter are useful because they tell us how much confidence we can have in the estimated parameter values.

## 24.5 Presenting results

If non-linear regression results are presented only in the text, it is often appropriate to specify the best fit parameter values and their standard errors. We might summarise the preceding analysis as follows:

> The logistic model captured the relationship between log protein concentration ($ng\ ml^{-1}$) and optical density in the ELISA assay. The asymptotic optical density was 2.2 (s.e. = 0.053). The midpoint protein concentration of optical density was 1.4 (s.e. = 0.064). The inverse sensitivity of the assay was 1.0 (s.e. = 0.035).

This allows the reader to understand the scientific relevance of the fitted non-linear model and use the equation in further calculations. When working with non-linear models, it is a good idea to also present the results as a figure showing the original data and the fitted curve.

### 24.5.1 Plotting the fitted line and the data

The process of creating a visual summary of a non-linear model is straightforward. There is a predict function that works with `nls` model objects, which means we can use the same workflow as the simple linear regression case.

The first step is to generate a sequence of values of the predictor (`pconc`) from which we will generate predictions, remembering to place the sequence in a data frame. We will make a sequence from 0.04 to 28 to span the range of protein concentrations in the data:

```r
pred_data <- data.frame(pconc = seq(0.04, 28, length.out = 1000))
```

We have to give the sequence the same name as the predictor variable (`pconc`) to ensure the `predict` function can find the column it needs when making predictions. We made a sequence with a very small distance between successive values by setting `length.out` to 1000. We did this because we will end up using a logarithmic scale for protein concentration, and we need to ensure the plotted curve is smooth at low concentrations.

Now we can use the model object and prediction data frame with `predict`:

```
predicted_density <- predict(elisa_mod1, pred_data)
```

The predicted values need to be kept in a data frame alongside their corresponding values of `pconc` to be used with **ggplot2**. The easiest way to do this is to add the predictions to `pred_data`:

```
pred_data <- mutate(pred_data, density = predicted_density)
```

We name the column containing the predictions `density` to ensure its name matches that used in the `elisa_assay` data.

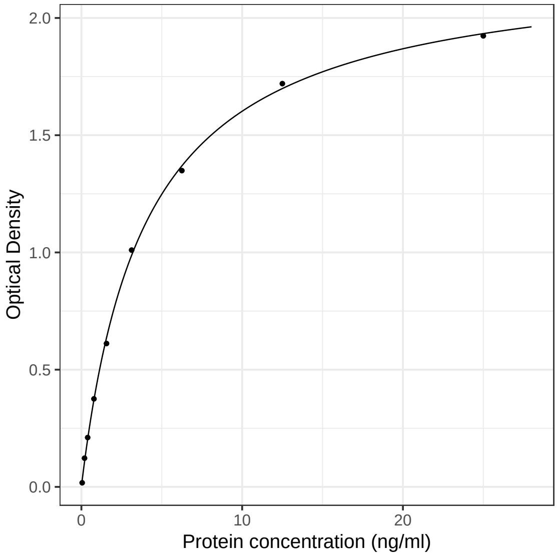
The `pred_data` object now has two columns called `pconc` and `density`, which contain the sequence of the protein concentrations and their predicted optical density values, respectively:

```
glimpse(pred_data)
```

```
## Rows: 1,000
## Columns: 2
## $ pconc   <dbl> 0.04000000, 0.06798799, 0.09597598, 0.12396396, 0.15195195, 0.~
## $ density <dbl> 0.02319109, 0.03895943, 0.05444924, 0.06968449, 0.08468013, 0.~
```

Now we can plot the predictions along with the data using `geom_line` and `geom_point`, respectively:

```
ggplot(pred_data, aes(x = pconc, y = density)) +
  geom_line() + geom_point(data = elisa_assay) +
  xlab("Protein concentration (ng/ml)") + ylab("Optical Density") +
  theme_bw(base_size = 16)
```

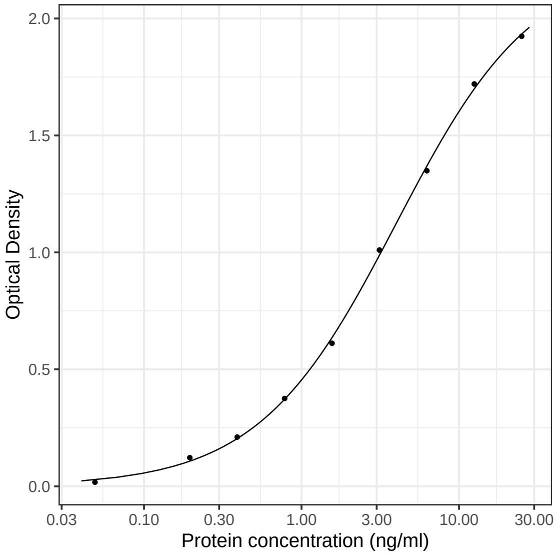


Setting up the column names of `pred_data` to match the response and predictor variable names means we only have to set up one aesthetic mapping for the raw data and model prediction layers. However, when adding the points, we need to tell **ggplot2** to look inside `elisa_assay` rather than use `pred_data`.

That worked fine, but it does not show protein concentration on a logarithmic scale. We used log-transformed values in the model, and in any case, the protein concentration values vary over several of orders of magnitude. Those are two good reasons to use a log scale on the x-axis.

There is more than one way to achieve this. For example, we could plot log-transform `pconc` and then plot the transformed values. Alternatively, we can get **ggplot2** to handle the transformation for us. The advantage of this second approach is that it provides nicer axis labels:

```
ggplot(pred_data, aes(x = pconc, y = density)) +
  geom_line() + geom_point(data = elisa_assay) +
  xlab("Protein concentration (ng/ml)") + ylab("Optical Density") +
  scale_x_continuous(trans = "log10", n.breaks = 6) +
  theme_bw(base_size = 16)
```



This uses the `scale_x_continuous` function set the x-axis to use a $\log_{10}$ scale and the number of breaks used for labelling that axis. That one extra line produces what we want. Now we have a nice publicationn-ready plot.

# Part IX

# Frequency Data and Non-parametric Tests

# Chapter 25

# Categorical data

Much of the time in biology, we are dealing with whole objects (plants, animals, cells, eggs, islands, etc.) or discrete events (attacks, matings, nesting attempts, etc.). We are often interested in making measurements of numeric variables (length, weight, number, etc.) and then either comparing means from samples (e.g. mean leaf size of plants from two habitat types) or investigating the association between different measurements (e.g. mean leaf size and herbivore damage).

However, we sometimes find a situation where the 'measurement' we are interested in is not a quantitative measure but is *categorical*. Categorical data are things like sex, colour or species. Such variables cannot be treated in the same way as numeric variables. Although we can 'measure' each object (e.g. record if an animal is male or female), we can't calculate numeric quantities such as the 'mean colour morph', 'mean species' or 'median sex' of animals in a sample. Instead, we work with the observed frequencies, in the form of counts, of different categories, or combinations of categories.

## 25.1 A new kind of distribution

There are quite a few options for dealing with categorical data[1]. We're just going to look at one option in this book: $\chi^2$ tests. This is pronounced, and sometimes written, 'chi-square'. The 'ch' is a hard 'ch', as in 'character'. They aren't necessarily the best tool for every problem, but $\chi^2$ tests are widely used in biology, so they are a good place to start.

[1] e.g. the 'log-linear model', 'Fisher's exact test', and the 'G-test'.

The material in this chapter is provided for readers who like to have a sense of how statistics works. Don't worry too much if it seems confusing—you don't have to understand it to use the tests we'll cover in the next two chapters. In any case, this chapter is very short.

The $\chi^2$ tests that we're going to study borrow their name from a particular theoretical distribution, called… the $\chi^2$ distribution. We won't study this in much detail. However, just as with the normal distribution and the *t*-distribution, it can be helpful to know a bit about it:

- The $\chi^2$ distribution pops up a lot in statistics. In contrast to the normal distribution, it isn't often used to model the distribution of data. Instead, the $\chi^2$ distribution is usually associated with a test statistic of some kind.
- The standard $\chi^2$ distribution is completely defined by only one parameter, called the degrees of freedom. This is closely related to the degrees of freedom idea introduced in the chapters on *t*-tests.
- The $\chi^2$ distribution is appropriate for positive-valued numeric variables. Negative values aren't allowed. This is because the distribution arises whenever we take one or more normally distributed variables, square these, and then add them up.

Let's take a look at the $\chi^2$ distribution with one degree of freedom:

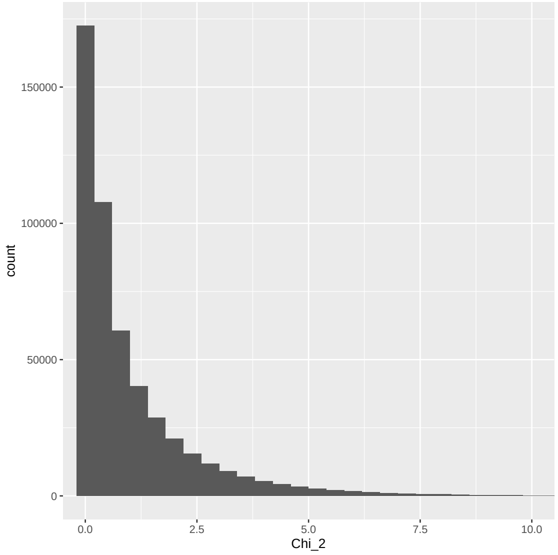


Figure 25.1: Distribution of a large sample of chi-square distributed variable with one degree of freedom

As we just noted, only positive values occur and most of these values lie between about 0 and 10. We can also see that the distribution is asymmetric. It is skewed to the right.

So why is any of this useful?

Let's look at the plant morph example again. Imagine taking repeated samples from a population when the purple morph frequency is 25% . If we take 1000 plants each time and the true frequency is 25%, we expect to sample 250 purple plants on average. We'll call this number the 'expected value'. We won't actually end up with 250 plants in each sample because of sampling error. We'll call this latter number the 'observed value'.

So far we're not doing anything we haven't seen before. We're just discussing what happens under repeated sampling from a population. Here's the new bit… Imagine that every time we sample the 1000 plants, we calculate the following quantity:

$$2 * \frac{(O-E)^2}{E}$$

…where $O$ is the observed value and $E$ is the expected value defined above. What does its distribution look like? We can find out by simulating the scenario in R and plotting the results:

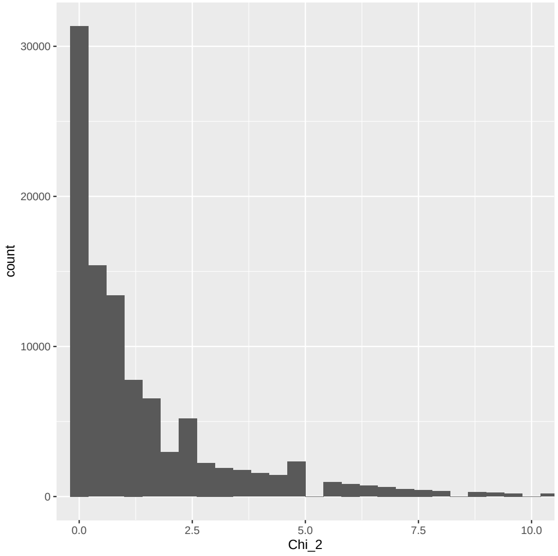


Figure 25.2: Distribution of the test statistic

That looks a lot like the theoretical $\chi^2$ distribution we plotted above. It turns out that observed frequencies ('counts') that have been standardised with respect to their expected values—via the $\frac{(O-E)^2}{E}$ statistic—have a $\chi^2$ sampling distribution (at least approximately). This result is the basis for using the $\chi^2$ distribution in various statistical tests involving categorical variables and frequencies or counts.

## 25.2 Types of test

We're going to learn about two different types of $\chi^2$ test. Although the two tests work on the same general principle, it is still important to distinguish between them according to where they are used.

### 25.2.1 $\chi^2$ goodness of fit test

A goodness-of-fit test is applicable when we have a single categorical variable and some hypothesis from which we can predict the expected proportions of observations falling in each category.

For example, we might want to know if there is any evidence for sex-related bias in the decision to study biology at Sheffield. We could tackle this question by recording the numbers of males and females in a cohort. This would produce a sample containing one nominal variable (Sex) with only two categories (Male and Female). Based on information about human populations, we know that the sex ratio among 18-year-olds is fairly close to 1:1[2]. We can thus compare the goodness of fit of the number of males and females in a sample of students with the expected value predicted by the 1:1 ratio.

If we had a total of 164 students, we might get this sort of table:

| | Male | Female |
|---|---|---|
| **Observed** | 74 | 90 |

With a 1:1 sex ratio, if there is no sex bias in the decision to study biology, we would expect 82 of each sex. It looks as though there might be some discrepancy between the expected values and those actually found. However, this could be entirely consistent with sampling variation. The $\chi^2$ goodness of fit test allows us to test how likely it is that such a discrepancy has arisen through sampling variation.

[2] Human sex-ratio is actually slightly biased toward males at birth, but since males experience a higher mortality rate in their teens, the sex ratio among 18 year olds is closer to 1:1.

### 25.2.2 $\chi^2$ contingency table test

A contingency table test is applicable when each object is classified according to more than one categorical variable. Contingency table tests are used to test whether there is an association between the categories of those variables.

Consider biology students again. We might be interested in whether eye colour was in any way related to sex. It would be simple to record eye colour (e.g. brown vs blue) along with the sex of each student in a sample. Now we would end up with a table that had two classifications:

| | Blue eyes | Brown eyes |
|---|---|---|
| **Male** | 44 | 20 |
| **Female** | 58 | 42 |

Now it is possible to compare the proportions of brown and blue eyes among males and females… The total number of males and females are 64 and 100, respectively. The proportion of males with brown eyes is $20/64 = 0.31$, and that for females $42/100 = 0.42$. It appears that brown eyes are somewhat less prevalent among males. A contingency table test will tell us if the difference in eye colour frequencies is likely to have arisen through sampling variation.

Notice that we are not interested in judging whether the proportion of males, or the proportion of blue-eyed students, are different from some expectation. That's the job of a goodness of fit test. We want to know if there is an association between eye colour and sex. That's the job of a contingency table test.

### 25.2.3 The assumptions and requirements of $\chi^2$ tests

Contingency tables and goodness-of-fit tests are not fundamentally different from one another in terms of their assumptions. The difference between the two types lies in the type of hypothesis evaluated. When we carry out a goodness-of-fit test, we have to supply the expected values. In contrast, the calculation of expected values is embedded in the formula used to carry out a contingency table test. That will make more sense once we've seen the two tests in action.

$\chi^2$ tests are sometimes thought of as **non-parametric** tests because they do not assume any particular form for the distribution of the data. In fact, as with any statistical test, there are some assumptions in play, but these are relatively mild:

- The data are independent counts of objects or events which can be classified into mutually exclusive categories.
- The expected counts are not very low. The general rule of thumb is that the expected values should be greater than 5.

The most important thing to remember about $\chi^2$ tests is that they must always be carried out on the actual counts. Although the $\chi^2$ is telling us how the proportions of objects in categories vary, the analysis should never be carried out on the percentages or proportions, only on the original count data, nor can a $\chi^2$ test be used with means.

# Chapter 26

# Goodness of fit tests

## 26.1 When do we use a chi-square goodness of fit test?

A $\chi^2$ goodness of fit test is appropriate in situations where we are studying a categorical variable, and we want to compare the frequencies of each category to pre-specified, expected values. Here are a couple of examples:

- We've already seen one situation where a goodness of fit test might be useful: the analysis of the sex ratio among biology undergraduates. We have a prior prediction about what the sex ratio should be in the absence of bias (1:1) and we wanted to know if there was any evidence for sex-related bias in the decision to study biology. We can use a goodness of fit test to compare the number of males and females in a sample of students with the predicted values to determine whether the data are consistent with the equal sex ratio prediction.

- Red campion (*Silene dioica*) has separate male (stamen bearing) and female (ovary and stigma bearing) plants. Both sexes can be infected by the anther smut *Ustillago violacea*. This smut produces spores in the plant's anthers, which are then transported to other host plants by insect vectors. *Ustillago* causes their sex to change on infecting the female flowers, triggering stamen development in the genetically female flowers. In populations of *Silene* in which there is no infection by *Ustillago* the ratio of male to female flowers is 1:1. Significant amounts of infection by the fungus may be indicated if there is an increase in the proportion of apparently male flowers relative to the expected 1:1 ratio.

The two examples considered above are as simple as things get: there are only two categories (Males and Females), and we expect a 1:1 ratio. However, the

$\chi^2$ goodness of fit test can be employed in more complicated situations:

- The test can be applied to any number of categories. For example, we might have a diet choice experiment where squirrels are offered four different food types in equal proportions, and the food chosen by each squirrel is recorded. The study variable would then have four categories: one for each food type.
- The expected ratio need not be 1:1. For example, the principles of Mendelian genetics predict that the offspring of two plants which are heterozygous for flower colour (white recessive, pink dominant) will be either pink or white flowered, in the ratio 3:1. Plants from a breeding experiment could be tested against this expected ratio.

## 26.2 How does the chi-square goodness of fit test work?

The $\chi^2$ goodness of fit test uses raw counts to address questions about expected proportions or probabilities of events[1]. As always, we start by setting up the appropriate null hypothesis. This will be question specific, but it must always be framed in terms of 'no effect' or 'no difference'. We then work out what a sampling distribution of some kind looks like under this null hypothesis and use this to assess how likely the observed result is (i.e. calculate a $p$-value).

We don't need to work directly with the sampling distributions of counts in each category. Instead, we calculate an appropriate $\chi^2$ test statistic. The way to think about this is that the $\chi^2$ statistic reduces the information in the separate category counts down to a single number.

Let's see how the $\chi^2$ goodness of fit test works using the *Silene* example discussed above. Imagine that we collected data on the frequency of plants bearing male and female flowers in a population of *Silene dioica*:

| | Male | Female |
|---|---|---|
| **Observed** | 105 | 87 |

We want to test whether the ratio of male to female flowers differs significantly from that expected in an uninfected population. The 'expected in an uninfected population' situation is the null hypothesis for the test.

**Step 1.** Calculate the counts expected when the null hypothesis is correct. This is the critical step. In the *Silene* example, we need to determine how many male

[1] We said it before, but it's worth saying again. Do not apply the goodness of fit test to proportions or means.

and female plants we expected to sample if the sex ratio really was 1:1.

**Step 2.** Calculate the $\chi^2$ test statistic from the observed and expected counts. We will show this later. However, the calculation isn't all that important, in the sense that we don't learn much by doing it. The resulting $\chi^2$ statistic summarises—across all the categories—how likely the observed data are under the null hypothesis.

**Step 3.** Compare the $\chi^2$ statistic to the theoretical predictions of the $\chi^2$ distribution to assess the statistical significance of the difference between observed and expected counts.

The interpretation of this $p$-value in this test is the same as for any other kind of statistical test: it is the probability we would see the observed frequencies, or more extreme values, under the null hypothesis.

### 26.2.1 Assumptions of the chi-square goodness of fit test

Let's remind ourselves about the assumptions of the $\chi^2$ goodness of fit test:

1. The data are independent counts of objects or events which can be classified into mutually exclusive categories. We shouldn't aggregate *Silene* sex data from different surveys unless we were certain each survey had sampled different populations.

2. The expected counts are not too low. The rule of thumb is that the expected values (not the observed counts!) should be greater than 5. If any of the expected values are below 5, the $p$-values generated by the test start to become less reliable.

## 26.3 Carrying out a chi-square goodness of fit test in R

> There is no need to download any data to work through this example. Although it is generally a good idea to keep data and code separate, the data used in a $\chi^2$ goodness of fit test are so simple we sometimes keep them in our R code. That's what we'll do here.

Let's see how to use R to carry out a $\chi^2$ goodness of fit test with the *Silene* sex data. .

The first step is to construct a numeric vector containing the *observed frequencies* for each category. We'll call this `observed_freqs`:

```r
observed_freqs <- c(105, 87)
observed_freqs
```

```
## [1] 105  87
```

Note that `observed_freqs` is a numeric vector, not a data frame. It doesn't matter what order the two category counts are supplied in.

Next, we need to calculate the *expected frequencies* of each category. Rather than expressing these as counts, R expects these to be proportions. We need to construct a second numeric vector containing this information. We'll call this `expected_probs`:

```r
# calculate the number of categories
n_cat <- length(observed_freqs)
# calculate the number in each category (equal frequencies in this e.g.)
expected_probs <- rep(1, n_cat) / n_cat
expected_probs
```

```
## [1] 0.5 0.5
```

Finally, use the `chisq.test` function to calculate the $\chi^2$ value, degrees of freedom and $p$-value for the test. The first argument is the numeric vector of observed counts (the data), and the second is the expected proportions in each category:

```r
chisq.test(observed_freqs, p = expected_probs)
```

```
## 
##  Chi-squared test for given probabilities
## 
## data:  observed_freqs
## X-squared = 1.6875, df = 1, p-value = 0.1939
```

The vectors containing the data and expected proportions have to be the same length for this to work. R will complain and throw an error otherwise.

The output is straightforward to interpret. The first part (`Chi-squared test for given probabilities`) reminds us what kind of test we did. The phrase 'for given probabilities' is R-speak informing us that we have carried out a goodness of fit test. The next line (`data:  observed_freqs`) reminds us what data we used. The final line is the one we care about: `X-squared = 1.6875, df = 1, p-value = 0.1939`. This output shows us the $\chi^2$ test statistic, the degrees of freedom associated with the test, and the $p$-value. Since $p > 0.05$, we conclude that the sex ratio is not significantly different from a 1:1 ratio.

**Degrees of freedom for a $\chi^2$ goodness of fit test**

We always need to know the degrees of freedom associated with our tests. This is $n$−1 in a $\chi^2$ goodness-of-fit test, where $n$ is the number of categories. This comes from the fact that we have to calculate one expected frequency per category (= $n$ frequencies), but since the frequencies have to add up to the total number of observations, once we know $n - 1$ frequencies, the last one is fixed.

### 26.3.1 Summarising the result

Having obtained the result we need to write the conclusion. As always, we always go back to the original question to write the conclusion. In this case the appropriate conclusion is:

> The observed sex ratio of *Silene dioica* does not differ significantly from the expected 1:1 ratio ($\chi^2 = 1.69$, d.f. $= 1$, $p = 0.19$).

### 26.3.2 A bit more about goodness of fit tests in R

There is a useful short cut that we can employ when we expect equal numbers in every category (as above). There is no need to calculate the expected proportions in this situation because R will assume we meant to use the 'equal frequencies' null hypothesis. So the following…

```
chisq.test(observed_freqs)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  observed_freqs
## X-squared = 1.6875, df = 1, p-value = 0.1939
```

…is exactly equivalent to the longer method we used first. We showed the first method because we do sometimes need to carry out a goodness of fit test assuming unequal expected values.

R will warn us if it thinks the data are not suitable for a $\chi^2$ test. Though it is often safe to ignore warnings produced by R, this is a situation where it is important to pay attention to it. We can see what the relevant warning looks like by using `chisq.test` with a fake data set:

```
chisq.test(c(2,5,7,5,5))
```

```
## Warning in chisq.test(c(2, 5, 7, 5, 5)): Chi-squared approximation may be
## incorrect
```

```
##
##  Chi-squared test for given probabilities
##
## data:  c(2, 5, 7, 5, 5)
## X-squared = 2.6667, df = 4, p-value = 0.6151
```

The warning `Chi-squared approximation may be incorrect` is telling us that there might be a problem with the test. What is the problem? The expected counts are below 5, which means the $p$-values produced by `chisq.test` will not be reliable.

### 26.3.3 Doing it the long way…

People tend not to carry out $\chi^2$ goodness of fit tests by hand these days. Why bother when a computer can do it for us? Nonetheless, we promised to show the calculation of the $\chi^2$ test statistic. It is found by taking the difference of each observed and expected count, squaring those differences, dividing each of these squared differences by the expected frequency, and finally, summing these numbers over the categories. That's what this formula for the $\chi^2$ test statistic means:

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

In this formula the $O_i$ are the observed frequencies, the $E_i$ are the expected frequencies, the $i$ in $O_i$ and $E_i$ refer to the different categories, and the $\Sigma$ means summation ('add up').

Does that look formula familiar? It looks a lot like the quantity we calculated when comparing plant morph counts to their expected value the previous categorical data chapter. That's no accident.

Here's how the calculation of the $\chi^2$ test statistic works with our current example:

1. Calculate out how many male and female plants we would have sampled on average if the sex ratio really were 1:1. We already did this—the numbers are: $(105 + 87)/2 = 192/2 = 96$ of each sex.

2. Calculate the $\frac{(O_i - E_i)^2}{E_i}$ term associated with each category. For the males, we have (105-96)$^2$ / 96 = 0.844, and for females we (87-96)$^2$ / 96 = 0.844[2]. The $\chi^2$ statistic is the sum of these two values.

Once we have obtained the $\chi^2$ value and the degrees of freedom for the test (d.f. = 1), we can calculate its *p*-value. We won't show how to do this—it's much simpler to get the whole job done in one step using `chisq.test`.

## 26.4 Determining appropriate expected values

A goodness of fit test can only be applied if we can determine some expected values with which to compare the observed counts. Equally obviously, almost anything can be made significant by using inappropriate expected values. This means we always need to have a justifiable basis for choosing the expected values. In many cases the experiment can be designed, or the data collected, in such a way that we would expect to find equal numbers in each category if whatever it is we are interested in is not having an effect. At other times, the expectation can be generated by knowledge, or prediction, of a biological process e.g. a 1:1 sex ratio, a 3:1 ratio of phenotypes. At other times the expectation may need a bit more working out.

[2] These are the same because there are only two categories in this example.

# Chapter 27

# Contingency tables

## 27.1 When do we use a chi-square contingency table test?

A $\chi^2$ contingency table test is appropriate in situations where we are studying *two or more* categorical variables—each object is classified according to more than one categorical variable—and we want to evaluate whether categories are associated. Here are a couple of examples:

- Returning to the data on biology students, we suggested that we might want to know if eye colour was related to sex in any way. That is, we want to know whether brown and blue eyes occur in different proportions among males and females. If we recorded the sex and eye colour of male and female students, we could use a $\chi^2$ contingency table test to evaluate whether eye colour is associated with sex.

- The two-spot ladybird (*Adalia bipunctata*) occurs in two forms: the typical form, which is red with black spots and the dark form, which has much of the elytral surface black, with the two spots red. We want to know whether the relative frequency of the colour morphs is different in industrial and rural habitats. We could address this question by applying a $\chi^2$ contingency table test to an aggregate sample of two-spot ladybirds taken from rural and industrial habitats.

Let's think about how these kinds of data look. Here are the biology student sex and eye colour data again, organised into a table:

| | Blue eyes | Brown eyes |
|---|---|---|
| **Male** | 22 | 10 |
| **Female** | 29 | 21 |

Table 27.2: Contingency tables without an association (left table), and with an association (right table), among two categorical variables

| | C1 | C2 | | C1 | C2 |
|---|---|---|---|---|---|
| **R1** | 10 | 20 | **R1** | 10 | 80 |
| **R2** | 40 | 80 | **R2** | 40 | 20 |

This is called a two-way contingency table. It is a *two-way* contingency table because it summarises the frequency distribution of two categorical variables at the same time[1]. If we had measured three variables, we would have a *three-way* contingency table (e.g. 2 x 2 x 2).

A contingency table takes its name from the fact that it captures the 'contingencies' among the categorical variables: it summarises how the frequencies of one categorical variable are associated with the categories of another. The term association is used here to describe the non-independence of categories among categorical variables. Other terms used to refer to the same idea include 'linkage', 'non-independence', and 'interaction'.

Associations are evident when the proportions of objects in one set of categories (e.g. R1 and R2) depends on a second set of categories (e.g. C1 and C2). Here are two possibilities:

- There is no evidence of association in the first table (left): the numbers in category R1 are 1/4 of those in category R2, whether the observations are in category C1 or C2. Notice that this reasoning isn't about the total numbers in each category—there are 100 category C2 cases and only 50 category C1 cases.

- There is evidence of an association in the second table (right): the proportion of observations in R1 changes markedly depending on whether we are looking at observations for category C1 or category C2. The R1 cases are less frequent in the C1 column relative to the C2 column. Again, it is the proportions that matter, not the raw numbers.

We can ask a variety of different questions of data in a contingency table, but they are usually used to test for associations between the variables they summarise. That's the topic of this chapter. There are different ways to carry out such tests of association. We'll focus on the most widely used tool—the 'Pearson's Chi Square' ($\chi^{2}$) contingency table test[2].

[1] This is called their 'joint distribution', in case you were wondering.

[2] This is the same Pearson who invented the correlation coefficient for measuring linear associations by the way.

## 27.2 How does the chi-square contingency table test work?

The $\chi^2$ contingency table test uses data in the form of a contingency table to address questions about the dependence between two or more different *kinds* of outcomes, or events. We start to tackle this question by setting up the appropriate null hypothesis. The null hypothesis is always the same for the standard contingency table test of association: it proposes that different events are independent of one another. This means the occurrence of one kind of event does not depend on the other type of event, i.e. they are *not* associated.

Once the null hypothesis has been worked out, the remaining calculations are no different from those used in a goodness of fit test. We calculate the frequencies expected in each cell under the null hypothesis, we calculate a $\chi^2$ test statistic to summarise the mismatch between observed and expected values and then use this to assess how likely the observed result is under the null hypothesis, resulting in the *p*-value.

We'll continue with the two-spot ladybird (*Adalia bipunctata*) example from the beginning of this chapter. The dark (melanic) form is under the control of a single gene. Melanic and red types occur at different frequencies in different areas. Two observations are pertinent to this study:

1. In London melanics comprise about 10%, whereas in rural towns in northern England the frequency is greater (e.g. Harrogate 63%, Hexham 75%).
2. The frequency of melanics has decreased in Birmingham since smoke control legislation was introduced.

It was thought that the different forms might be differentially susceptible to some toxic component of smoke, but this doesn't explain the geographic variation in the proportions of melanics. It turns out that the effect is a subtle one in which melanic forms do better in conditions of lower sunshine than red forms due to their greater ability to absorb solar radiation. That means melanics are favoured where the climate is naturally less sunny, though there will also be smaller-scale variations due to local environmental conditions that affect solar radiation, such as smoke.

A survey was carried out of *Adalia bipunctata* in a large urban area and the more rural surrounding areas to test whether this effect still occurs in industrial areas. The following frequencies of different colour forms were obtained.

| | Black | Red | Totals |
|---|---|---|---|
| **Rural** | 30 | 70 | *100* |
| **Industrial** | 115 | 85 | *200* |
| **Totals** | *145* | *155* | *300* |

We want to test whether the proportions of melanics are different between urban and rural areas. In order to make it a bit easier to discuss the calculations involved, we'll refer to each cell in the table by a letter…

| | Black | Red | Totals |
|---|---|---|---|
| **Rural** | $a$ | $b$ | $e$ |
| **Industrial** | $c$ | $d$ | $f$ |
| **Totals** | $g$ | $h$ | $k$ |

We can now step through the steps involved in a $\chi^2$ contingency table test:

**Step 1.** We need to work out the expected numbers in cells a-d, under the null hypothesis that the two kinds of outcomes (colour and habitat type) are independent of one another. Let's see how it works for the Black-Rural cell ('a'):

- Calculate the probability that a randomly chosen individual in the sample is from a rural location ($p$(Rural)). This is the total 'Rural' count ($e$) divided by the grand total ($k$):

$$p(\text{Rural}) = \frac{e}{k}$$

- Calculate the probability that a randomly chosen individual in the sample set is black ($p$(Black)). This is the total 'Black' count ($g$) divided by the grand total ($k$):

$$p(\text{Black}) = \frac{g}{k}$$

- Calculate the probability that a randomly chosen individual is both 'Rural' and 'Black', *assuming* these events are 'independent' . This is given by the product of the probabilities from steps 1 and 2:

$$p(\text{Rural, Black}) = \frac{e}{k} \times \frac{g}{k}$$

- Convert this to the expected number of individuals that are 'Rural' and 'Black', under the independence assumption. This is the probability from step 3 multiplied by the grand total ($k$):

$$E(\text{Rural, Black}) = \frac{e}{k} \times \frac{g}{k} \times k = \frac{g \times e}{k}$$

In general, the expected value for any particular cell in a contingency table is given by multiplying the associated row and column totals and then dividing by the grand total.

**Step 2** Calculate the $\chi^2$ test statistic from the four observed cell counts and their expected values. The resulting $\chi^2$ statistic summarises—across all the cells—how likely the observed frequencies are under the null hypothesis of no association (= independence).

**Step 3** Compare the $\chi^2$ statistic to the theoretical predictions of the $\chi^2$ distribution in order to assess the statistical significance of the difference between observed and expected counts. This *p*-value is the probability we would see the observed pattern of cell counts, or a more strongly associated pattern, under the null hypothesis of no association. A low *p*-value represents evidence against the null hypothesis.

### 27.2.1 Assumptions of the chi-square contingency table test

The assumptions of the $\chi^2$ goodness of fit test are essentially the same as those of the goodness of fit test:

1. The observations are independent counts. For example, it would not be appropriate to apply a $\chi^2$ goodness of fit test where observations are taken before and after an experimental intervention is applied to the same objects.

2. The expected counts are not too low. The rule of thumb is that the expected values (again, not the observed counts) should be greater than 5. If any of the expected values are below 5, the *p*-values generated by the test start to become less reliable.

The second assumption is not as critical as some people imagine. If we find ourselves in a situation where the expected value in one or more cells is less than 5 there is a simple fix—use a different test. We'll highlight one way to achieve this below.

## 27.3 Carrying out a chi-square contingency table test in R

We'll use the ladybird colour and habitat type example to demonstrate how to conduct a chi-square contingency table test in R.

Three versions of the ladybird data will be used: LADYBIRDS1.CSV, LADYBIRDS2.CSV, and LADYBIRDS3.CSV. These all contain the same information. They only differ in how it is organised. We assume that each of these has been read into tibbles called `lady_bird_df1`, `lady_bird_df2` and `lady_bird_df3`. For example:

```
lady_bird_df1 <- read_csv(file = "LADYBIRDS1.CSV")
lady_bird_df2 <- read_csv(file = "LADYBIRDS2.CSV")
lady_bird_df3 <- read_csv(file = "LADYBIRDS3.CSV")
```

Carrying out a $\chi^2$ contingency table test in R is very simple: we use the `chisq.test` function again. The only snag is that we have to ensure the data are correctly formatted. When we read data into R using a function like `read_csv` (or `read.csv`), we end up with a tibble (or ordinary data frame). Unfortunately, the `chisq.test` function is one of the few statistical functions not designed to work with data frames. We have to convert our data to a special contingency table object to use it with `chisq.test`.[3]

How do we construct a contingency table object? We use a function called `xtabs`. The `xtabs` function does categorical 'cross tabulation'—it sums up the number of occurrences of different combinations of categories among variables. We need to see how to use `xtabs` before running through the chi-square contingency table test.

## 27.3.1 Step 1. Get the data into the correct format

The usage of `xtabs` depends upon how the raw data are organised. There are possible cases.

### 27.3.1.1 Case 1: Raw data

Data suitable for analysis with a $\chi^2$ contingency table test are often represented in a data set with one column per categorical variable and one row per observation. This is the version of the ladybird data in the `lady_bird_df1`:

```
glimpse(lady_bird_df1)
```

```
## Rows: 300
## Columns: 2
## $ Habitat <chr> "Rural", "Rural", "Rural", "Rural", "Rural", "Rural", "Rural",~
## $ Colour  <chr> "Black", "Black", "Black", "Black", "Black", "Black", "Black",~
```

[3] The table objects produced by `xtabs` are **not** the same as the **dplyr** table-like objects: 'tibbles'. This is one reason why **dplyr** adopted the name 'tibble'. The overlap in names is unfortunate, but we'll have to live with it—there are only so many ways to name things that look like tables.

This data frame contains 300 rows (one for each ladybirds) and two columns (`Habitat` and `Colour`). Each column is a categorical variable containing information about the ladybirds. Each one has two unique values:

```
unique(lady_bird_df1$Habitat)
```

```
## [1] "Rural"      "Industrial"
```

```
unique(lady_bird_df1$Colour)
```

```
## [1] "Black" "Red"
```

We require a two-way table that contains the total counts in each combination of categories. This is what `xtabs` does. It takes two arguments: the first is a formula (involving the `~` symbol) that specifies the required contingency table; the second is the name of the data frame containing the raw data. When working with data in the above format—one observation per row—we use a formula that only contains the names of the categorical variables on the right hand side of the `~` (`~ Habitat + Colour`):

```
lady_bird_table <- xtabs(~ Habitat + Colour, data = lady_bird_df1)
```

When used like this, `xtabs` sums up the number of observations with each combination of `Habitat` and `Colour`.

We called the output `lady_bird_table` to emphasise that the data from `xtabs` are now stored in a contingency table. When we print this to the console we see that `lady_bird_table` does indeed refer to something that looks like a 2 x 2 contingency table of counts:

```
lady_bird_table
```

```
##             Colour
## Habitat       Black Red
##   Industrial   115  85
##   Rural         30  70
```

### 27.3.1.2 Case 2: Partial counts

Sometimes data suitable for analysis with a $\chi^2$ contingency table test are partially summarised into counts. For example, imagine that we visited five rural and five urban sites and recorded the numbers of red and black colour forms found at each site. This is the version of the ladybird data in the `lady_bird_df2`:

```
glimpse(lady_bird_df2)
```

```
## Rows: 20
## Columns: 4
## $ Habitat <chr> "Rural", "Rural", "Rural", "Rural", "Rural", "Rural", "Rural",~
## $ Site    <chr> "R1", "R2", "R3", "R4", "R5", "R1", "R2", "R3", "R4", "R5", "U~
## $ Colour  <chr> "Black", "Black", "Black", "Black", "Black", "Red", "Red", "Re~
## $ Number  <dbl> 10, 3, 4, 7, 6, 15, 18, 9, 12, 16, 32, 25, 25, 17, 16, 17, 23,~
```

It's a bit easier to see how the data are organised if we print the `lady_bird_df2` object:

```
lady_bird_df2
```

```
## # A tibble: 20 x 4
##    Habitat    Site  Colour Number
##    <chr>      <chr> <chr>   <dbl>
##  1 Rural      R1    Black      10
##  2 Rural      R2    Black       3
##  3 Rural      R3    Black       4
##  4 Rural      R4    Black       7
##  5 Rural      R5    Black       6
##  6 Rural      R1    Red        15
##  7 Rural      R2    Red        18
##  8 Rural      R3    Red         9
##  9 Rural      R4    Red        12
## 10 Rural      R5    Red        16
## 11 Industrial U1    Black      32
## 12 Industrial U2    Black      25
## 13 Industrial U3    Black      25
## 14 Industrial U4    Black      17
## 15 Industrial U5    Black      16
## 16 Industrial U1    Red        17
## 17 Industrial U2    Red        23
## 18 Industrial U3    Red        21
## 19 Industrial U4    Red         9
## 20 Industrial U5    Red        15
```

The counts at each site are in the `Number` variable, and the site identities are in the `Site` variable. We need to sum over the sites to get the total number within each combination of `Habitat` and `Colour`. We use `xtabs` again, but this time we also tell it which variable to sum over (`Number ~ Habitat + Colour`):

```
lady_bird_table <- xtabs(Number ~ Habitat + Colour, data = lady_bird_df2)
```

When working with partially summed counts data we have to use a formula where the name of the variable containing the counts is on left hand side of the `~`, and the names of the categorical variables to sum over are on the right hand side of the `~`. The `lady_bird_table` object produced by `xtabs` is no different than before:

```
lady_bird_table
```

```
##             Colour
## Habitat      Black Red
##   Industrial   115  85
##   Rural         30  70
```

Good! These are the same data.

#### 27.3.1.3 Case 3: Complete counts

Data suitable for analysis with a $\chi^2$ contingency table test are sometimes already summed into the required total counts. This is the version of the ladybird data in the `lady_bird_df3`:

```
glimpse(lady_bird_df3)
```

```
## Rows: 4
## Columns: 3
## $ Habitat <chr> "Industrial", "Industrial", "Rural", "Rural"
## $ Colour  <chr> "Black", "Red", "Black", "Red"
## $ Number  <dbl> 115, 85, 30, 70
```

Again, we can see how the data are organised by printing the `lady_bird_df3` object:

```
lady_bird_df3
```

```
## # A tibble: 4 x 3
##   Habitat    Colour Number
##   <chr>      <chr>   <dbl>
## 1 Industrial Black     115
## 2 Industrial Red        85
## 3 Rural      Black      30
## 4 Rural      Red        70
```

The total counts are already in the `Number` variable, meaning there is no need to sum over anything to get the total for each combination of `Habitat` and `Colour`.

However, we still need to convert the data to a contingency table object so we can feed it to `chisq.test`. One way to do this is by using `xtabs` again, exactly as before:

```
lady_bird_table <- xtabs(Number ~ Habitat + Colour, data = lady_bird_df3)
lady_bird_table
```

```
##             Colour
## Habitat      Black Red
##   Industrial   115  85
##   Rural         30  70
```

In this case `xtabs` doesn't change the data because it's just 'summing' over one value in each combination of categories. We're only using `xtabs` to convert it to a contingency table object. The resulting `lady_bird_table` object is the same as before.

### 27.3.2 Step 2. Do the test

Once we have the data in the form of a contingency table the associated $\chi^2$ test of independence between the two categorical variables is easy to carry out:

```
chisq.test(lady_bird_table)
```

```
## 
##  Pearson's Chi-squared test with Yates' continuity correction
## 
## data:  lady_bird_table
## X-squared = 19.103, df = 1, p-value = 1.239e-05
```

That's it. Just pass one argument to `chisq.test`: the contingency table.

This output should make sense in the light of what we saw in the previous chapter. R first prints a reminder of the test employed (`Pearson's Chi-squared test with Yates' continuity correction`) and the data used (`data: lady_bird_table`). We'll come back to the "Yates' continuity correction" bit in a moment. R then summarises the $\chi^2$ value, the degrees of freedom, and the $p$-value: `X-squared = 19.103, df = 1, p-value = 1.239e-05` The $p$-value is highly significant ($p$<0.001) indicating that the colour type frequency varies among the two kinds of habitats.[4]

[4] We could have summarised the result as: habitat type varies among the two colour types. This way of explaining the result seems odd though. Ladybirds are found within habitats, not the other way around. Just keep in mind that this is a semantic issue. The contingency table test doesn't make a distinction between directions of effects.

### Degrees of freedom for a $\chi^2$ contingency table test

We need to know the degrees of freedom associated with the test: in a two-way $\chi^2$ contingency table test these are $(n_A - 1) \times (n_B - 1)$, where $n_A - 1$ is the number of categories in the first variable, and $n_B - 1$ is the number of categories in the second variable. So if we're working with a 2 x 2 table the d.f. are 1, if we're working with a 2 x 3 table the d.f. are 2, if we're working with a 3 x 3 table the d.f. are 4, and so on.

What was that "Yates' continuity correction" all about? The reasoning behind using this correction is a bit beyond this book, but in a nutshell, it generates more reliable $p$-values under certain circumstances. By default, the `chisq.test` function applies this correction to all 2 x 2 contingency tables. We can force R to use the standard calculation by setting `correct = FALSE` if we want to:

```
chisq.test(lady_bird_table, correct = FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  lady_bird_table
## X-squared = 20.189, df = 1, p-value = 7.015e-06
```

Both methods give similar results in this example, though they aren't exactly the same—the $\chi^2$ value calculated when `correct = FALSE` is very slightly higher than the value found when using the correction.

Don't use the `correct = FALSE` option! The default correction is a safer option for 2 x 2 tables.

### What should we do when the expecetd values are below 5?

The expected cell counts all need to be greater than 5 when using the standard $\chi^2$ contingency table test. If thsi assumption is not met, the $\chi^2$ test statistic will deviate from its theoretical distribution and the test becomes unreliable.

Fortunately, if we find ourselves in a situation with low cell counts, `chisq.test` can be used to carry out an alternative test. We can access this alternative by setting the `simulate.p.value` argument to `TRUE`:

```
chisq.test(lady_bird_table, simulate.p.value = TRUE)
```

```
##
```

```
##  Pearson's Chi-squared test with simulated p-value (based on 2000
##  replicates)
##
## data:  lady_bird_table
## X-squared = 20.189, df = NA, p-value = 0.0004998
```

What just happened? R used a simulation approach to construct the null distribution for the test. The details don't matter too much, but logic of the approach is a lot like bootstrapping or permutation tests. R generates many hypothetical sets of cell counts based on the null hypothesis of no association, and then compares the distribution of their $\chi^2$ statistics to the value from the true data to calculate a $p$-value.

### 27.3.3 Summarising the result

We have obtained the result, so now we need to write the conclusion. As always, we go back to the original question to write the conclusion. In this case the appropriate conclusion is:

> There is a significant association between the colour of *Adalia bipunctata* individuals and habitat, such that black individuals are more likely to be found in industrial areas ($\chi^2 = 19.1$, d.f. $= 1$, $p < 0.001$).

Notice that we summarised the nature of the association alongside the statistical result. This is easy to do in the text when describing the results of a 2 x 2 contingency table test. It's much harder to summarise the association in written form when working with larger tables. Instead, we often present a table or a bar chart showing the observed counts.

## 27.4 Working with larger tables

Interpreting the results of a contingency table test is fairly straightforward in the case of a 2 x 2 table. We can certainly use contingency table tests with larger two-way tables (e.g. 3 x 2, 3 x 3, 3 x 4, etc), and higher-dimensional tables (e.g. 2 x 2 x 2), but the results become progressively harder to understand as the tables increase in size. If you get a significant result, it is often best to compare the observed and expected counts for each cell and look for the highest differences to try and establish what is driving the significant association. Visualising the data with a bar chart can also help with interpretation.

There are ways of subdividing large tables to make subsequent $\chi^2$ tests on individual parts of a table to establish specific effects, but these are not detailed here. Note that, unless we had planned to make more detailed comparisons before collecting the data, it is hard to justify such *post hoc* analysis.

# Chapter 28

# Non-parametric tests

## 28.1 What is a non-parametric test?

The majority of procedures we have been using to evaluate statistical significance require various assumptions about population distributions to be satisfied. We call these parametric methods because they are underpinned by a mathematical model of the population(s) (we discussed this idea in the Parametric statistics chapter). For this reason, the statistical tests associated with these methods—e.g. global significance tests and multiple comparisons tests in ANOVA—are called **parametric tests**.

**Non-parametric tests** are a class of statistical tests that make much weaker assumptions. The advantage of non-parametric tests is that they can be employed with a much wider range of forms of data than their parametric cousins. Although non-parametric tests are less restrictive in their assumptions, they are not, as is sometimes stated, assumption-free. The term non-parametric is just a catch-all term that applies to any test which doesn't assume the data are drawn from a *specific* distribution. We have already seen several examples of non-parametric tests:

- The bootstrap and permutation test procedures introduced in the first few chapters are non-parametric techniques.
- The $\chi^2$ goodness of fit and the $\chi^2$ contingency table tests make weak assumptions about the frequency data.

In this chapter, we are will look at some new non-parametric tests that perform analyses equivalent to $t$-tests, correlation analysis, and the global significance test in one-way ANOVA. But why do we need these in the first place?

## 28.2 Why use non-parametric tests ... and when?

### 28.2.1 What are the advantages?

Non-parametric tests make relatively weak assumptions about the distribution of the data, meaning they are useful in situations where our data are not well suited to parametric tests. When the assumptions of a particular parametric test are violated, a non-parametric version may be a better bet. For example, many non-parametric tests can be used to analyse data originally collected in ordinal, or rank, form. This is useful in investigations where the data cannot be collected any other way, for example:

- Subjects in a psychology experiment might be asked to rank a series of photographs of people in order of attractiveness.
- A panel of tasters judging the sweetness of wines may be able to score sweetness on a rank scale.
- Encounters between animals might be scored on the basis of the aggressive behaviours shown which can be put in rank order.

### 28.2.2 What are the disadvantages?

Given the advantages of non-parametric tests, there has to be a catch. Otherwise, everyone would use non-parametric tests all the time. In fact, they are usually used as a last resort. One problem with non-parametric tests is that if the data are actually appropriate for a parametric test the equivalent non-parametric test will be less powerful (i.e. less likely to find a difference even if there really is one). For this reason, an appropriate transformation followed by a parametric test will yield a more powerful analysis.

A second limitation of non-parametric tests is that they are less informative than parametric tests. For example, suppose we find a significant difference between group medians using a non-parametric test (see below). In that case, it can be difficult to understand which differences are driving the significant effect. On the other hand, if we can determine a suitable transformation and use an ANOVA model, we can deploy tools such as multiple comparison tests to understand the data better.

Parametric models make stronger distributional assumptions about the data. Still, they are much more flexible in some ways than non-parametric tests, i.e. there are non-parametric equivalents to some parametric tests. Still, there are many parametric tests for which there is no readily available non-parametric equivalent (e.g., the more complex designs of ANOVA). There is a non-parametric equivalent to ANOVA for complete randomised block design with one treatment factor, called *Friedman's test* (available via the `friedman.test` function in R), but beyond that, the options are very limited unless we know how to use advanced techniques such as the bootstrap.

## 28.3 How does a non-parametric test work?

The key thing about the non-parametric tests in this chapter is that the calculations are done using the rank order of the data, whatever the type of the original data—-ratio, interval, or ordinal. The general principle of such a test can be easily illustrated with the following example. Imagine that we want to compare two samples to determine whether they differ in their central tendency. That is, we want to know if the values in one sample tend to be larger or smaller than the values in a second sample. For example:

| | | | | |
|---|---|---|---|---|
| Sample A: | 1.3 | 2.8 | 4.1 | 3.2 |
| Sample B: | 7.2 | 3.0 | 4.2 | 6.2 |

If all the data (i.e. from both samples) are put in ascending order…

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Sample A: | 1.3 | 2.8 | | 3.2 | 4.1 | | | |
| Sample B: | | | 3.0 | | | 4.2 | 6.2 | 7.2 |

Then each number can be given a rank according to its place in the ordering…

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Sample A: | 1 | 2 | | 4 | 5 | | | |
| Sample B: | | | 3 | | | 6 | 7 | 8 |

It is easy to see now that if the rank values from each sample are added up, Sample A will have a lower value (sum = 12) than Sample B (sum = 24). This suggests that sample B has larger values than sample A.

If the samples had been completely non-overlapping, the totals would have been:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Sample A: | 1 + | 2 + | 3 + | 4 | | | | | = 10 |
| Sample B: | | | | | 5 + | 6 + | 7 + | 8 | = 26 |

On the other hand, if the samples had been largely overlapping, the rank totals would have been equal, or close to it:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Sample A: | 1 + | | | 4 + | | 6 + | 7 | | = 18 |
| Sample B: | | 2 + | 3 + | | 5 + | | | 8 | = 18 |

Obviously, the greater the difference in the rank totals, the less likely it is that the two samples could have come from the same distribution of values—i.e. the more likely it is that difference between them may be statistically significant.

The important thing to notice in such a procedure is that the original data could have been replaced with quite different values which, provided the ordering was the same, would have given exactly the same result. In other words the outcome of the test is reasonably insensitive to the underlying distribution of the data.

## 28.4 Non-parametric equivalents

Several non-parametric procedures provide similar types of tests to the simple parametric tests we have seen already. The actual calculations for the different tests are a little more involved than the general outline above, but they work from the same basic principle: find the rank order of all the data and then use the ranks in different groups to assess the statistical significance of the observed differences. As always, we have to specify some kind of null hypothesis (e.g. the medians of two samples are the same) to make the significance calculation work.

The four tests discussed here are:

1) **Wilcoxon signed-rank test**: This test is equivalent to a one-sample and paired-sample *t*-test. This test can be used to:
    - compare a sample to a single value, or
    - test for differences between paired samples.
2) **Mann-Whitney *U*-test**: This is equivalent to the two-sample *t*-test—it tests for differences central tendency of two unpaired samples.
3) **Kruskal-Wallis test**: This is equivalent to the global significance test in a one-way ANOVA—it tests for differences between the central tendency of several samples.
4) **Spearman's Rank correlation**: This is equivalent to Pearson's correlation—it tests for an association between two variables.

Notice that we haven't said anything about *what kind* of differences among samples these evaluate, i.e. we didn't specify the null hypothesis. We'll address this question as we discuss each test.

⚠ **So many names!**

One of the challenges that come with learning about non-parametric tests is that a particular test often has more than one name. For example:

1) **Wilcoxon signed-rank test**: This also goes by the name of the Wilcoxon one-sample test, the Wilcoxon matched-pairs test, or the Wilcoxon paired-sample test.

2) **Mann-Whitney *U*-test**: This also goes by the name of the Wilcoxon two-sample test, the Mann–Whitney–Wilcoxon, or the Wilcoxon rank-sum test.

Keep this in mind when reading about non-parametric tests. It often feels like there are many different tests out there, when in fact, they're all the same test under a different guise!

## 28.5 Wilcoxon signed-rank test

The most widely used non-parametric equivalent to the one-sample *t*-test is the Wilcoxon signed-rank test. The test can be used for any situation requiring a test to compare the median of a sample against a single value. However, it is almost always used to analyse data collected using a paired-sample design, so we'll focus on that particular application of the test. The Wilcoxon signed-rank test makes less stringent assumptions than the *t*-test, but it does make some assumptions:

- The variable being tested is measured on ordinal, interval, or ratio scales
- The (population) distribution of the variable is approximately symmetric[1].

The first assumption is simple enough—the variable can be anything but nominal. The second assumption essentially means that it doesn't matter what form the distribution of the variable looks like as long as it is symmetric—it would not be satisfied if the distribution were strongly right- or left-skewed.

As with the *t*-test, when applied to paired-sample data, the Wilcoxon signed-rank test starts by finding the differences between all the pairs and then tests whether these differences are significantly different from zero. The distribution under consideration is the *distribution of differences* between the pairs. Remember, this distribution will often be approximately normal even when the connected samples are not drawn from a normal distribution. This means that even if the samples have odd distributions, we may still find that we can use a paired-sample *t*-test if differences have a perfectly acceptable distribution. However, if the distribution of differences is not normal, then the Wilcoxon signed-rank test provides an alternative.

The following is an example of a situation where the Wilcoxon signed-rank test would be appropriate.

[1] There is an alternative to the Wilcoxon test which doesn't make an assumption about the distribution being symmetrical—called the sign test. This can be done in R, but there isn't a dedicated function for the sign test, and in any case, it is not very powerful with smaller sample sizes.

### 28.5.1 Leaf damage, plant defences and feeding by winter moth larvae

It has been hypothesised that plants respond to physical damage to their leaves from herbivores such as lepidopteran larvae by increasing the production of defensive compounds such as phenolics. In an experiment to test this effect, birch saplings were subjected to artificial leaf damage (hole punching) on half the leaves on selected branches. After 24 h, undamaged leaves from both the branches with hole-punched leaves and others distant from the damage site were collected and used in a choice test experiment with winter moth larvae. Twenty trials were carried out with a single caterpillar in each trial, offered one of each leaf type (i.e. one leaf from close to the damage site and one from an undamaged area). The percentage of each leaf consumed was estimated (to the nearest 5%) after 24h.

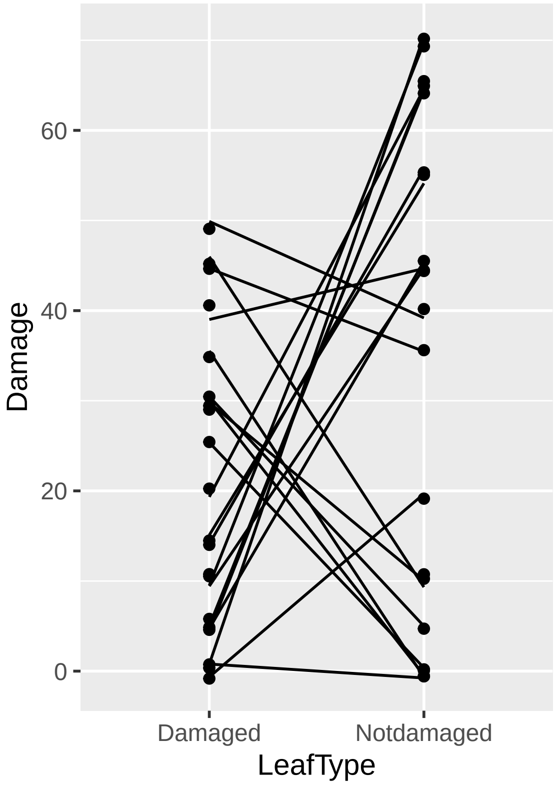
The data for this example live in the 'LEAF_DAMAGE.CSV' file. The code below assumes those data have been read into a tibble called `leaves`. Set that up if you plan to work along.

There are three variables: `ID` contains a unique identifier for each pair of leaves in a trial (1-20), `Leaf` identifies the type of leaf ('Damaged' vs. 'Undamaged'), and `Damage` contains the percent damage score:

```
## Rows: 40
## Columns: 3
## $ ID       <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18~
## $ LeafType <chr> "Notdamaged", "Notdamaged", "Notdamaged", "Notdamaged", "Notd~
## $ Damage   <dbl> 65, 0, 10, 45, 55, 45, 0, 10, 5, 70, 55, 40, 0, 70, 65, 35, 6~
```

As always, it's a good idea to visualise the data:

```
jitter <- position_jitter(width = 0, height = 1)
ggplot(leaves, aes(x = LeafType, y = Damage, group = ID)) +
  geom_point(position = jitter) +
  geom_line(position = jitter)
```

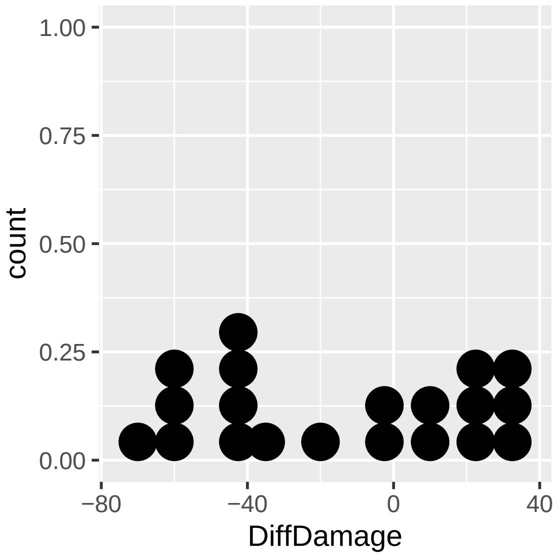


We used a new trick here: we 'jittered' the vertical position of the `Damage` values to deal with the overplotting (using the `position_jitter` function). This adds a bit of random noise to the y-axis position so that those with the same values end up being plotted in different places. This plot highlights that:

1. there is a lot of variation in the damage score—some larvae just eat more than others, and
2. there is a tendency for larvae to prefer the undamaged leaves, though the pattern is not overwhelming.

We need a statistical test. The data are paired, so a paired sample *t*-test or a Wilcoxon signed-rank test are our two options. We can visualise the damage distribution of *within larvae differences* to assess which test might be best:

```
# step 1 -- calculate the differences
leaves_diff <-
  leaves %>%
  group_by(ID) %>%
  summarise(DiffDamage = diff(Damage))
# step 2 -- make the plot
ggplot(leaves_diff, aes(x = DiffDamage)) +
  geom_dotplot(binwidth = 10)
```



There are large differences in both directions, and the distribution does not seem to be normal. The problem here is that caterpillars often spend time feeding on the first leaf they encounter, which is a more or less random choice; some may change leaves quickly in response to the quality of the leaf, others may feed for some time before leaving the leaf. We might still expect those on better-defended leaves to feed less, as the leaves should be less palatable, but they may not make the behavioural decision to switch leaf during the short period of the experiment (the experiment cannot go on for longer because the leaves change chemically the longer they are detached from the plant).

The distribution may not be normal, but it is roughly symmetrical. The symmetry assumption of the Wilcoxon signed-rank test seems to be satisfied, so we can use this test to examine whether there is any difference in caterpillar feeding on leaves from damaged areas and undamaged areas. We carry out the test using the `wilcox.test` function in R[2]. This works like the `t.test` function for a paired-sample design:

```
wilcox.test(Damage ~ LeafType, paired = TRUE, data = leaves)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

[2] There are two different ways to use the `wilcox.test` function for a paired design. These are analogous to the two methods used to carry out a paired sample *t*-test: the first uses the raw data and makes use of the R's formula system. The second supplies the function with a vector of differences

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with zeroes
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  Damage by LeafType
## V = 50.5, p-value = 0.07617
## alternative hypothesis: true location shift is not equal to 0
```

We must remember to set `paired = TRUE`. If we forget to do this, R will carry out the two-sample version of the test (the Mann-Whitney *U*-test discussed next). R produces a couple of warnings here, but we can usually ignore these—we can't do much about them even if we wanted to.

The elements of the output should be fairly easy to understand by now. At the top, we see a summary of the test and the data:

```
##        Wilcoxon rank sum test with continuity correction
## data:  Damage by LeafType
```

Next, we see the results: a test statistic and *p*-value. Although the test statistic is given as 'V' in the output, it is often quoted as 'W' when writing it in a report (we'll adopt this convention). This seems a bit odd, but it stems from the fact that there is some variation in the literature about what we report and how to report non-parametric tests.

The *p*-value is close to $p = 0.05$, but it doesn't quite fall below the threshold. It looks like there is no significant difference in feeding rates on the two types of leaf. We might report this as:

> When given a choice, winter moth larvae did not consume larger amounts of the leaves collected from areas of the tree that are undamaged than those from damaged areas (Wilcoxon matched-pairs test: W = 139.5, n = 20, p = 0.076).

## 28.6 The Mann-Whitney *U*-test

The Mann-Whitney *U*-test is the non-parametric equivalent to the independent two-sample *t*-test. The test can be used for any situation requiring a test to compare the median of two samples. The assumptions of the Mann-Whitney *U*-test are:

- The variable being tested is measured on ordinal, interval, or ratio scales
- The observations from both groups are independent of one another.

The first two assumptions are straightforward—data can be anything but nominal, and as with a paired-sample *t*-test, there must not be any dependence between the observations. Though not strictly necessary for the Mann-Whitney *U*-test to be valid, we usually add a third assumption:

- The distribution of the variable in each group is similar (apart from the fact that they have a different central tendency)

This assumption essentially means that it doesn't matter what the distributions of the two samples are like as long as they are roughly similar. For example, it would not be satisfied if we plan to compare data from a strongly right-skewed distribution with data from a strongly left-skewed distribution.

When all three of the above assumptions are satisfied, the Mann-Whitney *U*-test is used to look for differences between the central tendency of two distributions. A two-sample *t*-test evaluates the statistical significance of differences between two means. The null hypothesis of the Mann-Whitney *U*-test is that the two distributions have the same median. A significant p-value, therefore, indicates that the medians are likely to be different (not the means!).

### 28.6.1 Ant foraging

Let's use the Red wood ants example from the chapter on transformations to see how to carry out the Mann-Whitney *U*-test. Recall that this study measured the total biomass of prey being transported—the rate of food collection per ant per half hour—on three different tree species (rowan, sycamore and oak). Since the Mann-Whitney *U*-test only compares two samples, we will only use on the oak and sycamore cases to illustrate the test[3].

The data for this example live in the 'ANTS1.CSV' file. The code below assumes they have been read into a tibble called `ants`. We also assume the rowan cases have already been removed, using `filter`, for example:

Set all that up if you plan to work along.

The `Tree` variable contains the identities, and the `Food` variable contains the food collection rates:

```
glimpse(ants)
```

```
## Rows: 53
## Columns: 2
## $ Food <dbl> 20.1, 47.4, 85.6, 17.1, 5.7, 7.8, 28.8, 18.8, 12.1, 62.0, 4.9, 57~
## $ Tree <chr> "Oak", "Oak", "Oak", "Oak", "Oak", "Oak", "Oak", "Oak", "Oak", "O~
```

[3] Of course, if we really were interested in the difference between sycamore, oak *and* rowan we should use a different test!

Carrying out a Mann-Whitney *U*-test in R is simple enough. However, somewhat confusingly, we have to use the `wilcox.test` function again. This is because, as we noted above, the Mann-Whitney *U*-test is also known as a **two-sample Wilcoxon test**.

The `wilcox.test` function for an independent two-sample design works just like the corresponding analysis with the `t.test` function. The first argument should be a formula, and the second should be the data frame containing the relevant variables:

```
wilcox.test(Food ~ Tree, data = ants)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Food by Tree
## W = 453.5, p-value = 0.06955
## alternative hypothesis: true location shift is not equal to 0
```

The formula (`Food ~ Tree`) works in the same way as other statistical functions in R: the variable containing the values we wish to compare (`Food`) is on the left-hand side of the `~` and the variable containing the group identities (`Tree`) belongs to the right of it.

The output from a Mann-Whitney *U*-test is similar from that produced by the `t.test` function. At the top we see a summary of the test and the data:

```
##         Wilcoxon rank sum test with continuity correction
## data:  Food by Tree
```

This version of the Wilcoxon test is the same thing as a Mann-Whitney *U*-test, so there is nothing to worry about here. After this line we see a test statistic ('W'ilcoxon) and the associated *p*-value. Since *p*=0.07, we conclude the rates of removal of prey biomass were not significantly different between ants foraging in oak and in sycamore. In a report, the conclusion from the test could be summarised:

> The rates of removal of prey biomass were not significantly different between ants foraging in oak and in sycamore (Mann-Whitney *U*-test: U=453.5, $n_1$=26, $n_2$=27, *p*=0.07).

Notice how the statistics were reported. Because we are describing the test as a Mann-Whitney *U*-test, it is conventional to quote the test statistic as 'U', rather than 'W'. If we had decided to present the test as a two-sample Wilcoxon test, the test could be summarised:

> The rates of removal of prey biomass were not significantly different

> between ants foraging in oak and in sycamore (two-sample Wilcoxon test: $W$=453.5, $\mathrm{n}_1$=26, $\mathrm{n}_2$=27, $p$=0.07).

…remembering to change the name ('two-sample Wilcoxon test') and label ('W') used to describe the test statistic. In any case, we also have to provide the sample sizes associated with each tree group so that a reader can judge how powerful the test was.

## 28.7 The Kruskal-Wallis test

The Kruskal-Wallis is the non-parametric equivalent to one-way ANOVA. The Kruskal-Wallis test allows us to test for differences among more than two samples. Like the other rank-based tests we have encountered, it has some assumptions, but these are less restrictive than those for ANOVA. The assumptions are essentially the same as those for the Mann-Whitney $U$-test:

- The variable being tested is measured on an ordinal, interval, or ratio scale.
- The observations from both groups are independent of one another.
- The distribution of the variable in each group is similar (apart than the fact that they have a different central tendency)

The third assumption is important, particularly with respect to the skewness of the distributions. The test is at least reasonably robust to differences in the dispersion, but the Kruskal-Wallis test should not be used if the skewness of the variable is different among groups is very different. The reason for this is—just as with Mann-Whitney $U$-test—that a significant result is hard to interpret.

When all three of the above assumptions are satisfied, the Kruskal-Wallis is used to look for differences in the central tendency of two or more groups. A one-way ANOVA evaluates the statistical significance of differences between means of these groups. The null hypothesis of the Kruskal-Wallis test (if all three of the above assumptions) is that the groups have the same median. A significant $p$-value, therefore, indicates that the medians are likely to be different.

### 28.7.1 Learning in cuttlefish

In a study of cuttlefish learning, an experiment was conducted to determine how the length of exposure to a situation influenced the learning process. Cuttlefish feed on prawns. If they are presented with prawns in a glass tube, they strike at them but fail to capture the prey. Not surprisingly, after a period of this fruitless activity, they give up striking at the prawns.

In the experiment, 50 cuttlefish were divided at random into 5 groups of 10 and cuttlefish from each group were presented with prawns in glass tubes for different lengths of time: 2 min., 4 min., 8 min., 15 min., and 30 min. for the 5 groups respectively. After 24 hours, the same cuttlefish were presented with prawns again, and the number of strikes they made (over a fixed period) were recorded.

> The data for this example live in the 'CUTTLEFISH.CSV' file. The code below assumes those data have been read into a tibble called `cuttlefish`. Set that up if you plan to work along.

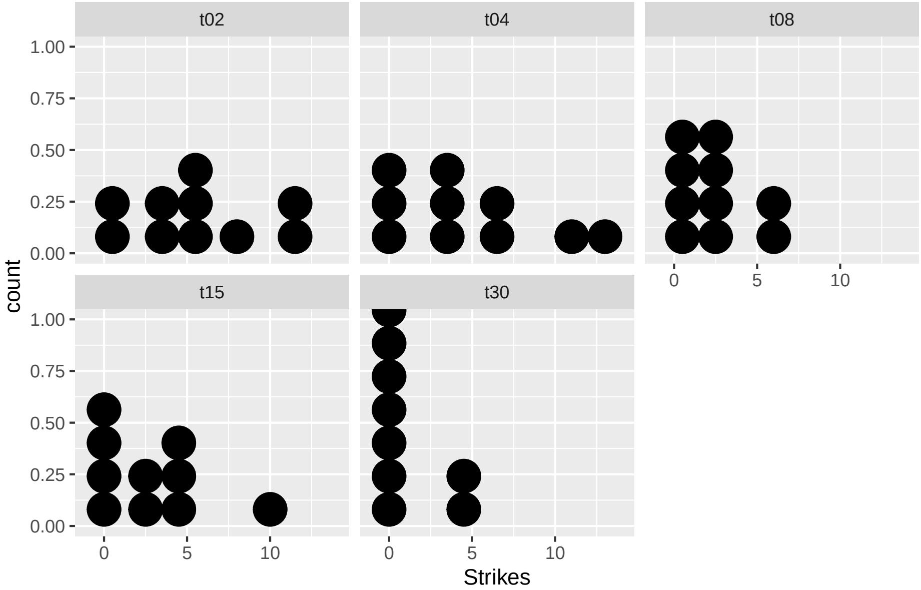
There are two variables: `Strikes` contains the number of strikes recorded and `Time` identifies the groups (period of previous exposure):

```
glimpse(cuttlefish)
```

```
## Rows: 50
## Columns: 2
## $ Strikes <dbl> 5, 12, 11, 4, 1, 6, 8, 3, 0, 5, 0, 13, 0, 3, 0, 7, 3, 4, 6, 11~
## $ Time    <chr> "t02", "t02", "t02", "t02", "t02", "t02", "t02", "t02", "t02",~
```

We need a visualisation to explore the data. Dot plots for each treatment work well here:

```
ggplot(cuttlefish, aes(x = Strikes)) +
  geom_dotplot(binwidth = 2) + facet_wrap(~ Time)
```



The data are very variable. The combination of the apparent skew and the fact that the data are generally small whole numbers with several equal values in each sample means that we may not be very successful in using a transformation to beat the data into shape. Let's try the Kruskal-Wallis test instead.

Predictably, the R function to carry out a Kruskal-Wallis test is called `kruskal.test`, and it is used in exactly the same way as every other statistical modelling function we have looked at:

```
kruskal.test(Strikes ~ Time, data = cuttlefish)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Strikes by Time
## Kruskal-Wallis chi-squared = 11.042, df = 4, p-value = 0.0261
```

And… one more time… the elements of the output should be easy to work out. These are statements of the test used and the data, followed by the results: a test statistic (another type of $\chi^2$ statistic), a degrees of freedom, and the all-important $p$-value.

We report all of these when writing up results of a Kruskal-Wallis test. However, there is some variation about how people report a Kruskal-Wallis test—some report the test statistic as a $\chi^2$, while others refer to it as an 'H' statistic. We will follow the common convention of reporting it as an 'H' value.

The test (as with ANOVA) tells us that there is at least one difference among the groups, but it doesn't tell where the difference or differences are. The output does not give the medians so we cannot judge how the samples are ordered. We can use `dplyr` to calculate the group-specific medians, though:

```
cuttlefish %>%
  group_by(Time) %>%
  summarise(Median = median(Strikes))
```

```
## # A tibble: 5 x 2
##   Time  Median
##   <chr>  <dbl>
## 1 t02      5
## 2 t04      3.5
## 3 t08      2.5
## 4 t15      2.5
## 5 t30      0
```

In this case, longer periods of exposure to the protected prawn do seem to result in fewer strikes in the later trial. Once we understand what is driving the significant result we're in a position to write a summary:

> The frequency with which the cuttlefish attacked the prawns was significantly affected by the length of time for which they had been exposed to protected prawns 24h earlier (Kruskal-Wallis test: H=11.04, d.f.=4, p<0.05), with longer prior exposure resulting in lower attack rates.

If it was important to know exactly which treatments were significantly different, then a multiple comparison test would be useful. There are no non-parametric multiple comparison tests available in base R, but they are implemented in the package called **nparcomp**.

## 28.8 Spearman's rank correlation

Spearman's rank correlation ($\rho$) tests for an association between two numeric variables. It is equivalent to Pearson's correlation. The advantages of using Spearman's rank correlation are: 1) the two variables do not need to be normally distributed, and 2) ordinal data can be used. This means we can use Spearman's rank correlation with data having skewed (or other odd) distributions, or with data originally collected on a rank/ordinal scale.

The key assumptions of Spearman's rank correlation are:

- Both variables are measured on ordinal, interval or ratio scales.
- There is a monotonic relationship between the two variables.

A monotonic relationship occurs when, in general, the variables increase in value together, or when the values of one variable increase, the other variable tends to decrease. What this means in practice is that we should not use Spearman's rank correlation if a scatter plot of the data forms a clear 'hill' or 'valley' shape.

Spearman's rank correlation is somewhat less powerful (roughly 91% in some evaluations) than Pearson's method when the data are suitable for the latter. Otherwise, it may even be more powerful. We'll work through an example to learn about Spearman's correlation.

### 28.8.1 Grouse lekking

Some bird species form 'leks'—gatherings of birds, with males each defending a small area of ground, displaying, and each mating with such females as he is successful in attracting. In general, in leks, a few birds secure many matings, and most birds secure relatively few. In a study of lekking in black grouse, a biologist is interested in whether birds that secure many matings in one season also do better the following year. Using a population with many colour-ringed birds, he gathers data for a reasonable number of males from two leks in successive years.

The data for this example live in the 'GROUSE.CSV' file. The code below assumes those data have been read into a tibble called `grouse`. Set that up if you plan to work along.

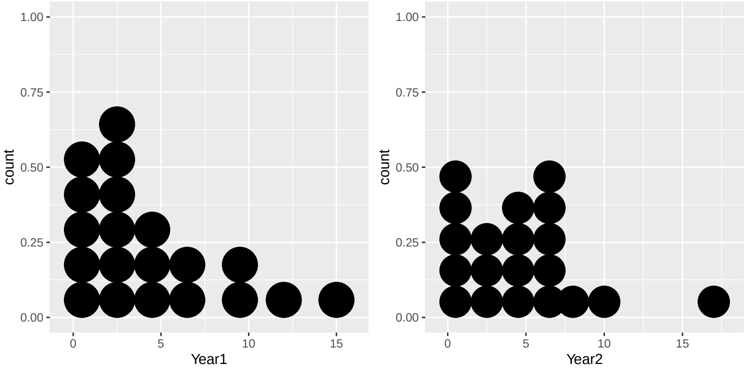
Each row of the data is the number of matings for a male in the two successive leks: `Year1` (year 1) and `Year2` (year 2):

```
glimpse(grouse)
```

```
## Rows: 20
## Columns: 2
## $ Year1 <dbl> 3, 15, 4, 3, 4, 9, 1, 0, 2, 7, 3, 2, 6, 0, 1, 12, 2, 1, 5, 10
## $ Year2 <dbl> 6, 8, 0, 0, 2, 10, 5, 4, 1, 4, 3, 7, 6, 2, 5, 17, 0, 0, 6, 7
```

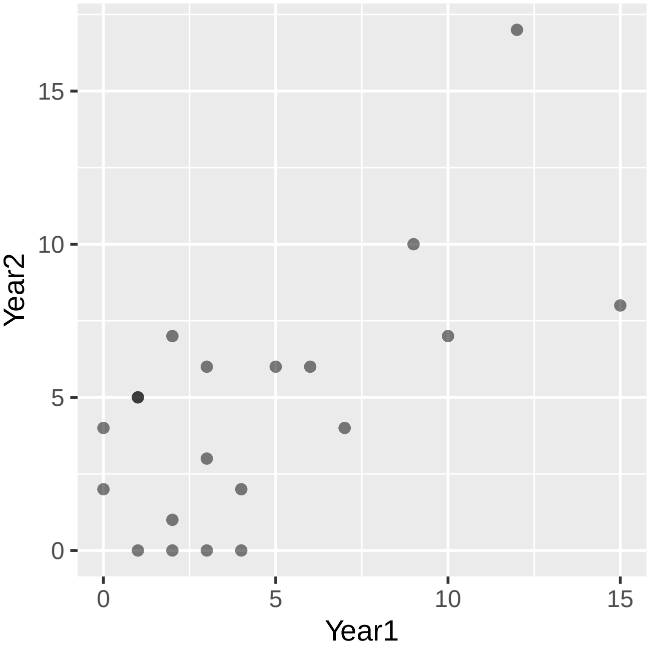
The first thing we might do is summarise the distribution of each number of matings variable:

```
ggplot(grouse, aes(x = Year1)) + geom_dotplot(binwidth = 2)
ggplot(grouse, aes(x = Year2)) + geom_dotplot(binwidth = 2)
```



Notice that the data are integer-valued because they are counts. These distributions seems to tie in with the biological observation that the distribution of matings is right-skewed: in both years there are only a few males that have high mating success, with most males securing only a handful of matings.

Now we need to visualise the association between the number of matings in successive years:



The data are integers, which means there is a risk of over-plotting (points will lie on top of one another). We made the points semi-transparent `alpha = 0.5` to pick this up where it occurs. It seems clear that mating success is positively associated, but we should confirm this with a statistical test. We'll base this on Spearman's correlation.

**How do we know to use a correlation analysis with these data?** Although there seems to be an association between the counts, it is not obvious that success in one year 'causes' success in another year and neither variable is controlled by the investigator. We're also not interested in using the success measure in year 1 to predict success in year 2. This indicates that correlation analysis is the appropriate method to evaluate the significance of the association.

**Why are we using Spearman's correlation?** The relationship appears roughly linear, so in that regard, Pearson's correlation might be appropriate. However, the distribution of each count variable is right-skewed, which means the normality assumption is probably suspect in this instance. We play it safe and use Spearman's correlation.

Carrying out a correlation analysis using Spearman's rank correlation in R is simple. Again, we use the `cor.test` function to do this:

```
cor.test(~ Year1 + Year2, method = "spearman", data = grouse)
```

```
## Warning in cor.test.default(x = mf[[1L]], y = mf[[2L]], ...): Cannot compute
## exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  Year1 and Year2
## S = 592.12, p-value = 0.01112
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.5547952
```

The only other thing we had to change, compared to the Pearson's correlation example, was to set `method = "spearman"` to specify the use of Spearman's rank correlation. Notice the warning message: 'Cannot compute exact p-value with ties'. This is generally not something we need to worry about for this particular test.

The output is very similar to that produced by `cor.test` when using Pearson's correlation. Once again, the `S = 592.12, p-value = 0.01112` line is the one we care about. The main difference is that instead of a *t*-statistic, we end up working with a different kind of test statistic ('*S*'). We aren't going to explain where this comes from because it's quite technical. Next we see the *p*-value. This is generated under the null hypothesis of zero correlation ($\rho = 0$). Since $p$ $< 0.05$, we conclude that there is a statistically significant correlation between mating success in successive years. Wait, where are the degrees of freedom? Simple—there aren't any for a Spearman's correlation test.

What is the correlation between mating success? That's given at the bottom of the test output again: +0.55. This says that there is a moderate, positive association between mating success in successive years, which is what we expect from the scatter plot.

When using the Spearman method, it is fine to report just the value of the correlation coefficient, the sample size, and the *p*-value (there is no need to report the test statistic). Here's how to report the results of this analysis:

> There is a positive association between the number of matings achieved by a particular male in one lek and the number the same male achieves in a subsequent lek (Spearman's rho=0.55, n=20, p < 0.05).

## 28.9 Parting words

Inappropriately distributed data can result in incorrectly high or low *p*-values. We should choose a statistical model based on how well our data align with the model's assumptions. On the principle that biological data rarely, if ever, fully comply with the assumptions of parametric tests, it is sometimes advocated that non-parametric tests should always be used. This is not very good advice. Use both approaches as appropriate and be aware of the strengths and weaknesses of each. There is more to statistics than just calculating *p*-values, and where possible, we prefer readily-interpretable models to more 'black box' approaches.

# Appendix A

# Choosing models and tests

## A.1 Introduction

One of the more difficult skills in data analysis is deciding which statistical models and tests to use in a particular situation. This book has introduced a range of different approaches and has demonstrated various biological questions that can be addressed with these tools. Here we draw together the statistical analyses we've encountered and explain how to match them to particular kinds of questions or hypotheses. However, there are a few things to consider before diving into an analysis.

### A.1.1 Do we need to carry out a statistical analysis?

This may seem like an odd question to ask, having just spent a considerable amount of time learning statistics. But it is an important one. There are many situations in which we don't need, or can't use, statistical tools. Here are two common ones:

1. *There are no statistical procedures that will allow us to analyse our data correctly*[1]. This happens sometimes. Even with careful planning, things don't always work out as anticipated, and we end up with data that cannot be analysed with a technique we know about. If the data can't be analysed sensibly, there is no point in doing any old analysis because we feel we have to produce a $p$-value. Instead, it's time to seek some advice.

2. *We could quite correctly apply a statistical test to your data, but it would be entirely superfluous.* We don't always need statistics to tell us what is going on. An effect may be exceptionally strong and clear, or it may be

[1] Alternatively, an appropriate technique exists, but we don't know about it!

that the importance of the result is not something captured by applying a particular statistical model or test[2]. This caveat is particularly relevant to exploratory studies, where the goal is to use the data to generate new hypotheses rather than test *a priori* hypotheses.

Also in the category of superfluous statistics are situations where we are using statistics in a technically correct way, but we're evaluating effects that are not interesting or relevant to the question at hand. This often arises from a misplaced worry that unless we have lots of statistics in our study, it somehow isn't 'scientific', so we apply them to everything in the hope that a reader will be impressed. Resist the temptation! This strategy will have the opposite effect—a knowledgeable reader will assume you don't know what you're doing if they see a load of pointless analyses.

## A.2 Getting started

If there is a need for statistical analysis, then the first thing to do is read the data into R and then… resist the temptation to start generating *p*-values! There are several things to run through before leaping into a statistical analysis:

1. Be sure to carefully review the data after importing it into R (functions like `View` and `glimpse` is good for this). There are a number of things to look out for:

    - *Check the data set 'tidy'* — Each variable should be one column and each row should correspond to one observation. The majority of statistical modelling tools in R expect the data to be organised in this format, as do packages like **dplyr** and **ggplot2**. A data set will need to be reorganised if it isn't already tidy. We can do this by hand, but it is almost always quicker to do it with R (the **tidyr** package is a good option).
    - *Understand how R has encoded the variables* — Examine the data using functions like `glimpse`. Pay close attention to the variable types—are they numeric, a character, or a factor? If a variable is not appropriate for the planned analysis, make any necessary changes. For example, if we plan to treat a variable as a factor but it has been read in as a number, we'd need to convert it to a factor before preceding.
    - *Check whether or not there are any missing values* — These appear as `NA` in R. If they are present, were they expected? If not, check the original data source to determine what has happened. If we're absolutely certain the missing values represent an error in the way the data were coded then it might be sensible to fix the source data.

[2] As the old joke goes: *What does a statistician call it when ten mice have their heads cut off and one survives? Not significant.*

However, if there is any doubt about how they arose, it is better to leave the source data alone and deal with the miscoded `NA`s in the R script.

2. Spend some time thinking about the variables in the data set. Which ones are relevant to the question in hand? If appropriate, decide which variable is the dependent variable (the 'y' variable) and which variable(s) is (are) the independent variable(s)? What kind of variables are we dealing with—ratio or interval scale numeric variables, ordinal or nominal categorical variables? It is much easier to determine which analysis options are available once these details are straightened out.

3. Make at least one figure to visualise the data. We have done this throughout this book. This wasn't to fill the time—it is a crucial step in any data analysis exercise. Informative figures allow us to spot potential problems with the data, and they give us a way to evaluate our question before diving into an analysis. If we can't see an appropriate way to visualise the data, we probably aren't ready to start doing the statistics!

Steps 1-3 are all critical components of 'real world' data analysis. It may be tempting to skip these and just get on with the statistics, especially when pressed for time. Don't do this! The 'skip to the stats' strategy will waste time because we fail to spot potential problems and carry out an inappropriate analysis.

## A.3 A key to choosing statistical models and tests

The choice of appropriate statistical model/test is affected by two things:

1. The kind of question we are asking.

2. The nature of data we have:

    - what type of variables: ratio, interval, ordinal or nominal?
    - are the assumptions of a particular model or test satisfied by the data?

The schematic key (below) provides an overview of the statistical models and tests we've covered in this book, structured in the form of a key. The different choices in the key are determined by a combination of the type of question being asked, and the nature of the data under consideration.

The notes that follow the key expand on some of the issues it summarises and explain some of the trickier elements of deciding what to do. The key is quite large, so it is hard to read easily on a web browser (it also doesn't render very well in Firefox for some reason). Either download a PDF copy of the key or open it on its own in a new tab and zoom in.

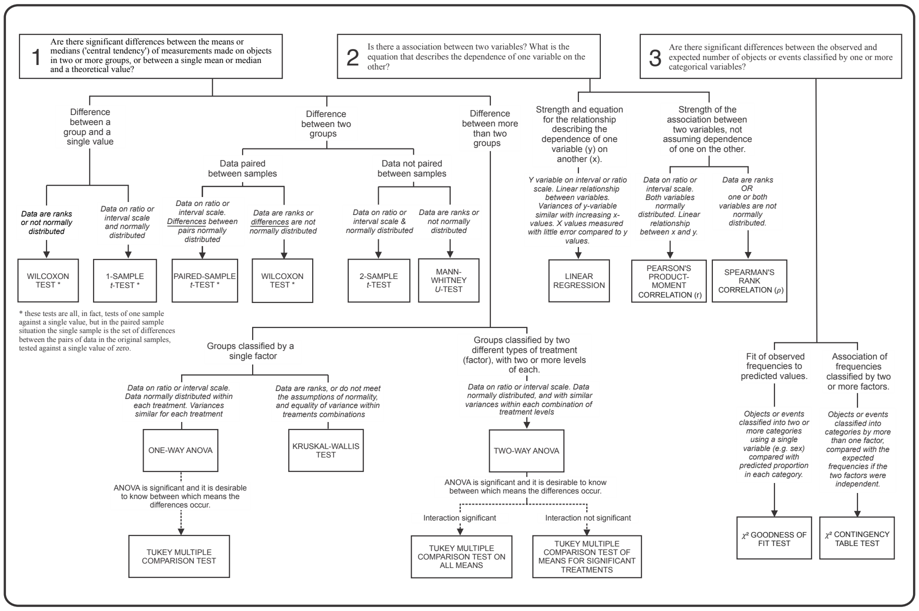

1 Are there significant differences between the means or medians ('central tendency') of measurements made on objects in two or more groups, or between a single mean or median and a theoretical value?
2 Is there a association between two variables? What is the equation that describes the dependence of one variable on the other?
3 Are there significant differences between the observed and expected number of objects or events classified by one or more categorical variables?
Difference between a group and a single value
Difference between two groups
Difference between more than two groups
Data paired between samples
Data not paired between samples
Data are ranks or not normally distributed
Data on ratio or interval scale and normally distributed
Data on ratio or interval scale. Differences between pairs normally distributed
Data are ranks or differences are not normally distributed
Data on ratio or interval scale & normally distributed
Data are ranks or not normally distributed
WILCOXON TEST *
1-SAMPLE t-TEST *
PAIRED-SAMPLE t-TEST *
WILCOXON TEST *
2-SAMPLE t-TEST
MANN-WHITNEY U-TEST
* these tests are all, in fact, tests of one sample against a single value, but in the paired sample situation the single sample is the set of differences between the pairs of data in the original samples, tested against a single value of zero.
Strength and equation for the relationship describing the dependence of one variable (y) on another (x).
Strength of the association between two variables, not assuming dependence of one on the other.
Y variable on interval or ratio scale. Linear relationship between variables. Variances of y-variable similar with increasing x-values. X values measured with little error compared to y values.
Data on ratio or interval scale. Both variables normally distributed. Linear relationship between x and y.
Data are ranks OR one or both variables are not normally distributed.
LINEAR REGRESSION
PEARSON'S PRODUCT-MOMENT CORRELATION (r)
SPEARMAN'S RANK CORRELATION (ρ)
Groups classified by a single factor
Groups classified by two different types of treatment (factor), with two or more levels of each.
Data on ratio or interval scale. Data normally distributed within each treatment. Variances similar for each treatment
Data are ranks, or do not meet the assumptions of normality, and equality of variance within treaments combinations
Data on ratio or interval scale. Data normally distributed, and with similar variances within each combination of treatment levels
ONE-WAY ANOVA
KRUSKAL-WALLIS TEST
TWO-WAY ANOVA
ANOVA is significant and it is desirable to know between which means the differences occur.
ANOVA is significant and it is desirable to know between which means the differences occur.
Interaction significant
Interaction not significant
TUKEY MULTIPLE COMPARISON TEST
TUKEY MULTIPLE COMPARISON TEST ON ALL MEANS
TUKEY MULTIPLE COMPARISON TEST OF MEANS FOR SIGNIFICANT TREATMENTS
Fit of observed frequencies to predicted values.
Association of frequencies classified by two or more factors.
Objects or events classified into two or more categories using a single variable (e.g. sex) compared with predicted proportion in each category.
Objects or events classified into categories by more than one factor, compared with the expected frequencies if the two factors were independent.
χ² GOODNESS OF FIT TEST
χ² CONTINGENCY TABLE TEST

Figure A.1: A key for choosing statistical models and tests.

## A.4 Four main types of question

**Question 1**: Are there significant differences between the means or medians ('central tendency') of a variable in two or more groups, or between a single mean or median and a theoretical value?

This first question is relevant when we have measurements of one variable (e.g. plant height) on each experimental unit (e.g. individual plants), and experimental units are in different groups. If there is more than one group, one or more variables (factors) are used to encode group membership (given by the factor levels). Remember that these grouping factors are distinct from the variable being analysed—they essentially describe the study design. This type of question includes anything where a comparison is being made between the variable in one group and:

- a single theoretical value,
- another group whose values are independent of those in the first group (independent design),
- more than one other group, or
- a second group which has values that form logical pairs with those in the first group (paired design)[3].

[3] In a paired design the two groups aren't really separate, independent entities, in the sense that pairs of measurements have been taken from the same physical 'thing' (site, animal, tree,

The measurement scale of the variable in these situations may be ratio, interval, or ordinal. The only variable for which the statistical tools described here would not be suitable are categorical variables.

**Question 2**: Is there an association between two variables? What is the equation that describes the dependence of one variable on the other?

Question 1 is concerned with comparing distinct groups where we have measurements of one variable on each experimental unit. In contrast, Question 2 occurs where we've taken two different measurements for each experimental unit (e.g., plant size *and* number of seeds produced). Here we are interested in asking whether, and perhaps how, those two variables are associated with each other.

Here, again, the measurement scale of the variable in these situations may be ratio, interval, or ordinal.

**Question 3**: Are there significant differences between the observed and expected number of objects or events classified by one or more categorical variables?

Question 3 is the only question that is focused on the analysis of categorical variables. Here we have situations where the 'measurements' on objects can be things like colour, species, sex, etc. In these situations, we analyse the data by counting how many of the objects fall into each category or combination of categories. The frequency of counts across the categories can then be tested against some predicted pattern.

## A.5 Question 1 — Comparison of group means or medians

### A.5.1 Question 1 How many groups?

Within the set of situations covered by Question 1, there are some further subdivisions: we need to decide whether: we have one group only (and a theoretical or expected value to compare it against), we have two groups, or we have more than two groups. Usually, this should be a fairly straightforward decision.

### A.5.2 [Question 1] Single group

When we have a **single group** (**1a**) the only thing that remains to be done is check the type and distribution of the variable. If the data are approximately normally distributed, then the obvious test is a **one-sample *t*-test**. If the data are not suitable for a *t*-test, even after transformation, then we could use a

etc).

**Wilcoxon test** (we studied this in terms of a paired design, but remember, a paired experimental design is ultimately reduced to a one-sample test).

### A.5.3 [Question 1] Two groups

If we have two groups, then there is a further choice: whether there is a logical pairing between the variable measured in the two groups or whether the data are independent. This sometimes causes problems, particularly where the pairing is not of the obvious sort. One useful rule-of-thumb is to ask whether the data could be 'paired' in more than one way. If there is any uncertainty about how the pairing should be done, that is probably an indication that it isn't a paired situation. The most common problem, however, is failing to recognise pairing when it exists.

When faced with paired design, the test involves first deriving a new variable from the differences among pairs and then using this variable in a one-sample test. Either a **one-sample *t*-test** or **Wilcoxon paired-sample test** is required, depending on whether the new variable (the differences) is approximately normally distributed or not.

If the data are independent, then a **two-sample *t*-test** or **Mann-Whitney -test** will be the best approach.

### A.5.4 [Question 1] More than two groups

The first decision here is about the structure of the data. This sometimes causes problems. There are various situations in which we may be interested in testing for differences between several means (or perhaps medians). These will often involve either a one-way comparison in which each observation can be classified as coming from one set of treatments (one factor), or a two-way comparison in which each value comes from a combination of two different sets of treatments (two factors). It is easy to mix these situations up if we're not paying attention.

One way to try and establish the structure of the data is to draw up a table. If the data fit neatly into the sort of table below, where there is one factor (e.g. factor 'A') that has two or more levels (e.g. level 1, 2, 3) then we have a *one-way design*. The treatments designated by the levels of the factor in this situation are typically related in some way (e.g. concentrations of pesticide, temperatures). The only question it makes sense to address with these data is whether there are differences among the means of the three treatments.

| **FACTOR A** | |
|---|---|
| **Treatment 1** | 1,4,6,2,9 |
| **Treatment 2** | 7,3,8,9,4 |
| **Treatment 3** | 5,3,7,6,4 |

**Two-way design?**

If there are two different types of treatment factor (factors ‘A’ and ‘B’), and within each factor, there are two or more levels, then we have a *two-way design*. The treatments designated by the levels of a particular factor are typically related in some way, but the set of treatments associated with each factor are not related to one another. We could not draw up a table like the one above and fit these data into it because each observation occurs simultaneously in one treatment level from each of the two factors:

| | **Treatment B1** | **Treatment B2** | **Treatment B3** |
|---|---|---|---|
| **Treatment A1** | 1,4,6 | 3,9,1 | 2,2,7 |
| **Treatment A2** | 7,3,8 | 2,3,6 | 9,3,4 |
| **Treatment A3** | 5,3,7 | 1,8,6 | 2,2,6 |

If we have data from such a two-way design we can still use ANOVA to compare means, but we can’t use one-way ANOVA to do this. We need to use something called... a **two-way ANOVA**. This book hasn’t considered two-way ANOVA, but it is such a common design we mention it here to help you spot it in your work.

Having established whether we have a one-way design, we need to determine whether the data are likely to satisfy the assumptions of the model we’re presented with. We could start this evaluation by plotting the raw data (e.g. using a series dot plots). However, it’s much better to fit the ANOVA model and use regression diagnostics to check whether the assumptions are satisfied. Do that!

In the case of a one-way design, we have the option of a non-parametric **Kruskal-Wallis test** when the data are not suitable for ANOVA. Otherwise, we need to find a suitable transformation and then use a **one-way ANOVA**. Remember, before turning to a Kruskal-Wallis test, it is good to see if the data can be made suitable for ANOVA by transformation.

Finally, we should consider the option of multiple comparison tests. These should only be used if the global significance test from an ANOVA is significant.

## A.6 Question 2 – Associations between two variables?

Assuming it is an association we’re after—not a difference between groups (see box below)—then the main decision we need to make is whether to use a correlation or regression analysis.

### A.6.1 [Question 2] Testing $y$ as a function of $x$, or an association between $x$ and $y$?

The choice between regression and correlation techniques depends on the nature of the data *and* the purpose of the analysis. Suppose the purpose of the analysis is to describe (mathematically) the relationship that captures the dependence of one variable ($y$) on another ($x$). This scenario points to **simple regression** being the most appropriate technique. On the other hand, if we only want to know whether there is an association between two variables, then this would suggest **correlation**.

However, it is not just the goal of an analysis that matters, unfortunately! The two techniques also make different assumptions about the data. For example, regression assumes that the $x$-variable is measured with little error relative to the $y$-variable, but doesn't require the $x$-variable to be normally distributed. Pearson's correlation assumes that both $x$ and $y$ are normally distributed (or our sample is very large).

The final decision about which method to use may depend on trying to match up both the question and the nature of the data. It sometimes happens that we want to ask a question that requires a regression approach, but the data are not suitable. In this situation, it can be worth proceeding with regression, bearing in mind that the answer we get may be less accurate than it should be (though careful use of transformations may improve things).

Once we decide a correlation is appropriate, then the choice of parametric or non-parametric test should be based on the extent to which the data match the assumptions of the parametric test.

A final point here that can cause difficulties is the issue of the 'dependence' of one variable on another. In biological terms, we are often interested in the relationship between two variables, one of which we know is biologically dependent on the other. However, designating one variable the dependent variable ($y$) and the other the independent variable ($x$) does not imply that we think 'y depends on x', or that 'x causes y'.

An example will clarify this idea. Tooth wear in mammals is dependent on age. However, imagine that we have collected a number of samples of mammalian teeth from individuals of a species that have died for various reasons, and for which we also have an estimate of age at death. We may want to find an equation for the relationship between age ($y$) and tooth wear ($x$). Why? It could be useful to be able to use measurements of tooth wear to estimate the age of individuals in a new sample, for for which we do not have an estimate of age at death. We'd need an equation that describes the 'dependence' of age on tooth wear to do this. So here, the direction of dependence in the analysis is not the same as the causal dependence of the two variables.

The point is that the choice of analysis does not determine the direction of biological dependence—it is up to us to do the analysis in a way that makes sense for the purpose of the study.

⚠ **Question 1 or Question 2?**

Although it seems straightforward to choose between Question 1 and Question 2, it does sometimes cause a problem in the situation there are two groups, the data are paired, and the same variable has been measured in each 'group'. Because two sets of measurements on the same objects (say individual organisms) fit the structure a paired-sample *t*-test or a regression (or correlation), it is very important to identify clearly *the effect we want to test for*. A concrete example will help make this clearer…

The situation where confusion most easily arises is when the same variable has been measured in both groups. Imagine we've collected data on bone strength from males and females in twenty families, where the males and females are siblings—a brother and sister from each family. The pairing clearly makes sense because the siblings are genetically related and likely to have grown up in similar environments. Consider the following two situations…

- If our goal is to test whether males and females tend to have different bone strengths, then a paired-sample *t*-test makes sense: it compares males with females controlling for differences due to relatedness and environment.
- If our goal is to test whether bone strength runs in families, then the *t*-test is of no use. In this case, it makes sense to evaluate whether there is a correlation in the bone strength of sibling pairs (i.e. if one sibling has high bone strength, then does the other as well?).

So while the data can be analysed in either way, it is the question we're asking that is the critical thing to consider. Just because data *can* be analysed in a particular way doesn't mean that analysis will tell us what we want to know. One way to do tackle this sort of situation is to imagine what the result of each test using those data might look like and think about how one might interpret that result. Does that interpretation answer the right question?

## A.7 Question 3 — Frequencies of categorical data

This kind of question relates to categorical or qualitative data, (e.g. the number of male versus female offspring, the number of black versus red ladybirds, the number of plants of different species). The data are counts/frequencies of the number of objects or events belonging to each category. The principle of testing such data is that the observed frequencies in each category are compared with expected (predicted) frequencies in each category.

Deciding between goodness of fit tests and contingency tables is generally fairly straightforward once we've determined whether counts are classified by a single categorical variable, or more than one categorical variable. If there is more than one categorical variable, then it should be possible to classify each observation into one category of each categorical variable, and that allocation should be unique. Each observation should fit in only one combination of categories.

There is a further difference which may also affect our choice. In the case of a single factor, the question we ask is whether the numbers of objects in each category differ from some expected value. The expected values might be that there are equal numbers in each category, but could be something more complicated—it is entirely dependent on the question we're asking. In the case of the two-variable classification, the test addresses one specific question: is there an association between the two factors? The expected numbers are generated automatically based on what would be expected if the frequencies of objects in the different categories of one factor were unaffected by their classification by the other factor.

## A.8 Variables or categories?

One final issue that sometimes causes questions is when a variable is treated as categorical when it is really a continuous measure or when a continuous variable is made into categories. There is indeed some blurring of the boundaries here. Two situations are discussed below.

### A.8.1 ANOVA vs. regression

There are many situations in which data may be suitable for analysis by regression or one-way ANOVA, even though they are different kinds of models. For example, if a farmer wishes to determine the optimal amount of fertiliser to add to fields to achieve maximum crop yield, she might set up a trial with 5 control plots and 5 replicate plots for each of 4 levels of fertiliser treatment: 10, 20, 40 and 80 kg ha NPK (nitrogen, phosphorus and potassium fertiliser) and measure the crop yield in each replicate plot at the end of the growing season (kg ha year).

If we are simply interested in determining whether there is a significant difference in yields from different fertiliser treatments, and if so, which dose is best, then ANOVA (and multiple comparisons) is probably the best technique. On the other hand, we might be interested in working out the general relationship between fertiliser dose and yield, perhaps to be able to make predictions about the yield at other doses than those we have tested. We could use regression to determine whether there was a significant relationship between the two and describe it mathematically[4].

### A.8.2 Making categories out of continuous measures

Sometimes you will have data that are, at least in principle, continuous measurements (e.g. the abundance of an organism at different sites) but have been grouped into categories (e.g. abundance categories such as 1-100, 101-200, 201-300, etc. ). One question is whether these count as categories and whether, for example, you could look at the association between abundance categories and habitat using a contingency table test. The answer is that this would indeed be a legitimate procedure (though it may not be the most powerful).

It is good not to group numeric data into categories if we can avoid it because this throws away information. However, it isn't always possible to make a proper measurement, though we can assign observations to ordinal categories. For example, when observing flocks of birds, we might find that it's impossible to count them properly, but we can reliably place the numbers of birds in a flock into abundance categories (1-100, 101-200, etc. ). Many variables that we treat as ordinal categories could, in principle, be measured in a more continuous form: 'unpigmented', 'lightly pigmented', 'heavily pigmented'; 'yellow', 'orange', 'red'; 'large', 'small'… These are all convenient categories, but in one sense, they are fairly arbitrary. This doesn't mean that we can't construct an analysis using these categories. However, one thing to bear in mind is that when we divide a continuous variable into categories, decisions about where to draw the boundaries will affect the pattern of the results.

[4] One additional potential advantage of regression in this kind of situation is that it *might* result in more a powerful statistical test of fertiliser effects than ANOVA. This is because a regression model only 'uses up' two degrees of freedom—one for each of the intercept and slope—while ANOVA uses four (n-1). A regression makes stronger assumptions about the data though, because it assumes a linear relationship between crop yield and fertiliser.

# Appendix B

# Writing a scientific report

> *"The question is," said Alice, "whether you can make words mean so many different things."*
>
> "The question is," said Humpty Dumpty, "which is to be master – that's all."
>
> Lewis Carroll, *Through the Looking Glass* (1871)

## B.1 Introduction

Scientific information is communicated in various ways: through talks and seminars, through posters at meetings, but mainly through scientific papers. Papers published in books or journals provide the main route by which scientific findings are made available to others for evaluation and subsequent use. Over time the scientific paper has developed into a reasonably formal method of communication, with certain structures, styles and conventions. Information is presented in a standardised way, meaning particular bits of information can be extracted more easily by a knowledgeable reader.

In this chapter, we will examine the structure and conventions of a biological paper, using an example of a field study of the territorial behaviour of a damselfly. We aim to illustrate the typical form and content of a scientific paper. Of course, papers vary in their exact requirements, and no one example can cover all the possibilities. Read recent papers in a relevant subject area and analyse their styles and structures to see which work best.

The conventions discussed below are not rules and should be flexibly interpreted under the guiding principle that the aim is to present information as clearly, concisely and unambiguously as possible. Although taking the scientific paper as a model, the principles apply to other, less formal project write-ups and reports.

## B.2 The structure of a scientific report

The normal scientific report has a standard structure (parts in parentheses are optional):

1. Title
2. Abstract / Summary
3. Introduction
4. Methods
5. Results
6. Discussion
7. (Acknowledgements)
8. Literature cited
9. (Appendix)

### B.2.1 Title

Although not really a section of the paper, it is worth giving the title some thought. Aim for something that gives a fairly specific description of the topic of the paper, and possibly the essential result, but without being too long:

> Diurnal changes in the depth distribution of copepods in lakes with and without planktivorous fish: evidence of a predator avoidance mechanism?
>
> An experimental study of the effect of food supply on laying date in the coot.
>
> The distribution and altitudinal limits of bracken (*Pteridium aquilinum*) in the North York Moors National Park.
>
> Reverse transcription-PCR detection of LaCrosse virus in mosquitoes and comparison with enzyme immunoassay and virus isolation.

The important thing to note is that the titles contain a good deal of specific information—we have a pretty good idea of what the paper is about before we read it. Avoid vague titles such as…

> A study of damselfly behaviour

…when in fact, we have looked at is the mating and oviposition behaviour of damselflies of a particular species in relation to the current speed in different areas of the river, and what we want to say is:

> The influence of river flow rate on mating behaviour and oviposition
> in the damselfly *Calopteryx splendens*

Don't put irrelevant specific information in the title. It might be irrelevant to say that we carried out a study in a particular river—that detail is probably not important for the question we're asking. However, the reference to the North York Moors above is relevant because the study is of an area-specific problem (the study is primarily of use to people who want to know about bracken in that area).

### B.2.2 Abstract or Summary

The purpose of an abstract is to present a factual summary of the main purpose, results and conclusions of the report which is short and makes sense on its own. Often it is best (and some journals require it) to do this as 3-6 numbered points comprising some, or all, the following:

- The scope and purpose of the study
- Methods (not always necessary)
- Result 1
- Result 2…
- Conclusion

For example:

> - The territorial behaviour, mating frequency and oviposition of *Calopteryx splendens* (Charpentier) (Odonata: Calopterygidae) were studied in relation to the water flow rate in the territories (weed patches) of individual males.
> - Weed patches with faster flow rates appeared to be preferentially selected by males and more vigorously defended. Weed patches in slow or still water were often unoccupied. Experimental reduction of flow rate in individual patches caused males to desert previously defended territories.
> - Males had greater mating success on territories with higher flow rates, and more ovipositions were observed in these patches.
> - It is not known why weed patches with faster flows seem to be better quality sites for *Calopteryx* oviposition, but possible reasons include higher oxygen levels for developing eggs and better protection from egg parasitoids.

### B.2.3 Introduction

The introduction should:

- Set the background to the question, using the literature (Why is it interesting? Why is it important?).
- State the question, hypotheses and predictions. (What is it that we're actually investigating?)
- Briefly state what the study does (What is in this paper?)

Start with brief general statements to put the study into its broader context:

> Oviposition site selection by female insects can be a critical factor in offspring survival, and hence, fitness (Smith 1981). In some insects, notably many of the Odonata, males occupy or defend oviposition sites and mate with arriving females before allowing them to oviposit at that site. Males in such systems benefit in two ways from defending high quality sites: mating with all females ovipositing at the site ensures their offspring will have higher survival, and by occupying high quality sites, they will have access to more females (Jones 1976).

Then move on to more specific detail about the type of system:

> In calopterygid damselflies females oviposit in the submerged stems of aquatic plants in streams and small rivers (Hines 1956, Norman 1968). Males defend patches of weed...

Then develop the question:

> It has been repeatedly observed that many weed patches are always occupied and are the subject of much territorial dispute amongst males, whilst others remain unoccupied or uncontested (Gateman and Nunn 1978, Speake 1982, Mollison 1987). This suggests substantial differences in patch quality, but the basis of this difference is not known. Since the larvae may disperse after hatching, the underwater environment of a weed patch seems most likely to be important for survival and development of the eggs. One important physical factor which could influence the environment in a weed patch, and which may vary considerably in different parts of the river channel, is flow rate. We therefore hypothesised that flow rates could be an important determinant of patch quality.

Say what the study actually does:

> In this study, we investigated the physico-chemical differences between 'good' and 'poor' quality patches of weed as defined by the behaviour of the damselfly *Calopteryx virgo* Linnaeus. We also tested the assumption that males on more vigorously defended patches have greater mating success.

Don't separate out the question, hypothesis and predictions as special statements in bold or whatever, or put them under separate headings. Although they provide vitally important context to a study, they should simply appear where necessary as part of the normal text.

### B.2.4 Methods

The Methods section should provide enough information about how the study was carried out to enable the reader to evaluate the validity of the results.

- What was done?
- Where (usually necessary for fieldwork) ?
- When (may be necessary for seasonally dependent studies) ?
- Why (may be necessary to justify the use of a particular approach) ?

It is often said that we should write the methods so that someone could repeat what we've have done exactly. This is OK in principle, but often takes an excessive amount of space and shouldn't be the overriding principle. The emphasis should be on giving the reader sufficient information to evaluate the results of a study. Focus on the important detail: i.e., it doesn't matter that we sorted our sample into Petri dishes, or which make of microscope we used, but it does matter that we worked at 20x magnification, because that may determine how likely it is that we missed very small items. The main exception to this is if we're reporting a novel technique that other people are likely to want to use. In that case, more detail than normal will be required.

Be concise. There is no need to explain the details of standard procedures. If we're using a procedure described by someone else then summarise the essential features and just cite the reference for the method. It is not necessary to state which statistical tests were used unless they are non-standard or require particular discussion (e.g. it is useful to state that the data were transformed before analysis). Similarly, we don't need to state what statistics package were used for standard statistical procedures. Avoid 'padding' sentences that waste space, such as:

> The data were analysed statistically and by plotting graphs to see what the results were.

The standard style in scientific reports is to write in the third person ("Experimental plots were marked out" rather than "We marked out experimental plots"). This an area where accepted conventions vary between different disciplines. In some, the use of the first person is permitted and even encouraged where it enhances readability of the text. Judicious use of 'I' or 'we' can improve the clarity and readability of text and should be used where appropriate. But keep in mind that the use of the first person is not accepted in some disciplines. If in doubt, it is safest to stick with the third person approach.

In any case, try to use the active voice, i.e., avoid this style…

> It was found that males always defended single weed patches.

… and and use this style instead:

> Males always defended single weed patches.

A final point is that when a study is made up of several experiments, or sets of observations, it is a good idea to use appropriate subheadings to make it easier for the reader to follow, both within a particular section (such as the Methods) but then also using the equivalent subheadings to organise the Results and possibly the Discussion. e.g.,

> MATERIALS AND METHODS
>
> **Study site**
>
> **Territory occupancy by males**
>
> **Oviposition behaviour**
>
> **Experimental manipulations of flow rate**

### B.2.5 Results

The central goal of a results section is to provide a clear account of the material factual findings of the investigation, using a combination of text, summarised data, and figures. If different parts of the study are covered under different subheadings in the Methods, then use the same subheadings to organize the Results.

The Results section should focus on explaining clearly what the results are, but should not contain discussion of the biological implications of the results.

Results are presented in a variety of different ways:

**Text**. The text part is important. This must include clear statements of the results. No result should just be presented just as a figure or a table with no corresponding statement in the text. It is important to lead the reader through the information, bringing out the important features. This does not mean that we should duplicate information in text and figures, but if a figure is used, there should be a reference to that figure in the text that summarises the key results.

**Data**. Numerical data are normally presented in tables (see below), but sometimes ‘stand-alone’ simple numerical results can be given in the text. In either case, the data is normally be presented in summarised form only (e.g. means and standard deviations).

Don’t include big tables of raw data. Presenting raw data may be appropriate if there is a legitimate need to discuss the values of specific data points, but this is rarely the case. If we want include the raw data (e.g. because the data set may be of use to others), they should be supplied as an appendix or some kind of stand-alone data file. Alternatively, if our goal is to share our data with the rest of the world we might consider using an online data repository like Figshare.

**Statistical summaries**. The results should be where most or all of the statistical results appear. There are three places to include summaries of statistics:

1. In the text…
2. In figure legends…
3. In tables. If there are large numbers of tests to present which would clutter the text, e.g. the analysis involves 10 regressions of the same kinds of variables, for a number of different taxa, then it may be convenient to summarise the slopes, intercepts and significance of the relationships in a table rather than trying to put all ten in the text (although a figure would be even better).

Be sure to report statistical results in full: include the test statistic, degrees of freedom (both of them, when reporting an *F*-test), and the *p*-value.

**Figures and Tables**. Any type of graphical presentation is a figure. A table has just text and numbers. All figures and tables must be referred to in the text of the Results (or elsewhere).

Tables…

- contain just text and numbers
- have the legend at the top
- use just horizontal lines as separators
- are labelled: Table 1, Table 2, etc.

Table B.1: **Table 2**. The flow rates of the manipulated patches, and mean simultaneous number and turnover (number of different males per day) of territorial males on experimental patches. (Values in parentheses are standard errors for each mean).

| Experimental treatment | Mean flow rate (m $s^{-1}$) | Mean number of males per patch | Mean turnover of males per patch |
|---|---|---|---|
| Increased flow | 0.45 (0.11) | 1.20 (0.22) | 1.2 (0.4) |
| Increased flow | 0.18 (0.09) | 0.18 (0.09) | 3.1 (0.6) |
| Increased flow | 0.02 (0.01) | 0.10 (0.03) | 5.9 (0.9) |

Figures…

- have a legend at the bottom
- should be labelled: Figure 1, Figure 2, etc.

- can comprise a single graph or diagram, but a single figure can also comprise several graphs – in which case each should be labelled: a, b, c, etc.

### B.2.6 Discussion

The function of the discussion is to consider the meaning of the results and the light they throw on the original question; to assess the results in the context of other studies; and, if appropriate, to consider the limitations of the work and future directions for study. The exact structure and content of the Discussion will vary somewhat depending on the particular study and what the results show, but usually the following components should be included.

It is usually helpful to start the discussion with a short paragraph, or so, summarizing the key results. e.g.,

> *Calopteryx* females exhibit a distinct preference for weed beds in faster flowing water as oviposition sites. Males recognise such good quality sites and occupy and defend them against other males, ignoring weed beds in slower water. This results in more copulations for males which occupy the fast flowing sites. The assessment and response of males to flow rate changes can occur within a few hours

Next, we usually consider the whether the results support the hypothesis or suggest it requires modification or rejection. e.g.,

> The male damselflies' preferential occupancy and vigorous defence of weed patches with faster flow rates, combined with a clear positive relationship between flow rate and oviposition frequency, provides strong support for the view that the underwater environment is an important determinant of oviposition site quality.

It may be important to discuss the limitations of the study and the appropriate direction for further work, but these are not always required. Don't pad out the discussion with endless text considering every possible wrinkle in the study. When appropriate, a discussion of the limitations should be brief and to the point.

> Although the results do implicate flow rate as a determinant of oviposition site, it is not clear whether females are responding directly to flow rate, or whether males are assessing flow rate and females are selecting the higher quality males (presumably those that occupy the best patches) assessed in some other way. This would require a separate experiment where females were allowed to select oviposition sites in the absence of males.

Don't just grumble and don't make stock criticisms without good reason (e.g., don't automatically say it would have been better to have a larger sample size—this may be true, but it may not—large sample sizes don't solve everything).

There may be unresolved, or unsolvable, problems. Be honest about these, but also be positive: if the author don't seem to be sure that a study is worth reporting, how will anyone else be convinced? A report does not require a section headed 'Experimental Error'. Similarly don't attribute any problems that can't be explained to experimental error; everyone knows measurements aren't perfect so it doesn't explain anything.

Finally, bring out the wider implications (but be realistic about the significance of the work) and future directions, e.g.

> These results indicate selection of oviposition sites, by females, on the basis of flow rate, but the reasons for such selectivity are not known. Flow rate has been implicated in other studies of aquatic insects as being of importance for preventing low oxygen conditions developing (a stress to which developing eggs may be particularly sensitive) (Armherst 1989). High flow may also reduce the ability of egg parasitoids to search the plants (Girton and Jenner 1976). A critical part of assessing the basis of site choice, and evaluating the role of the underwater environment will be measurement of egg and larval survival in weed beds of different flow rates. It seems likely that the patterns observed in *Calopteryx* in a single section of the river may also be important in determining choice of habitat between different river sections or even different rivers with high or flow rates. This also raises the unwelcome possibility that quite subtle changes in flow caused by water abstraction and river regulation (a problem on a neighbouring stream to the study site) could cause marked interference with *Calopteryx* breeding and even loss of the species from a river system.

The Discussion should not contain new results (except occasionally for small additional analyses of the data that have arisen as a direct consequence of interpretation of the main results - and that shed light upon the questions in the paper). Also avoid over-extending the implications of what was found. A slight trend in the results from one particular experiment may not be an entirely sound basis from which to challenge the fundamental tenets of evolutionary biology. (On the other hand it just could be; the skill is in spotting the few occasions when it is!).

Overall, keep the focus of the discussion firmly on the results, don't wander off into ten pages of philosophical discourse on the state of the field in general. And keep the volume and depth of the discussion in proportion to the rest of the paper, and to the significance (biological rather than statistical) of the results.

### B.2.7 Acknowledgements

This is the place to acknowledge persons or organizations who have made significant contributions to the execution of the work. For example: funding bodies, people who have contributed ideas or assisted with some of the actual work, landowners giving access to sites, specialists who have made identifications and people who have read and commented on the manuscript. Don't get carried away—there's no need to thank every friend, relation and loved one for general help through life's little crises.

## B.2.8 Literature cited / References

This section should provide a complete listing of all, and only, references cited in the text of the report. There are three things to consider here:

- What to cite
- How to cite it in the text
- How to construct a reference list

### B.2.8.1 What to cite

We should cite appropriate references wherever a point of substance (fact, or opinion) is made that is not our own or may not be regarded as common knowledge. e.g.

Facts:

> Several species in the genus Calopteryx perform a complex 'wing floating' display as part of the courtship behaviour (Malmquist 1956)

Opinion of others:

> This behaviour is generally considered to be a display of male quality (Fredenholm 1978, Summers 1991).

Absence of citation is taken to indicate either the author's own view or a result generated by the present study:

> The function of this behaviour may be to signal the flow rate, and hence quality of a patch, to a female.

...or something sufficiently well known to be regarded as common knowledge:

> Damselflies are predatory both in the larval and adult stages.

### B.2.8.2 Styles of citation

If writing a manuscript for publication in a scientific journal, obviously use the style of the journal in question (exactly—including punctuation). When writing any other type of report, we can choose our own style, but if in doubt the easiest approach is probably to follow the style of a major journal in the appropriate subject area. There are two main styles in widespread use:

The most common (and most straightforward) cites references in the text using names and dates, and lists all references alphabetically in the reference list.

In the text, e.g.

> Wide fluctuations in temperature reduce egg viability (Smith 1987).

OR

> Smith (1987) found that wide variations in temperature reduced egg viability.

And in the reference list:

> Smith, A. J. (1987) The effect of temperature on egg development and survival in the damselfly *Calopteryx virgo. J. Zool. (Lond.)* 47: 231-243.

Note that the necessary information about the journal is the: journal title, the volume number (47) and the pages of the article (231-243). Journals often also have a part number, e.g., volume 47(2). There is no to include this in the citation; the page numbers should be sufficient.

The list should be in alphabetical order by first author. If there is more than one reference by the same author then order them by date. If there are papers with the same first author but different second/third authors then these come after the single author papers by the first author, and in alphabetical order by second, third, etc. authors, e.g.,

> Smith A J (1987)... Smith A J (1989)... Smith A J, Girton S and Mackay R H (1984)... Smith A J and Wallis K C (1983)... Smith A J and Wallis K C (1985)...

If several citations by the same author in the same year are in the list, then denote them with letters e.g.

In the text:

> Smith (1987a), Smith (1987b)

In the list:

> Smith, A. J. (1987a) The effect of temperature on egg development and survival in the damselfly *Calopteryx virgo. J. Zool. (Lond.)* 47: 231-243 Smith, A. J. (1987b) The oviposition behaviour of *Calopteryx virgo* (Odonata: Zygoptera). *Anim. Behav.* 27: 197-209

The other main style is to use numerical superscripts (or equivalent) in the text, numbering the references in the order in which they are mentioned in the text, and ordering the final reference list in the same way, e.g.

In the text:

> Wide fluctuations in temperature reduce egg viability[23]. Smith[23] found that wide variations in temperature reduced egg viability.

In the reference list:

> 22. Wilcove H, Papapangiotou L A and Lu, X (1978) Mating strategies in a calopterygid damselfly. *Anim. Behav.* 16:21-30
> 23. Smith A J (1987) The effect of temperature on egg development and survival in the damselfly *Calopteryx virgo. J. Zool. (Lond.)* 47:231-243
> 24. Morris L L (1991) A model of territory switching behaviour. *Am. Nat.* 230:390-395

In many journals using this system, the titles of the references in the list are also omitted e.g.

> 23. Smith A J (1987) *J. Zool. (Lond.)* 47: 231-243

This is done purely to save space, so unless we are specifically asked to do this it is best to include the complete reference. Although such numerical systems usually require the reference list to be ordered by number, it is possible (and much more convenient) to use an alphabetical listing even if numbers are used in the text (alphabetically ordered references are numbered in order and then the numbers used in the text instead of names). The advantage to a numbering system is that is saves space in the text, the disadvantage is that the numbers don't tell a reader which paper is being referred to as they read—they have to keep looking them up in the list.

Some final points to bear in mind about references and their citation: 1) every reference cited in the text must appear in the reference list, and every reference in the list must appear in the text; 2) we should never cite something we've not seen. If we need to cite something we have seen discussed or cited somewhere else, but haven't seen our self (and cannot get hold of) we should make it clear that we're citing someone else's interpretation of the original reference, e.g.,

> ...Jones (1928—cited in Smith 1987)

In the list we should then give the full citation for Smith (1987), not Jones (1928).

There are standard abbreviations for journal names. These are often given in the journal itself, and are available on a list in the Library, or can be found by looking up the journal on Biological Abstracts. If we don't know what the standard abbreviation is, and it is not obvious, then it's best to use the full name rather than making up a new abbreviation.

**Use a reference manager!**

Managing citations and generating reference lists can be a painful process, especially when everything is done ‘manually’ (cut and paste, cut and paste, cut and paste…). These days a number of reasonably good reference managers (software packages) are available to help with the process of managing references, inserting citations, formatting, and generating reference lists. Endnote, Paperpile and Zotero are among the most popular, but there are many different options. Choose one, and learn how to use it. This will save a huge amount of effort in the long run.

### B.2.9 Appendices

Use appendices for large amounts of raw data, long species lists, detailed mathematical or laboratory working, of a non-standard method, or (short) program listings, but only where the inclusion of such information markedly enhances the usefulness of the paper. Normally such appendices are not required. Avoid using them just to show how much work went into a study!

## B.3 Presenting species names

A final thing worth noting, is the correct way to present species names (there is an example of it in the passage above). This causes a great deal of confusion, largely because it is not always appreciated that specific meaning attaches to the conventions used for presenting species names. The name above, *Calopteryx virgo* Linnaeus, has several distinct elements in its presentation (italics, upper case initial letter(s) etc.). **These elements matter**. The full meanings of each of the various elements one might find in a scientific name are too extensive to cover here, but the following guidelines should cover most situations.

Presentation of common names is less fixed by convention than of scientific binomials but, in general, common names can be written with lower case initial letters unless the name itself contains a proper name (e.g. Norway spruce). Common names are written in the same typeface as the normal text. Common names can be used in reports, but the scientific binomial is a unique identifier that provides a standard, internationally recognized, label for a species. A report should always include the scientific name of the species we’re dealing with.

Obviously it is important ensure the scientific name is spelled correctly. Fortunately systematists’ scientific latin is fairly simple phonetically, but nonetheless it is best to check the name from a reliable source when writing it for the first time (try searching for 'calopterix' on the web!). So now the spelling is right let's look at the parts of the name and how to present them. In typeset documents we usually see:

> *Calopteryx virgo*

The first name (the genus) should begin with an upper case letter, and the second name (the species) should begin with a lower case letter. This is not a style choice, it is a rule! Both names are usually written in italics, but sometimes they're written in normal type and underlined. Underlining is a less-used alternative that derives from the fact that single underlining was the printers' instruction to a typesetter to set the text in italic. This convention was widely used in the days before word-processors, as italic text was tricky to produce on a typewriter.

Sometimes there will be more than just the genus and species names:

> *Calopteryx virgo* (Odonata: Calopterygidae)

The names on the right (though they could equally well be on the left) are the higher taxonomic classification (order and family in this case) and are sometimes presented to enable a reader to easily place the organism. Just having a species name is not always very informative unless the reader already knows what sort of organisms are being discussed. These are written in normal text, but with an upper case initial letter. Just to confuse things though, when we write the informal derivative version of such names such as 'odonate' or 'calopterygid', e.g.

> …calopterygids, unlike other odonates…

…then they have a lower case initial letter.

If, as is occasionally the case, we're describing the subspecies of an organism (e.g. *Calopteryx splendens xanthostoma*) then the sub-species name (*xanthostoma*) is formatted the same way as the species name.

In the passage above the name of the damselfly is followed by a name: Linnaeus. This is the **authority**, the name of the taxonomist responsible for naming the species (in this case the famous Swedish systematist Carolus Linnaeus). Sometimes a year is also shown after the authority. Taxonomy changes as groups are revised and new classifications developed, and so species names are often not the same now as the ones they were originally given. This results in a complicated system of having more than one authority, dates, and authorities appearing in different sorts of brackets and parentheses, sometimes abbreviated, sometimes not, e.g.,

> *Calopteryx virgo* Linnaeus 1758 *Calopteryx splendens xanthostoma* (Charpentier) *Althea rosea* (L.) Cavanille

To present things correctly in a report there is no need to know exactly what all these different arrangements mean. The important thing to remember is that things like the arrangements of parentheses mean something specific – don't just stick them in to make it look tidy. And when authorities are abbreviated (e.g. Linnaeus to L.) these abbreviations are fixed. We can't just decide to abbreviate an authority to something that looks sensible. If these esoteric details are really needed then copy them carefully from a reliable source. When should we include the authority? In a scientific paper it is conventional to include the authority when the species is first mentioned (in the main text, not the abstract), and leave it out thereafter. However, for most other purposes there is no need to include the authority.

Finally, abbreviation of names. Once we've given the full name of a species it is often convenient to refer to it in an abbreviated form later in the report:

> Females of *C. virgo* were regularly observed…

There is only one correct way of abbreviating the name: shorten the genus to its initial letter (plus full stop) and keep the full specific name; **never do the reverse** (*Calopteryx v.*). If there is a subspecies name then it's fine to abbreviate both generic and specific names, e.g., *C. s. xanthostoma.*

## B.4 Approaches to writing

Everyone tackles the task of writing a report in different ways, dictated by a combination of personal preference and practical constraints. That said, we can still provide some useful suggestions for approaching a scientific writing task.

Most importantly, separate the process of writing, editing and revising—these are not the same thing. Begin with a clear written plan of what to say before starting, but expect to go through several cycles of writing, editing and revising a report before it is satisfactory (it is impossible to get everything right the first time):

- When writing, focus on writing! Follow the plan and get the key ideas down, even if the text ends up a little ugly in places, and keep going. Avoid the urge to double back and polish up text.
- Editing is essentially a sentence-level process—it addresses problems with the minutiae such as spelling, grammar or word choice. This step doesn't involve large, structural changes to a document.
- Revising concerns higher level criticism. Are the questions clearly elaborated? Is there a logical flow of ideas? Are the Results and Discussion connected? This step often involves moving (or removing) paragraphs, extending or narrowing text, or rewriting confusing text.

Word processors greatly ease the task of editing and revising manuscripts, but it is sometimes better to get a draft down and then work on it on paper, rather than agonising too long in front of the screen. It is much easier to get an overview of the structure, and to spot errors, from printed copy rather than trying to do it entirely on the computer, where we're always peering at a fragment of the manuscript though a little window.

Many people think that starting with the Results and/or Methods sections is the easiest approach—these require straightforward reporting of factual information, and the pattern and presentation of results will be clearly established before it's time to move onto the Discussion and Introduction. If writing the Discussion and Introduction is still a struggle, try writing the Abstract first—this will crystallize the key messages.

Keep the writing simple, clear and concise. Science writing is about clear communication, not verbal acrobatics. Explain things to an appropriate level for the intended audience. This is harder than it sounds. By the time we reach the writing stage, we have usually spent a good deal of time thinking about the problem we're discussing. Things that seem 'obvious' may not be so obvious to a naive reader.

Find a critical friend to read a draft when it's ready (and be prepared to accept their honest comments!). It is of value to have the manuscript read by people who know the field and those who don't; they will pick up different things.

Finally, whenever possible, it is worth putting the report away for a couple of weeks and doing something else, then going back and rereading it. A report's faults are much easier to spot after a break from working on it.

### B.4.1 A last piece of advice...

No discussion can cover all the different sorts of report a scientist may be required to write at one time or another, but the ideas above provide a guide to one of the commonest. It is only a guide, and some circumstances will require a different approach or structure.

Writing isn't easy. Like any complex skill, it takes a great deal of time and effort to develop. People learn to write in different ways, and what works for one person may not work for another. Nonetheless, two pieces of advice from experienced writers crop up time and time again:

**Read actively.** It's impossible to become a competent writer without first becoming a regular and active reader. Reading provides an opportunity to develop scientific writing skills by critically evaluating the papers we read, not just in terms of the science they present but the effectiveness of their presentation. We can decide what works well and what doesn't and then adopt good ideas. Of course, this only works if we spend time reading the scientific literature in the first place!

**Write often.** Many good writers will say that although they don't always write a lot, they write often (every day!). Spread writing tasks out and set aside time to get them right. Rather than waiting until just before a deadline is looming to begin a project report or essay, start early and spend a little time every day working on it. Write regularly, but also write with effort—be prepared to critically evaluate, edit and revise written work.