icebreakeR: An Introduction to R

Andrew Robinson

Department of Mathematics & Statistics
University of Melbourne

Outline

Introduction

What is R?

- R is a programming language that has been optimized for data analysis and modeling.
- R can be used as an object-oriented programming language, or as a statistical environment within which sets of instructions can be performed automatically.

Why use R?

1. R runs on Windows, Mac-OS, and Unix variants;
2. R provides a vast number of useful statistical tools;
    1. many of which have been painstakingly tested;
3. R produces publication-quality graphics in a variety of formats;
4. R plays well with LaTeX via the `Sweave` package;
5. R plays well with FORTRAN, C, and shell scripts;
6. R scales, making it useful for small and large projects;
7. R is object-oriented;
8. R eschews the GUI.

Why avoid R?

Frustration!

1. R cannot do everything;
2. R will not hold your hand;
3. The documentation can be opaque;
4. R can drive you crazy, or age you prematurely;
5. The contributed packages have been exposed to varying degrees of testing and analysis;
6. R stores objects in RAM;
7. R eschews the GUI.

A contrast

1. R is object-oriented;
2. SAS is PROCedure-oriented;

Infrastructure

Organization

File structure: project

- data, documents, graphics, images, notes, scripts.

Exercise 1

Getting going and stopping

Getting Going

- Starting R
- Adding Packages

Exercise 2

Stopping

- Cancelling operations
- Quitting

Exercise 3

Working Directory

Where it all begins.

Relevant commands

- `getwd()`
- `setwd("`*new working directory*`")`

Exercise 4

Showcase

Work through the exercise in the Showcase chapter. Reflect on the commonalities between the exercise and what you need from R.

Getting Local Help

R comes with internal help. Use the examples.

Relevant commands

- `help(`*object*`)`
- `?`*object*
- `help.search("`*phrase*`")`
- `help.start()`

Exercise 5

Remote Help

The Internet is a vast repository of advice. Most of it is good.

Relevant commands

- `RSiteSearch()`
- Google
  - R-help . . .

Responsive Help

A list-server exists. Information is available at:
`https://stat.ethz.ch/mailman/listinfo/r-help`

Relevant issues

- Post questions as a last resort.
- `http://www.r-project.org/posting-guide.html`

Exercise 6

Writing Scripts

Why write scripts?

Relevant issues

- `source("script-name.R", echo=TRUE)`
- Comments: #

Exercise 7

Work Spaces

The workspace is the container of all your objects.

Relevant commands

- `ls()`
- `rm(`*object*`)`
- `rm(list=ls())`
- `save.image()`
- `save(`*object*`)`
- `load()`

Exercise 8

History

The past.

Relevant commands

- `savehistory()`
- `loadhistory()`

Exercise 9

Extensibility

R starts speedily.

Relevant commands

- `require(`*package*`)`
- `installed.packages()`
- `available.packages()`
- `install.packages(`*package*`)`

Interface

Import and Export

Import

- `read.`*xxx*`()`

Exercise 10

Export

- `write.`*xxx*`()`
- `pdf("`*pdf-name.pdf*`") ... dev.off()`

Exercise 11

R is hard work

Work hard!

R is hard work

Work hard!

That means:

- Read widely.
- Use the resources available.
- Experiment patiently and flexibly.
- Keep scripts.
- Comment generously.

What is an Object?

Characteristics

- Everything.
- Objects are realizations of a class.
- All objects have attributes.

Why use Objects?

Using objects simplifies many complicated problems.

1. Communication.
2. Comparison.
3. Coercion.

But you have to put them somewhere!

Assignment

Relevant commands

- *name* <- *definition*
- class(*object*)

Every object that is followed by () is a function, being called. Every object that is followed by [] is being sub- or super-setted.

Exercise 12

Object Types

Atomistic

- Numeric.
- String.
- Factor.
- Integer.
- Logical.
- Missing.

Exercise 13

Object Types

Containers

- Vector
  - Vectorization

Exercise 14

- Dataframe.

Exercise 15

- Matrix.
- Array.
- List.

Messing with Data

`merge()`

`reshape()`

`order()`

Write your own functions.

- Flow control
- Scoping
- Class control

Exercise 16

Data Descriptions

Simple data descriptions are readily available.

- Univariate
  - Numerical
  - Categorical
- Multivariate
  - Numerical/Numerical
  - Numerical/Categorical
  - Categorical/Categorical

Exercise 17

ALL THE LITTLE PIECES ...

- `plot(`x`, `y`, ...)`
- xlim=, ylim=
- xlab=, ylab=
- main=
- col=
- pch=, lty=, ...

Graphics: Organization

- `par(...)`
- `las=1`
- `mfrow=c(2,2)`
- `mar=c(4,4,3,2)`
- `new=TRUE`

Graphics: Augmentation

- `plot(`x`, `y`, ...)`
- `points()`
- `axis()`
- `mtext()`
- `box()`
- `legend()`

Graphics: Permanance

Right-click the graphic, copy as windows metafile, and paste to a document. Or . . .

- `pdf("pdf-name.pdf")`
- `plot(x, y, ...)`
- `dev.off()`

Graphics: Challenges

- Error Bars
- Colour by groups

Graphics: Contributions

- trellis (`lattice` package)
- grammar of graphics (`ggplot`, `ggplot2` packages)

Exercise 18

Exercise 19

Linear Models

```
name <- lm(y ~ x)
```

Useful Options

- `data = dataframe`
- `na.action = na.exclude`

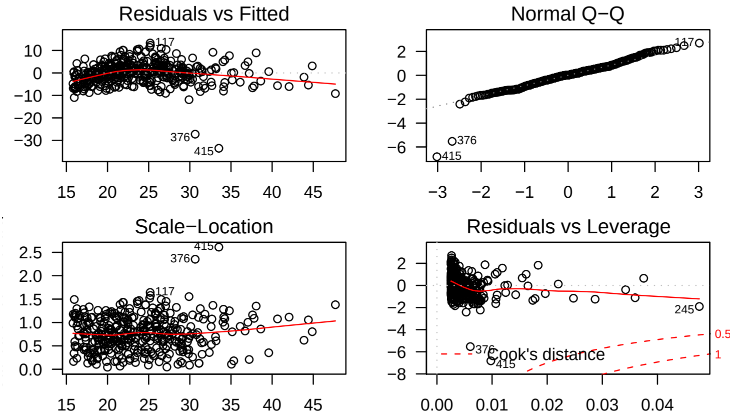
Occasional Options

- `subset = logical or index`
- `weights = weights`
- `formula = terms(y ~ x, keep.order = TRUE)`

Diagnostics Code

```
plot(model)
```

Diagnostics Interpretation



Estimation 1

```
> summary(hd.lm)

Call:
lm(formula = height.m ~ dbh.cm, data = ufc)

Residuals:
     Min       1Q   Median       3Q      Max
-33.5257  -2.8619   0.1320   2.8512  13.3206

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 12.67570    0.56406   22.47   <2e-16 ***
dbh.cm       0.31259    0.01388   22.52   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.941 on 389 degrees of freedom
Multiple R-Squared: 0.566, Adjusted R-squared: 0.5649
F-statistic: 507.4 on 1 and 389 DF,  p-value: < 2.2e-16
```

PREDICTION

`predict(`*model*`, newdata=`*new-dataframe*`, ...)`

Inference

```
> anova(hd.lm)

Analysis of Variance Table

Response: height.m
           Df  Sum Sq Mean Sq F value    Pr(>F)
dbh.cm      1 12388.1 12388.1  507.38 < 2.2e-16 ***
Residuals 389  9497.7    24.4
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Exercise 20

Mixed-effects models incorporate *two kinds* of predictor variables.

- Fixed effects - speak for themselves.
- Random effects - represent a population.

Necessity

Natural resources data commonly have hierarchical structure.

- Trees within plots within stands within forests.
- Times within trees . . .

Mixed-effects models enable the modeling of correlated data *without* violation of important regression assumptions.

Necessity

Natural resources data commonly have hierarchical structure.

- Trees within plots within stands within forests.
- Times within trees . . .

Mixed-effects models enable the modeling of correlated data *without* violation of important regression assumptions.

Regression assumptions.

- True relationship is linear.
- Residuals are normally distributed.
- Residuals have identical distribution (variance).
- Residuals are independent.

Utility

Mixed effects models allow the estimation of useful quantities.

- Variance components.
- Intra-class correlation.

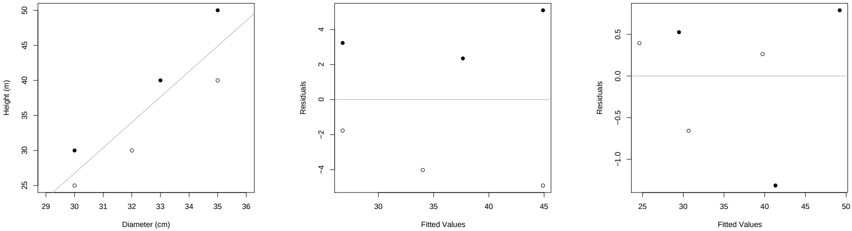
Yet another perspective

Construct a height-diameter relationship using two randomly selected plots in a forest, and that we have measured three trees on each.

Growing conditions are quite different on the plots, leading to a systematic difference between the height-diameter relationship on each.

If we fit a simple regression to the trees then we obtain a residual/fitted value plot.

If we fit a simple regression to the trees with an intercept for each plot then we obtain a residual/fitted value plot.


Height (m)
Diameter (cm)
Residuals
Fitted Values
Residuals
Fitted Values

Decomposition 1

Note that the model specification implies that:

$$y_{ij} - \hat{y}_{ij} = \hat{\epsilon}_{ij} \tag{1}$$

and

- The true relationship is linear.
- $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$
- The ϵ_i are independent.

Clearly not true.

Decomposition 2

What if we could make:

$$y_{ij} - \hat{y}_{ij} = \hat{b}_i + \hat{\epsilon}_{ij} \tag{2}$$

Then we merely need to assume that:

- The true relationship is linear.
- $b_i \sim \mathcal{N}(0, \sigma_b^2)$
- $\epsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$
- The ϵ_{ij} are independent.

Much more tenable!

Decomposition 3

The assumptions are satisfied because the systematic differences between the plots, which previously produced correlation, are now accounted for by the new random effects.

However, when the time comes to use the model for prediction, we do not need to know the plot identity, as the fixed effects do not require it.

Data - Height/Diameter from Stage (1963)

A brief synopsis: a sample of 66 trees was selected in national forests around northern and central Idaho. According to Stage (*pers. comm.* 2003), the trees were selected purposively.

The habitat type and diameter at 4'6" were also recorded for each tree, as was the national forest from which it came. Each tree was then split, and decadal measures were made of height and diameter inside bark at 4'6".

Scatterplot

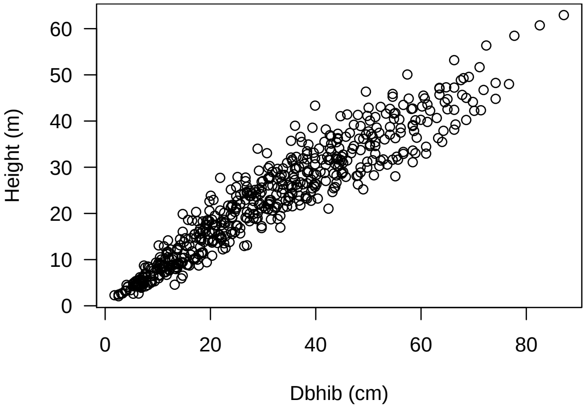


Figure: Al Stage's Grand Fir stem analysis data: height (m) against diameter (cm). These were dominant and co-dominant trees.

Another look

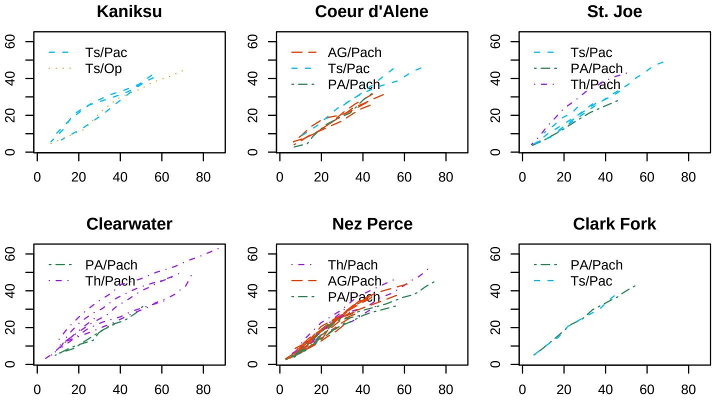


Figure: Al Stage's Grand Fir Stem Analysis Data: height (ft, vertical axes) against diameter (inches, horizontal axes) by National Forest. These were dominant and co-dominant trees.

Model for getting it wrong in R

$$h_i = \beta_0 + \beta_1 \times d_i + \epsilon_i \tag{3}$$

Model for getting it wrong in R

$$h_i = \beta_0 + \beta_1 \times d_i + \epsilon_i \quad (3)$$

Regression assumptions.

- True relationship is linear.
- $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$
- $Cov(\epsilon_i, \epsilon_j) = 0$ for $i \neq j$

DIAGNOSTICS FOR GETTING IT WRONG IN R

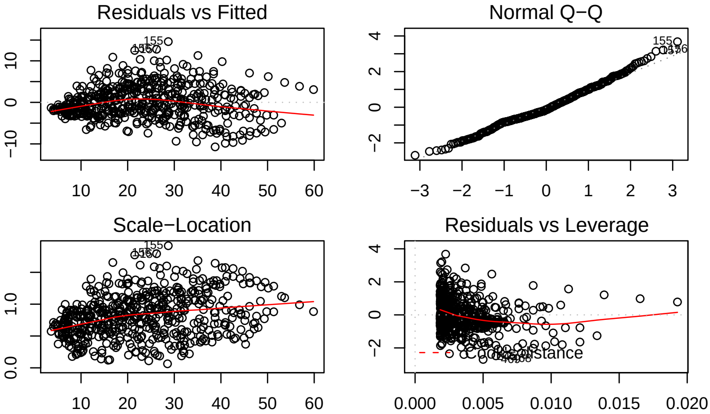


FIGURE: Popular regression diagnostics from R.

Model for getting it less wrong in R

$$h_{it} = \beta_0 + (\beta_1 + b_{1i}) \times d_{it} + \epsilon_{it} \tag{4}$$

Model for getting it less wrong in R

$$h_{it} = \beta_0 + (\beta_1 + b_{1i}) \times d_{it} + \epsilon_{it} \tag{4}$$

Regression assumptions.

- True relationship is linear.
- $b_{1i} \sim \mathcal{N}(0, \sigma^2_{b_1})$
- $\epsilon_{it} \sim \mathcal{N}(0, \sigma^2)$
- $Cov(\epsilon_{it}, \epsilon_{jt}) = 0$ for $i \neq j$
- $Cov(\epsilon_{it}, \epsilon_{ig}) = 0$ for $t \neq g$

Assumptions for getting it less wrong in R

Now, the key assumptions that we're making are that:

1. the model structure is correctly specified
2. the tree and forest random effects are normally distributed,
3. the tree random effects are homoscedastic within the forest random effects.
4. the inner-most residuals are normally distributed,
5. the inner-most residuals are homoscedastic within and across the tree random effects.
6. the innermost residuals are independent within the groups.

Diagnostics for getting it less wrong in R

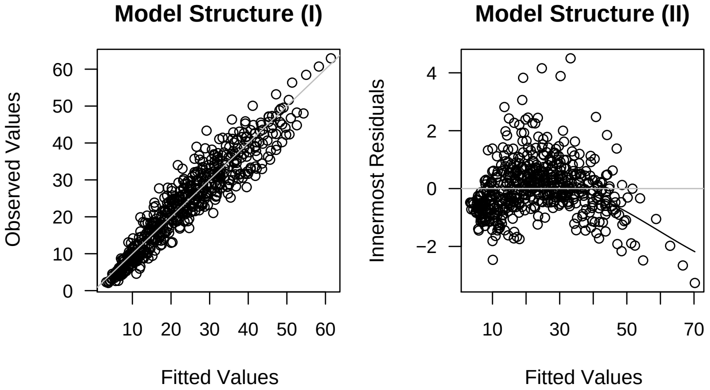


Figure: Useful regression diagnostics from R.

Diagnostics for getting it less wrong in R

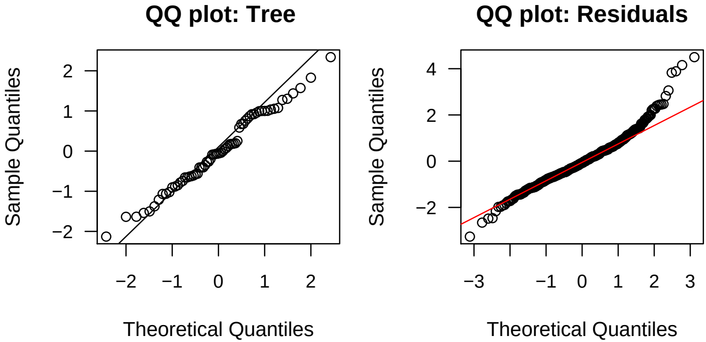


Figure: More useful regression diagnostics from R.

Diagnostics for getting it less wrong in R

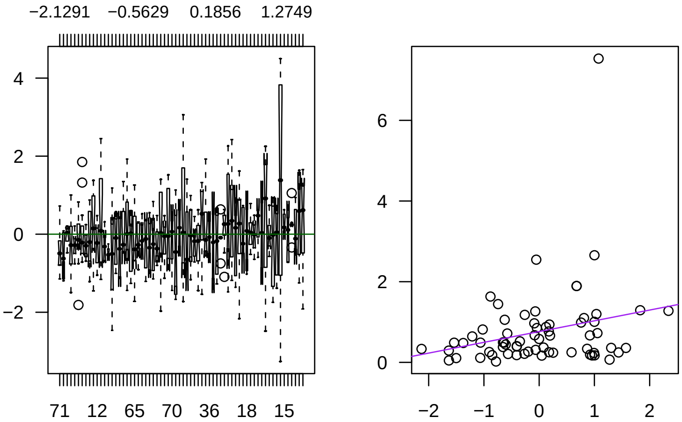


Figure: More useful regression diagnostics from R.

Diagnostics for getting it less wrong in R

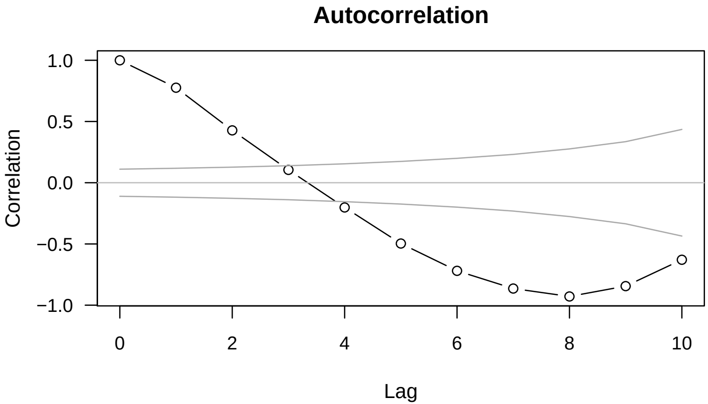


Figure: More useful regression diagnostics from R.

The roles differ

For the design,

- fixed effects represent *themselves*;
- random effects represent *a population*.

The roles differ

Within the model,

- fixed effects *explain* variation;
- random effects *organize* unexplained variation.

Random effects are effects that common sense says will explain variation, but you don't want to have to know them in order to be able to apply the model.

Modelling is much more involved

Add a new dimension to your flow chart!

A Modelling Strategy

The modeling strategy depends on the modelers intention.

1. Fit baseline model.
    1. Include the meaningful fixed effects.
    2. Include the design random effects.
2. Check the assumption diagnostics.
3. Add or modify random components until diagnostics are satisfied.
    1. a heteroskedastic variance structure (several candidates)
    2. a correlation structure (several candidates)
    3. extra random effects (e.g. random slopes)
4. Consider adding more fixed effects.
5. Re-examine the diagnostics, add/modify random effects, etc.

Basic Model Statement

$$\mathbf{Y} = \mathbf{X}\beta + \mathbf{Z}\mathbf{b} + \epsilon$$

$$\begin{aligned} \mathbf{b} &\sim \mathcal{N}(\mathbf{0}, \mathbf{D}) \\ \epsilon &\sim \mathcal{N}(\mathbf{0}, \mathbf{R}) \end{aligned}$$

Basic Model Statement

$$\mathbf{Y} = \mathbf{X}\beta + \mathbf{Zb} + \epsilon$$

$$\begin{aligned} \mathbf{b} &\sim \mathcal{N}(\mathbf{0}, \mathbf{D}) \\ \epsilon &\sim \mathcal{N}(\mathbf{0}, \mathbf{R}) \end{aligned}$$

Design Matrices

- **X** allocates the fixed effects.
- **Z** allocates the random effects.

Basic Model Statement

$$\mathbf{Y} = \mathbf{X}\beta + \mathbf{Z}\mathbf{b} + \epsilon$$

$$\begin{aligned} \mathbf{b} &\sim \mathcal{N}(\mathbf{0}, \mathbf{D}) \\ \epsilon &\sim \mathcal{N}(\mathbf{0}, \mathbf{R}) \end{aligned}$$

Design Matrices

- **X** allocates the fixed effects.
- **Z** allocates the random effects.

Covariance Matrices

- **D** describes the random effects covariance.
- **R** allocates the residuals covariance.

$\mathbf{Y} = \mathbf{X}\beta + \mathbf{Zb} + \epsilon$

$$\text{Var}\left(\mathbf{Y} \mid \mathbf{X}, \mathbf{Z}, \beta, \mathbf{b}\right) = \mathbf{R}$$

$\mathbf{Y} = \mathbf{X}\beta + \mathbf{Zb} + \epsilon$

$$\text{Var}\left(\mathbf{Y} \mid \mathbf{X}, \mathbf{Z}, \beta, \mathbf{b}\right) = \mathbf{R}$$

$$\text{Var}\left(\mathbf{Y} \mid \mathbf{X}, \beta\right) = \mathbf{ZDZ}' + \mathbf{R} = V$$

Log Likelihood

$$\mathcal{L}\left(\beta, \mathbf{V} \mid \mathbf{Y}, \mathbf{X}\right) = -\frac{1}{2}\ln\left(|\mathbf{V}|\right) - \frac{n}{2}\ln\left(2\pi\right) - \frac{1}{2}\left(\mathbf{Y} - \mathbf{X}\beta\right)' \mathbf{V}^{-1}\left(\mathbf{Y} - \mathbf{X}\beta\right)$$

Profile β out

$$\hat{\beta} = (\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{V}^{-1}\mathbf{Y}$$

β is gone!

$$\mathcal{L}(\beta, \mathbf{V} \mid \mathbf{Y}, \mathbf{X}) = f(\mathbf{V}, \mathbf{Y}, \mathbf{X}, \mathbf{Z}, \mathbf{D}, \mathbf{R})$$

β is gone!

Estimate $\hat{V}$ by maximization and then $\hat{\beta}$ by substitution.

ReML

Maximum likelihood estimators of covariance parameters are usually negatively biased.

ReML

Briefly, ReML involves applying ML, but replacing

- $\mathbf{Y}$ with $\mathbf{KY}$;
- $\mathbf{X}$ with $\mathbf{0}$;
- $\mathbf{Z}$ with $\mathbf{K'Z}$; and
- $\mathbf{V}$ with $\mathbf{K'VK}$

where $\mathbf{K}$ is such that $\mathbf{K'X} = 0$.