Statistical Learning and Data Mining – An Example-Based Introduction with R

John Maindonald

October 16, 2010

Contents

A Warning Note:
A big computer, a complex algorithm and a long time does not equal science.
[SSC 2003, Halifax, June 2003]

Preface

Data mining may be seen as a response, in the first place from the computer science community, to ways in which advances in computing technology – both software and hardware – have transformed the collection and use of data. These changes have affected science, commerce and government.

Since the 1960s, each passing decade has set new records set in the size of the largest of the data sets that are analysed. Often also, both in science and commerce, there has been an increase in complexity. The primary challenge is, often, data management.

There are important classes of problem where the analysis can to a large extent be automated, or at least handled without great attention to statistical considerations. It is here that the general style of approach that has been typical of data mining can be successful. Much of the statistical theory that supports data mining falls comes, broadly, within the framework of statistical learning.

Notwithstanding emphases and origins that differ somewhat from those of traditional applied statistics, data mining makes demands of the data analyst that are entirely comparable to those of traditional statistical analysis. Just as with statistical analysis, what is done should be driven by the questions that are asked, by prior knowledge, and by the demands of the data themselves. Typically, the aim is to make comments, or reach conclusions, that are valid beyond the particular data that are analysed. Statisticians are likely to speak of extracting information from data. The data mining literature may prefer to speak of extracting patterns from data.

Commonly, data mining has prediction as its aim. Data analysis demands that can be addressed from this limited purview do however often morph into a demand to identify the main factors that are driving the prediction. This is an exercise that is fraught with hazard, with challenges that go well beyond those of prediction. This present account makes some limited attempt to draw attention to those hazards and challenges.

This text has only limited coverage of the extensive statistical theory that offers important insights on all data analysis, and that can be important for understanding the benefits and limitations of automation. Consistent with this:

- There will often be recourse to a relatively informal ideas-based approach that makes little explicit use of mathematical formulae.
- Empirical approaches, e.g., to assessing accuracy, will be emphasized at the expense of modeling approaches that rely more heavily on statistical theory.
- Hints will be given on areas of statistical theory that it will be useful to master, in parallel with this text or following on from it.

Statistical theory, as it affects practical data analysis, is currently developing rapidly. This is a result of a synergy between new theoretical developments, and the computational power (software and hardware) of modern computer systems. The R system, used for the computations that are described here, is a product of such a synergy.

A Text in Eight Parts

The text is structured as follows:

I: Overview of Major themes

II: Data Summary Appropriate forms of summary can often provide useful insight. Be careful however of the potential for misleading forms of summary. Summary may also be a useful or necessary preliminary to further analysis.

III: Populations, Samples & Sample Statistics These chapters give an overview of statistical theory and ideas that will be important in the later discussion.

Part I
Overview of Major Themes

1 Advance and Change in Science, Commerce and Technology

Changes that the data mining literature emphasizes, a result of or made possible by computer technology, are:

- Huge data sets[1] have become common. Often these hold new types of data – web pages, medical images, expression arrays, genomic data, NMR spectroscopy, sky maps, ...
- Sheer size brings challenges. In the first place, these are challenges for data management. Challenges for data analysis are less simply described.
- New algorithms; "algorithmic" models.
  - Examples are trees, random forests, Support Vector Machines, ...
  - They are algorithmic in the sense that the motivation was algorithmic.
- Automation, especially for data collection

The effects of these changes have spread across commerce, government and society, as well as across science. The same or similar technology may be used across these different areas of application.

Database issues, size of dataset, and new algorithms, are strongly emphasized in the data mining literature. Changes that get much less attention in the data mining literature are:

- Data set size is not necessarily a useful guide to the amount of information in the data.
  - A key question is whether there are many observations, or many variables, or both.
  - Where the number of observations is large, there is often a structure (eg, changes in time) that requires attention.
- While there has been huge progress in automating much of the detail of data analysis, there are severe limits to the automation that is, currently, satisfactorily possible.
  - Except in limited areas of application, complete automation remains a pipe dream. In those areas where it has proved effective, automation typically has a large setup and maintenance cost.
  - Automation is most feasible in applications where mistakes can be tolerated, where it is not necessary to be consistently correct.
- There is a synergy between the development of computing power, the challenge from new types of data, and the development of new theory. Much of the new development in theoretical statistics has been driven and facilitated by the demand to take advantage of new computational power.
- New data analysis methodologies often allow analyses that make better use of the data, and are more directly attuned to the questions of scientific interest, than was readily possible 15 years ago.
- The account given here will emphasize the importance of statistical issues and insights. Advances in statistical methodology have widened the gap between those whose statistical knowledge has not advanced much in the past decade, and those professionals who are fully au fait with modern methods.

[1] Issues of data set size have generated some modest smount cf hype, as the frequent reference to *Big Data* in Weiss and Indurkha (1997)

- Algorithms become important once the required analysis task has been closely prescribed. Algorithms are required that will be efficient in extracting the information that is required from the data.
- New statistical "meta-analysis" approaches that combine data from multiple studies into a single analysis may allow the detection of patterns that were not apparent from the individual studies. They may resolve some discrepancies between the separate analyses, while raising further questions. Note that meta-analysis typically has complications that make automation hazardous.

While emphasizing the synergy with new algorithmic development, the data mining literature has pretty much ignored the synergy with new developments in statistical theory.

Other responses to the changes, with much in common with data mining, have the names "machine learning", and "analytics". Machine learning has grown out of an engineering context, while the name "analytics" is widely used in a business context. Other such names are in use also, most of them with a focus on a specific area of application.

Data mining is sometimes presented as a collection of algorithms. Validation issues are, largely, left to one side. The present account will widen the scope, to describe an enterprise that builds on advances in data collection and data analysis technology in ways that pay serious regard to model validation issues. In this account, statistical considerations have a large role.

Maindonald, J.H. (2006) comments, from a somewhat different perspective, on a number of the issues that are raised below.

2 Mining, Learning and Training

Mining is used in two senses:

- Mining for data
- Mining to extract meaning is a scientific/statistical sense.

The pre-analysis data extraction & processing often relies heavily on computer technology. Additionally, there are design of data collection issues that demand more attention than has been common.

In mining to extract meaning, statistical considerations come to the fore. Computer power does however provide a major part of the "how".

Learning & Training

- The (computing) machine *learns* from data.
- Use *training* data to train the machine or software.

Modeling (or a machine?) that learns from the data

The reference is to modeling in which the data have a substantial role in determining the form of model. The demand for such models arises, in part, from the size of the data sets (many observations) that are now commonly available. Deviations from the strict form of model that are described by theory are more readily detectable. Statistical learning approaches are then used to accommodate deviations from the theoretical model. The plot of residuals in Figure 1 suggests that, for the data displayed there, a statistical learning approach may be appropriate.

For classification models, theory rarely gives much help in deciding on the form of model. A statistical learning approach is, to a greater or lesser extent, inherent in the nature of the modeling problem.

In the applications of statistical learning that are prominent in the data mining literature, the aim is usually prediction rather than the obtaining of interpretable model parameters. Hence the name "predictive modeling". Interpretation of model parameters raises additional issues that will be the subject of brief comment in a later section.

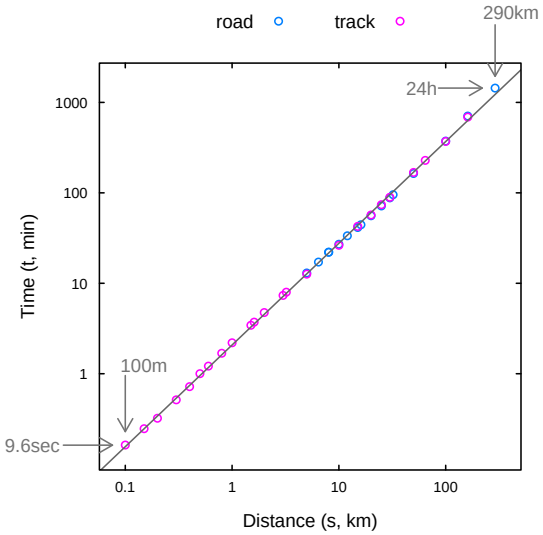


Figure 1: Record times versus distances, with both scales logarithmic, for track and road athletic races. With a ratio of largest to smallest time that is ~3000, differences from the line have to be large to be visually obvious. The plot of residuals shows that, for the longest race, the difference from the line is >15%. (Differences on a scale of natural logarithms, if small, are a little less than fractional differences.) The next figure shows a clear systematic pattern in the residuals.

2.1 Statistical Learning Example

2.1.1 Continuous Outcome

The first example (Figure 1) is for a continuous outcome variable. Data are world record times for athletic track and road races, as at October 2006. The range of distances and times is huge, from 100m in 9.6sec to 292.2km in 24h.

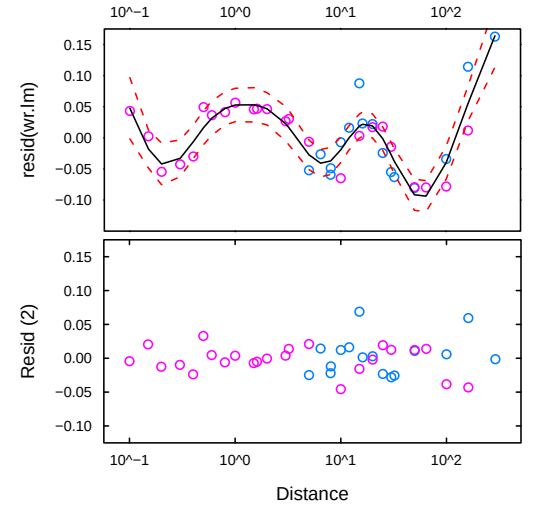


Figure 2: Here a smooth curve has been fitted to the residuals, using a routine that does the job pretty much automatically. It is assumed that residuals from the curve are independent. This assumption is especially crucial for the 95% pointwise confidence limits about the curve. The lower panel shows residuals from the smooth.

We can fit a curve rather than a line (Figure 2), in what is a statistical learning approach. Here, a curve will be fitted to the residuals – this corrects for the biases in the line.

Questions are:

- Will the pattern be the same in 2030?
- Is it consistent across geographical regions?
- Does it partly reflect greater attention paid to some distances?
- So why/when the smooth, rather than the line?

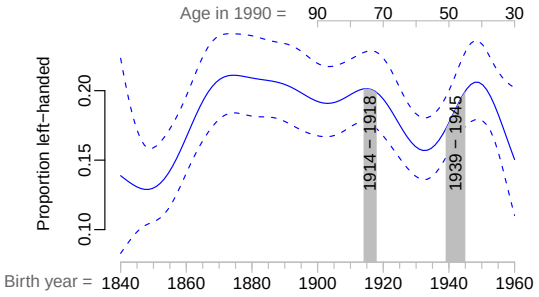


Figure 3: Variation with birth year in the proportion of first class cricketers who used their left arm for bowling.

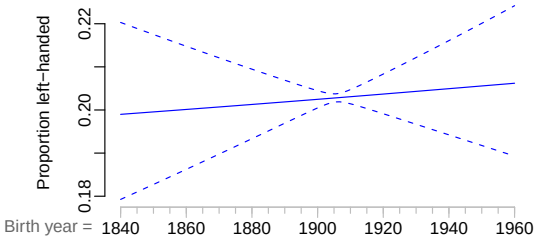


Figure 4: Result from use of simulated data, in which the probability that a birth will yield a first class cricketer is a constant 20%

Clearly the smooth curve (line, with 'corrections' from the line) would be useful to race organizers who wished to estimate the time at which a race winner could be expected to appear.

This is a very simple example of the use of the methodology. It generalizes, allowing the fitting of curves and surfaces, in principle in an arbitrary number of dimensions.[2] The ability to fit such curves and surfaces automatically is remarkable, relative to what was available a decade ago.

2.1.2 Binary data

Data, extracted by John Aggleton (now at Univ of Cardiff), are from records of UK first class cricketers born 1840 – 1960. Variables are

Year of birth
Years of life (as of 1990)
1990 status (dead or alive)
Cause of death: killed in action / accident / in bed
Bowling hand – right or left

The key assumption is that bowling arm is independent between cricketers. This assumption would be vitiated if for example data were comprised entirely of identical twins, with the two members of a twin pair generally expected to use the same bowling hand!

Given this assumption, the changes in the proportion of left-handers must then reflect changing external conditions, perhaps changing opportunities for left-handed players to join clubs and get good coaching.

We can check that the methodology does not have any tendency to produce curves where there are none. Figure 4 was obtained by simulating a situation where, each time a new cricketer is born, the probability of left-handedness is 0.2.

[2]Note in particular the abilities in the R package *mgcv*, documented in detail in Wood (2006).

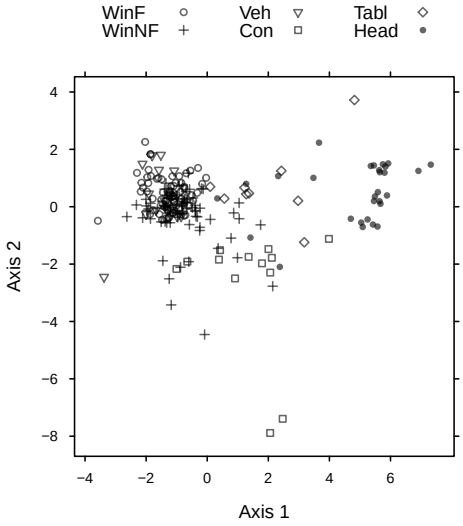


Figure 5: Visual representation of a classification rule, derived using *linear discriminant analysis*, for the forensic glass data. A six-dimensional pattern of separation between the categories has been collapsed down to two dimensions. Some categories may therefore be better distinguished than is evident from this figure.

2.1.3 Classification example – forensic glass identification:

Now consider a classification example. intended to help illustrate some of the important issues. As is common in many of the examples that are the stock-in-trade of the data mining literature, the interest is in prediction rather than interpretation of model parameter of estimates.

The example relates to glass fragments that were collected in the course of forensic work. Glass was of the following types. Numbers of pieces of glass of each of the different types are given:

Window float (70)	Window non-float (76)	Vehicle window (17)
Containers (13)	Tableware (9)	Headlamps (29)

Variables are %'s of `Na`, `Mg`, ..., plus refractive index. In all there are 214 rows of data (observations) × 10 columns (variables).

The aim is to find a rule that predicts the type of any new piece of glass. Figure 5 is a visual summary of the result from the use of a simple form of classification methodology, with the name *linear discriminant analysis*.

Questions, for any use of the results (e.g., to identify glass on a suspect)

How/when were data generated? (1987)

- Are the samples truly representative of the various categories of glass? (To make this judgement, we need to know how data were obtained.)

  Are they relevant to current forensic use? (Glass manufacturing processes and materials have surely changed since 1987.)

  What are the prior probabilities? (Would you expect to find headlamp glass on the suspect's clothing?)

These data are probably not a good basis for making judgements about glass fragments found, in 2008, on a suspect's clothing. Too much is likely to have changed since 1987. We'd want data that are a better match with the glass fragments that one might currently expect to find. We can then generalize with confidence, from the sample from which results have been obtained to some wider population.

In practice, that may be an almost impossible ask. We may have to be content with data that are from a population that is a less than perfect match to the population to which results are to be applied.

Analysis results: The overall classification error, from use of the random forests algorithm that will be described later, 20.1%. The confidence with which different glass types can be classified does however vary greatly from one type to another. For glass types other than Window float and Window non-float, the evidence is rather scanty.

The classification error rates were:

	Classification error
Window float ('WinF': 70)	0.10
Window non-float ('WinNF': 76)	0.22
Vehicle window ('Veh': 17)	0.59
Containers ('Con': 13)	0.23
Tableware ('Tabl': 9)	0.22
Headlamps ('Head': 29)	0.14

2.2 Some further reflections – What is data mining?

Daryl Pregibon's definition of data mining as "Statistics at scale and speed" may be as apt as any. Scale and speed create, inevitably, a large demand for automation. The skill lies in knowing what to automate, when to call on the skill of the human expert, and in the use of tabular and graphical summaries that will assist the judgment of skilled data analysts or call attention to features of the data that might not otherwise be obvious. The demand for scale, speed and automation has created many opportunities for researchers from a computer science tradition to take a lead role.

Data mining, and indeed all data analysis, draws both from statistics and from computing:

- Statistics contributes: methods for the design of data collection, models, the distinction between signal and noise, attention to issues of generalization, well-tested modeling approaches, and a long tradition of experience in the analysis of data.
- Computing has contributed the means for managing data, for automating large parts of computations, for maintaining an audit of all steps in an analysis, and some novel algorithms and algorithmic approaches.

Comments in Witten and Frank (2000), with respect to machine learning. seem relevant also to data mining:

> In truth, you should not look for a dividing line between machine learning and statistics, for there is a continuum, and a multidimensional one at that, of data analysis techniques. . . . Right from the beginning, when constructing and refining the initial data set, standard statistical methods apply: visualisation of data, selection of attributes, discarding of outliers, and so on. Most learning algorithms use statistical tests . . . (p.26).[3]

There are then several alternative names for disciplines, or traditions, that operate in the same general arena as statistics – including especially machine learning and data mining. Another name that has some currency is *analytics*, witness Davenport and Harris's text that *Competing on Analytics*. I will use *data analysis* as a name for activites that attract one or more of these names.

[3]Be careful, though, what you do with outliers! Unless demonstrably erroneous, they should, although perhaps omitted from the main analysis, be reported and included in graphs. In some analyses the interest may be in a small number of points that lie away from the main body of the data.

3 Purpose, Context, Interpretation and Generalization

A data mining style of analysis, if it is done properly, offers exactly the same range of challenges as data analysis more generally. Key issues for any study are:

1. Why am I undertaking this investigation?
2. What is the intended use of results?
3. How widely is it hoped that results will apply?
4. What limitations, arising from the manner of collection or from the incompleteness of the information, may constrain that intended use?

When the analysis is complete, a key question will be: "What is the relevance of these results?"

3.1 Purpose

The following is a (perhaps incomplete) list of the purposes that a data analysis may aim to serve:

1. Data collection and summarization may be an end in itself. A business needs to have accurate accounts just so that it can know whether it is making a profit.
2. Prediction; i.e., the aim is to make statements that generalize beyond the circumstances that generated the particular data that are under study.
3. Understanding – the elucidation of pattern. To be of interest, the pattern must usually be relevant beyond the immediate data in which it was found, i.e., generalization is an issue here also.

It is then important to ask which of these apply.

The answers may have strong implications for the any decision on how to handle the data. Data mining exercises typically are mainly interested in predictive accuracy. Questions of what interpretation can be placed on model parameters do however often arise, often as an afterthought to the main analysis. It is therefore important to understand the potential for misinterpretation.

Most (all?) data mining analyses involve an element of generalization. In predictive modeling, generalization is an explicit concern. The nature of the generalization will typically have large implications for the investigations that are to be undertaken, of a kind that this module will explore.

Is an hypothesis essential?

The hypothesis testing approach to inference, while in wide use in some areas of statistical application, seems relatively uncommon in the data mining literature. Certainly, it offers a means for making statements that apply beyond the specific data used to generate and/or test them. It is not however always the best or most appropriate approach for this purpose.

3.1.1 Example –the different uses of Australian Bureau of Statistics data

Note the variety of uses of data that are collected by the the Australian Bureau of Statistics. By explicit use of samples, or (less often) census data, statements will be made that apply to one or other Australian population – to humans, sheep, farms, or whatever. Results may be used directly to allocate resources, e.g., the distribution of GST revenue to states. They are also a resource that will be used by researchers (statisticians, data miners) to find that patterns that will guide decision-making. As those decisions will affect the future, the interest is in those patterns that can be expected to persist into the future, i.e., there is a predictive element.

3.1.2 Exercises:

Set out aims for analysis for the studies that have generated the following data:

The forest cover type data set, available from the web site noted in connection with Blackard (1998). See the file **covtype.info** for details of these data.

The data set `ant111b` that gives yield of corn for each of four blocks at each of eight sites on the island of Antigua in the Caribbean, in a single year.[4]

The data set on tinting of car windows (`tinting` (also in `DAAG`).

The attitudes to science data set (`science`,*DAAG*).

Data on diet-disease associations, with the food frequency questionnaire as the diet measurement instrument.

Data on diet-genotype associations, with SNP (single nucleotide polymorphism) information for each of a number of positions on the chromosome used to indicate genotype.

Studies and/or associated data sets that may be encountered in remaining modules of the course.

3.2 Analysis, and interpretation

This section will note some of the issues that become important for any detailed analysis. These are phrased as questions:

- There are many different methods/algorithms. How should the analyst choose between them? What are good ways to assess the performance of one or other algorithm?
- Often, the analyst would like to know which data columns (variables, or features) were important for the classification. Could some of them be omitted without loss?
- The analyst may want to attach an interpretation to one or more coefficients? Does the risk of heart attack increase with the amount that a person smokes?
- Above, I jumped directly into fitting a classification model, with no preliminary scrutiny of the data. This is risky. What sorts of preliminary scrutiny can be used to identify problems with the data, or issues that ought to be addressed?
- I offered a two-dimensional summary of the results, allowing some insight into the classification result. What can be learned from such a plot? What other investigations might give useful insight on the analysis results?

3.2.1 Analysis methodology

The discussion to date has focused on regression with a continuous outcome, and classification, with a categorical outcome. In a broader view of regression classification is however a type of regression – a regression where the outcome is a classification rather than an outcome values for a continuous variable. The two types of problem have important common features, as well as important differences. Where the focus is on features that they have in common, it makes sense to consider them together in the same discussion. When the differences seem more important than the common features, they will be considered together.

The data mining literature places a great deal of emphasis on the new "algorithmic" methods – tree-based methods (including ensemble methods such as random forests), methods that use "bagging" and "boosting", neural nets, and support vector machines. These (some of them at least) are undoubtedly useful additions to the analyst's kit. Their usefulness, in preference to alternatives, should however be

[4] These data are included in the `DAAG` package for R. Several of the data sets that appear in illustrative examples in these notes are from *DAAG*.

demonstrated in the context of the demands of one or other data-based investigation. That includes, in particular, the demand to demonstrate that the result has a relevance become the particular circumstances that generated the data that were used.

3.2.2 The Interpretation of Model Parameters

Consider data[5] that gives record times for Northern Ireland mountain races.

The "obvious" simple model has log(*time*) as a linear function of log(*dist*) and log(*climb*).

$$\widehat{\log(\text{time})} = -5.0 + 0.68\ \log(\text{dist}) + 0.47\ \log(\text{climb})$$

Note the coefficients 0.68 (for log(*dist*)) and 0.47 (for log(*climb*))! Do they make sense? Thus, the coefficient 0.68 for log(*dist*) implies that the relative rate of increase of time with distance is, if climb is held constant, 68% of the relative rate of increase of distance. If a second kilometer is added to a 1 kilometer race, the time per unit distance will be better than for the 1 kilometer race.

The clue is that the coefficient predicts what will happen if climb is held constant. The one kilometer race then involves much steeper climbing (and decent) than the two kilometer race.

More interpretable coefficients can be obtained by regressing on log(*dist*) and log(*climb*/*dist*). The comparison between different distances is then fair.

For a meaningful interpretation of model parameters, it is necessary to be sure that:

- All major variables or factors that affect the outcome have been accounted for.
- Those variables and factors operate, at least to a first order of approximation, independently.

Rosenbaum (2002) suggests approaches that are often useful in the attempt to give meaningful interpretations to coefficients that are derived from observational data.

3.2.3 Accuracy assessment

Primarily, the accuracy assessment methods that are discussed here assume that the target population is essentially the same as the source population from which the data have been obtained. Even for this limited purpose, there is serious scope for getting answers that can be grossly optimistic.

Accuracy assessment is important for its own sake. It is helpful to know what the finally fitted model has been able to achieve. Unless however there is effective accuracy assessment, it will not be possible to fit a good model:

- Many methods work by starting with an initial model, which is successively refinement. Too much refinement (over-fitting) will lead to a model with reduced predictive power. It is necessary to know when to stop.
- Good accuracy assessments are required so that a model fitted using one methodology can be compared with a model fitted using another methodology.

Here are further, more specific, comments:

Continuous outcome data: For regression with a continuous outcome, normal theory accuracy estimates can, if the independent normal error assumptions are not too badly wrong, work quite well. Note however that:

- If the model is selected from a wide class of models, or if there is extensive variable selection (e.g., select the best 3 explanatory variables out of 10), the accuracy estimates may be grossly optimistic.
- If observations are not independent, accuracy estimates may again be wrong, usually optimistic. Note however that that the situation can in special cases be rescued by choosing a more realistic model for the "error". Some of the possibilities are:

[5] from `http://www.nimra.org.uk/calendar.asp`

- For data that are collected over time, models are available that can account for the likely sequential correlation.
- Variation is often multi-layered – variation between different countries, variation between humans in an individual country, variation between clinical assessment made on the same human, and so on. Again modeling approaches are available that can account for such different sources of variation.
- Spatial models are another possibility.

Empirical methods for accuracy assessment can in principal be adapted for use where there is a complex error structure. This does however require a clear understanding of the theoretical issues, and is not straightforward.

Categorical outcome data: Here, the theoretical accuracy estimates that are available for certain of the methods rely on asymptotic approximations. For 'algorithmic' methods, including tree-based methods and random forests, theoretical results have limited relevance. Accuracy assessment almost inevitably relies on empirical methods.

The empirical methods do if used correctly cope with the effects of model and variable selection.

The accuracy that matters

As has been pointed out, the source population from which data have been obtained will often not be exactly the same as the target population. There are two important issues here:

- Where predictive modeling of a comparable type is being carried out repeatedly, the analyst should keep a record of the comparison between after-the-event model performance and predicted performance, e.g., from cross-validation on the original data.
- Often, predictions are successively made ahead in time. If a long enough data series is available, time series methods may be apprropriate. In effect, past changes from one time to the next are used as a guide to likely future changes.

The modeling of changes over time is in principle a good idea. Do not however put too much faith in the model. A lesson from the recent financial crisis is, surely, that it is unwise to put much faith in any financial model that does not allow for occasional "shocks". The warnings in Taleb (2004) merit attention.

3.2.4 When are rough and ready methods enough?

Rough and ready methods are fine, if they do the job. How does one know whether the job is done?

- Watch for source/target (eg, 2008/2009) differences.
- Allow for effects of model and variable selection.
- For interpretation of individual model parameters, know the traps.
- Are there dependence issues (time series, ...)?

Rough and ready methods may yield useful clues that make a start on gaining understanding in new areas. Greater finesse will almost invevitably be needed in order to make progress once the low-hanging fruit have been harvested.

Work with expression array data provides an example. Leek and Storey (2007) argue that in expression array analyses involving samples from biological organisms, there is commonly a dependence structure that arises because some samples (observations) share common features that are not accounted for by available covariates. This is, for example, likely to be the case for cancer tissue samples. The SVD (singular value decomposition) can provide a low rank approximation that accounts for most of the dependence structure. This leads to changes in the set of sequences that are identified as differentially expressed. The list, and the order, are at the same time more stable with respect to sampling variability.

Automation

As much as is reasonable, it makes sense to automate. Attention can then be focused on those aspects of the investigation that are not susceptible to automation.

Automation can however only be properly effective when the science is well understood. Analysis methodology can be effectively automated, to a smaller or larger extent, once the analysis has been run a number of times with similar data, and results validated. Even then, it is necessary to be open to new insights, or new wrinkles that emerge in the course of the analysis.

4 Challenges from Size of Dataset

Datasets may be large because there are a large number of observations. Or they may be large because there are many variables (features).

4.1 Many Observations

- Additional structure often comes with increased size – data may be less homogeneous, span larger regions of time or space, ...
- Or there may extensive information about not much!
  - e.g., temperatures, at second intervals, for a day.
  - SEs from modeling that ignores this structure may be misleadingly small.
- In large homogeneous datasets, spurious effects are a risk
  - Small SEs increase the risk of detecting spurious effects that arise, e.g., from sampling bias (likely in observational data) and/or from measurement error.

Sheer size brings challenges. In the first place, these are challenges for data management. The analysis challenges that result from huge numbers of observations are not, however, primarily those of scaling up regression and related models for use with large datasets.

The analysis challenges are most important for models with a complex error structure – repeated measures and times series, spatial models, and so on. They are much less important for the use of regression models.

Where a straightforward use of a regression model really does seem appropriate, there can be advantages in structuring the analysis as a series of separate regressions:

- As an example, consider data that have been collected over a non-trivial interval of time. It is then sensible, as a check, to do separate analyses for separate times.
- Where there is no time or other such structure in the data, the analysis can usefully be repeated for separate random samples of the data. Variation in model parameters and predictions under such repeated sampling provides a check on theoretically based estimates of standard errors. If the empirical standard errors are larger, it is those that should be believed.

4.2 Many variables (features)

Huge numbers of variables spread information thinly! Strong assumptions are needed:

(a) most variables have no effect, and/or:

(b) variables act in concert.

Common approaches are to select, or to penalize (eg, $\lambda \|b\|_p$, where $0 \leq p \leq 2$)

- Either way, there are selection effects – with enough lottery tickets, prizes are pretty much inevitable

- A pattern based on the 'best' 15 features, out of 500, may well be meaningless!

Note that any over-fitting with respect to the target is likely to reduce real accuracy!

Part II
Data Summary

Data analysis has as its end point the use of forms of data summary that will convey, fairly and succinctly, the information that is in the data. Considerable technical skill may be required to extract that information.

5 Weighting Effects in Summary Tables

Several examples will be given. Care is required, when summary tables are formed, that the information in the data not misrepresented. The effects of interest were very evident in the `UCBAdmissions` data. In tables of counts, they are associated with Simpson's paradox, alternatively known as the Yule-Simpson effect or, in the genetic context, as epistasis.

5.1 Bias from addition over unequally weighted sub-categories

Here is a contrived example; data are admissions to a fictitious university:

	Engineering		Sociology		Total	
	Female	Male	Female	Male	Female	Male
Admit	10	30	30	15	*40*	*45*
Deny	10	30	10	5	*20*	*35*

Summing over the two separate tables is equivalent, for purposes of calculating overall admission rates, to the following:

Females: $\frac{10}{20} \times \frac{20}{60} + \frac{30}{40} \times \frac{40}{60}$ [0.33 (Eng) : 0.67 (Soc)]

Males: $\frac{30}{60} \times \frac{60}{80} + \frac{15}{20} \times \frac{20}{80}$ [0.75 (Eng) : 0.25 (Soc)]

The Overall Rates are:

- females ($\frac{2}{3}$): bias (0.33:0.67) is towards the Sociology rate (0.75)
- males ($\frac{45}{80}$): bias is (0.75:0.25) towards the Engineering rate (0.5).

Several further examples, of this same general character, appear below.

Simpson's paradox and epistasis

In population genetics, Simpson's paradox type effects are known as epistasis. Most human societies are genetically heterogeneous. In San Francisco, any gene that is different between the European and Chinese populations will be found to be associated with the use of chopsticks! If a disease differs in frequency between the European and Chinese populations, then a naive analysis will find an association between that disease and any gene that differs in frequency between the European and Chinese populations.

Such effects are a major issues for gene/disease population association studies. It is now common to collect genetic fingerprinting data that should identify major heterogeneity. Providing such differences are accounted for, large effects that show up in large studies are likely to be real. Small effects may well be epistatic.

5.1.1 The UCB Admissions Data

Data are admission frequencies, by sex, for the six largest departments at the University of California at Berkeley in 1973. For a reference to a web page that has the details; see the belp page for `UCBAdmissions`. Type

```
> help(UCBAdmissions)     # Get details of the data
> example(UCBAdmissions)  # Example code gives tabular and graphical
```

Note the margins of the table:

```
> str(UCBAdmissions)
```

```
 table [1:2, 1:2, 1:6] 512 313 89 19 353 207 17 8 120 205 ...
 - attr(*, "dimnames")=List of 3
  ..$ Admit : chr [1:2] "Admitted" "Rejected"
  ..$ Gender: chr [1:2] "Male" "Female"
  ..$ Dept  : chr [1:6] "A" "B" "C" "D" ...
```

Notice that what we have is a 3-way table, with margins `Admit`, `Gender` and `Dept`.

Here, we will calculate overall admission rates separately for males and females, admission rates by department separately for males and females, and for each department the number of males and females applying, as a proportion of the total number of the relevant `Gender`. The reasoning is that, if different genders have different departmental preferences, overall admission rates for males will be biased towards admission rates for departments that are popular with males, while overall admission rates for females will be biased towards admission rates for departments that are popular with females.

Two functions that will be important for the calculations are `margin.table()` and `prop.table()`.

- The following are the overall admission rates:

  ```
  > alltab <- margin.table(UCBAdmissions, margin=c(1,2))
  > alltab
  ```

  ```
             Gender
  Admit       Male Female
    Admitted 1198    557
    Rejected 1493   1278
  ```

  Now calculate, for each Gender (margin 2), the proportions admitted and rejected. We require a table with the margin `Gender` (=2). Proportions are calculated across the elements of the remaining margin of the table, which is `Admit`. The proportion admitted provides all the needed information. Hence the restriction to row 1 (`[1, ]`).

  ```
  > round(prop.table(alltab, margin=2)[1, ], 3)
  ```

  ```
    Male Female
   0.445  0.304
  ```

- The following are the admission rates for the different departments. We require a table with margins `Gender` (=2) and `Dept` (=3). Proportions are calculated across the elements of the remaining margin of the table. We require only the proportion admitted. Hence the restriction to row 1 (`[1, , ]`), which at the same time gives a compact table:

  ```
  > round(prop.table(UCBAdmissions, margin=2:3)[1, , ], 3)
  ```

  ```
          Dept
  Gender       A    B     C     D     E     F
    Male   0.621 0.63 0.369 0.331 0.277 0.059
    Female 0.824 0.68 0.341 0.349 0.239 0.070
  ```

- Now calculate the numbers of males and females applying to each department. The margins that we require in the table are `Gender` (=2) and now `Dept` (=3).

  ```
  > (totbydept <- margin.table(UCBAdmissions, margin=c(2,3)))
  ```

```
        Dept
Gender     A   B   C   D   E   F
  Male   825 560 325 417 191 373
  Female 108  25 593 375 393 341
```

Proportions will now be calculated for the margin `Dept` of the table `totbydept` just obtained:

```
> round(prop.table(totbydept, margin=1), 3)
```

```
        Dept
Gender       A     B     C     D     E     F
  Male   0.307 0.208 0.121 0.155 0.071 0.139
  Female 0.059 0.014 0.323 0.204 0.214 0.186
```

Do the data provide evidence, across the University as a whole, of sex-based discrimination?

A relatively small proportion of females (5.9%) applied to department A where admission rates were relatively high, while a high proportion (32.3% and 21.4% respectively) applied to departments C and E where admission rates were relatively low. The very high number of males applying to departments A and B has biased the male rates towards the relatively high admission rates in those departments, while the relatively high number of females applying to departments C, D and F biased the overall female rates towards the low admission rates in those departments. The overall bias arose because males favored departments where there were a relatively larger numbers of places.

What model is in mind? Is the aim to compare the chances of admission for a randomly chosen female with the chances of admission for a randomly chosen male? The relevant figure is then the overall admission rate of 30.4% for females, as against 44.5% for males. Or, is the interest in the chances of a particular student who has decided on a department? The female had a much better chance than a male in department A, while a male had a slightly better chance in departments C and E.

Here, information was available on the classifying factor on which it was necessary to condition. This will not always be the case. In any such tabulation, it is always possible that there is some further variable that, when conditioned on, can reverse or otherwise affect an observed association.

The results that give the overall proportions are, for these data and depending on the intended use, an unsatisfactory and potentially misleading summary. The phenomenon that they illustrate, known as Simpson's paradox or as the Yule-Simpson effect, is discussed in Aldrich (1995); Simpson (1951).

5.2 Analysis of a substantial dataset – US accident data

Each year the National Highway Traffic Safety Administration (NHTSA) in the USA collects, using a random sampling method, data from all police-reported crashes in which there is a harmful event (people or property), and from which at least one vehicle is towed. The data frame `nassCDS` (*DAAG*) is derived from NHTSA data.[6]

The data are a sample, for the years 1997 – 2002. The use of a complex sampling scheme has the consequence that the sampling fraction differs between observations. Each point has to be multiplied by the relevant sampling fraction, in order to get a proper estimate of its contribution to the total number of accidents. The column `weight` (`national` = *national inflation factor* in the SAS dataset) gives the relevant multiplier.

Meyer (2006) argues that on balance (over the period when their data were collected) airbags cost lives. In order to obtain a fair comparison, it is necessary to adjust, not only for the effects of seatbelt use, but also for speed of impact. When this is done, airbags appear on balance to be dangerous, with the most serious effects in high impact accidents, but the effect is at the level of statistical error.

[6] They hold a subset of the columns from a corrected version of the data analyzed in the Meyer (2005) paper that is referenced on the help page for `nassCDS`. More complete data are available from one of the web pages
`http://www.stat.uga.edu/~mmeyer/airbags.htm` (SAS transport file)
or `http://www.maths.anu.edu.au/~johnm/datasets/airbags/` (R image file).

Strictly, the conclusion is that, conditional on involvement in an accident that was sufficiently serious to be included in the database (at least one vehicle towed away from the scene), there is a suggestion that airbags are harmful.

Farmer (2006) argued that these data have too many uncertainties and potential sources of bias to give reliable results when analyzed as will be done here. He presented a different analysis, based on the use of front seat passenger mortality as a standard against which to compare driver mortality, and limited to cars without passenger airbags. In the absence of any effect from airbags, the ratio of driver mortality to passenger mortality should be the same, irrespective of whether or not there was a driver airbag. In fact the ratio of driver fatalities to passenger fatalities was 11% lower in the cars with driver airbags.

5.2.1 From Highly to Mildly Misleading Analyses

The analyses presented here will be for a subset of the data that are further restricted. The oldest vehicles with airbags, represented in these data, were from 1986. In an analysis that does not allow for age of vehicle, this risks biasing results for vehicles without airbags towards results for older vehicles. If there is an adjustment for age of vehicle, vehicles that are older than 1986 do not contribute useful information, for purposes of assessing the effectiveness of airbags. In addition to omitting vehicles older than 1986, observations with weight 0, and one observation where the year of vehicle was unknown. This omits 2726 records out of the total of 26217, leaving 23491 records.

```
> library(DAAG)
> nassnew <- subset(nassCDS, !is.na(yearVeh) & yearVeh>=1986 & weight>0)
```

The following uses `xtabs()` to estimate numbers of front seat passengers alive and dead, classified by airbag use:

```
> library(DAAG)
> (abtab <- xtabs(weight ~ dead + airbag, data=nassnew))
```

```
       airbag
dead          none     airbag
  alive 4357429.74 6614169.17
  dead    29897.41   25919.11
```

The function `prop.table()` can then be used to obtain the proportions in margin 1, i.e., the proportions dead, according to airbag use:

```
> round(prop.table(abtab, margin=2)["dead", ], 4)
```

```
  none airbag
0.0068 0.0039
```

The above might suggest that the deployment of an airbag substantially reduces the risk of mortality. Consider however:

```
> abSBtab <- xtabs(weight ~ dead + seatbelt + airbag, data=nassnew)
> ## Take proportions, retain margins 2 & 3, i.e. airbag & seatbelt
> round(prop.table(abSBtab, margin=2:3)["dead", , ], 4)
```

```
        airbag
seatbelt   none airbag
  none   0.0180 0.0155
  belted 0.0039 0.0021
```

The results are now much less favorable to airbags. The clue comes from examination of:

```
> margin.table(abSBtab, margin=2:3)  # Add over margin 1
```

```
        airbag
seatbelt      none    airbag
  none    916169.2  885635.3
  belted 3471157.9 5754452.9
```

In the overall table, the results without airbags are mildly skewed (~4.12:1.37) to the results for `belted`, while with airbags they are highly skewed (~57.6:8.86) to the results for `belted`.

5.2.2 Taking Account of Estimated Force of Impact

Now take account, additionally, of estimated force of impact (`dvcat`):

```
> ASdvtab <- xtabs(weight ~ dead + seatbelt + airbag + dvcat,
                       data=nassnew)
> ## Use ftable to get a compact, flattened version of the table
> round(ftable(prop.table(ASdvtab, margin=2:4)["dead", , , ]), 6)

                dvcat  1-9km/h    10-24    25-39    40-54      55+
seatbelt airbag
none     none         0.000000 0.002583 0.020300 0.040323 0.204534
         airbag       0.004023 0.004873 0.010982 0.075990 0.269959
belted   none         0.000000 0.000380 0.005743 0.028141 0.139204
         airbag       0.000000 0.000195 0.003331 0.022666 0.157394
```

It will be apparent that differences between `none` and `airbag` are now below any reasonable threshold of statistical detectability.

5.2.3 Changes in the Risk

The package *DAAG* includes the function `excessRisk()`. Run it with the default arguments, i.e. type

```
> excessRisk()

  seatbelt none_dead none_tot airbag_dead airbag_tot noneProp airbagProp
1     none     24067  1366089       13760     885635   0.0176     0.0155
2   belted     15609  4118833       12159    5762975   0.0038     0.0021

  Difference: airbag_dead - none_dead
1                           -1842.471
2                           -9681.088
Differences in expected number of deaths are calculated
relative to the level 'none' of the factor 'airbag'.
```

Here are several exercises.

1. Classify according to `dvcat` as well as `seatbelt`. All you need do is add `dvcat` to the first argument to `excessRisk()`. What is now the total number of excess deaths?
   [The categories are 0-9 kph, 10-24 kph, 25-39 kph, 40-54 kph, and 55+ kph]
2. Classify according to `dvcat`, `seatbelt` and `frontal`, and repeat the calculations. What is now the total number of excess deaths?

Explain the dependence of the estimates of numbers of excess deaths on the choice of factors for the classification.

5.2.4 More Variables Still

There are at least two other variables that may affect the risk of death. These are the year of manufacture of the vehicle, and the age of the occupant. Possibly also the year of the accident might be important, but the data do not have enough information to allow this effect to be modeled in addition to all the others. These will be investigated in Subsection 5.2.

5.3 Summary of Continuous Outcome Data

Unequal subgroup weights create exactly the same potential, as with binary (or categorical) outcome data, for misleading summary.

Unequal subgroup weights with continuous data – an example

Figure 5.3 relates to data collected in an experiment on the use of painkillers.[7]. Notice that the overall comparison (average for baclofen versus average for no baclofen) goes in a different direction from the comparison for the two sexes separately.

Researchers had been looking for a difference between the two analgesic treatments, without and with baclofen. When the paper was first submitted for publication, an alert reviewer spotted that some of the treatment groups contained more women than men, and proposed a re-analysis to determine whether this accounted for the results.[8] When the data were analysed to take account of the gender effect, it turned out that the main effect was a gender effect, with a much smaller difference between treatments.

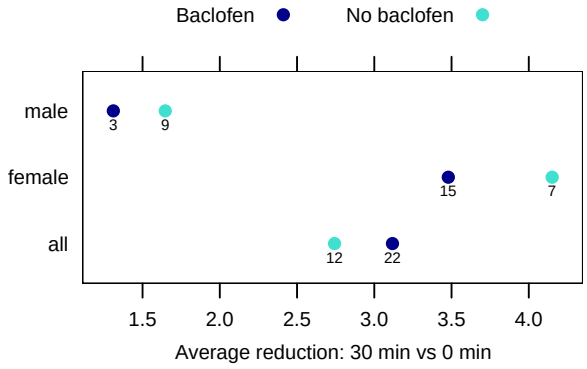


Figure 6: Does baclofen, following operation (additional to earlier painkiller), reduce pain? Subgroup numbers, shown below each point in the graph, weight the overall averages when sex is ignored.

The overall averages in Figure 5.3 reflect the following subgroup weighting effects:

Baclofen: 15f to 3m, i.e. $\frac{15}{18}$ to $\frac{3}{18}$ (a little less than f average)
No baclofen: 7f to 9m, i.e. $\frac{7}{16}$ to $\frac{9}{16}$ ($\approx \frac{1}{2}$-way between m & f)

This is still only part of the story. More careful investigation revealed that the response to pain has a different pattern over time. For males, the sensation of pain declined more rapidly over time.

Strategies

(i) Simple approach Calculate means for each subgroup separately.
Overall treatment effect is average of subgroup differences.
Effect of baclofen (reduction in pain score from time 0) is:

Females: 3.479 - 4.151 = -0.672 (-ve, therefore an increase)

Males: 1.311 - 1.647 = -0.336

[7]Gordon, N. C. et al.(1995): "Enhancement of Morphine Analgesia by the GABAB against Baclofen". Neuroscience 69: 345-349

[8]Cohen, P. 1996. Pain discriminates between the sexes. New Scientist, 2 November, p. 16.

Average over male and female = -0.5 × (0.672+0.336) = -0.504

(ii) Fit a model that accounts for sex and baclofen effects y = overall mean + sex effect + baclofen effect + interaction
(At this point, we are not including an error term).

Why specify a model?

It makes assumptions explicit.

5.3.1 Cricket – Runs Per Wicket:

	1st innings		2nd innings		Overall	
	Runs	Wickets	Runs	Wickets		
Bowler A	40	4	240	6	280	10
Bowler B	70	5	50	1	120	6

Table 1: Runs per wicket for each bowler in the two innings.

The runs per wicket are:

	1st innings	2nd innings
Bowler A	10.00	40.00
Bowler B	14.00	50.00

Table 2: Runs per wicket for each bowler in the two innings.

Observe that although Bowler A does better than bowler B in each innings, his overall average is worse – 28 runs per wicket as opposed to 20.

A fair way to make the comparison is to model the effects both of bowler and of innings, using a linear model.

5.4 Further Examples

5.4.1 Do the left-handed die young

A number of papers, in Nature, in the psychological literature and in the medical literature, have argued that left-handed people have poorer survival prospects than right-handers. It turns out that, in a large cross-sectional sample of the British population that was studied in the 1970s, the proportion of left-handers declined from around 15% for ten-year-olds to around 5% for 70-year olds. If average age at death is compared between left-handers and right-handers, left-handers will be over-represented among those dying young, and over-represented among those dying in older years. Hence the average age will be be lower for left-handers than for right-handers. Disturbingly it has been easier to get this nonsense published than to get refutations published.

Again survival analysis methods are required for a proper analysis. Once the effect noted above has been removed, there may be a small residual effect from left-handedness. See Bland & Altman (2005).

5.4.2 Hormone replacement therapy

Cohort and other population based studies have suggested hormone replacement therapy (HRT) reduces the risk of coronary heart disease (CHD). A large meta-analysis of what were identified as the best quality observational studies found a relative reduction in risk of 50% from any use of HRT.

A large randomized controlled trial found an increase in hazard, from use of CRT, of 1.29 (95% CI 1.02–1.63), after 5 years of follow-up. Thus, so far from reducing CHD risk, it increases the risk. The conclusion given in a 2006 ABC Health Report interview is that:

> Hormone therapy, both oestrogen combined with progesterone and oestrogen alone, increase risk of cardio vascular disease, stroke, blood clots and the hormone therapy that was combined meaning oestrogen and progesterone increase risk of breast cancer.
> [This is taken from: `http://www.abc.gov.au/rn/healthreport/stories/2006/1530042.htm`]

This was an especial puzzle because the results of the observational studies have been consistent with the results of randomized trials for other outcomes – breast cancer (increased risk for the combined oestrogen/progesterone HRT; for a 50-year old from 11 in 1000 to maybe 15 in 1000), colon cancer (reduced risk), hip fracture (reduced risk, but diet, exercise and other drugs can achieve the same or better results) and stroke (increased risk; for a 50-year old from 4 in 1000 to 6 in 1000). See the ABC web page just noted and, e.g., Rossouw et al. (2002) for further details and references.

A recent analysis by Hérnan et al. (2008) of the observational data gave the following factors by which the average risk is multiplied: These effects are assumed to add.

Years of follow-up	0 - 2	>2 - 5	>5
Multiply risk by	1.5	1.3	0.67
Years since menopause	<10	10 - 20	>20
Multiply risk by	0.89	1.24	1.65

The observational data included some individuals with long follow-up times, whereas the nature of a randomized trial (after randomization, there is a limited follow-up time) rules out long follow-up times. Moreover, in order to make up numbers, the randomized trials included many women with long times following menopause. Both these factors increase the average estimated risk for the randomized trials, relative to the observational data. The analysis will appear later this year, in a paper in the journal *Epidemiology*.

In part, the issue is that both the randomized trials and the observational studies yielded averages for populations that were heterogeneous in ways that gave different relative weights to relevant sub-populations. Earlier analyses failed to identify important relevant covariates.

5.5 Biases from omission of features – further comments

Some of the possibilities that it may be necessary to contemplate, for this specific example and more generally, are:

1. The issue is one of design of data collection, as well as analysis. If information has not been collected on relevant variables, the analyst cannot allow for their effect(s).

2. If the data are observational, there may be crucial variables on which it is impossible to collect information. Or there may be no good understanding of what the relevant variables are.

3. Providing the problem is understood and handled appropriately, large effects are unlikely, in large data sets, to arise from differences between sub-populations.

4. Small effects are highly likely, and should always be treated with scepticism. Small effects that are artefacts of the issues noted here show up more readily than small effects that are genuine. This is because the effects that will be noted here will almost invitably skew estimates of genuine effects, either exaggerating the effect or (just as likely) reversing the direction of its apparent effect.

Part III
Populations, Samples & Sample Statistics

6 Populations and Samples

The available data rarely comprises a total population. At best, it is likely to be a sample, preferably a random sample in which all population values appear with equal probability, from the population.

This is likely to be true even if the sample comprises all the data that were available at the time. Results will typically be applied in some new context, later in time, where the available data have changed. At best, changes between the original data and the later point in time for which predictions will be made will be rather similar to changes between one sample and another. This is however a best case scenario. Commonly there will be changes similar to those between one sample and another, plus systematic changes in time.

Thus a bank will have, in principle at least, complete information on financial transactions with current customers. As a guide to future financial transactions for those same customers, this is a sample of customer behavior that may or may not be a good guide to future transactions.

6.1 Samples

The function `sample()` takes samples from a set of data values. Samples may be taken without (the default) or with replacement. In without replacement sampling from a set $\{x_i, i = 1, \ldots n\}$ each element x_i can appear at most once in the sample.

The R function `sample()` models what should happen when a random sample is taken. Each successive value is chosen at random from the values that remain in the population, indepndently of the values previously chosen. In without replacement sampling (the default for `sample()`) from a finite population, values that are removed are not available for selection when the next sample value is chosen. In with replacement sampling, the same value can be selected any number of times.

```
> sample(1:10, size=5)                # Yields 5 distinct values
> sample(1:10, size=5, replace=TRUE)  # Values can be repeated
```

An important use of with replacement sampling is for bootstrap sampling. Here, the sample is usually chosen to be of the same size (i.e., the same number of values) as the set of values from' which the sample is taken.

6.2 Continuous distributions

6.2.1 Contexts in which continuous distributions appear

The following are some of the contexts in which it may be useful to characterize and/or compare continuous distributions:

- Before fitting a classification model, it is desirable to do exploratory analyses that compare the groups with respect to both discrete and continuous variables.
- For regression modeling, various prior checks are desirable on dependent as well as on explanatory variables.
- Categories will in some instances be formed by discretizing a continuous variable. Where possible, comparisons on the continuous scale should precede or accompany the discrete comparisons.
- For each category A, suppose that p_A is the probability, assessed independently of the data for an observation, that an observation belongs to category A. Many classification algorithms model $\log(p_A/(1-p_A))$, as a function of the explanatory factors and variables. The distribution of $\log(p_A/(1-p_A))$ is then of interest.

6.2.2 Empirical vs theoretical distributions

Consider now data that give the heights of 118 female students attending a first year statistics class at the University of Adelaide. Figure 7 plots a histogram and overlays it with a density plot. (The parameter setting `prob=TRUE` for the histogram is needed so that the units on the vertical scale are the same both for the histogram and for the density plot.) Vertical bars above the x-axis give the positions of the actual points. The function `na.omit()` omits missing values.

The vertical scale is chosen so that multiplying the height of each rectangle by the width of its base (5cm in each case) gives an estimate of the proportion of data values in that range. The same scale is used for the density plots, except that the density now changes continuously. It estimates, at each point, the proportion of values per unit interval.

```
> library(MASS)             # MASS has the survey data set
> library(lattice)
> heights <- na.omit(survey[survey$Sex=="Female", "Height"])
> ## NB: For consistency with the density plot, the vertical scale
> ## for the histogram must be a density scale (freq=FALSE),
> hist(heights, freq=FALSE,
         xlab=paste("Heights (cm)"),
         main="", cex.axis=1.25, cex.lab=1.25)
> lines(density(heights))
> ## Show data values along the x-axis
> rug(heights, side=1)
```

The data set `survey` is included with the `MASS` package.

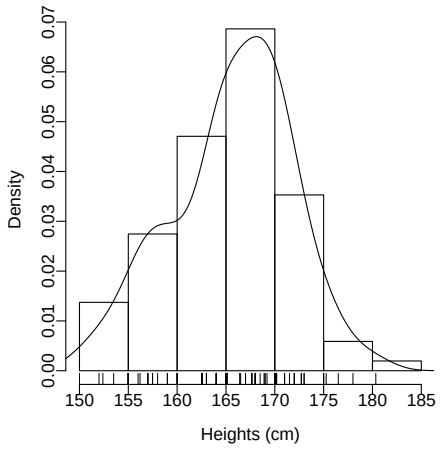


Figure 7: Vertical bars along the axis show the heights of 118 female students in a first year statistics class at the University of Adelaide. Alternative summaries of the distribution are a histogram, the overlaid density plot (solid curve), and a fitted normal curve (dashed).

Observe that:

- The vertical bars that show the distribution of data values are not very informative.
- The top of each histogram estimates the relative number of points (students) per unit along the x-axis, within the class boundaries of that histogram. That estimate changes suddenly at the class boundaries; this is an unsatisfactory feature of the histogram.
- The density curves give smooth estimates of the relative number of points (students) per unit along the x-axis. This is much preferable. However there is still an issue of the choice of bandwith for the smoother. This corresponds to the need to choose, for the histogram, the class width.
- The solid curve is a density estimate that makes very limited assumptions about the population density. The appearance of the curve will, depending on the sample size, be quite strongly

affected by sampling variation. Try repeating the plot with random samples of size 102 (the same size as the sample of Adelaide students) from the normal distribution.

- The dashed curve makes the strong assumption that the population distribution is normal.

Histograms are almost never used for formal inference, i.e., for reaching precisely formulated conclusions about the population from which the data have come. At best, they give a rough indication of the population distribution. To give more than a very crude indication of the population distribution, the sample size must however be large, perhaps several hundred. A density curve does somewhat better. Even so, it is easy to over-interpret density curves from small samples.

6.2.3 Boxplots, and the inter-quartile range:

The boxplot is a widely used summary of a distribution of data values that focuses on key features only. Visual comparison of boxplots is much easier than visual comparison of density plots.

Figure 8 shoes such a plot. The key features are:

- an upper "whisper"; points larger than this are plotted separately; they are identified as possible outliers.
- the upper quartile, dividing off the lower 75% of the distribution from the upper 25%.
- the median, dividing off the lower 50% of the distribution from the upper 50%.
- the lower quartile, dividing off the lower 25% of the distribution from the upper 75%.
- a lower "whisper"; points smaller than this are plotted separately; they are identified as possible outliers.

The central rectangular box thus extends from the lower quartile to the upper quartile, and takes in the central 50% of the data.[9]

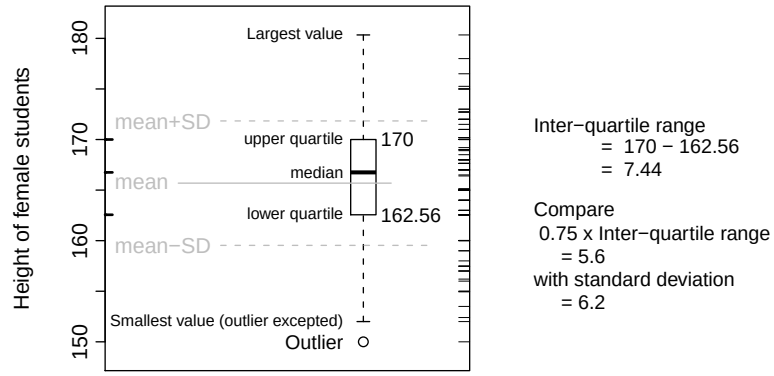


Figure 8: Boxplot, with annotation that explains boxplot features. Lines in gray show mean-SD, mean, and mean+SD. Data are heights of 118 female students in a first year statistics class at the University of Adelaide.

For a normal distribution, around one point in 100 can be expected to lie below the lower whisker, or above the upper whisker. Do not too readily identify points as outliers. If the distribution is skewed to the right, i.e. values are more spread out above than below the median, some points are likely for

[9]The range from the lower quartile to the upper quartile is known as the inter-quartile range, abbreviated to IQR.

this reason to appear beyond the upper whisker. The present data are slightly skewed to the left, and it is therefore not surprising that one point has appeared below the lower whisker.

The following code reproduces a simplified version of Figure 8:

```
> attach(survey)
> boxplot(Height[Sex=="Female"])
> detach(survey)
```

6.2.4 Samples from an empirical distribution

Here, the interest is in taking repeated with replacement samples, or *bootstrap* samples, from a sample that is the immediate subject of investigation. In with replacement sampling, each sampled element is placed back in the population before taking the next element. This is equivalent to sampling without replacement from the infinite population obtained by specifying a uniform distribution on the sample values. Try

```
> sample(1:8, size=5)          # replace=FALSE
> sample(1:8, replace=TRUE)    # permutes the original data
>   # By default, as there were 8 values, size=8
> sample(c(2,8,6,5,3), replace=TRUE)
```

6.2.5 How accurate is the density curve or boxplot?

Resampling can be a good way to get an indication of the accuracy of a density curve or boxplot. Take repeated random with replacement samples, of the same size as the initial sample, from the one available sample. Fit a density curve to each such sample. The differences between these density curves should give a good idea of the range of variation that could be expected in repeated samples from the population.

We will take five bootstrap samples from the data on Adelaide University statistics students. Or equivalently, take one sample of five times the size of the source sample, and split it into five.

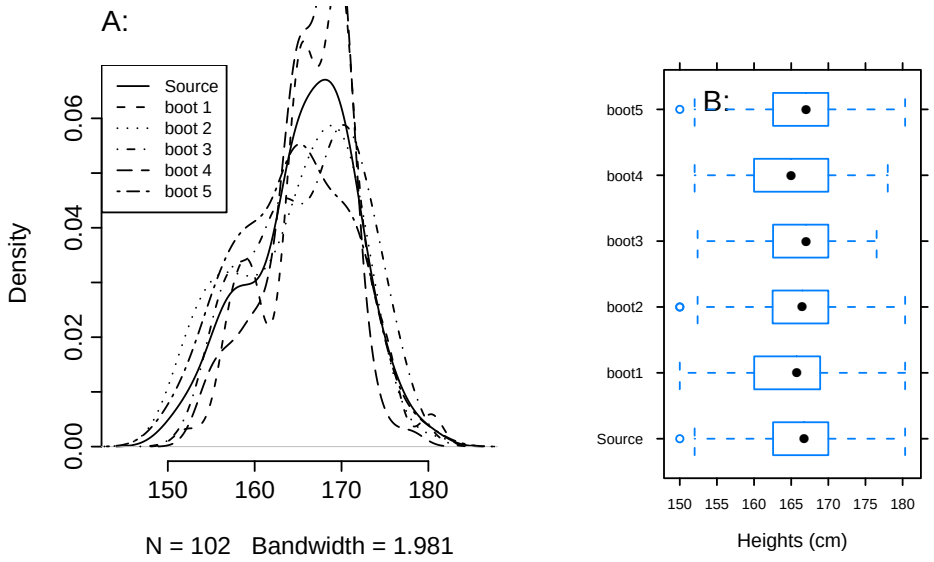


Figure 9: Panel A shows heights (cm) of female 1st year Adelaide University statistics students. The density curve from the sample is shown with a solid line, while the 5 bootstrap samples are shown with dashed lines. Panel B shows boxplot representations of the data and the same five bootstrap samples.

Here is the code

```
> library(MASS)             # MASS has the survey data set
> library(lattice)
> heights <- na.omit(survey[survey$Sex=="Female", "Height"])
> sampsize <- length(heights)
> par(fig=c(0,0.65, 0,1))
> plot(density(heights), main="", bty="n")
> xdf <- matrix(0, ncol=6, nrow=sampsize)
> xdf[,1] <- heights
> leg <- c("Source", paste("boot", 1:5))
> for(i in 2:6){xb <- sample(heights, replace=TRUE)
                  xdf[,i] <- xb
        lines(density(sample(heights, replace=TRUE)), lty=i, xpd=TRUE)
                  }
> legend(x="topleft", legend=leg, lty=1:6, cex=0.75)
> mtext(side=3, line=1.0, "A:", adj=0)
> par(fig=c(0.65,1, 0,1))
> mtext(side=3, line=1.0, "B:", adj=0)
> sampleID <- factor(rep(c("Source", paste("boot", 1:5, sep="")),
                         each=sampsize))
> xdf <- data.frame(x=as.vector(xdf), Sample=sampleID)
> bplt <- bwplot(Sample ~ x, data=xdf,
                  xlab="Heights (cm)", auto.key=list(columns=3),
                  par.settings=simpleTheme(lty=c(1, rep(2,5)))
                  )
> print(update(bplt, scales=list(tck=0.5)), pos=c(0.6,0,1,1), newpage=FALSE)
```

6.2.6 Use of the sample to make inferences about the population

For reaching conclusions about the population from which the data have come there are two common approaches:

1. Resampling approaches work with the actual data values.
2. Reasoning may proceed as though the population distribution is normal, i.e., use the dashed density curve.

Proceeding as though the population distribution is normal is fine provided that

- Inferences will be based on the sample mean.
- The population distribution is not too far from normal. (NB: Greater deviations from normality can be tolerated for larger sample sizes).

6.3 Theoretical probability distributions

This subsection will introduce mathematical and computing tools that are important for working with theoretical distributions. As has been hinted, the normal distribution has a particular importance.

Models that are commonly used for population distributions include the normal (heights, weights and other morphometric measures, preferably on a logarithmic scale), exponential (lifetimes of components, where the probability of failure is unchanged over time), uniform, binomial (number of female children in a family of size N), and Poisson (failures in some fixed time interval, where the probability of failure is unchanged over time). Even if not a completely accurate model, one of these distibutions may be a reasonable starting point for investigation.

6.3.1 Mathematical definition

A probability distribution for a random variable X that has a subset of the real line as support, defines for all x_1 and x_2 in its support:

$$\Pr[x_1 < X \leq x_2].$$

In a discrete population, each value has a probability (or probability mass) associated with it. In a continuous population, each value x has an associated density $f(x)$, such that for any two values a and b in the support of $f()$,

$$\Pr[a < X <= b] = \int_a^b f(x)dx$$

A very important continuous distribution is the normal. This has

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{(x-\mu)^2}{2\sigma^2})$$

where μ is the mean, and σ^2 is the variance.

Set $Z = \frac{X-\mu}{\sigma}$. Then if X is distributed as normal with mean μ and variance σ^2, Z has a standard normal distribution, i.e.

$$f(z) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{x^2}{2})$$

Mean and variance will be defined shortly, in Subsection 6.3.3. In a symmetric distribution such as the normal, the mean lies at the axis of symmetry. The square root of the variance, which is the standard deviation, is a measure of variability about the mean.

Note: More generally, a distribution may have both continuous and discrete components. Any discrete components are commonly called *probability masses* or *spikes*.

For example, modeling of the distribution of 1978 income in the data frame `nswdemo` (*MASS*) requires a spike at 0.

6.3.2 Density Curves and Cumulative Distribution Functions

These may be defined either by a density function, or by a cumulative distribution curve.

```
> curve(pnorm(x), from = -3, to = 3)
```

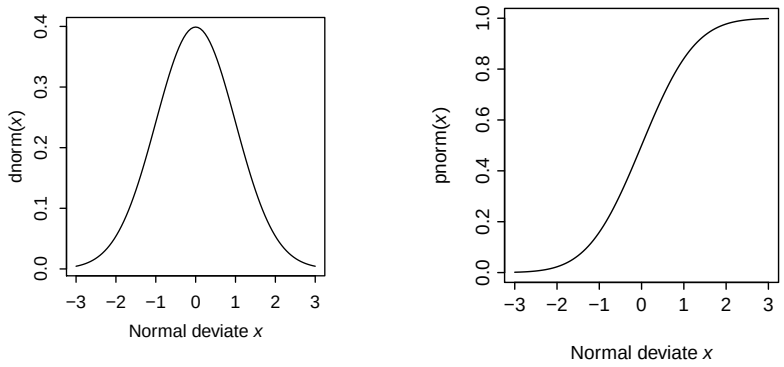


Figure 10: Normal density curve, with cumulative distribution function alongside.

The following plots the density of a normal distribution with a mean of 0 and SD=1:

```
> curve(dnorm(x), from = -3, to = 3)
```

Why were the limits for the curve taken to be -3 and 3?

The height of the curve is the probability density. For a small interval of width h including the point, the probability is

$$h \times \text{normal density}$$

The area under the curve between $x = x_1$ and $x = x_2$ is the probability that the random variable X will lie between $x = x_1$ and $x = x_2$.

Cumulative probability curves The following plots the cumulative probability curve of a normal distribution with a mean of 0 and SD=1 (these are the defaults):

The ordinates of the cumulative density curve give the cumulative probabilities, i.e., the height of the curve at x is $\Pr[X \le x]$. It follows that

$$\Pr[x_1 < X \le x_2 = \Pr[X \le x_2] - \Pr[X \le x_1]$$

Thus, suppose that X has a normal distribution with a mean of 0 and a standard deviation equal to 1. The probability that X is between -1 and 1 can be calculated as:

```
> pnorm(1) - pnorm(-1)

[1] 0.6826895
```

6.3.3 The mean and variance of a population

See Section 22 for the definition of the expectation of a random variable. The population mean is

$$\mu = \mathrm{E}[X] = \int x f(x) dx$$

while the variance is

$$\sigma^2 = \mathrm{E}[(X - \mu)^2] = \int (x - \mu)^2 f(x) dx$$

6.3.4 Samples from a population – R functions

Unless stated otherwise, "sample" will mean "simple random sample".

The R functions `rnorm()` (normal), `rexp()` (exponential), `runif()` (uniform), `rbinom()` (binomial), and `rpois()` (Poisson), all take samples from infinite distributions.

```
> rnorm(n=10, mean=11, sd=2)
> rnorm(n=10)
> runif(n=10)
```

6.4 Displaying the distribution of sample values – some further comments

Examination of a the sample distribution may allow an assessment of whether the sample is likely to have come, e.g., from a normal population distribution. There is an art to making this comparison. In the sequel, some of the different ways in which the comparion might be made will be investigated.

6.4.1 Estimated density curve – the choice of bandwidth

Earlier, in Figure 7, we fitted a density curve to the distribution of heights of 118 female students attending a first year statistics class at the University of Adelaide. We now continue that discussion.

First, repeat the plot with a wider smoothing window, or bandwidth. In the figure, I've added marks on the horizontal axis that show the actual heights. Also marked off, in gray lines, are the mean, mean-SD and mean+SD.

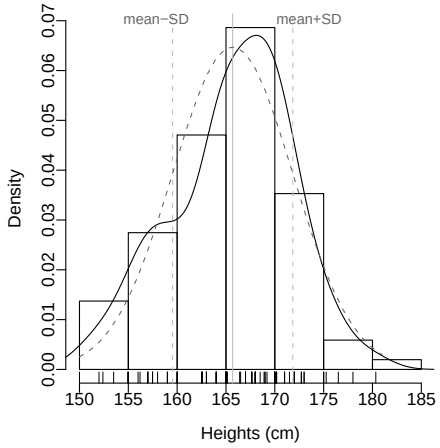


Figure 11: Density plot, now with a larger smoothing window (`bw`) and with a gaussian (normal) kernel, showing the distribution of heights of 118 female students in a first year statistics class at the University of Adelaide. A normal density curve has been added. Marks on the horizontal axis show the actual heights. Also marked off, in gray lines, are the mean, mean-SD and mean+SD.

Here is the code

```
> heights <- na.omit(survey[survey$Sex=="Female", "Height"])
> ## NB: The vertical scale for the histogram must be a density scale.
> ## for consistency with the density plot.
> ## bw is the bandwidth of the smoother, in x-axis units
> heights <- na.omit(survey[survey$Sex=="Female", "Height"])
> ## NB: For consistency with the density plot, the vertical scale
> ## for the histogram must be a density scale (freq=FALSE),
> hist(heights, freq=FALSE, main="", , cex.axis=1.25, cex.lab=1.25,
        xlab=paste("Heights (cm)"))
> lines(density(heights))
> ## Show data values along the x-axis
> rug(heights, side=1)
> av <- mean(heights); sdev <- sd(heights)
> abline(v=av, col="gray")
> abline(v=av-sdev, col="gray", lty=2)
> abline(v=av+sdev, col="gray", lty=2)
> xval <- pretty(heights, n=40)
> den <- dnorm(xval, mean=av, sd=sdev)
> lines(xval, den, col="gray40", lty=2)
> ytop <- par()$usr[4]-0.25*par()$cxy[2]
> text(av-sdev, ytop, adj=0.75, labels="mean-SD", col="gray40", xpd=TRUE)
> text(av+sdev, ytop, adj=0.25, labels="mean+SD", col="gray40", xpd=TRUE)
```

Note: If data have a sharp lower or upper cutoff (a sharp lower cutoff at zero is common), parameters `from` and/or `to` can be set to ensure that this sharp cutoff is reflected in the fitted density.

Exercise: Draw a random sample of size 20 from an exponential distribution with `rate = 1`. Plot an estimated density curve.

6.4.2 Normal and other probability plots

Although preferable to histograms, density plots are not in general an ideal tool for judging whether the sample is likely to have come from one or other theoretical distribution, most often the normal distribution. The appearance depends too much on the choice of bandwith. They do provide visual cues that might be used to identify differences from the theoretical distribution, but those clues and hard to interpret.

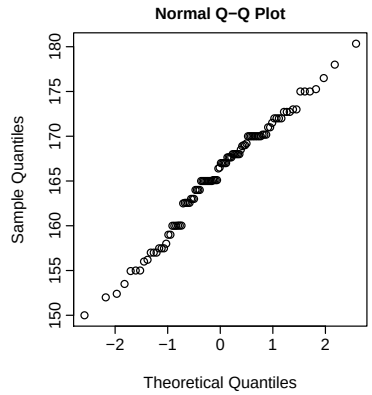


Figure 12: Normal probability plot for the distribution of heights of 118 female students in a first year statistics class at the University of Adelaide.
Code is

```
> y <- with(survey, na.omit(Height[Sex=="Female"]))
> qqnorm(y)
```

A much better tool is the Q-Q plot, which is a form of cumulative probability plot. Here, the focus will be on the comparison with a normal distribution, and the relevant Q-Q plot is a normal probability plot, using the function `qqnorm()`. Figure 12 shows a normal probability plot for the distribution of heights of the 118 female students in a first year statistics class at the University of Adelaide.

If data are from a normal distribution, points should lie close to a line. For a small sample size, quite large deviations from a line can be accepted. If the sample is large, points should lie close to a line. It is useful to draw repeated Q-Q plots with random samples of the same size from a normal distribution, in order to calibrate the eye. The function `qreference()` from the `DAAG` package may be useful for this purpose. For example:

```
> y <- na.omit(survey[survey$Sex=="Female", "Height"])
> qreference(y, nrep=6)
```

6.4.3 A further note on density estimation – controlling smoothness

We can control the smoothness of the density plot. There are various ways to do the smoothing. By default, with a normal "kernel", a mixture of normal densities is used.

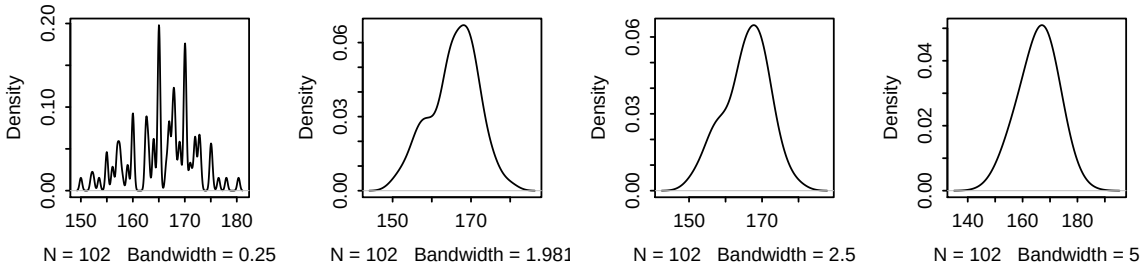


Figure 13: Density curves for Adelaide female student heights. Curves are shown for three different choices of bandwidth: 0.25, 1.98 (the default for these data), 2.5 and 5.0. The normal kernel (the default) is used in each case, so that increasing the bandwidth forces the curve closer to normal.

Increasing the bandwidth makes the estimated density more like the density that is used as the kernel. Thus increasing the bandwith, with a `"gaussian"` kernel, is alright providing that the sample really is from a normal distribution. Figure 13 shows, for the Adelaide female student data, the effect of varying the bandwidth. The default bandwith usually gives acceptable results. Experimentation with different choices of bandwidth is sometimes insightful. The code is:

```
> plot(density(rnorm(50), kernel="rectangular", bw=0.5), type="l")
> plot(density(runif(50), kernel="rectangular", bw=0.5), type="l")
> plot(density(runif(50), kernel="gaussian", bw=0.5), type="l")
```

7 Sample Statistics and Sampling Distributions

7.1 Variance and Standard Deviation:

In a sample, the *variance* is the average of the sum of squares of the deviations from the mean. If n is the sample size then, to correct for the fact that deviations are measured from the sample mean (rather than from the true mean), the sum of squares of deviations from the mean is usually divided by $n-1$. Thus, given sample values $x_1, x_2, \ldots, x_n$, the usual estimate of the variance σ^2 is

$$\widehat{\sigma^2} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}$$

Why divide by $n-1$ rather than by n. A sample of one gives no information on the variance. Every value additional to the first gives one additional piece of information.

The standard deviation (SD) is the square root of the variance. The standard deviation is widely used, both in statistical theory and for descriptive purposes, as a measure of variability. The most obvious intuitive interpretations of the SD assume a normal population, or a random sample from a normal population. If data are from a normal population, then 68% of values will on average be within one standard deviation either side of the mean.

A key idea is that sample statistics have a sampling distribution – the distribution of values that would be observed from repeated random samples. This is an idea that will be illustrated in laboratory exercises.

Sample survey theory is one of several areas where there has been a strong tradition of basing all inferences on variances. This works well when inferences are mostly for means or totals and samples are large. The reason for this will become apparent below, in the discussion of the sampling distribution of the mean. There are however important small sample applications where it does not work well, and sample survey analysts are now moving away from the former relatively exclusive reliance on variance based inferences.

7.2 The Standard Error of the Mean (SEM):

The standard deviation estimates the variability for an individual sample value. This variability does not change (though the estimate will) as the sample size increases. On the other hand, the sample mean does become less susceptible to variability as the sample size increases. If σ is the standard deviation then, for a random sample, the standard error of the mean is $\sigma/\sqrt{n}$.

Here are calculations that give, for the student heights, the mean, the standard deviation and the standard error:

```
> attach(survey)
> y <- na.omit(Height[Sex=="Female"])
> sd(y)

[1] 6.151777

> sd(y)/sqrt(length(y))

[1] 0.6091167

> detach(survey)
```

The standard error of the mean is, with a sample of 118, less than a tenth the size of the standard deviation. This result relies crucially on the i.i.d. assumption. This will be an important issue for multi-level models.

7.3 The sampling distribution of the mean:

We have just one sample, and therefore just one mean. The standard error of the mean relates to the distribution of means that might be expected if multiple samples (always of size 118) could be taken from the population that provided the sample.

It is however possible to simulate the taking of such repeated samples. There are two ways that this sampling distribution might be obtained:

1. As the sample distribution seems close to normal, the use of repeated samples of size 118 from a normal distribution may be reasonable. This simulation might assume a a mean of 165.69, as for the sample, and an SD of 6.15 as for the sample.

2. An alternative is to take repeated samples, with replacement, from the original sample itself. This is equivalent to sampling from a population in which each sample value is repeated an infinite number of times. if no use is made of theoretical results or approximations, this repeated sampling from the one available sample is the best approximation that is available to repeated sampling from the original population.

Figure 14A was obtained using the simulation approach, while Figure 14B used the bootstrap approach. In Panel A, the normal density function from which the samples are taken is in gray. In Panel B, the estimated density function for the sample in gray.

Alternatively, and

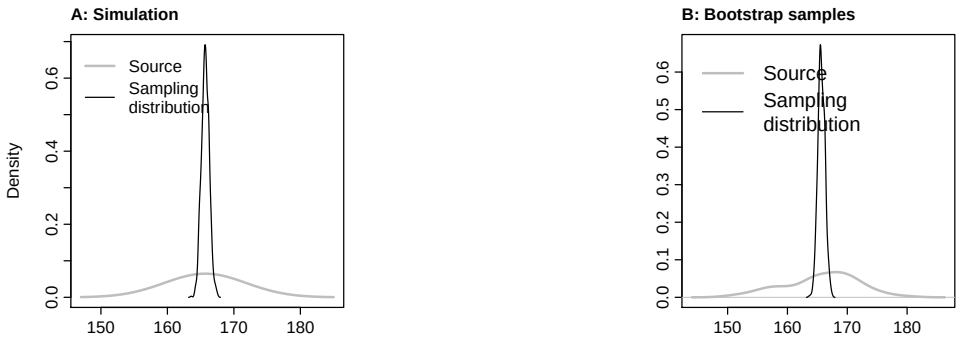


Figure 14: Panel A shows a density plot for a simulated distribution of the mean, for samples of size 118 from a normal distribution with mean=165.7 and SD=6.15, as for the sample of University of Adelaide students. Panel B shows the estimated sampling distribution when bootstrap samples of size 118 are taken from the sample of University of Adelaide students.

Code for Panel A (simulation) is:

```
> av <- numeric(1000)
> for (i in 1:1000)
        av[i] <- mean(rnorm(118, mean=165.69, sd=6.15))
> avdens <- density(av)
> xval <- pretty(c(165.69-3*6.15, 165.69+3*6.15), 50)
> den <- dnorm(xval, mean=165.69, sd=6.15)
> plot(xval, den, type="l", xlab="", ylab="Density",
        ylim=c(0,max(avdens$y)), col="gray", lwd=2)
> lines(avdens)
> title(main="A: Simulation", adj=0)
> legend("topleft", legend=c("Source", "Sampling\ndistribution"),
          col=c("gray", "black"), lty=c(1,1), lwd=c(2,1), bty="n")
```

Code for Panel B (the bootstrap) is:

```
> av <- numeric(1000)
> for (i in 1:1000)
        av[i] <- mean(sample(y, size=length(y), replace=TRUE))
> avdens <- density(av)
> plot(density(y), ylim=c(0, max(avdens$y)), xlab="", ylab="",
        col="gray", lwd=2, main="")
> lines(avdens)
> title(main="B: Bootstrap samples", adj=0)
> par(oldpar)
> legend("topleft", legend=c("Source", "Sampling\ndistribution"),
          col=c("gray", "black"), lty=c(1,1), lwd=c(2,1), bty="n")
```

The sampling distribution for the mean looks, in the company of the distribution of sample values. like a veritable Eiffel tower!

A practical consequence of the Central Limit Theorem is that the sampling distribution will for a sample of this size be much the same (close to a normal distribution) irrespective of the distribution from which the sample is taken, providing that the distribution is roughly symmetric and not unduly spread out in the tails. Try the following, which takes samples from a uniform distribution on the interval (0,1):

```
> par(mfrow=c(1,2))
> av <- numeric(1000)
> xval <- pretty(c(-.5, 1.5), 500)
> plot(xval, dunif(xval), type="l")
> for (i in 1:1000) av[i] <- mean(runif(n=118))
> plot(density(av))
> par(mfrow=c(1,1))
```

All statistics have sampling distributions. For example, there is a sampling distribution for the median. Unlike the distribution of the mean, this is strongly affected by the distribution from which the sample is drawn. Coefficients in linear or other models have sampling distributions.

Elegant simulations that demonstrate the Central Limit Theorem can be viewed at `http://animation.yihui.name/prob:central_limit_theorem` and `http://animation.yihui.name/prob:law_of_large_numbers`. The R code for running theses simulations, using the package *animation*, is available from theses same web pages.

Exercise 1: Try varying the sizes of the samples for which the averages are calculated. Even with `n` as small as 5 or 6, the distribution will be quite close to normal. Try also varying the number of samples that are taken. Taking some number of samples greater than 1000 will estimate the distribution more accurately; with fewer samples the estimate will be less accurate.

Exercise 2: Repeat, but now sampling from: (a) a uniform distribution, and (b) an exponential distribution.

8 Accuracy Assessment

Having trained a model, we would like to know how well the model has performed. If model A performs better than model B we will, other things being equal, prefer model A.

An ideal is that predictions should be accurate for test data that accurately reflect the context in which the model will be used. Accuracy measures that are widely used are:

1. For a regression model with a continuous outcome, define the prediction error to be the difference between the model prediction and the observed value. The root mean square prediction error is then a measure of accuracy.

2. For a classification model, the percentage of correct classifications is often a suitable measure of accuracy. The deviance, or another "information" measure may by used, in some computational and theoretical contexts, as a proxy for percentage of correct classifications.

In practice predictive accuracy is commonly assessed, using the above or other accuracy measures, for the same population from which the sample is derived. Assessment of the extent to which results are relevant to the target population is then a matter for separate investigation. This is a key issue, that is too often ignored.

The initial focus will be on model accuracy, which is an over-riding requirement. This will be followed by discussion of the accuracy of parameter estimates. Accuracy of parameter estimates has additional complications, beyond those involved in assessing accuracy of model predictions.

Elegant simulations that demonstrate the bootstrap and cross-validation be viewed by going to `http://animation.yihui.name/dmml:start`. The R code for running these simulations, using the package *animation*, is available from the web pages for the individual animations.

8.1 Mechanisms for assessing predictive accuracy

Mechanisms that can be used for such assessment include:

1. Derivation of a theoretically based estimate, e.g., for the error mean square for an `lm()` linear model.

2. The training/test set approach, using a random split into training and test set.

3. Cross-validation, in which each of k parts of the data become in turn the test set, with remaining data ($k-1$ out of k parts) used for training.

4. Bootstrap approaches can be used in much the same way as cross-validation, c.f., the approach used by the *randomForest* package. Observations that for the time being serve as test data are said to be "out-of-bag", or OOB.

In many statistical learning contexts, theoretical accuracy assessments are of limited usefulness. It is frequently necessary to rely on empirical methods. This is especially true for classification models. Several important methods of this type will now be described.

The final three methods are "resampling" methods, i.e., they rely on taking some form of sample from the one original available sample. They may be adapted in various ways to make smaller or greater use of theoretical assumptions. In any case all methods, as described here, assume that observations have been sampled independently.

The training/test approach is in principle the most general. If test data can be found that accurately reflect the target population, the training/test approach is to be preferred. If however the test data are derived from a random split of data from the source population, the other methods are in general preferable, because they at some point to all the data, both for training and for testing.

Here now are some further details of the methods that have been noted.

Simulation: A model is proposed that generated the data. What are its statistical properties? One answer is repeated simulations. Simple uses of this idea are:

- Simulate repeated sampling of values that follow a normal distribution.

- Simulate repeated sampling of values that follow a non-normal distribution, perhaps an exponential distribution.

- Marks are placed on the circumference of a roulette wheel that divide it into three perhaps unequal parts. Labeling the outcomes A, B and C, one can simulate, eg, a result from 1000 spins.

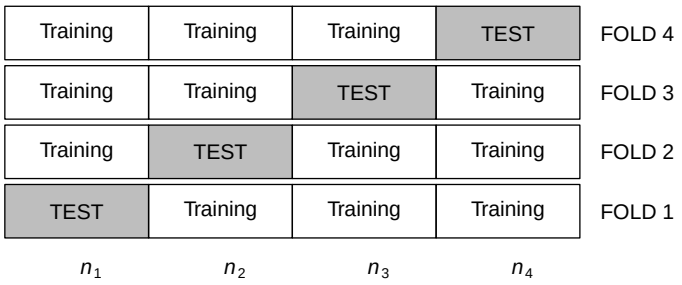


Figure 15: Schematic, designed to explain cross-validation. Data are first split, randomly, into k nearly equal parts. Here, for illustration, $k = 4$, with n_1, n_2, n_3, and n_4 observations in the respective. parts. At the first iteration (*fold*), the n_1 observations in the first part are set aside for testing, with remaining observations used for training, and so on. The split into a part that is used for testing, and the remaining observations that are used for testing, has the name *fold*. Once calculations for all 4 folds are complete, predicted values are available for all observations, in each case from a model that was trained independently of those observations.

Training/Test:

- Split data into training and test sets
- Train (NB: steps **1 & 2**); use test data for assessment.

This is the most widely applicable methodology. The test data can in principle be taken from the target population. If however data from the actual target is available when the model is fitted, one would want to use this for model fitting. Thus, in practice, testing on data from the actual target often has to be an "after-the-event" kind of check.

When there is very adequate data, a training/test split of the data achieves all that is needed. It is not necessary to look for a method, such as will now be described, that makes better use of the data.

Cross-validation Simple version: Train on subset 1, test on subset 2
Then; Train on subset 2, test on subset 1

More generally, data are split into k parts (eg, $k = 10$). Use each part in turn for testing, with other data used to train.

Cross-Validation Steps are:

- Split data into k parts (in Figure 15, k=4)
- At the ith repeat or *fold* ($i = 1, \ldots k$) use:
  the ith part for *testing*, the other k-1 parts for *training*.
- Combine the performance estimates from the k folds.

Bootstrap Sampling Bootstrap samples are with replacement samples, of the same size as the initial sample.

Here are two bootstrap samples from the numbers 1 ... 10

```
1 3 6 6 6 6 6 8 9 10 (5 6's; omits 2,4,5,7)
2 2 3 4 6 8 8 9 10 10 (2 2's, 2 8's, 2 10's; omits 1,5,7)
1 1 1 1 3 3 4 6 7 8 (4 1's, 2 3's; omits 2,5,9,10)
```

Here is how bootstrap sampling can be used in practice:

- Take repeated (with replacement) random samples of the observations, of the same size as the initial sample.
- Repeat analysis on each new sample (NB: In the example above, repeat **both** step 1 (selection) & 2 (analysis).
- Variability between samples indicates statistical variability.
- Combine separate analysis results into an an overall result.

8.2 Methods for assessing accuracy of parameter estimates

Quite stringent conditions are necessary to ensure that accuracy estimates for a regression or classification model will be unbiased or have negligible bias. The model must be correct. Section 5 illustrates, with examples, some of the issues.

Some of the possibilities are:

1. Estimates that depend heavily on distributional assumptions may be calculated from the one available sample. The standard errors, t-statistics, and related statistics that are included in the output from R's `lm()` linear modelling function have this character.
2. Bootstrap samples can be used to derive the sampling distributions of some of the statistics that may be of interest – means, means and regression coefficients. This approach does however have limitations, which can be serious. For extreme quantiles, it will fail.
3. In a limited range of circumstances, permutation methods may be used for tests of statistical significance.

As described here, all methods assume that observations have been sampled independently from the relevant population. Exact theoretically based results are available for models with iid normal errors. If the distribution is not normal results are, under relatively weak independence assumptions, valid asymptotically, i.e., it is valid in the limit as the sample size goes to infinity.

Bootstrap and permutation methods do not rely, directly, on normality assumptions. Some assumptions are however necessary if results are to be susceptible to ready interpretation. How does one interpret the result of a bootstrap version of a t-test for comparing two means, if the two distributions have a markedly different shape?

Laboratory Notes 8 demonstrate bootstrap samping and a permutation distribution approach, for the comparison of two means. It is assumed that there are no other factors that might, in part or whole, account for any difference.

8.3 Examples

8.3.1 Comparing two populations

Cuckoos lay eggs in the nests of other birds. The eggs are then unwittingly adopted and hatched by the host birds. Newton (1893-1896, p. 123) makes the claim that the eggs that cuckoos lay in the nests of other birds tend to match the eggs of the host bird in size, shape and color. Latter (1902) collected extensive cuckoo egg data, in order to investigate these claims.

Figure 16A is a a summary of the data. Eggs laid in the nests of wrens are clearly much smaller, both in length and breadth, than the eggs laid in the nests of other birds. Visually, it is hard to see much distinction between eggs laid in the nests of these other species. For now, it therefore seems reasonable to examine the comparison between eggs laid in the nests of wrens, and eggs laid in the nexts of other birds, as in Figure 16B. Observe that Figure 16B tells much the same story, whether we focus on length or on breadth.

There are two types of questions that these data might be used to answer:

1. Are the apparent differences, between eggs found in the nests of wrens and eggs found in the nests of other birds, reproducible. If another sample of cuckoo eggs was collected, similarly a mix of eggs from wrens' nests and eggs from the nests of other birds, is it likely that similar differences would again be found?
2. Given a sample of cuckoo eggs is it possible to predict, with some reasonable accuracy, which eggs are from cuckoos and which from other birds?

The first question is often of interest in a data mining context, but is not usually the question of most direct interest. The second question is the one that is more commonly the focus of direct interest.

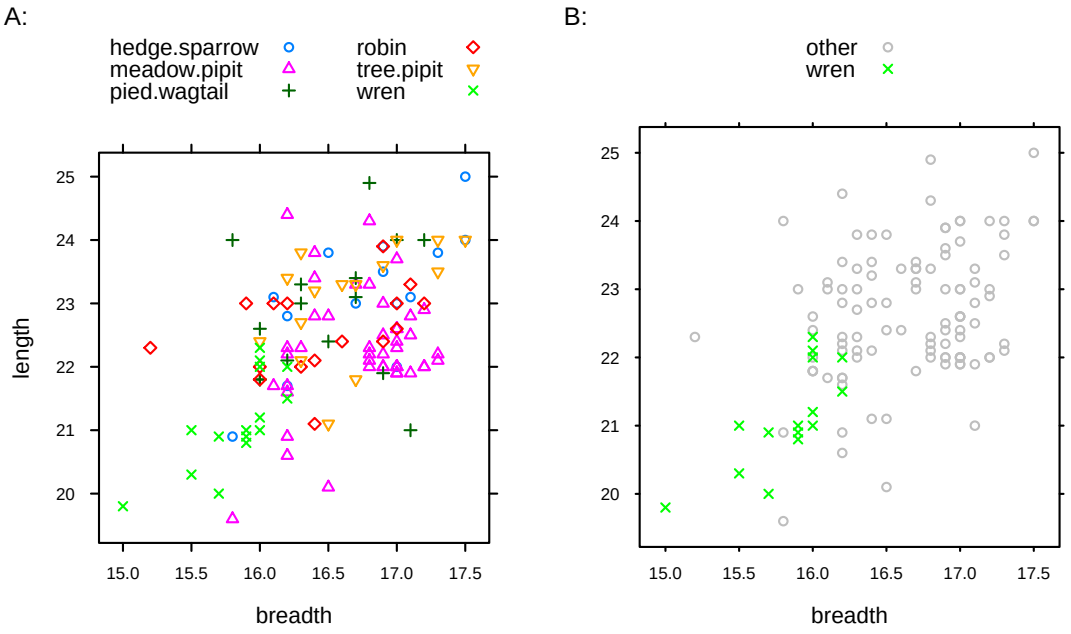


Figure 16: Length versus breadth of cuckoo eggs, identified according to the species of host bird in whose nest the eggs were laid.

For the moment, the focus will be on the first question, first using informal graphical comparisons, then moving to more formal methods.

8.3.2 Comparisons for individual variables

Figure 16B provided what is perhaps the most obvious form of graphical comparison. But might the difference between eggs laid in wren nests and eggs laid in other nests be merely a result of chance?

First, consider how we might do this separately for length. Figure 17 shows the comparison.

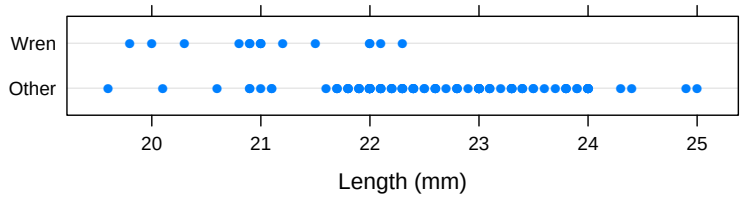


Figure 17: Dotplot comparison between lengths of eggs laid in wren nests and eggs laid in other nests.

Code is:

```
> dotwren <- dotplot(species %in% "wren" ~ length, data=cuckoos,
                     scales=list(y=list(labels=c("Other", "Wren"))),
                     xlab="Length (mm)")
> print(dotwren)
```

8.3.3 A check that uses the bootstrap

The bootstrap (resampling with replacement) may be used to check whether the difference is likely to be more than noise. Repeated pairs of with replacement samples are drawn, the first member of the pair from "other" (non-wren), and the second member from eggs laid in wren nests. For each pair, calculate the difference between the means. Repeat a number of times (here 100, so that points stand out clearly, but 1000 would be better), and plot the differences. Figure 18 shows the result of one such sampling experiment.

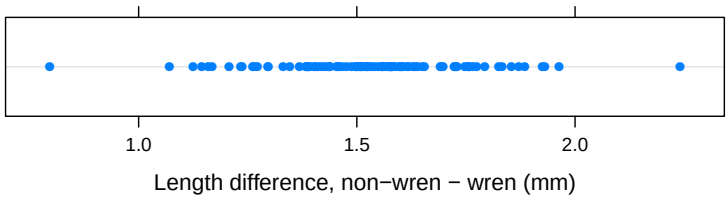


Figure 18: Differences, in successive bootstrap samples, between mean length of eggs in non-wren nests, and eggs in wren nests.

Suitable code is:

```
> avdiff <- numeric(100)
> for(i in 1:100){
      avs <- with(cuckoos, sapply(split(length, species %in% "wren"),
                                  function(x)mean(sample(x, replace=TRUE))))
      avdiff[i] <- avs[1] - avs[2]  # FALSE (non-wren) minus TRUE (wren)
    }
> dotdiff <- dotplot(~ avdiff, xlab="Length difference, non-wren - wren (mm)")
> print(dotdiff)
```

Observe that none of the differences are anywhere near zero. This is convincing evidence that the length differences are unlikely to be due to chance.

8.3.4 A check that uses simulation (the parametric bootstrap)

The difference here is that the random samples are drawn from normal distributions with the same mean and variance as in the samples.

If the variances can be assumed equal, the relevant distribution (when an infinite number of bootstrap samples are taken) can be determined theoretically, and except as a learning exercise there is not much point in such a simulation. If variances are unequal, the situation is more complicated. The standard theoretical approaches do however have simulation counterparts.

For a *t*-text comparison that allows for unequal variances, proceed thus:

```
> id <- as.numeric(with(cuckoos, species %in% "wren"))+1
> Species <- c("non-wren", "wren")[id]
> with(cuckoos, t.test(length[Species=="non-wren"],
                       length[Species=="wren"]))

        Welch Two Sample t-test

data:  length[Species == "non-wren"] and length[Species == "wren"]
t = 7.0193, df = 21.244, p-value = 5.872e-07
alternative hypothesis: true difference in means is not equal to 0
```

```
95 percent confidence interval:
 1.069984 1.970016
sample estimates:
mean of x mean of y
    22.64     21.12
```

Part IV
Linear Models, GLMs and GAMs

GLMs are Generalized Linear Models, while GAMs are Generalized Additive Models.

Most accounts of linear models assume that errors are independently and identically distributed (i.i.d.). That assumption is by no means necessary. In real world examples, it is often patently false. It will however be our starting point, for several reasons:

- There are a wide range of situations where the i.i.d. errors assumption is a reasonable approximation.
- It is enough to deal with one complication at a time.

Generalized linear models (GLMs) are an extension of linear models. An important special case is models with a binary outcome. These are essentially classification models where there are two possible outcomes.

9 Linear Models

The base R system and the various R packages provide, between them, a huge range of model fitting abilities. In these notes, the major attention will be on the model fitting function is `lm()`, where the `lm` stands for linear model. Here, we fit a straight line, which is very obviously a linear model! This simple starting point gives little hint of the range of models that can be fitted using R's linear model `lm()` function. Later discussion will build on these simple ideas to present a more expansive view of linear models.

R's implementation of linear models uses a symbolic notation Wilkinson & Rogers (1973), that gives a straightforward means for describing elaborate and intricate models.

9.1 Straight line Regression

	weight	depression
1	1.90	2.00
2	3.10	1.00
3	3.30	5.00
4	4.80	5.00
5	5.30	20.00
6	6.10	20.00
7	6.40	23.00
8	7.60	10.00
9	9.80	30.00
10	12.40	25.00

Table 3: Data showing depression in lawn (mm.), for various weights of roller (t)

A straight line regression model for the data in Table 3 can be written

$$\text{depression} = \alpha + \beta \times \text{weight} + \text{noise}.$$

Writing y in place of `depression` and x in place of `weight`, we have:

$$y = \alpha + \beta x + \varepsilon.$$

Subscripts are often used. Given observations (x_1, y_1), $(x_2, y_2), \ldots, (x_n, y_n)$, we may write

$$y_i = \alpha + \beta x_i + \varepsilon_i.$$

The $\alpha + \beta x$ term is the deterministic component of the model, and ε is the random noise. Greatest interest usually centers on the deterministic term.

The line is chosen so that the sum of squares of residuals is as small as possible. Thus, it is chosen to minimze

$$\sum_{i=1}^{n}(y_i - a - b_i x_i)^2$$

where a is an estimate of the intercept α and b is an estimate of the slope β. The R function `lm()` provides a ready way to obtain the estimates a and b.

Given estimates a and b we can pass the straight line

$$\widehat{y} = a + bx$$

through the points of the scatterplot. Fitted or predicted values are calculated using the above formula, i.e.

$$\widehat{y_1} = a + bx_1,\ \widehat{y_2} = a + bx_2, \ldots .$$

By construction, the fitted values lie on the estimated line. The line passes through the cloud of observed values. Useful information about the noise can be gleaned from an examination of the residuals, which are the differences between the observed and fitted values,

$$e_1 = y_1 - \widehat{y_1},\ e_2 = y_2 - \widehat{y_2}, \ldots . \tag{1}$$

In particular, a and b are estimated so that the sum of the squared residuals is as small as possible, i.e., the resulting fitted values are as close (in this "least squares" sense) as possible to the observed values. The residuals are shown as vertical lines, gray for negative residuals and black for positive residuals, in Figure 19.

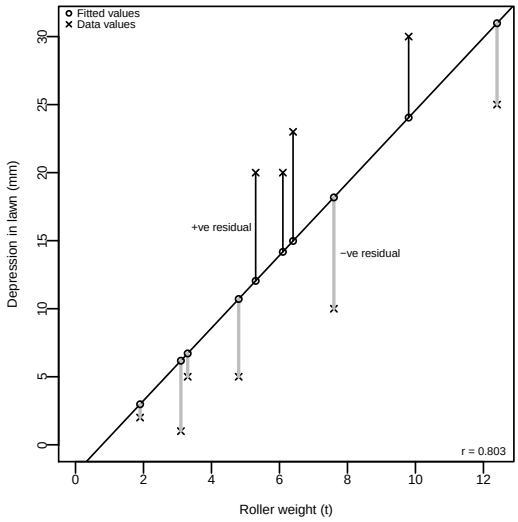


Figure 19: Lawn depression for various weights of roller, with fitted line. The fitted line is designed to minimize the sum of squares of residuals, i.e., the sum of squared lengths of the vertical lines, joining x's to o's, that are shown on the graph.

In standard analyses, we assume that the ε_i are independently and identically distributed as normal variables with mean 0 and variance σ^2.

9.2 Why minimize the sum of squares?

A more fundamental principle is maximum likelihood. The joints density of n independent Normal random variables U_1, U_2, $\ldots U_n$, each with mean μ and variance σ^2, is:

$$\begin{aligned} f(u_1, u_2, \ldots u_n) &= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{(u_i-\mu)^2}{2\sigma^2}) \\ &= \frac{1}{\sigma^n (2\pi)^{\frac{n}{2}}} \exp(-\sum_{i=1}^{n} \frac{(u_i-\mu)^2}{2\sigma^2}) \end{aligned}$$

NB: Independence allows us to multiple the individual normal densities, to obtain the joint density.

Replacing the u_i by the errors e_i in equation 1 yields the likelihood:

$$\begin{aligned} L(a, b, \sigma) &= \frac{1}{\sigma^n (2\pi)^{\frac{n}{2}}} \exp(-\sum_{i=1}^{n} \frac{e_i^2}{2\sigma^2}) \\ &= \frac{1}{\sigma^n (2\pi)^{\frac{n}{2}}} \exp(-\frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - a - bx_i)^2) \end{aligned}$$

As log() is a monotonic function, maximizing the log likelihood is exactly equivalent to maximizing the likelihood. The log likelihood is:

$$log(L) = -\frac{n}{2}\log(2\pi) - n\log\sigma - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - a - bx_i)^2$$

Equivalently, we can minimize

$$-log(L) = \frac{n}{2}\log(2\pi) + n\log\sigma + \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - a - bx_i)^2$$

For any given σ, the problem reduces to that of minimizing

$$\sum_{i=1}^{n} (y_i - a - bx_i)^2$$

i.e., to least squares. The argument is quite general. The argument carries through in exactly the same way, if $y_i - a - bx_i$ is replaced by $y_i - g(\boldsymbol{x}_i; \boldsymbol{b}_i)$, where $\boldsymbol{x}_i$ and $\boldsymbol{b}_i$ are vector valued.

Least squares does not tell us how to estimate σ^2. The maximum likelihood estimate for σ^2 is

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - a - bx_i)^2$$

This is biased. The theory simplifies if an unbiased estimate is used. If p is the number of parameters (here there are two parameters, a and b) over which the sum of squares has been minimized, division is by $n - p$.

The assumptions of independence, identical distribution and normality are crucial. Least squares will not in general yield maximum likelihood estimates if:

- Variances are not homogeneous (use weighted least squares if it is known how the variances change with x_i, of if the pattern of change can be inferred with some reasonable confidence)
- Observations are not independent;
- Data are not from a normal distribution.

9.3 Syntax – model, graphics and table formulae:

The syntax for `lm()` models that will be demonstrated here is used, with modification, throughout the modeling functions in R. A very similar syntax can be used for specifying graphs and for certain types of tables.

The following plots the data in the data frame `roller` (shown in Table 3) that is in the *DAAG* package.

```
> library(DAAG)
> plot(depression ~ weight, data=roller)
```

The formula `depression ~ weight` can be used either as a graphics formula or as a model formula. Just to see what happens, try fitting a straight line, and adding it to the above plot:

```
> lm(depression ~ weight, data=roller)

Call:
lm(formula = depression ~ weight, data = roller)

Coefficients:
(Intercept)       weight
     -2.087        2.667

> abline(lm(depression ~ weight, data=roller))
```

The different components of the model are called `terms`. In the above, there is one term only on the right, i.e., `weight`.

9.4 The technicalities of linear models

9.4.1 The model matrix – straight line regression example

The quantity that is to be minimized can be written:

$$\sum_{i=1}^{10}(y_i - a - bx_i)^2$$

Now observe how this can be written in matrix form. Set

$$\mathbf{X} = \begin{pmatrix} 1 & 1.9 \\ 1 & 3.1 \\ 1 & 3.3 \\ 1 & 4.8 \\ 1 & 5.3 \\ 1 & 6.1 \\ 1 & 6.4 \\ 1 & 7.6 \\ 1 & 9.8 \\ 1 & 12.4 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} 2 \\ 1 \\ 5 \\ 5 \\ 20 \\ 20 \\ 23 \\ 10 \\ 30 \\ 25 \end{pmatrix} \quad \mathbf{e} = \mathbf{y} - \mathbf{Xb} = \begin{pmatrix} 2-(a+1.9b) \\ 1-(a+3.1b) \\ 5-(a+3.3b) \\ 5-(a+4.8b) \\ 20-(a+5.3b) \\ 20-(a+6.1b) \\ 23-(a+6.4b) \\ 10-(a+7.6b) \\ 30-(a+9.8b) \\ 25-(a+12.4b) \end{pmatrix} \quad \text{where} \quad \mathbf{b} = \begin{pmatrix} a \\ b \end{pmatrix}$$

Here a and b are chosen to minimize the sum of squares of elements of $\mathbf{e} = \mathbf{y} - \mathbf{Xb}$, i.e., to minimize

$$\mathbf{e}'\mathbf{e} = (\mathbf{y} - \mathbf{Xb})'(\mathbf{y} - \mathbf{Xb})$$

The least squares equations can be solved using matrix arithmetic. For our purposes, it will be sufficient to use the R function `lm()` to handle the calculation:

```
> lm(depression ~ weight, data=roller)
```

```
Call:
lm(formula = depression ~ weight, data = roller)

Coefficients:
(Intercept)       weight
     -2.087        2.667
```

Both `weight` and `depression` are variables, i.e., they take values on the real line. They have, within R, class "numeric".

Recap, and Next Steps in Linear Modeling

The straight line regression model is one of the simplest possible type of linear model. We have shown how to construct the model matrix that R uses when it fits such models. Here, it had two columns only. Omission of the intercept term will give an even simpler model matrix, with just one column.

Regression calculations in which there are several explanatory variables are handled in the obvious way, by adding further columns as necessary to the model matrix. This is however just the start. There is a great deal more that can be done with model matrices, as will be demonstrated.

9.4.2 What is a linear model?

The models discussed here are linear, in the sense that predicted values are a linear combination of a finite set of basis functions. The basis functions can be nonlinear functions of the features, allowing the modeling of systems in which there can nonlinear components that enter additively. The technical mathematical apparatus of linear models has a wider importance than linear models per se. It is a fundamental component of many of the algorithms that have been developed by machine learners, by data miners, and by statisticians.

Data that are intended for regression calculations consist of multiple observations (or instances, or realizations) of a vector $(x_1, x_2, \ldots, x_k, y)$ of real numbers, where the x_is are explanatory variables and y is the dependent variable.

Given $x_1, x_2, \ldots, x_k$, which take values on the real line, a first step (which in the simplest case maps the x_is onto themselves), is the formation of basis' functions

$$\phi_1(x_1, x_2, \ldots x_k), \phi_2(x_1, x_2, \ldots x_k), \ldots, \phi_p(x_1, x_2, \ldots x_k)$$

In the simplest case $p = k$ and $\phi_1(x_1, x_2, \ldots x_p) = x_1$, $\phi_2(x_1, x_2, \ldots x_p) = x_2$, $\ldots$, $\phi_p(x_1, x_2, \ldots x_p) = x_p$.

Then any function with values on the real line such that

$$f(x_1, x_2, \ldots x_k) = b_1\phi_1(x_1, x_2, \ldots x_k) + b_2\phi_2(x_1, x_2, \ldots x_k) + \ldots + b_p\phi_p(x_1, x_2, \ldots x_k)$$

where the elements of $\mathbf{b} = (b_1, b_2, \ldots b_p)$ are the only unknowns, specifies a linear model.

The model is linear in the values that the ϕ's take on the sample data. It is not, in general, linear in the x_i's. **Here endeth our brief excursion that has defined the term** *linear model.*

The random part of the model: The statistical output (standard errors, p-values, t-statistics) from the `lm()` function assumes that the random term is i.i.d. (independently and identically distributed) normal. Least squares estimation is them equivalent to maximising the likelihood.

What if the i.i.d. assumption is false? Depending on the context, this may or may not matter. In general, it is unwise to assume that it does not matter!

If the i.i.d. normal errors assumption is false in ways that are to some extent understood, then it may be possible to make use of functions in one or other of the R packages that are designed to facilitate the modeling of the random part of the model. Typically, these fit the model by maximising the likelihood. Note especially the R packages *nlme* and *lme4*, for handling multilevel and related models, and *arima* and related functions in the *stats* package that fit time series models.

9.4.3 Model terms, and basis functions:

In the very simple model in which depression is modeled as a linear function of `weight`, there the one term (`weight` generates two basis functions: $\phi_1(x) = 1$ and $\phi_2(x) = x$ which mapped values of `weight` into itself. (Basis functions seem an unnecessary complication, for such a simple example.)

9.5 Multiple Regression

In multiple regression, the model matrix has one column for the constant term (if any), plus one column for each additional explanatory variable. Thus, multiple regrssion is an easy extension of straight line regression. Further flexibility is obtained by transforming variable values, if necessary, before use of the variable in a multiple regression equation.

In the next example, there are multiple explanatory variables. We start with simple multiple linear regression model, and then look to see whether there is a case to replace the linear terms by polynomial or spline terms. Polynomial and spline terms extend the idea of "linear model", with the result that the dependence upon the variables in the model may be highly nonlinear! The `lm()` function will fit any model for which the fitted values are a linear combination of basis functions. Each basis function can in principle be an arbitrary transformation of one or more explanatory variables. "Additive models" may be better terminology.

The dataset `nihills` in the `DAAG` package has record times for Northern Ireland mountain races. First, get a few details of the data:

```
> str(nihills)
'data.frame': 23 obs. of  4 variables:
 $ dist : num  7.5 4.2 5.9 6.8 5 4.8 4.3 3 2.5 12 ...
 $ climb: int  1740 1110 1210 3300 1200 950 1600 1500 1500 5080 ...
 $ time : int  3090 1680 2531 3739 1948 1740 1982 1669 1619 7017 ...
 $ timef: int  3832 2243 3193 4371 2295 2119 2526 2331 2187 8930 ...
```

First, get a few details of the data:

```
> str(nihills)
'data.frame':        23 obs. of  5 variables:
 $ dist    : num  7.5 4.2 5.9 6.8 5 4.8 4.3 3 2.5 12 ...
 $ climb   : int  1740 1110 1210 3300 1200 950 1600 1500 1500 5080 ...
 $ time    : num  0.858 0.467 0.703 1.039 0.541 ...
 $ timef   : num  1.064 0.623 0.887 1.214 0.637 ...
 $ gradient: num  232 264 205 485 240 ...
```

A scatterplot matrix, which plots every column against every other column and shows the result in the layout used for correlation matrices, is useful for an initial look at the data. The scatterplot matrix is a graphical counterpart of the correlation matrix.

For identifying the axes for each panel

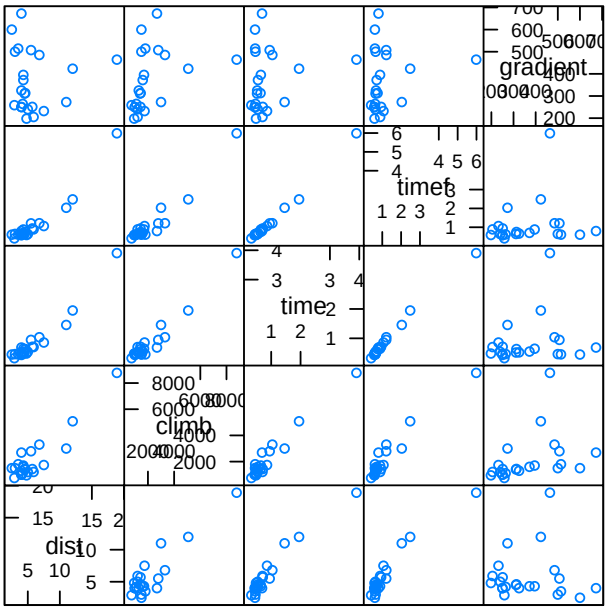
- look along the row to the diagonal to identify the variable on the vertical axis.
- look up or down the column to the diagonal to identify the variable on the horizontal axis.

Note that the data are positively skewed, i.e., there is a long tail to the right, for all variables. For such data, a logarithmic transformation often gives more nearly linear relationships.

```
> ## Create a data frame that holds the logged data
> lognihills <- log(nihills)
> names(lognihills) <- c("ldist", "lclimb", "ltime", "ltimef")
```

The relationships between explanatory variables, and between the dependent variable and explanatory variables, are closer to linear when logarithmic scales are used. The log transformed data are consistent with a form of parsimony that is advantageous if we hope to find a relatively simple form of model. We will see that this also leads to more readily interpretable results. Also the distributions for individual variables are more symmetric.

```
> ## Create scatterplot matrix
> library(lattice)
> print(splom(nihills, par.settings=size10))
```



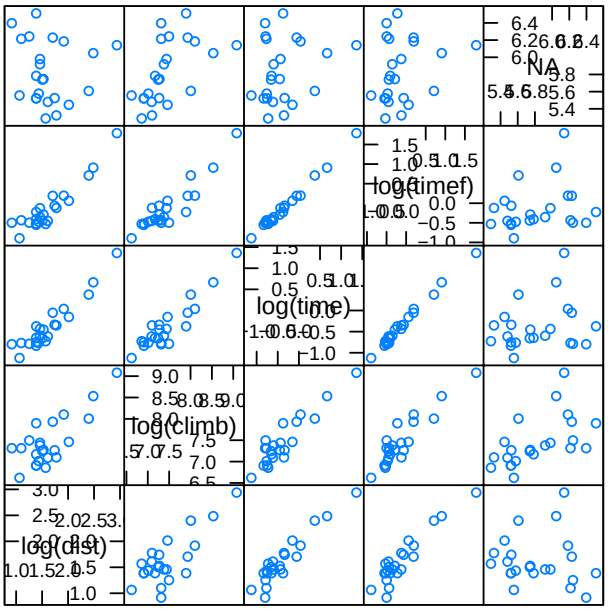
Scatter Plot Matrix

Figure 20: Scatterplot matrix for the Northern Ireland mountain racing data, with the correlation matrix given alongside. The function `cor()` takes a matrix or data frame as argument, by default giving the Pearson linear correlation.

The correlation matrix is:

```
> ## Correlation matrix
> round(cor(nihills), 2)
```

	dist	climb	time	timef	gradient
dist	1.00	0.91	0.97	0.95	0.03
climb	0.91	1.00	0.97	0.96	0.40
time	0.97	0.97	1.00	1.00	0.20
timef	0.95	0.96	1.00	1.00	0.19
gradient	0.03	0.40	0.20	0.19	1.00



Scatter Plot Matrix

Figure 21: Scatterplot matrix for the Northern Ireland mountain racing data, with logarithmic scales.

```
> print(splom(lognihills,
        varnames=c("log(dist)",
                   "log(climb)",
                   "log(time)",
                   "log(timef)"),
        par.settings=size10))
> round(cor(lognihills), 2)
```

	ldist	lclimb	ltime	ltimef	<NA>
ldist	1.00	0.78	0.95	0.93	-0.07
lclimb	0.78	1.00	0.92	0.92	0.57
ltime	0.95	0.92	1.00	0.99	0.24
ltimef	0.93	0.92	0.99	1.00	0.25
<NA>	-0.07	0.57	0.24	0.25	1.00

9.5.1 The regression fit

The following fits a regression plane, with logarithmic scales for all variables:

```
> lognihills <- log(nihills)
> names(lognihills) <- paste("l", names(nihills), sep="")
> lognihills.lm <- lm(ltime ~ ldist + lclimb, data=lognihills)
> round(coef(lognihills.lm),3)
```

```
(Intercept)       ldist      lclimb
     -4.961       0.681       0.466
```

This translates to:

$$\begin{aligned}\widehat{\text{time}} &= e^{3.205} \times \text{dist}^{0.686} \times \text{climb}^{0.502} \\ &= 24.7e^{3.205} \times \text{dist}^{0.686} \times \text{climb}^{0.502}\end{aligned}$$

Thus for constant `climb`, the prediction is that time per mile will decrease with increasing distance. Shorter races with the same climb will involve steeper ascents and descents; thus this seems reasonable.

A result that is easier to interpret can be obtained by regressing `log(time)` on `log(dist)` and `log(gradient)`, where `gradient` is `dist/climb`.

```
> nihills$gradient <- with(nihills, climb/dist)
> lognihills <- log(nihills)
> names(lognihills) <- paste("l", names(nihills), sep="")
> lognigrad.lm <- lm(ltime ~ ldist + lgradient, data=lognihills)
> round(coef(lognigrad.lm),3)
```

```
(Intercept)       ldist   lgradient
     -4.961       1.147       0.466
```

Thus, with `gradient` held constant, the prediction is that `time` will increase at the rate of $\text{dist}^{1.147}$. This makes good intuitive sense.

We pause to look more closely at the model that has been fitted. Does `log(time)` really depend linearly on the terms `ldist` and `log(lclimb)`? The function `termplot()` gives a good graphical indication (Figure 22).

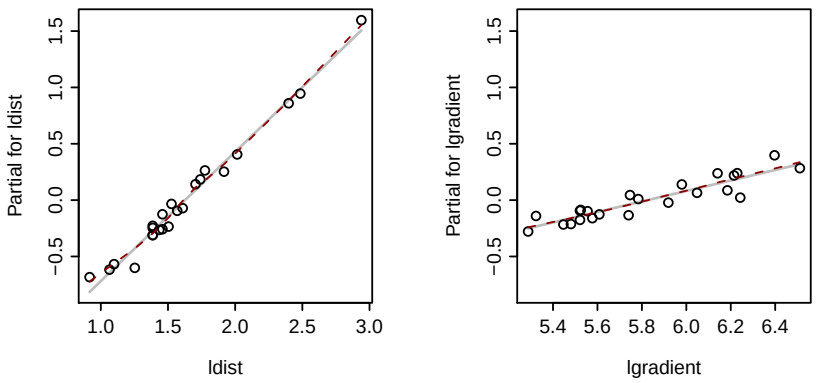


Figure 22: In these "term plots" the vertical scales in both panels show log(time), but centered to a mean of zero. The left panel shows partial residuals for `ldist`, while the right panel shows partial residuals for `lgradient`, i.e., `log(climb/dist)`. Smooth curves (dashes) have been passed through the points.

The code is

```
> par(mfrow=c(1,2))   # Ask for a 2 by 1 layout
> ## Plot the terms in the model
> termplot(lognigrad.lm, col.term="gray",
           partial=TRUE, col.res="black",
           smooth=panel.smooth)
```

Sugar yield data

	weight	trt
1	82.00	Control
2	97.80	Control
3	69.90	Control
4	58.30	A
5	67.90	A
6	59.30	A
7	68.10	B
8	70.80	B
9	63.60	B
10	50.70	C
11	47.10	C
12	48.90	C

Model matrix

(Intercept)	trtA	trtB	trtC
1	0	0	0
1	0	0	0
1	0	0	0
1	1	0	0
1	1	0	0
1	1	0	0
1	0	1	0
1	0	1	0
1	0	1	0
1	0	0	1
1	0	0	1
1	0	0	1

Table 4: The data frame `sugar` is shown in the left panel. The right panel has R's default form of model matrix that is used in explaining the yield of `sugar` as a function of treatment (`trt`)

The vertical scales show changes in `ltime`, about the mean of `ltime`. The lines show the effect of each explanatory variable when the other variable is held at its mean value. The lines, which are the contributions of the individual linear terms ("effects") in this model, are shown in gray so that they do not obtrude unduly. The dashed curves, which are smooth curves that are passed through the residuals, are the primary feature of interest in these plots. Notice that, in the plot for `ldist`, the smooth dashed line does not quite track the fitted line; there is a small but noticeable indication of curvature. Note also that until we have modeled effectively the clear trend that seems evident in this plot, there is not too much point in worrying about possible outliers. The trend can be very adequately modeled with a quadratic curve.

9.6 Modeling qualitative effects – a single factor

The `sugar` data frame (*DAAG* package) compares the amount of sugar obtained from an unmodified wild type plant with the amounts from three different types of genetically modified plants. In Table 4, the data are shown, with a model matrix alongside that may be used in explaining the effect of plant type (`Control`, or one of the three modified types `A` or `B` or `C`) on the yield of `sugar`.

In the model matrix in Table 4, `Control` is the baseline, and the yields for `A`, `B` and `C` are estimated as differences from this baseline. Then for each of the three treatments `A`, `B` and `C` there is an indicator variable that is 1 for that treatment, and otherwise zero. There are three basis functions that are used to account for the four levels of the factor `trt`.

The code used to fit the model is:

```
> library(DAAG)        # sugar is in DAAG package
> sugar.lm <- lm(weight ~ trt, data=sugar)
> summary(sugar.lm)

Call:
lm(formula = weight ~ trt, data = sugar)

Residuals:
     Min       1Q   Median       3Q      Max
-13.3333  -2.7833  -0.6167   2.1750  14.5667

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   83.233      4.473  18.609 7.17e-08
```

```
trtA         -21.400      6.325  -3.383 0.009597
trtB         -15.733      6.325  -2.487 0.037680
trtC         -34.333      6.325  -5.428 0.000625

Residual standard error: 7.747 on 8 degrees of freedom
Multiple R-squared: 0.7915,        Adjusted R-squared: 0.7133
F-statistic: 10.12 on 3 and 8 DF,  p-value: 0.004248
```

`Control` was taken as the baseline; the fitted value is 83.23, which is given as `(Intercept)`. The vales that are given for remaining treatments are differences from this baseline. Thus the fitted value (here equal to the mean) for treatment A is 83.23-21.40, that for B is 83.23-15.73, while that for C is 83.23-34.33.

The termplot summary

Again, termplots can be an excellent way to summarize results. Here is the termplot summary for the analysis of the cuckoo egg length data:

```
> termplot(sugar.lm, partial.resid=TRUE, se=TRUE)
```

The dotted lines show one standard deviation limits either side of the mean.

In the above model there was just one term, i.e., species, and hence just one graph. This one graph brings together information from the values of the six basis functions that correspond to the term `species`. The vertical scale is labeled to show deviations of egg lengths from the overall mean.

In this example the so-called "partial residuals" are the deviations from the overall mean. The dashed lines show one standard error differences in each direction from the species mean. (The standard error of the mean measures the accuracy of the mean, in the same way that the standard deviation measures the accuracy of the of an individual egg length.)

A note on factors: The names for the different values that a factor can take are the "levels".

```
> levels(bowler)
> levels(innings)
```

Internally, factors are stored as integer values. Each of the above factors has two levels. A lookup table is used to associate levels with these integer values.

Other things to try: The function `expand.grid()` can be helpful for setting up the values of the factors. We use `xtable()` to check that this gives the correct table:

```
> ## Use expand.grid() to set up the values of the factors
> y <- c(10, 14, 40, 50)
> Z <- expand.grid(bowler=c("A","B"), innings=c("one","two"))
> ## Check that this gives the correct table
> xtabs(y ~ bowler+innings, data=Z)

      innings
bowler one two
     A  10  40
     B  14  50
```

Other parameterizations

1. Above we used the default "corner" parameterization, which R calls the "treatment" parameterization. There are alternatives. The most commonly used alternative parameterization is the "anova" parameterization, which R calls the "sum" parameterization. Use it thus:

```
> options(contrasts=c("contr.sum", "contr.poly"))
> model.matrix(~ trt, data=sugar)

   (Intercept) trt1 trt2 trt3
1            1    1    0    0
2            1    1    0    0
3            1    1    0    0
4            1    0    1    0
5            1    0    1    0
6            1    0    1    0
7            1    0    0    1
8            1    0    0    1
9            1    0    0    1
10           1   -1   -1   -1
11           1   -1   -1   -1
12           1   -1   -1   -1
attr(,"assign")
[1] 0 1 1 1
attr(,"contrasts")
attr(,"contrasts")$trt
[1] "contr.sum"

> lm(weight ~ trt, data=sugar)

Call:
lm(formula = weight ~ trt, data = sugar)

Coefficients:
(Intercept)         trt1         trt2         trt3
     65.367       17.867       -3.533        2.133
```

These are called the "sum" contrasts (i.e., a particular form of parameterization) because they are constrained to sum to zero. The sum contrasts have been favoured in texts on analysis of variance.

2. There can be interactions between factors, or between factors and variables.

9.6.1 Grouping model matrix columns according to term

Quite generally, the basis functions $\phi_1, \phi_2, \ldots, \phi_p$ may be further categorized into groups, with one group for each term the model, thus:

$$\underbrace{\phi_1, \ldots, \phi_{m_1}}_{\text{Term1}}, \underbrace{\phi_{m_1+1}, \ldots, \phi_{m_2}}_{\text{Term2}}, \ldots$$

In the above, the basis functions for one factor formed just one termx. More generally, thare may be one group of basis functions for each of several factors. In the later discussion of spline terms, several basis functions will be required to account for each spline term in the model.

9.7 *Linear models, in the style of R, can be curvilinear models

We want to model y as a curvilinear function of x, where the form of the curve is chosen automatically. The idea is to express the response as a linear combination of curves. The curves comprise a set of basis functions for a vector space. Two general styles of curve will be described – orthogonal polynomials, and splines.

9.7.1 Polynomials and orthogonal polynomials

For fitting a polynomial function

$$y = b_0 + b_1 x + b_2 x^2 + \ldots + b_k x^k$$

of degree k in x, it is enough to have a model matrix that has, in addition to the initial column of 1s, columns that have values of x, x^2, $\ldots$, x^k.

Orthogonal polnomials, illustrated in Figure 23 with x the column `juice` in the data frame `fruitohms`, are however preferable. All the information needed to assess the degree of polynomial required can be obtained by fitting the maximum degree k of polynomial that is judged reasonable. Then:

- Coefficients of lower order basis functions do not change when higher order basis functions are removed.
- Standard errors of coefficients all change by the same constant multiplier when higher order basis functions are removed.

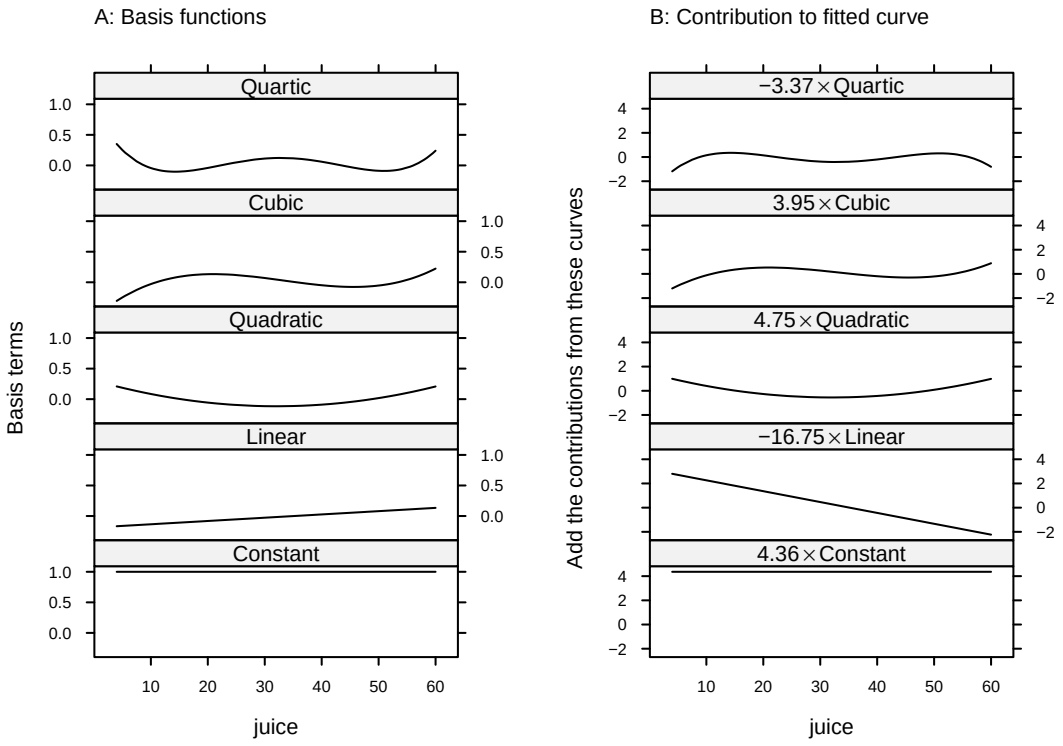


Figure 23: Panel A shows constant, linear, quadratic, cubic and quartic orthogonal polynomial basis functions, for a model that has the column `juice` in the `fruitohms` data frame as the explanatory variable.

10 An introduction to logistic regression

The data that will be used for illustration are from the data frame `bronchit` in the *SMIR* package. Figure 24 shows two plots – one of `poll` (pollution level) against `cig` (number of cigarettes per day), and the other of `poll` against `log(poll)`. In each case, points are identified as with or without bronchitis.

Code for panel A is

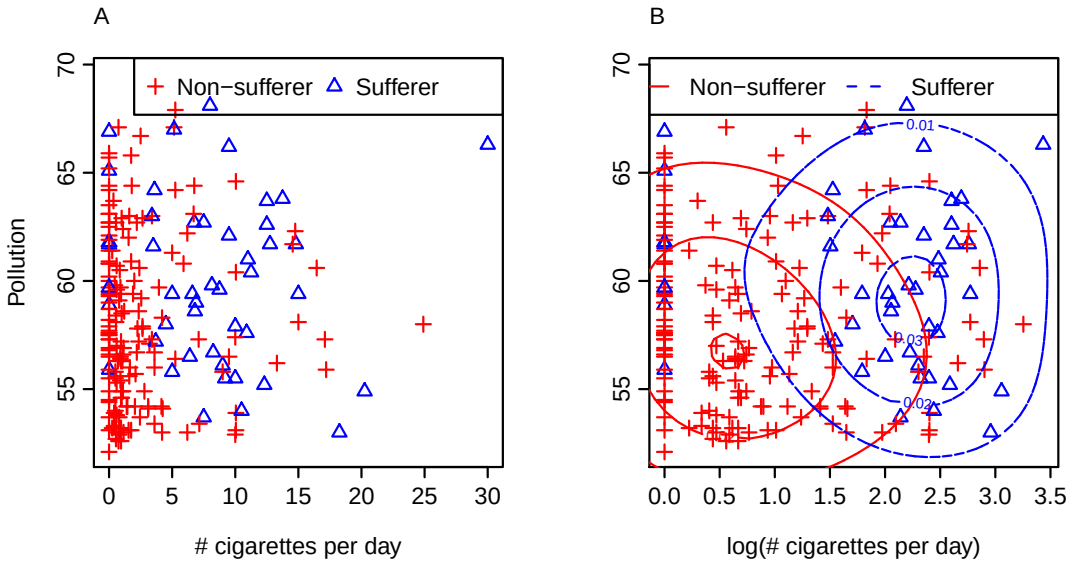


Figure 24: Panel A plots `poll` (pollution level) against `cig` (number of cigarettes per day). In panel B, the x-scale shows the logarithm of the number of cigarettes per day.

```
> library(SMIR); data(bronchit); library(KernSmooth)
> par(fig=c(0,.525, 0,1))
> ylim <- range(bronchit$poll)+c(0,1.5)
> plot(xlab="# cigarettes per day", ylab="Pollution", poll ~ cig,
       col=c(2,4)[r+1], pch=(3:2)[r+1], data=bronchit, ylim=ylim)
> legend(x="topright", legend=c("Non-sufferer","Sufferer"), ncol=2,
         pch=c(3,2), col=c(2,4))
> mtext(side=3, line=1.0, "A", adj=0)
```

Code for panel B is

```
> par(fig=c(.475,1, 0,1), new=TRUE)
> plot(poll ~ log(cig+1), col=c(2,4)[r+1], pch=(3:2)[r+1],
       xlab="log(# cigarettes per day)", ylab="",
       data=bronchit, ylim=ylim)
> xy1 <- with(subset(bronchit, r==0), cbind(x=log(cig+1), y=poll))
> xy2 <- with(subset(bronchit, r==1), cbind(x=log(cig+1), y=poll))
> est1 <- bkde2D(xy1, bandwidth=c(0.7, 3))
> est2 <- bkde2D(xy2, bandwidth=c(0.7, 3))
> lev <- pretty(c(est1$fhat, est2$fhat),4)
> contour(est1$x1, est1$x2, est1$fhat, levels=lev, add=TRUE, col=2)
> contour(est2$x1, est2$x2, est2$fhat, levels=lev, add=TRUE, col=4, lty=2)
> legend(x="topright", legend=c("Non-sufferer","Sufferer"), ncol=2,
         lty=1:2, col=c(2,4))
> mtext(side=3, line=1.0, "B", adj=0)
```

The logarithmic transformation spreads the points out in the x-direction, in a manner that is much more helpful for prediction than the untransformed values in panel A. The contours for non-sufferer and sufferer in panel B have a similar shape. The separation between non-sufferer and sufferer is stronger in the x-direction than in the y-direction. As one indication of this, the contours at a density of 0.02 overlap slightly in the x-direction, but strongly in the y-direction.

```
> par(mfrow=c(1,2))
> termplot(cig2.glm, se=TRUE, ylim=c(-2,4))
> par(mfrow=c(1,1))
```

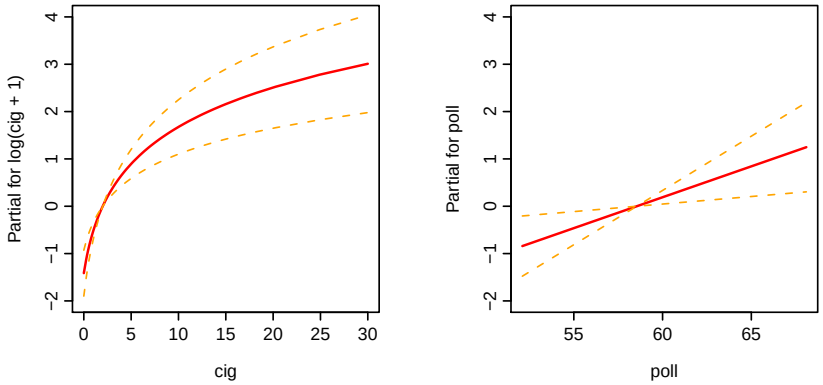


Figure 25: The panels show the contributions that the respective terms make to the fitted values, when the other term is held constant.

Logistic regression calculations

Figure 24 made it clear that the distribution of number of cigarettes had a strong positive skew. Thus, we might fit the model:

```
> cig2.glm <- glm(r ~ log(cig+1) + poll, family=binomial, data=bronchit)
> summary(cig2.glm)

Call:
glm(formula = r ~ log(cig + 1) + poll, family = binomial, data = bronchit)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6105  -0.5856  -0.3620  -0.2387   2.6529

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -10.78772    2.98851  -3.610 0.000307
log(cig + 1)   1.28823    0.22078   5.835 5.38e-09
poll           0.13057    0.04937   2.645 0.008169

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 221.78  on 211  degrees of freedom
Residual deviance: 168.76  on 209  degrees of freedom
AIC: 174.76

Number of Fisher Scoring iterations: 5
```

Termplots (Figure 25) provide a useful check that the effects of the covariates really are plausibly linear.

10.1 Logistic regression for the US accident data

Now consider the use of a logistic regression model to fit, in addition to `seatbelt`, `airbag` and `dvcat` as in Subsection 5.2, effects that are linear in `yearVeh` and `ageOFocc`. In order to get unbiased estimates,

the column `weight` provides weights. Each point is however a single observation – the weights do not reflect information content. A consequence is that the SEs that are given by the logistic regression analysis are meaningless:

```
> nassnew <- subset(nassCDS, !is.na(yearVeh) & yearVeh>=1986 & weight>0)
> nassnew.glm <- glm(dead ~ seatbelt + airbag + dvcat + yearVeh + ageOFocc,
                     weights=weight, family = quasibinomial, data=nassnew)
> summary(nassnew.glm)

Call:
glm(formula = dead ~ seatbelt + airbag + dvcat + yearVeh + ageOFocc,
    family = quasibinomial, data = nassnew, weights = weight)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-52.3406   -1.3832   -0.6847   -0.3243  143.9870

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -57.814010  57.374257  -1.008    0.314
seatbelt1     0.621012   0.081695   7.602 3.04e-14
airbag1       0.092832   0.124688   0.745    0.457
dvcat.L       5.293175   0.950856   5.567 2.62e-08
dvcat.Q       0.158468   0.806005   0.197    0.844
dvcat.C      -0.258352   0.505234  -0.511    0.609
dvcat^4       0.380526   0.244482   1.556    0.120
yearVeh       0.025830   0.028773   0.898    0.369
ageOFocc      0.036087   0.003899   9.255  < 2e-16

(Dispersion parameter for quasibinomial family taken to be 309.1294)

    Null deviance: 701450  on 23490  degrees of freedom
Residual deviance: 478605  on 23482  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 7
```

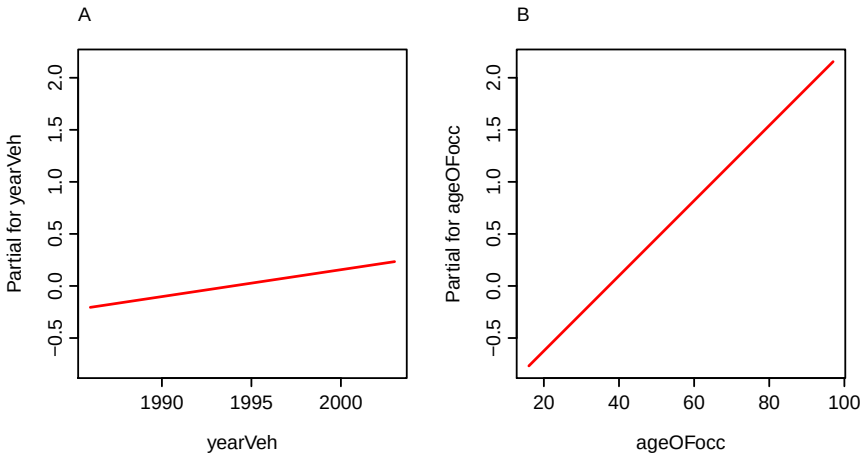


Figure 26: Term plots showing the contributions of the separate terms to the fitted values, in the GLM model.

A better approach may be to use the function `svyglm()` in the `survey` package to get estimates, together with SEs that are plausible:

```
> library(survey)
> des <- svydesign(ids = ~0, weights = ~weight, data = nassnew)
> summary(svyglm(dead ~ seatbelt + airbag + dvcat + yearVeh + ageOFocc,
      family = binomial, design = des))

Call:
svyglm(dead ~ seatbelt + airbag + dvcat + yearVeh + ageOFocc,
    family = binomial, design = des)

Survey design:
svydesign(ids = ~0, weights = ~weight, data = nassnew)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -57.81401   69.71333  -0.829   0.4069
seatbelt1     0.62101    0.09296   6.680 2.43e-11
airbag1       0.09283    0.12379   0.750   0.4533
dvcat.L       5.29317    0.45963  11.516  < 2e-16
dvcat.Q       0.15847    0.39378   0.402   0.6874
dvcat.C      -0.25835    0.26509  -0.975   0.3298
dvcat^4       0.38053    0.19233   1.978   0.0479
yearVeh       0.02583    0.03499   0.738   0.4604
ageOFocc      0.03609    0.00423   8.531  < 2e-16

(Dispersion parameter for binomial family taken to be 0.6582727)

Number of Fisher Scoring iterations: 10
```

These analyses are suspect, primarily because the weightings that are given for the observations are suspect. A more robust methodology, which uses data for drivers as well as for front seat passengers, is described in Farmer (2005).

11 Generalized Additive Models (GAMs)

11.1 Introduction

In the account that will be given here, Generalized Additive Models (GAMs) extend linear and generalized linear models to include smooth functions of explanatory variables with the smoothness determined by either

a parameter that directly controls the smoothness of the curve, or

estimated predictive accuracy.

In the present discussion, the chief attention will be on smoothing terms that are spline functions of a single explanatory variable. Such functions can themselves be constructed as linear combinations of spline basis terms.

The account will proceed as follows:

1. An account of regression splines, which work with cubic spline basis terms of chosen degree. For this, a linear combination of spline basis terms is chosen that gives a curve that best fits the data.

2. The use of a basis that allows a high degree of flexibility in the chosen curve, but increasing the residual sum of squares by a roughness penalty that is some multiple λ of the integral of the squared first derivative.

- Smoothing splines place a knot at each data point.
- Penalized splines aim only to ensure that knots are well spread each data.

3. Use of generalized cross-validation (GCV) to determine the choice of λ.
4. The extension to generalized linear models (GLMs), in particular logistic regression models (for 0/1 data) and Poisson regression models (count data).

For an account of the detailed computations, see the document
`http://wwwmaths.anu.edu.au/%7Ejohnm/r-book/xtras/lm-compute.pdf`.
See Wood (2006) for a comprehensive account of GAM models as implemented in R's *mgcv* package.

11.2 Splines

A spline curve is a is piecewise polynomial curve, i.e., it joins two or more polynomial curves. The locations of the joins are known as "knots". In addition, there are boundary knots which can be located at or beyond the limits of the data. There are theoretical reasons for use of smoothly joining cubic splines. Smooth joining implies that the second derivatives agree at the knots where the curves join. Two types of splines are in common use:

> Natural splines have zero second derivatives at the boundary knots. As a consequence, the curves extrapolate as straight lines.
>
> B-splines are unconstrained at the boundary knots,

Spline curves of any given degree can be formed as a linear combination of basis functions. The *splines* package has two functions that may be used to generate basis terms – `bs()` which generates B-spline basis terms, and `ns()` which generates natural spline basis terms. In either case there are many different choices of basis functions.

Natural splines will be the major focus of attention. Let $g(x)$ be an arbitrary function that is formed from k cubic curves that join smoothly, with zero second derivatives at the boundary knots. Then there exists a basis $\phi_1(x), \phi_2(x), \ldots, \phi_k(x)$, such that:

$$g(x) = b_0 + b_1\phi_1(x) + b_2\phi_2(x) + \ldots + b_k\phi_k(x)$$

The basis terms span a vector space that has, after allowing one degree of freedom for the constant term, k degrees of freedom. If, instead, $\phi_1(x), \phi_2(x), \ldots, \phi_k(x)$ is a B-spline basis, the basis spans a vector space that, after allowing for the constant term, has $k + 2$ degrees of freedom.

The following uses the abilities of the *splines* package, with data from the data frame `fruitohms` in the *DAAG* package. First `ohms` is plotted against `juice`. The function `ns()` (*splines* package) is then used to set up the basis functions for a natural spline with 3 degrees of freedom (`ns(juice, 3)`) and fit the curve. Figure 27 shows the plot:

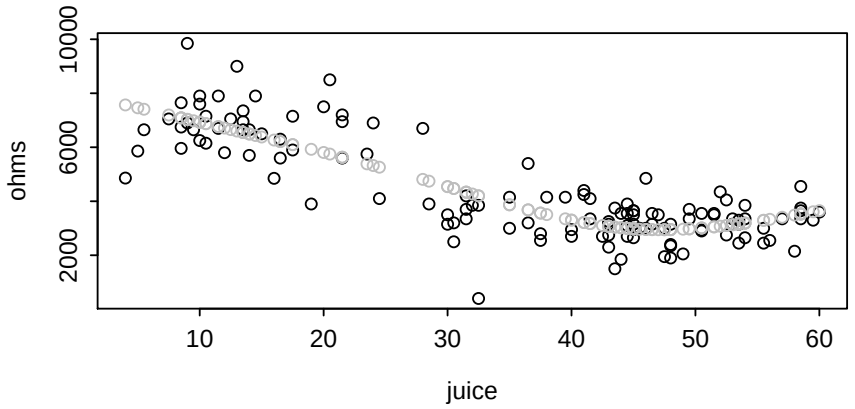


Figure 27: Smoooth curve, with pointwise 95% CI limits, fitted to the plot of resistance against apparent juice content. A three degree of freedom natural spline basis was used.

```
> library(DAAG)
> plot(ohms ~ juice, data=fruitohms)
> library(splines)
> fitohms <- fitted(lm(ohms ~ ns(juice, df=3), data=fruitohms))
> points(fitohms ~ juice, data=fruitohms, col="gray")
```

The parameter `df` (degrees of freedom) controls the smoothness of the curve. A large `df` allows a very flexible curve, e.g., a curve that can have multiple local maxima and minima. Clearly. the choice of a 3 degree of freedom curve, rather than 2 or 4, was arbitrary. The later discussion of Generalized Additive (GAM) models in Subsection 11.4 will show how this arbitrariness in choice of smoothing parameter can be avoided, providing data values can be assumed independent.

The advantage of regression splines is that they stay within the linear model (`lm()`) framework, with the same linear model theory and computational methods as any other linear model.

The `termplot()` function can be used to assess the result of a regression spline fit, just as for any other linear model fit. There is an option that allows, also, one standard error limits about the curve:

```
> ohms.lm <- lm(ohms ~ ns(juice, df=3), data=fruitohms)
> termplot(ohms.lm, partial=TRUE, se=TRUE)
```

The labeling on the vertical axis shows differences from the overall mean of `ohms`. In this example the *partial* is just the difference from the overall mean.

Natural spline basis functions, and their contributions to the fit

Figure 28A shows basis functions, both for natural splines of degree 3 (dashed curves) and of degree 4 (solid curves). Here, knots are placed at points that are equally spaced through the data. Notice that, in moving from a degree 3 natural spline curve to a degree 4 natural spline curve, the basis functions all change.

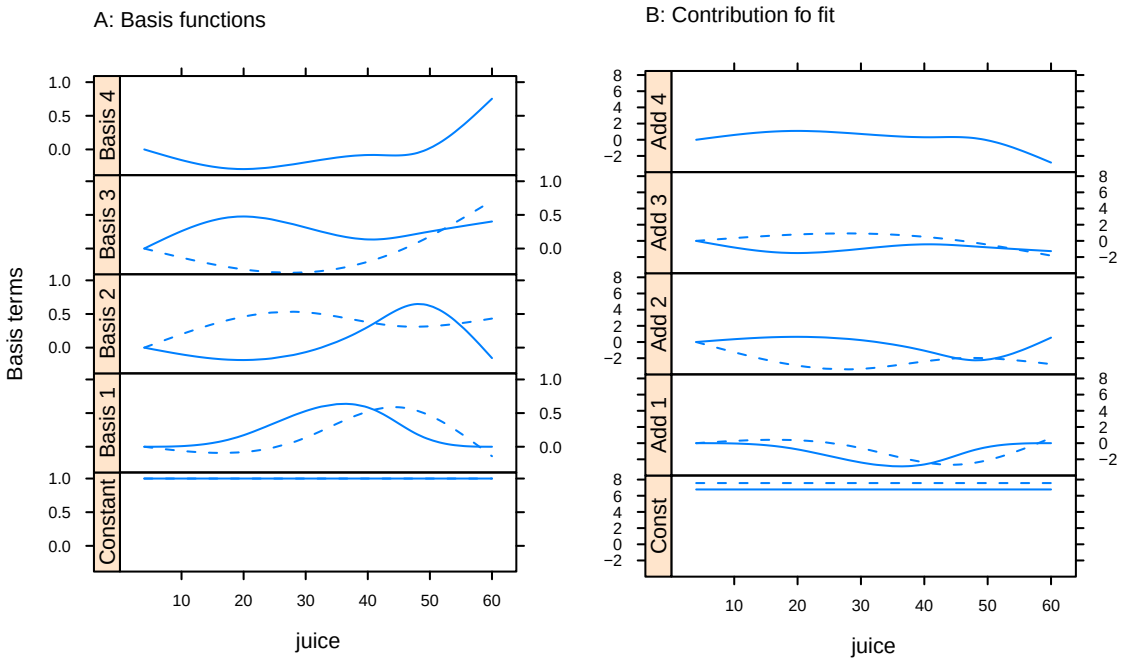


Figure 28: Panel A shows natural spline basis functions: (i) for a natural cubic spline of degree 3, with knots at the 33.3% and 66.7% quantile of the data (dashed curves); and (ii) for a natural cubic spline of degree 4, with knots at the 25%, 50% and 75% quantile of the data. Panel B shows the contributions of the basis functions to the fitted natural spline curve, in the regression of `kiloohms` on `juice`.

The two sets of basis functions can be extracted thus:

```
> matohms3 <- model.matrix(with(fruitohms, ~ ns(juice, 3)))
> matohms4 <- model.matrix(with(fruitohms, ~ ns(juice, 4)))
```

Spline basis elements

It is insightful to extract and plot the elements of the B-spline basis. Suitable code is:

```
> par(mfrow=c(2,2))
> basismat <- model.matrix(ohms.lm)
> for (j in 2:4) plot(fruitohms$juice, basismat[,j])
```

The first column of the model matrix is the constant term in the model. Remaining columns are the spline basis terms. The fitted values are determined by adding a linear combination of these four curves to the constant term.

Splines in models with multiple terms

For present purposes, it will be enough to note that this is possible. Consider for example

```
> loghills2k <- log(hills2000[, ])
> names(loghills2k) <- c("ldist", "lclimb", "ltime", "ltimef")
> loghill2k.lm <- lm(ltime ~ ns(ldist,2) + lclimb, data=loghills2k)
> par(mfrow=c(1,2))
> termplot(loghill2k.lm, col.term="gray", partial=TRUE,
           col.res="black", smooth=panel.smooth)
> par(mfrow=c(1,1))
```

11.3 Smooth functions of one explanatory variable

Smoothing spline smooths of a single variable place a knot at each data point. A penalty, some multiple λ of the integral of the squared second derivative of y with respect to x, is however added to the residual sum of squares, penalizing steep slopes. Consider a small interval δx over which the second derivative $f''(x)$ of the smoother $f(x)$ is approximately constant. The contribution of that interval to the penalty is then $\lambda f''(x)^2 \delta x$.

The total penalty is

$$\lambda \int f''(x)^2 dx$$

where the integral is over the range of x. The effect of the the penalty is to reduce the effective degrees of freedom. An adaptation of cross-validation – generalized cross-validation – is used to choose λ.

As noted above, the contributions of several variables can be added. There is then one λ_i, $i = 1, \ldots p$, for each of the p variables.

The placing of a knot at each data point has the result that the number of basis functions is one less than the number of data points. Even with just one explanatory variable, this can computationally expensive. A reasonable solution is to work, as in the penalized spline approach, with some smaller number of knots that are spread evenly through the data. Alternatively, use may be made of a low rank approximation to the space spanned by the complete set of basis terms.

A further challenge is to define and fit general smoothing functions of several explanatory variables. Thin plate splines are one approach. A set of basis functions emerges directly from the demand to minimize the residual sum of squares, plus a smoothness penalty that is a multiple λ of the multivariate integral over the space spanned by the explanatory variables of a suitable smoothness function.

These smoothers have as many parameters as there are unique predictor combinations, so that the computational cost is proportional to the cube of the number of of variables. Minimization over the full set of basis functions can be therefore computationally demanding, or may be intractable. Use of

a low rank approximation to the space spanned by the thin plate spline basis may be essential. Wood (2006) calls the resulting basis a thin plate regression spline basis.

The approaches that are described generalize for use with the error terms that are available for GLM models, now working with a penalized likelihood. The function `gam()`, in the *mgcv* package, handles the fitting in a highly automatic manner.

11.4 GAM models with normal errors

Fitting a GAM model with a single smoothing term

In Figure 29, residuals were calculated from a linear regression of `log(Time)` on `log(Distance)`, with data from the `worldRecords` dataset in the *DAAG* package. Then the function `gam()` was used to pass a smooth curve through the residuals.

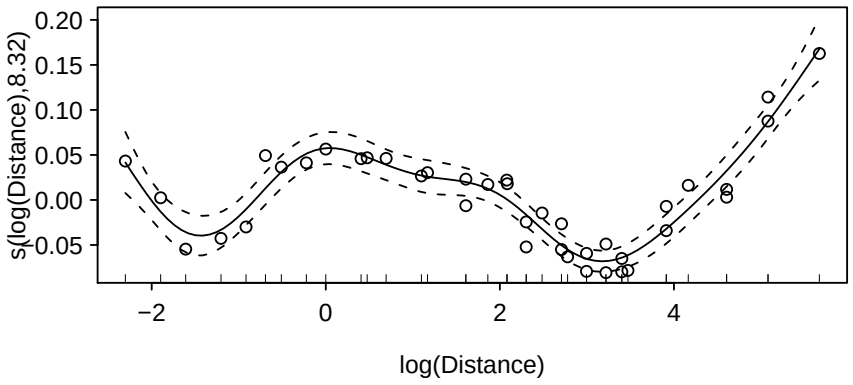


Figure 29: Residuals from the regression of `log(Time)` on `log(Distance)`, with smooth curve fitted. Data are from the `worldRecords` dataset (*DAAG* package).

Code that handles the calculations and plots the result is:

```
> library(mgcv)
> res <- resid(lm(log(Time) ~ log(Distance), data=worldRecords))
> wr.gam <- gam(res ~ s(log(Distance)), data=worldRecords)
> plot(wr.gam, residuals=TRUE, pch=1, las=1)
```

As `Time` is on a scale of natural logarithms, a residual of 0.1 corresponds to a deviation from the line of $\exp(0.1) - 1 \simeq 10.5\%$, i.e., just a little larger than 10%. A residual of 0.15 corresponds to a deviation from the line of ~16.2%. The magnitude of these deviations may seem surprising, given that the graph shows the points lying very close to the regression line of `log(Time)` on `log(Distance)`. The reason is that the times span a huge range, with the largest time more than 8800 × the smallest time. Set against a change by a factor of 8800, a change of 15% is very small.

If the preference is to fit a smooth curve to the initial data, the following code may be used:

```
> wrdata.gam <- gam(log(Time) ~ s(log(Distance)), data=worldRecords)
> plot(wrdata.gam, residuals=TRUE, pch=1)
```

On the graph that results, the 95% pointwise confidence bounds are hard to distinguish from the line.

With models such as these, it is good practice to check whether any consistent pattern appears with random data. As will be shown below, this can happen when the density of points varies along the *x*-axis, with the result that a single smoothing parameter is inappropriate.

Among the possibilities for creating "random" data are:

- Permute the residuals.
- Take a bootstrap sample of the residuals.

- Take random normal data from a population with mean 0 and standard deviation the same as that of the residuals.

Figure 30 shows the plots from 6 random permutations of the residuals.

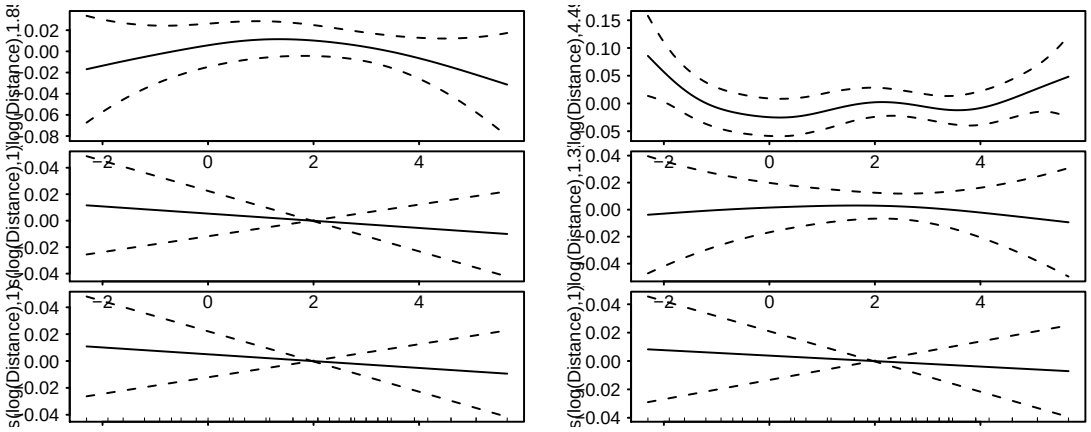


Figure 30: Plots from GAM models, fitted to successive random permutations of the residuals from the straight line fit of `log(Time)` to `log(Distance)`, for the `worldRecords` data. Note that none of thse plots shows more than a hint of a pattern.

Reassuringly, none of these show a pattern that stands out clearly, relative to the 95% pointwise error limits.The code used is:

```
> opar <- par(mfrow=c(3,2), mar=c(0.25, 4.1, 0.25, 1.1))
> res <- resid(lm(log(Time) ~ log(Distance), data=worldRecords))
> for(i in 1:6){
    permres <- sample(res)  # Random permutation
    # 0 for left-handers
    # 1 for right
    perm.gam <- gam(permres ~ s(log(Distance)), data=worldRecords)
    plot(perm.gam, las=1, rug=if(i<5)FALSE else TRUE)
  }
> par(opar)
```

Fitting a GAM model to climate data – two smooth terms

Time series data are likely to be correlated between years, creating potential issues for the use of smoothing methodology. In fortunate circumstances, any underlying trend will stand out above the error, but this should not be taken for granted. Simple forms of relationship are more plausible than complicated forms of relationship. The error term in regression relationships are used to explain synchrony between series is less likely to be less affected by autocorrelation than errors in the separate series. With these cautions, we proceeed to examination of a time series regression relationship.

Figure 31 fits annual rainfall, in the Murray-Darling basin of Australia, as a sum of smooth functions of `Year` and `SOI`. Figure 31 shows the estimated contributions of the two model terms.

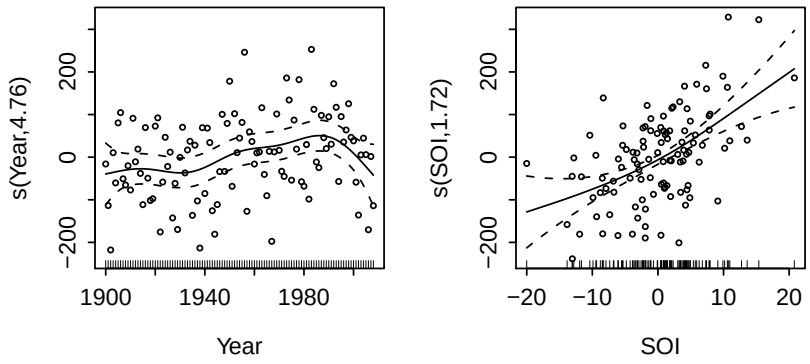


Figure 31: Contributions of the model terms to `mdbRain`, in a GAM model that is the sum of smooth terms in `Year` and `Rain`. The dashed curves show pointwise 2-SE limits, for the fitted curve. Note the downturn in the trend of `mdbRain` after about 1985.

Code is:

```
> par(mfrow=c(1,2))
> mdbRain.gam <- gam(mdbRain ~ s(Year) + s(SOI), data=bomregions)
> plot(mdbRain.gam, residuals=TRUE, se=2, pch=1, cex=0.5, select=1)
> plot(mdbRain.gam, residuals=TRUE, se=2, pch=1, cex=0.5, select=2)
> par(mfrow=c(1,1))
```

11.5 Smoothing terms with time series data – issues of interpretation

Now consider the data series `Erie`, giving levels of Lake Erie from 1918 to 2009 will be used for illustration.[10] They are available in versions ≥ 0.7-8 of the *DAAGxtras* package, as the column `Erie` in the multivariate time series object `greatLakes`. It is convenient to start by extracting the column that will be used:

```
> Erie <- greatLakes[,"Erie"]
```

An ideal would be to find a covariate or covariates than can largely explain the year to year changes. For the series that we now examine, no such explanation is available.

The unsmoothed data

Figure 32 shows a plot of the series:

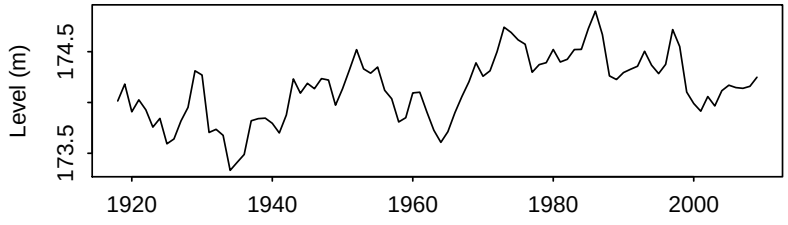


Figure 32: Level of Lake Erie, in meters.

```
> ## Code
> plot(Erie,
       xlab="",
       ylab="Level (m)")
```

In the absence of identfying a direct cause for the year to year changes, the best that can be done is to find a correlation structure that drives much of the year to year change. A re-run of the process (a new *realization*) will produce a different series, albeit one that shows the same general tendency to move up and down.

[10] Data are from `http://www.lre.usace.army.mil/greatlakes/hh/greatlakeswaterlevels/historicdata/greatlakeshydrographs/`

The plots that are show in Figure 33 are good starting points for investigation of the corrrelation structure. Panel A shows lag plots, up to a lag of 3. Panel B shows estimates of the successive correlations, in this context are called autocorrelations.

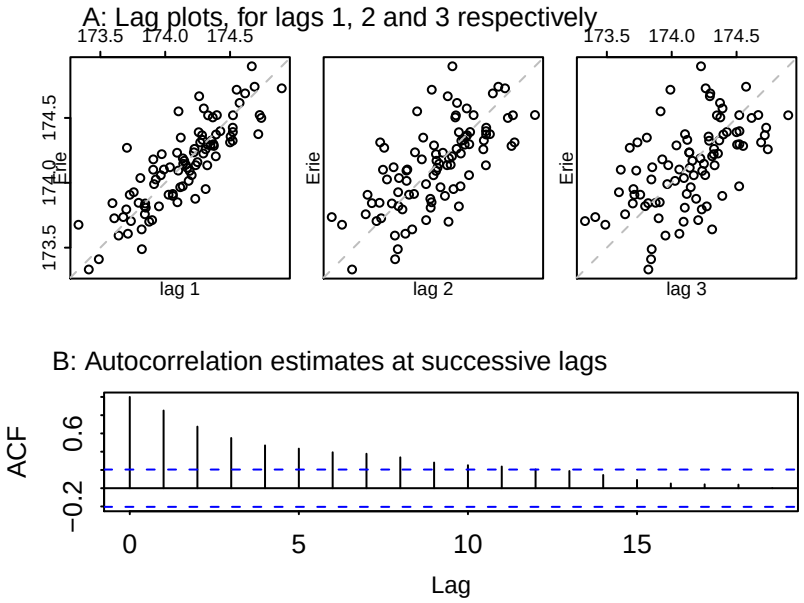


Figure 33: Panel A plots lake levels vs levels at lags 1, 2 and 3 respectively, for Lake Erie. Panel B shows the autocorrelations at lags 0 (= 1), 1 (1^{st} graph in panel A), 2 (2^{nd} graph), ... A large autocorrelation at lag 1 is followed by smaller autocorrelations at later lags.

```
> ## Panel A
> lag.plot(Erie, lags=3,
           do.lines=FALSE,
           layout=c(1,3))
> ## Panel B
> acf(Erie)
```

There is a strong correlation at lag 1, a strong but weaker correlation at lag 2, and a noticeable correlation at lag 3. Such a correlation pattern is typical of an autoregressive process where most of the sequential dependence can be explained as a flow-on effect from a dependence at lag 1.

In an autoregressive time series, an independent error component, or "innovation" is associated with each time point. For an order p autoregressive time series, the error for any time point is obtained by taking the innovation for that time point, and adding a linear combination of the innovations at the p previous time points. (For the present time series, initial indications are that $p = 1$ might capture most of the correlation structure.)

11.5.1 Smooth, with automatic choice of smoothing parameter

What do we see if we fit a GAM smoothing term to the `Erie` series? Recall that the smooth assumes independently and identically distributed data, and in particular that there is no serial correlation.

Figure 34 shows the result:

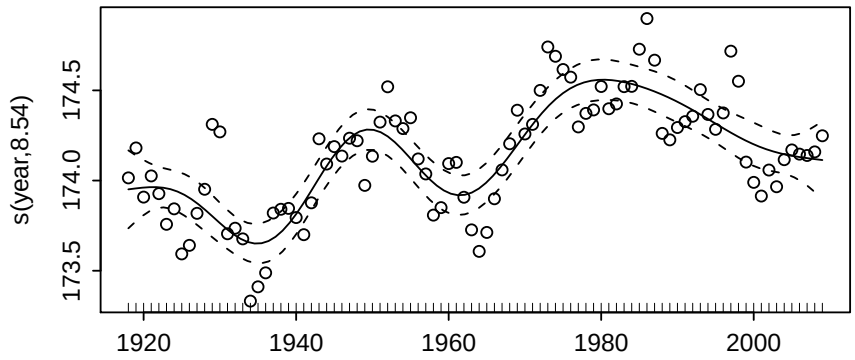


Figure 34: GAM smoothing term, fitted to the Lake Erie Data. The curve has removed the autocorrelation structure from the series, leaving residuals that are independent.

The code is:

```
> df <-  data.frame(height=as.vector(Erie), year=time(Erie))
> obj <- gam(height ~ s(year), data=df)
> plot(obj, shift=mean(df$height), residuals=T, pch=1, xlab="")
```

The curve may be useful as a broad summary of the pattern of change over time. The pointwise confidence limits would be meaningful if the processes that generated the sequential correlation structure could be identified and used to explain the curve. Without such an understanding, they are meaningless. All that is repeatable is the process that generated the curve, not the curve itself. Time series models, such as will now be considered, are designed to account for such processes.

Fitting and use of an autoregressive model

An autoregressive model gives insight that makes it possible to estimate the level of the lake a short time ahead, and to put realistic confidence bounds around those estimates. For the Lake Erie data, an autoregressive correlation structure doesw a good job of accounting for the pattern of change around a mean that stays constant.

Once an autoregressive model has been fitted, the function `forecast()` in the *forecast* package can be used to predict future levels, albeit with very wide confidence bounds.

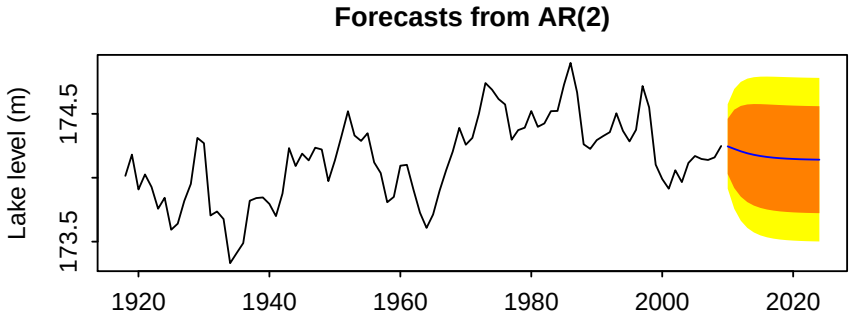


Figure 35: Predictions, 15 years into the future, of Lake Erie levels (m). The shaded areas give 80% and 95% confidence bounds.

The code is:

```
> erie.ar <- ar(Erie)
> library(forecast)
> plot(forecast(erie.ar, h=15), ylab="Lake level (m)")   # 15 time points ahead
```

To see the paramteters of the model that has been fitted, type:

```
> erie.ar
```

Fitting smooth curves to simulations of an autoregressive process

In order to reinforce the points just made, consider results from fitting smooth curves to repeated simulations of an autoregressive process, here with a lag 1 correlation of 0.7:

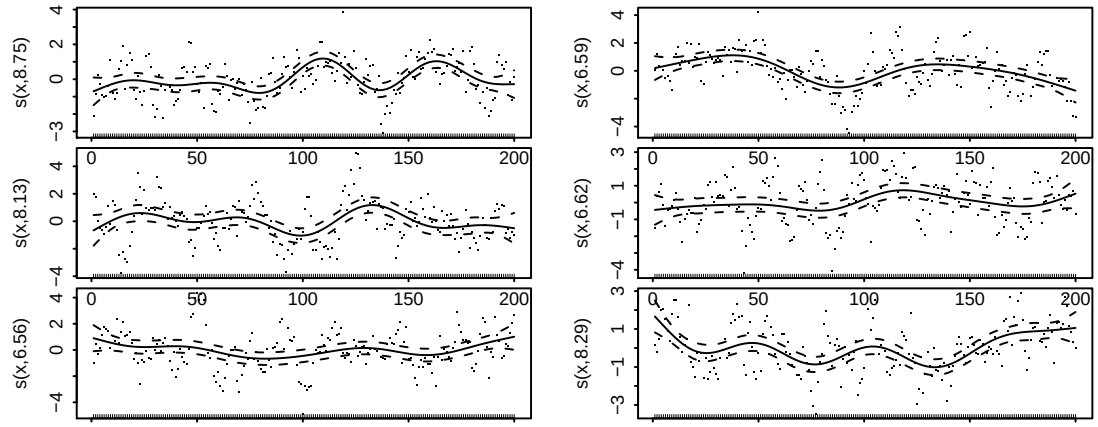


Figure 36:

Code for one of these plots is:

```
> df <- data.frame(x=1:200, y=arima.sim(list(ar=0.7), n=200))
> df.gam <- gam(y ~ s(x), data=df)
> plot(df.gam, residuals=TRUE)
```

The smooth curves, fitting assuming independent errors, are different on each occasion. They serve no useful purpose, if the aim is to generalize beyond the particular realization that generated them.

This brief excursion into a simple form of time series model is intended only to indicate the limitations of automatic smooths. Among several recent introductions to time series that include R code for the computations, note Hyndman et al. (2008), which is designed for use with the *forecasting* bundle of packages.

11.6 Logistic regression with GAM smoothing term

Several articles in medical journals have used data on first class cricketers in the UK to suggest that left-handers, i.e., cricketers who used their left hand for bowling, have a shorter life-span than right-handers. Those articles did not however account for changes over time in the proportions of left-handers. Similar studies have been done for basketballers, again ignoring systematic changes over time in the proportions of left-handers.

For cricketers born between 1840 and 1960, the total numbers are:

```
> ## The cricketer dataset is in the DAAG package
> table(cricketer$left)

right  left
 4859  1101
```

The proportion of left-handers is a little under 18.5%.

A GAM model, with binomial link, will show how the proportion may have changed. Here, the independence assumption is very plausible. There may be occasional father and son successions of left-handers, but these are likely to make only a very small contribution to the total data.

The following does the calculations

```
> library(mgcv)
> hand <- with(cricketer, as.vector(as.vector(unclass(left)-1)))
>   # 0 for left-handers
>   # 1 for right
> hand.gam <- gam(hand ~ s(year), data=cricketer, family=binomial)
```

Figure 37 plots the result:

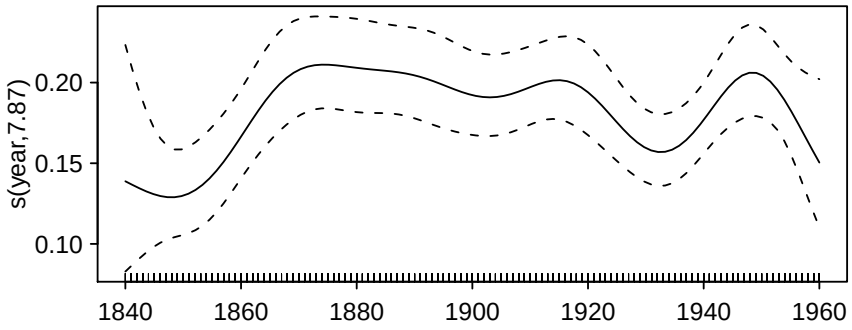


Figure 37: Plot from a GAM model in which the proportion of left-handed cricketers has been modeled as a smooth function of year of birth. The dashed lines are approximate two standard error limits.

The code that did the plotting is:

```
> plot(hand.gam, las=1, xlab="",
       trans=function(x)exp(x)/(1+exp(x)),
       shift=mean(predict(hand.gam)))
```

As a check, Figure 38 does several fits in which left-handers are generated by a random process, with a constant 18.5% proportion:

The code used is:

```
> opar <- par(mfrow=c(5,1), mar=c(0.25, 4.1, 0.25, 1.1))
> for(i in 1:5){
  hand <- sample(c(0,1), size=nrow(cricketer), replace=TRUE,
                 prob=c(0.185, 0.815))
    # 0 for left-handers
    # 1 for right
  hand.gam <- gam(hand ~ s(year),
                  data=cricketer)
  plot(hand.gam, las=1, xlab="",
       rug=if(i<5)FALSE else TRUE,
       trans=function(x)exp(x)/(1+exp(x)),
       shift=mean(predict(hand.gam)))
  }
> par(opar)
```

Occasionally, one or more of these plots will show an apparent pattern.

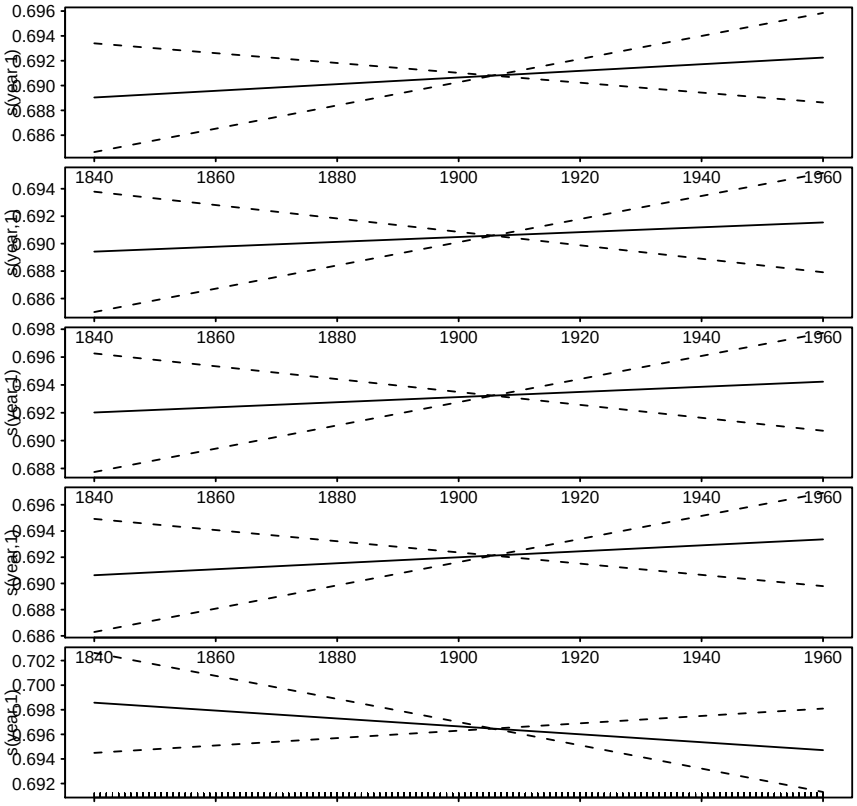


Figure 38: Plots from GAM models, fitted to successive random draws from a population in which the proportion of lefthanded cricketers is as constant 18.5%, irrespective of date of birth.

11.7 Poisson regression with GAM smoothing term

There may be some insight to be gained from changes in the numbers of left-handers and right-handers separately. For this, we assume that left-handers and right-handers are generated by separate Poisson processes, according to rates that vary smoothly over time. The numbers of left-handers and right-handers can be summed for each year, when these annual numbers also follow a Poisson process.

The following code fits the model:

```
> rtlef <- data.frame(with(cricketer, as(table(year, left),"matrix")))
> rtlef$year <- as.numeric(rownames(rtlef))
> denright <-  gam(right ~ s(year), data=rtlef, family=poisson)
> denleft <-  gam(left ~ s(year), data=rtlef, family=poisson)
> fitright <- predict(denright, type="response")
> fitleft <- predict(denleft, type="response")
```

Code that does the plotting, as in Figure 39 is:

```
> opar <- par(mar=c(2.1,4.6,3.1,0.6), mgp=c(2.65, .5,0))
> plot(fitright ~ year, col="blue", main="", type="n", xlab="",
       ylab="Number of cricketers\nborn in given year",
       ylim=c(0, max(rtlef$right)), data=rtlef)
> with(rtlef, lines(fitright ~ year, col="blue", lwd=2))
> with(rtlef, lines(fitleft ~ year, col="purple", lwd=2))
> with(rtlef, lines(I(4*fitleft) ~ year, col="purple", lty=2))
> legend("topleft", legend=expression("Right-handers", "Left-handers",
      "Left-handers "%*%" 4"),
         col=c("blue", "purple", "purple"), pch=c(1,1), lty=c(1,1,2),
         lwd=c(2,2, 1),bty="n", inset=0.01)
> par(opar)
```

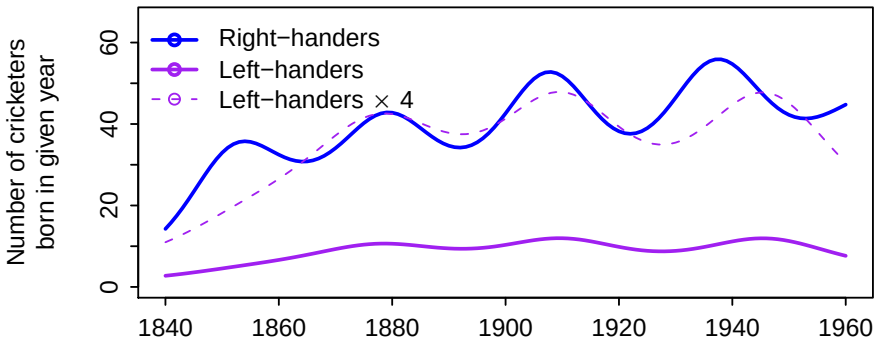


Figure 39: Numbers of left and right-handers have been separately modeled as smooth functions of year of birth. The dashed curve scales up the number of left-handers by a factor of 4, for easy comparison of the patterns of variation of the two curves.

Both curves can be fitted at once, thus:

```
> num.df <- with(cricketer, as.data.frame((table(year, left))))
> num.df$year <- as.numeric(as.character(num.df$year))
> num.gam <-  gam(Freq ~ left + s(year, by=left), data=num.df, family=poisson)
> par(mfrow=c(1,2))
> plot(num.gam)
> par(mfrow=c(1,1))
```

Observe that in `Freq ~ left + s(year, by=left)`, the factor `left` has appeared twice – giving separate offsets (constant terms) for the two curves, and separate patterns of variation about the respective offsets.

Counts that are quasipoisson

To do!

11.8 Exercises

1. Below, we start with an underlying curve to an underlying curve $mu = \frac{\sin x}{x}$ and add one or other of: (i) random normal noise, or (ii) autocorrelated random data. The underlying curve is:

   ```
   > x <- seq(from=-10, to =10, by=0.55)  ## NB: Best avoid x=0
   > mu <- sin(x)/x
   ```

   Consider the model simulations A and B

   ```
   > ## A: Add white noise (random normal noise) with mean 0, specify sd
   > makewhite <- function(mu, sd){
       mu + rnorm(n=length(mu), sd=sd)
     }
   > ## B: Add a series with mean 0, sd=0.2, autocorrelation 0.6
   > ## NB: sd is the standard deviation of the innovations.
   > makeAR1 <- function(mu, sd, ar=0.5){
       mu + arima.sim(list(ar=ar), n=length(mu), sd=sd)
     }
   ```

   The following plots the data for three realisations of each of series A and series B, then using the `gam()` function to fit a smooth curve and add the smooth to the plots:

   ```
   > opar <- par(mfrow=c(2,3))
   > for(i in 1:3){
   ```

```
    y <- makewhite(mu, sd=0.15)
    white.gam <- gam(y ~ s(x))
    plot(white.gam, resid=TRUE, shift=white.gam$coef[1],
         pch=1, se=FALSE, xlab="")
    if(i==1)mtext(side=3, line=0.4, "Add white noise")
  }
> for(i in 1:3){
    y1 <- makeAR1(mu, sd=0.15, ar=0.5)
    ar1.gam <- gam(y1 ~ s(x))
    plot(ar1.gam, resid=TRUE, shift=ar1.gam$coef[1],
         pch=1, se=FALSE, xlab="")
    if(i==1)mtext(side=3, line=0.4, "Add autoregressive 'noise'")
  }
> par(opar)
```

Repeat the plots with `sd=0.45` and with `sd=0.75`. Under what circumstances is the autocorrelated error least likely to distort the smooth? Under what circumstances is a spurious smooth likely?

12 Errors in x

In the discussion so far, it has been assumed, either that the explanatory variables are measured with negligible error or that the interest is in the regression relationship given the observed values of explanatory variables.

This subsection is designed to draw attention to the likely effect of errors in the explanatory variables on regression slope. Discussion will be limited to a relatively simple "classical" errors in x model. For this model the error in x, if large, reduces the chances that the estimated slope will appear statistically significant. Additionally, it reduces the expected magnitude of the slope, i.e., the slope is attenuated. Even with just one explanatory variable x, it is not possible to estimate the magnitude of the error or consequent attenuation from the information shown in a scatterplot of y versus x. For estimating the magnitude of the error, there must be a direct comparison with values that are measured with negligible error.

The discussion will now turn to as study on the measurement of dietary intake. The error in the explanatory variable, as commonly measured, turned out to be larger and of greater consequence than most researchers had been willing to contemplate.

12.1 Measurement of dietary intake

The 36-page Diet History Questionnaire is a Food Frequency Questionnaire (FFQ) that was developed and evaluated at the U.S. National Cancer Institute, for use in large-scale trials that look for dietary effects on cancer and on other diseases. Given the huge scale of some of these trials, some costing US$100,000,000 or more, it has been important to have an instrument that is relatively cheap and convenient. Unfortunately, as the study that is reported in Schatzkin et al (2003) demonstrates, the FFQ seems too inaccurate to serve its intended purpose.

This FFQ queries frequency of intake over the previous year for 124 food items, asking details of portion sizes for most of them. There are supplementary questions on such matters as seasonal intake and food type. More detailed food records may be collected at specific times, which can then be used to calibrate the FFQ results. One such instrument is a 24-hour dietary recall, questioning participants on their dietary intake in the previous 24 hours. The accuracy of the 24-hour dietary recall was a further concern of the Schatzkin et al (2003) study. Doubly Labeled Water, which is a highly expensive biomarker, was used as an accurate reference instrument.

Schatzkin et al (2003) reported measurement errors where the standard deviation for estimated energy intake was seven times the standard deviation, between different individuals, of the reference. Additionally, Schatzkin et al (2003) found a bias in the relationship between FFQ and reference that

further reduced the attenuation factor, to 0.04 for women and to 0.08 for men. For the relationship between the 24 hour recalls and the reference, the attenuation factors were 0.1 for women and 0.18 for men, though these can be improved by the use of repeated 24-hour recalls. These errors were much larger than most researchers had been willing to contemplate.

The results reported in Schatzkin et al (2003) raise serious questions about what such studies can achieve, using instruments such as those presently available that are sufficiently cheap and convenient that they can be used in large studies. The measurement instrument and associated study design issues have multi-million dollar implications. Carroll (2004) gives an accessible summary of the issues.

This is a multi-million dollar issue. The following prospective studies that use such instruments are complete or nearly complete:

NHANES:	n = 3,145 women aged 25-50
(National Health and Nutrition Examination Survey)	
Nurses Health Study:	n = 60,000+
Pooled Project:	n = 300,000+
Norfolk (UK) study:	n = 15,000+
AARP:	n = 250,000+

Only 1 prospective study has found firm evidence suggesting a fat and breast cancer link, and 1 has found a negative link. The lack of consistent (even positive) findings led to the Women's Health Initiative Dietary Modification Study in which 60,000 women have been randomized to two groups: healthy eating and typical eating. Objections to this study are:

- Cost ($100,000,000+)
- Can Americans can really lower % fat calories from to 20%, from the current 35%
- Even if the study is successful, difficulties in measuring diet mean that we will not know what components led to the decrease in risk.

12.2 A simulation of the effect of measurement error

Suppose that the underlying regression relationship that is of interest is

$$y_i = \alpha + \beta z_i + \epsilon_i \ (i = 1, \ldots, n)$$

and that the measured values are

$$x_i = z_i + \eta_i$$

where

$$\text{var}[\epsilon_i] = \sigma^2; \ \text{var}[\eta_i] = \tau^2$$

Figure 40 shows the effect. If τ is 40% of the standard deviation in the x direction, i.e., $\tau = 0.4 s_z$, the effect on the slope is modest. If $\tau = 2 s_z$, the attenuation is severe.

The expected value of the attenuation in the slope is, to a close approximation

$$\lambda = \frac{1}{1 + \tau^2 / s_z^2}$$

where $s_z = \sqrt{\sum_{i=1}^{n} (z_i - \bar{z})^2}$. If $\tau = 0.4 s_z$, then $\lambda \simeq 0.86$.

Whether a reduction in slope by a factor of 0.86 is of consequence will depend on the nature of the application. Often there will be more important concerns. Very small attenuation factors (large attenuations), e.g. less than 0.1 such as were found in the Schatzkin et al (2003) study, are likely to have serious consequences for the use of analysis results.

Points to note are:

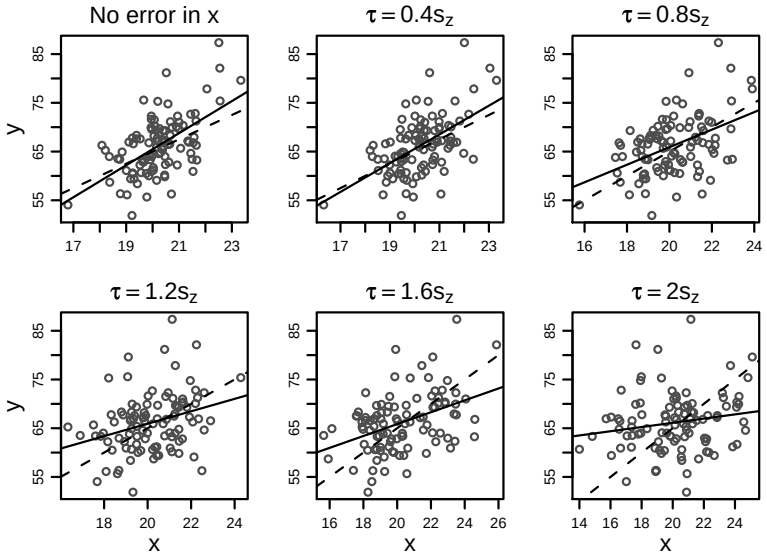


Figure 40: The solid line in each panel is the fitted regression line for y on x. Different panels have different choices τ for the SD of the independent errors in x. The underlying relationship, (dashed line) is in each instance $y = 15 + 2.5\,z$. The SD of values of z (measured without error), is $s_z = \sqrt{\sum_{i=1}^{n}(z_i - \bar{z})^2}$.

- From the data used in the panels of Figure 40, it is impossible to estimate τ, or to know the underlying z_i values. This can be determined only from an investigation that compares the x_i with an accurate, i.e., for all practical purposes error-free, determination of the z_i. The Schatzkin et al (2003) study that will be discussed below made use of a highly expensive reference instrument, too expensive for standard use, to assess and calibrate the widely used cheaper measuring instruments.
- A test for $\beta = 0$ can be undertaken in the usual way, but with reduced power to detect an effect that may be of interest.
- The t-statistic for testing $\beta = 0$ is affected in two ways; the numerator is reduced by an expected factor of λ, while the standard error that appears in the numerator increases. Thus if $\lambda = 0.1$, the sample size required to detect a non-zero slope is inflated by more than the factor of 100 that the effect on the slope alone would suggest.

In social science, the ratio τ^2/s_z^2 has the name *reliability*. As Fuller (1987) points out, a better name is reliability ratio.

12.3 Errors in variables – multiple regression

Again, attention will be limited to the classical errors in x model. Where one only of several variables is measured inaccurately, its coefficient may on that account not appear statistically significant, or be severely attenuated. For remaining variables (measured without error) possible scenarios include: the coefficient suggests a relationship when there is none, or the coefficient is reversed in sign. Where several variables are measured with error, there is even more room for misleading and counter-intuitive coefficient values.

13 Further issues for the use and interpretation of regression models

13.1 Data collection biases

Large biases can arise from the way that data have been collected. The Literary Digest poll that was taken prior to the US 1936 Presidential election, where Roosevelt had 62% of the vote rather than the

predicted 43%, is an infamous example. The estimate of 43% was based on a sample, highly biased as it turned out, of 2.4 million!

The problems that arise can be exacerbated by more directly statistical problems, i.e., issues that it is important to note even if random samples are available. Estimates of regression coefficients, or other model parameters, cannot necessarily be taken at their face value.

13.2 Model and/or variable selection bias

13.2.1 Model selection

When the model is fitted to the data used to select the model from a set of possible models, the effect is anti-conservative. Thus, standard errors will be smaller than indicated by the theory, and coefficients and t-statistics larger. Such anti-conservative estimates of standard errors and other statistics may, unless the bias is huge, nevertheless provide the useful guidance. Use of test data that are separate from data used to develop the model deals with this issue.

There is a further important issue, that use of separate test data does not address. Almost inevitably, none of the models on offer will be strictly correct. Mis-specification of the fixed effects, and to a lesser extent of the random effects, is likely to bias model estimates, at the same time inflating the error variance or variances, i.e., it may to some extent work in the opposite direction to selection effects.

13.2.2 Variable selection and other multiplicity effects

Variable selection has the same, or greater, potential for bias as model selection. This is an especial issue for the analysis of microarray and other genomic data, where a small number of gene expression measures, perhaps of the order of 5 - 20, may be selected from 10,000 or more. See Ambroise and McLachlan (2001) for a critique of papers where the authors have fallen prey to this trap. This can also be an issue for graphs that are based on the data that remain after selection.

Empirical accuracy assessments seem the only good way to address the major issues that can arise here. There are traps for data analysts who have not taken adequate account of the implications of selecting, for use in a regression or discriminant or similar analysis, a small number of variables ("features") from a much larger number. Maindonald (2003) gives a relatively elementary account of this matter, which should be accessible to non-specialists. The paper Ambroise and McLachlan (2001) is a careful examination of several examples, all concerned with the use of discriminant methods in connection with microarray data, from the literature. The same effects can arise from model tuning. Cross-validation is a key tool in this context. This, or the bootstrap, seems the only good way to allow for the skewing of results that can arise from potentially huge variable selection effects. Any model tuning and/or variable selection must be repeated at each cross-validation fold.

13.3 Does screening reduce deaths from gastric cancer?

The issue here is that of comparing groups who may differ in respects other then the respect that is under investigation. In other words, there are likely to be hidden variables.

Patients who had surgery for gastric cancer were divided into two groups – those who had presented with cancer at a hospital or doctor's surgery, and those who had been diagnosed with cancer as a result of screening. Mortality was assessed in the 5 years following surgery:

	Mortality	Number
Unscreened Group	41.9%	352
Screened Group	28.2%	308

Table 5: Mortality in five-year period following surgery for cancer, classified according to whether patients presented with cancer, or cancer was detected by screening.

What are the possible explanations for the higher mortality in the unscreened group?

Screening may be catching cancer early, thus reducing the risk of death.

Cancers detected by screening may be at an earlier stage of development, and thus less immediately fatal.

Some cancers detected by screening may be of a less dangerous type, that progress slowly, or may never progress to become fatal.

All three effects may contribute to the difference.

Question: What are likely/possible missing variables/factors, for these data?

The appropriate approach is to identify several large groups of patients, randomly assigning groups for screening or no screening. Study participants are then followed for, e.g., the next decade. One study[11] classified 24,134 survey recipients as screened or unscreened, according as they had been screened, or not, in the previous year. It then followed them up for 40 months:

	Male		Female	
	Unscreened (n = 6,536)	Screened (n = 4,934)	Unscreened (n = 8,456)	Screened (n = 4,208)
Gastric cancer				
No. of deaths	19	8	9	4
Mortality rate	86.8	53.0	31.0	40.2
All causes				
No. of deaths	473	237	403	97
Mortality rate	2,199.0	1,593.1	1,370.7	829.4

Table 6: Mortality rates (deaths per 100,000 person years), from gastric cancer and from all causes.

Question: What are likely/possible missing variables/factors, for these data?

13.4 Alcohol consumptions and risk of coronary heart disease

Consider now observational studies of the effects of modest wine-drinking on heart disease (Jackson et al., 2005). There are a large number of factors that affect heart disease – genetic, lifestyle, diet, and so on. Any analysis of observational data that tries to account for their joint effect will inevitably be simplistic. The assumptions made about the form of the response (usually, a straight line on a suitably transformed scale) will be simplistic. Simplistic assumptions will be made about interaction effects (how does alcohol intake interact with other dietary habits?), and so on.

[11] used in: Inaba et al. 1999: Evaluation of a Screening Program on Reduction of Gastric Cancer Mortality in Japan: Preliminary Results from a Cohort Study. Preventive Medicine 29: 102-106

No. of events (mortality/CHD)	All-cause mortality	Coronary heart disease
Men		
Never drink (16/43)	2.3 (1.2 – 3.8)	1.8 (1.3 – 2.5)
Special occasions (33/76)	1.4 (0.9 – 2.2)	1.1 (0.8 – 1.4)
1–2 times/month (37/93)	1.5 (1.0 – 2.2)	1.0 (0-8 – 1.3)
1–2 times/week (82/306)	1 (baseine)	1.0 (baseline)
Almost daily (52/219)	0.9 (0.7 – 1.3)	0.9 (0.8 – 1.1)
Twice a day or more (22/41)	2.5 (1.5 – 4.1)	1.1 (0.8 – 1.5)
Women		
Never drink (9/43)	1.5 (0.7 – 3.5)	1.8 (1.3 – 2.8)
Special occasions (40/127)	1.5 (0.7 – 3.5)	1.2 (0.9 – 1.5)
1–2 times/month (14/61)	1.7 (1.0 – 2.9)	1.0 (0-8 – 1.8)
1–2 times/week (26/137)	1 (baseine)	1.0 (baseline)
Almost daily (18/59)	1.3 (0.7 – 2.4)	0.8 (0.6 – 1.2)
Twice a day or more (5/7)	4.8 (1.8 – 12.7)	1.3 (0.6 – 2.8)

Table 7: Increased risk of mortality, relative to baseline, according to frequency of alcohol consumption. Factors for which adjustment was made were age, smoking, employment grade, blood cholesterol, blood pressure, body mass index, and general health as measured by a score from a questionnaire. CHD was recorded as an outcome if there was an episode of fatal or non-fatal cornonary heart disease.

Here, there are may factors for which there should be an adjustment. After adjusting for the effects of other factors. how does level of alcohol consumption affect risk of death? The method of analysis used is survival analysis, which will not be covered in this course. Think of it as an extension of the regression methodology that will be considered later in the course, with the risk of death relative to the baseline as the outcome. (Risk is expressed as a probability density; in this context it has the name "hazard" rate.)

Britton & Marmot (2004) report on an 11-year follow-up of a study of 10,308 London-based civil servants aged 35-55 years at baseline (33% female). Adjustments were made for age, smoking, employement grade, blood cholesterol, blood pressure, body mass index, and general health as measured by a score from a questionnaire. Table 7 shows the estimated ratio of risk relative to the baseline line, i.e., to the risk from all other factors.

Thus, it looks as though modest levels of alcohol consumption may be beneficial. However the results remain controversial. There may for example be lifestyle factors, associated with levels of alcohol consumption, for which factors such as employment have not made adequate adjustment. If such factors are correlated with frequency of drinking, this might in part explain the result. See especially Jackson et al. (2005).

Note also another source of evidence, derived from so-called Mendelian randomization studies. (Mendelian dose assignment would be a more accurate description than "Mendelian randomization".) Half of the Japanese population is homozygous or heterozygous for a non-functional variant of the gene ALDH2, making them unable to metabolise alcohol properly, with unpleasant consequences. The effect is more serious for the homozygotes than for the heterozygotes. The result is that homozygotes heavily curtail their alcohol consumption and heterozygotes curtail it to some lesser extent. The incidence of CHD closely reflects results predicted by Britton & Marmot (2004). At the same time, no association was apparent between genotype and other factors implicated in CHD. See Davey Smith & Ebrahim (2005).

13.5 Freakonomics

Several of the studies that are discussed in Leavitt and Dubner (2005), some with major public policy relevance, relied to an extent on regression methods – usually generalized linear models rather than linear models. References in the notes at the end of their book allow interested readers to pursue technical details of the statistical and other methodology. The conflation of multiple sources of insight

and evidence is invariably necessary, in such studies, if conclusions are to carry conviction. Ignore the journalistic hype, obviously the responsibility of the second author, in the preamble to each chapter.

13.6 Further reading

See Rosenbaum (1999) and Rosenbaum (2002) for a comprehensive overview of issues that commonly arise in the analysis of observational data, and of approaches that may be available to handle some of the major sources of potential difficulty.

Part V
Discrimination and Classification

The methods described here have the character of regression models where the outcome is categorical, one of g classes. For example, the `fgl` dataset has measurements of each on nine physical properties, for 214 samples of glass that are classified into six different glass types.

Linear Discriminant Analysis (LDA), which will be discussed first, may be contrasted with the strongly non-parametric random forest method that uses an ensemble of trees. See Maindonald & Braun (2010, Section 11.7).

A good strategy for getting started is to fit a linear discriminant model with main effects only, comparing the accuracy with that from a random forest analysis. If the random forest analysis gives little or no improvement, the linear discriminant model may be hard to better. There is much more that can be said, but this is often a good starting strategy.

See Ripley (1996); Venables and Ripley (2002); Maindonald & Braun (2010, Section 12.2).

14 Linear Methods for Discrimination

Notation and types of model

Observations are rows of a matrix $\mathbf{X}$ with p columns. The vector $\mathbf{x}$ is a row of $\mathbf{X}$, but in column vector form. The outcome is categorical, one of g classes.

Methods discussed here will all use as predictors continuous non-linear functions of the columns of $\mathbf{X}$. There are several mechanisms for such modeling that involve the use of spline basis terms.

As before, observations are rows of a matrix $\mathbf{X}$ with p columns. The vector $\mathbf{x}$, is a row of $\mathbf{X}$, but in column vector form.

The outcome is categorical, one of g classes, where now g may be greater than 2. The matrix $\mathbf{W}$ estimates the within class variance-covariance matrix, while $\mathbf{B}$ estimates the between class variance-covariance matrix. Details of the estimators used are not immediately important. Note however that they may differ somewhat between computer programs.

14.1 `lda()` and `qda()`

The functions that will be used are `lda()` and `qda()`, from the *MASS* package. The function `lda()` implements linear discriminant analysis, while `qda()` implements quadratic discriminant analysis. Quadratic discriminant analysis is an adaptation of linear discriminant analysis to handle data where the variance-covariance matrices of the different classes are markedly different. For $g = 2$ the logistic regression model, fitted using R's `glm()` function, is closely analagous to the linear discriminant model that is fitted using `lda()`. The difference can however be important.

An attractive feature of `lda()` is that the search for a discriminant rule leads to a representation of a subspace of the column space of $\mathbf{X}$ in r-dimensional space. Providing that the rank of $\mathbf{X}$ is at least $g-1$, $r = g-1$. Use of a spectral decomposition leads to r sets of scores, where each set of scores explains a successively smaller (or at least, not larger) proportion of the sum of squares of differences of group means from the overall mean. The r sets of scores can be examined using a pairs plot.

With three groups, two dimensions will account for all the variation. A scatterplot is then a geometrically complete representation of what the analysis has achieved. With larger numbers of groups, it will often happen that a two or at most three dimensions will account for most of the variation.

The plots that it yields are a major part of the appeal of `lda()`. Where `lda()` does not work well, they may hint at what type of alternative method might be preferred. They can be useful for identifying subgroups of the orginal g groups, and for identifying points that may be misclassified

14.1.1 `lda()` and `qda()` – theory

The functions `lda()` and `qda()` in the *MASS* package implement a Bayesian decision theory approach. Points to note are:

- The methodology is implemented within a Bayesian framework. By default, the prior probabilities for the various categories are taken to be the relative frequencies for those categories. The classification rule changes if the frequencies are changed from the default.
- For any given classification rule, the overall accuracy (proportion correctly classified) changes if the prior probabilities are changed.
- For estimating the accuracy for a given target population, the prior probabilities should be the proportions in that population, not the proportions in the sample.

More specifically:

- A prior probability π_c is assigned to the cth class ($c = 1, \ldots g$).
- The density $p(\mathbf{x}|c)$ of $\mathbf{x}$, conditional on the class c, is assumed multivariate normal, i.e., rows of $\mathbf{X}$ are sampled independently from a multivariate normal distribution.
- For linear discrimination, classes are assumed to have a common covariance matrix Σ, or more generally a common $p(\mathbf{x}|c)$. For quadratic discrimination, different $p(\mathbf{x}|c)$ are allowed for different classes.
- Use Bayes' formula to derive $p(c|\mathbf{x})$. The allocation rule that gives the largest expected accuracy chooses the class with maximal $p(c|\mathbf{x})$; this is the Bayes' rule.
- More generally, assign cost L_{ij} to allocating a case of class i to class j, and choose c to minimize $\sum_i L_{ic} p(i|\mathbf{x})$.

Note that `lda()` and `qda()` use the prior weights, if specified, as weights in combining the within class variance-covariance matrices.

Using Bayes' formula

$$\begin{aligned} p(c|\mathbf{x}) &= \frac{\pi_c p(\mathbf{x}|c)}{p(\mathbf{x})} \\ &\propto \pi_c p(\mathbf{x}|c) \end{aligned}$$

The Bayes' rule maximizes $p(c|\mathbf{x})$. For this it is sufficient, for any given $\mathbf{x}$, to maximize

$$\pi_c p(\mathbf{x}|c)$$

or, equivalently, to maximize

$$\log(\pi_c) + \log(p(\mathbf{x}|c))$$

Now assume $p(\mathbf{x}|c)$ is multivariate normal, i.e.,

$$p(\mathbf{x}|c) = (2\pi)^{\frac{p}{2}} \mid \Sigma_c \mid)^{-\frac{1}{2}} exp(-\frac{1}{2} Q_c)$$

where

$$Q_c = (\mathbf{x} - \mu_c)^T \Sigma_c^{-1} (\mathbf{x} - \mu_c)$$

Then

$$\log(\pi_c) + \log(p(\mathbf{x}|c)) = \log(\pi_c) - \frac{1}{2} Q_c + \frac{p}{2} \log(2\pi) - \frac{1}{2} \log(\mid \Sigma_c \mid)$$

Leaving off the $\log(2\pi)$ and multiplying by -2, this is equivalent to minimization of

$$Q_c + \log(\mid \Sigma_c \mid) - 2\log(\pi_c) = (\mathbf{x} - \mu_c)^T \Sigma_c^{-1} (\mathbf{x} - \mu_c) + \log(\mid \Sigma_c \mid) - 2\log(\pi_c)$$

The observation $\mathbf{x}$ is assigned to the group for which

$$(\mathbf{x} - \mu_c)^T \Sigma_c^{-1} (\mathbf{x} - \mu_c) + \log(\mid \Sigma_c \mid) - 2\log(\pi_c)$$

is smallest.

Set $\mu_c = \bar{\mathbf{x}}_c$, and replace $\mid \Sigma_c \mid$ by an estimate $\widehat{\Sigma_c}$.

[Note that the usual estimate of the variance-covariance matrix (or matrices) is positive definite, providing that the same observations are used in calculating all elements in the variance-covariance matrix and $\mathbf{X}$ has no redundant columns.]

Then $\mathbf{x}$ is assigned to the group to which, after adjustments for possible differences in π_c and $\mid \Sigma_c \mid$, the Mahalanobis distance

$$(\mathbf{x} - \bar{\mathbf{x}}_c)^T \widehat{\Sigma_c}^{-1} (\mathbf{x} - \bar{\mathbf{x}}_c)$$

of $\mathbf{x}$ from $\mathbf{x}_c$ is smallest.

If a common variance-covariance matrix $\Sigma_c = \Sigma$ can be assumed, a linear transformation is available to a space in which the Mahalanobis distance becomes a Eucliean distance. Replace $\mathbf{x}$ by

$$\mathbf{z} = (U^T)^{-1}\mathbf{x}$$

and $\bar{\mathbf{x}}_c$ by $\bar{\mathbf{z}}_c = (U^T)^{-1}\bar{\mathbf{x}}_c$ where U is an upper triangular matrix such that $U^T U = \widehat{\Sigma}$. Then

$$(\mathbf{x} - \mu_c)^T \mathbf{W}^{-1} (\mathbf{x} - \mu_c) = (\mathbf{z} - \bar{\mathbf{z}}_c)^T (\mathbf{z} - \bar{\mathbf{z}}_c)$$

which in the new space is the squared Euclidean distance to from $\mathbf{z}$ to $\bar{\mathbf{z}}_c$.

A result of the `lda` calculations is thus to determine, for each observation, a distance from each of the g group means. In general, these means define a hyperplane in $g-1$ dimensional space. Three group means define a plane, four group means define a 3-dimensional hyperplane, and so on.

Note: For estimation of the posterior probabilities, the simplest approach is that described above. Thus, replace $p(c|\mathbf{x};\theta)$ by $p(c|\mathbf{x};\hat{\theta})$ for calculation of posterior probabilities (the 'plug-in' rule). Here, θ is the vector of parameters that must be estimated. The functions `predict.lda()` and `predict.qda()` offer the alternative estimate `method="predictive"`, which takes account of uncertainty in $p(c|\mathbf{x};\hat{\theta})$. Note also `method="debiased"`, which may be a reasonable compromise between `method="plugin"` and `method="predictive"`

14.1.2 Canonical discriminant analysis

Here we assume a common variance-covariance matrix. As described above, replace $\mathbf{x}$ by

$$\mathbf{z} = \mathbf{U}^{\mathbf{T}^{-1}}\mathbf{x}$$

where $\mathbf{U}$ is an upper triangular matrix such that $\mathbf{U}^{\mathbf{T}}\mathbf{U} = \widehat{\Sigma}$. The estimated variance-covariance matrix of $\mathbf{z}$ is then the identity matrix. Observe that

$$\begin{aligned}
\widehat{\mathrm{var}[\mathbf{z}]} &= \mathrm{E}[\mathbf{U}^{\mathbf{T}^{-1}}(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T \mathbf{U}^{-1}] \\
&= \mathbf{U}^{\mathbf{T}^{-1}} \mathrm{E}[(\mathbf{x} - \boldsymbol{\mu}(\mathbf{x} - \boldsymbol{\mu})^T] \mathbf{U}^{-1} \\
&= \mathbf{U}^{\mathbf{T}^{-1}} \widehat{\Sigma} \mathbf{U}^{-1} \\
&= \mathbf{I}_p, \quad \text{where } \mathbf{I}_p \text{ is the } p \times p \text{ identity matrix.}
\end{aligned}$$

The between classes variance-covariance matrix becomes

$$\tilde{\mathbf{B}} = \mathbf{U}^{\mathbf{T}^{-1}} \mathbf{B} \mathbf{U}^{-1}$$

The ratio of between to within class variance of the linear combination $\alpha^T \mathbf{z}$ is then

$$\begin{aligned} \mathrm{E}[\alpha^T(\mathbf{z}-\boldsymbol{\mu})(\mathbf{z}-\boldsymbol{\mu})^T\alpha &= \frac{\alpha^T\tilde{\mathbf{B}}\alpha}{\alpha^T\alpha} \\ &= \alpha^T\tilde{\mathbf{B}}\alpha, \text{ subject to the constraint } \| \alpha \| = 1. \end{aligned}$$

The matrix $\tilde{\mathbf{B}}$ admits the principal components decomposition

$$\tilde{\mathbf{B}} = \lambda_1\mathbf{u_1}\mathbf{u_1^T} + \lambda_2\mathbf{u_2}\mathbf{u_2^T} + \ldots + \lambda_r\mathbf{u_r}\mathbf{u_r^T}$$

The choice $\alpha = \mathbf{u_1}$ maximizes the ratio of the between to the within group variance, a fraction λ_1 of the total. The choice $\alpha = \mathbf{u_2}$ accounts for the next largest proportion λ_2, and so on.

The vectors $\mathbf{u}_1, \ldots \mathbf{u}_r$ are known as "linear discriminants" or "canonical variates". Scores, which are conveniently centered about the mean over the data as a whole, are available on each observation for each discriminant. These locate the observations in r-dimensional space, where r is at most $\min(g-1, p)$. A simple rule is to assign observations to the group to which they are nearest, i..e., the distance d_c is smallest in a Euclidean distance sense.

For plotting in two dimensions, one takes the first two sets of discriminant scores. A point $\mathbf{z}_i$ that is represented as

$$\zeta_{i1}\mathbf{u}_1 + \zeta_{i2}\mathbf{u}_2 + ... + \zeta_{ir}\mathbf{u}_r$$

is plotted in two dimensions as (ζ_{i1}, ζ_{i2}), or in three dinsensions as $(\zeta_{i1}, \zeta_{i2}, \zeta_{i3})$. The amounts by which the original columns of $\mathbf{x}_i$ need to be multiplied to give ζ_{i1} are given by the first column of the list element `scaling` in the `lda` object. For ζ_{i2}, the elements are those in the second column, and so on. See the example below.

As variables have been scaled so that within group variance-covariance matrix is the identity, the variance in the transformed space is the same in every direction. An equal scaled plot should therefore be used to plot the scores.

14.1.3 Linear Discriminant Analysis – Fisherian and other

Fisher's linear disciminant analysis was a version of canonical discriminant analysis that used a single discriminant axis. The more general case, where there can be as many as $r = \min(g-1, p)$ discriminant functions, is described here.

The theory underlying `lda()` assigns $\mathbf{x}$ to the class that maximizes the likelihood. This is equivalent to choosing the class c that minimizes $d_c + \log(\pi_c)$, where if the same estimates are used for $\mathbf{W}$ and $\mathbf{B}$, d_c is the distance as defined for Fisherian linear discriminant analysis. Recall that π_c is the prior probability of class c.

The output from `lda()` includes the list element `scaling`, which is a matrix with one row for each column of $\mathbf{X}$ and one column for each discriminant function that is calculated. This gives the discriminant(s) as functions of the values in the matrix $\mathbf{X}$.

There are two ways that one can run `lda()` and/or `qda()`:

- With the argument `CV=TRUE`, leave-one-out cross-validation is used to return a list with components `class` (the class assigned by the cross-validation) and `posterior` (the posterior probabilities).
- For purposes other than leave-one-out cross-validation, use the argument `CV=FALSE`, which is the default.

In the sequel, we will need the *MASS* package, which has functions for linear and quadratic discriminant analysis. It also has the `fgl` dataset.

```
> library(MASS)
```

14.2 Example – analysis of the forensic glass data

The data frame `fgl` in the *MASS* gives 10 measured physical characteristics for each of 214 glass fragments that are classified into 6 different types. As noted above, the data frame `fgl` has 10 measured physical characteristics for each of 214 glass fragments that are classified into 6 different types.

First, fit a linear discriminant analysis, and use leave-one-out cross-validation to check the accuracy, thus:

```
> fglCV.lda <- lda(type ~ ., data=fgl, CV=TRUE)
> tab <- table(fgl$type, fglCV.lda$class)
> ## Confusion matrix
> print(round(apply(tab, 1, function(x)x/sum(x)), digits=3))
```

```
        WinF WinNF   Veh   Con  Tabl  Head
  WinF  0.729 0.237 0.647 0.000 0.111 0.034
  WinNF 0.229 0.684 0.353 0.462 0.222 0.069
  Veh   0.043 0.000 0.000 0.000 0.000 0.000
  Con   0.000 0.039 0.000 0.462 0.000 0.034
  Tabl  0.000 0.026 0.000 0.000 0.556 0.000
  Head  0.000 0.013 0.000 0.077 0.111 0.862
```

Now run the function with `CV=FALSE`, and examine the output:

```
> opt <- options(digits=2)
> fgl.lda <- lda(type ~ ., data=fgl)
> fgl.lda
```

```
Call:
lda(type ~ ., data = fgl)

Prior probabilities of groups:
 WinF WinNF   Veh   Con  Tabl  Head
0.327 0.355 0.079 0.061 0.042 0.136

Group means:
          RI Na   Mg  Al Si    K   Ca     Ba    Fe
WinF   0.718 13 3.55 1.2 73 0.45  8.8 0.0127 0.057
WinNF  0.619 13 3.00 1.4 73 0.52  9.1 0.0503 0.080
Veh   -0.036 13 3.54 1.2 72 0.41  8.8 0.0088 0.057
Con    0.928 13 0.77 2.0 72 1.47 10.1 0.1877 0.061
Tabl  -0.544 15 1.31 1.4 73 0.00  9.4 0.0000 0.000
Head  -0.884 14 0.54 2.1 73 0.33  8.5 1.0400 0.013

Coefficients of linear discriminants:
     LD1   LD2  LD3    LD4   LD5
RI  0.31 0.029 0.36  0.247 -0.80
Na  2.38 3.165 0.46  6.924  2.40
Mg  0.74 2.986 1.57  6.850  2.80
Al  3.34 1.725 2.20  6.419  0.94
Si  2.45 3.006 1.70  7.542  0.96
K   1.57 1.862 1.29  8.076  2.82
Ca  1.01 2.373 0.65  6.697  3.71
Ba  2.31 3.443 2.60  6.438  4.41
Fe -0.51 0.217 1.20 -0.045 -1.30
```

```
Proportion of trace:
  LD1   LD2   LD3   LD4   LD5
0.815 0.117 0.041 0.016 0.011

> options(opt)
```

Observe that 93% of the information, as measured by the trace, is in the first two discriminants. We can plot scores on these discriminants, one against the other, as in Figure 41:

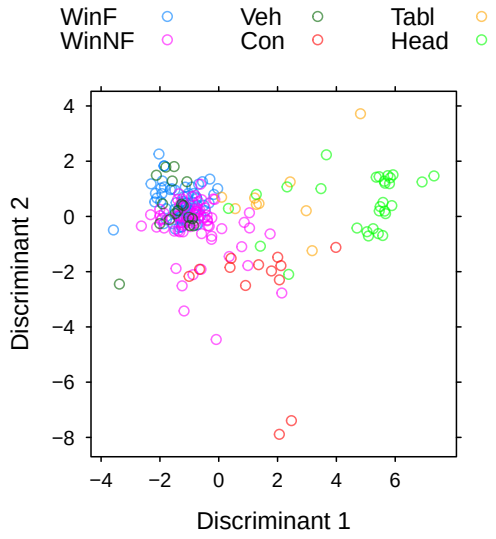


Figure 41: Visual representation of scores derived from *linear discriminant analysis*, for the forensic glass data. A six-dimensional pattern of separation between the categories has been collapsed down to two dimensions. Some categories may therefore be better distinguished than is evident from this figure.

The code for Figure 41 is:

```
> library(lattice)
> scores <- predict(fgl.lda)$x
> gph <- xyplot(scores[,2] ~ scores[,1], groups=fgl$type,
                xlab="Discriminant 1", ylab="Discriminant 2",
                aspect=1, scales=list(tck=0.4), auto.key=list(columns=3),
                par.settings=simpleTheme(alpha=0.6),
                title="Plot shows first two linear discriminant scores")
> print(gph)
```

The discriminant functions

The following demonstrates the use of the information, giving details of the linear discriminant functions, in the component `scaling` of the model object `fgl.lda`:

```
library(MASS)
fgl.lda <- lda(type ~ ., data=fgl)
scores <- predict(fgl.lda, dimen=5)$x  # Default is dimen=2
## Now calculate scores from other output information
checkscores <- model.matrix(fgl.lda)[, -1] %*% fgl.lda$scaling
## Center columns about mean
checkscores <- scale(checkscores, center=TRUE, scale=FALSE)
plot(scores[,1], checkscores[,1])  # Repeat for remaining columns
```

93% of the information, as measured by the trace, is in the first two discriminants.

14.2.1 Two groups – comparison with logistic regression

Logistic regression, which can be handled using R's function `glm()`, is a special case of a Generalized Linear Model (GLM). The approach is to model $p(c|\mathbf{x};\hat{\theta})$ using a parametric model that may be the same logistic model as for linear and quadratic discriminant analysis.

In this context it is convenient to change notation slightly, and give $\mathbf{X}$ an initial column of ones. In the linear model and generalized linear model contexts, $\mathbf{X}$ has the name "model matrix".

The vector $\mathbf{x}$ is a row of $\mathbf{X}$, but in column vector form. Then if π is the probability of membershipÂăin the second group, the model assumes that

$$\log(\pi/(1-\pi) = \beta'\mathbf{x}$$

where β is a constant.

Compare logistic regression with linear discriminant analysis:

- Inference is conditional on the observed $\mathbf{x}$. A model for $p(\mathbf{x}|c)$ is not required. Results are therefore more robust against the distribution $p(\mathbf{x}|c)$.
- Parametric models with "links" other than the logit $f(\pi) = \log(\pi/(1-\pi)$ are available. Where there are sufficient data to check whether one of these other links may be more appropriate, this should be done. Or there may be previous experience with comparable data that suggests use of a link other than the logit.
- Observations can be given prior weights.
- There is no provision to adjust predictions to take account of prior probabilities, though this can be done as an add-on to the analysis.
- The fitting procedure minimizes the deviance, which is twice the difference between the log-likelihood for the model that is fitted and the loglikelihood for a 'saturated' model in which predicted values from the model equal observed values. This does not necessarily maximize predictive accuracy.
- Standard errors and Wald statistics (roughly comparable to t-statistics) are provided for parameter estimates. These are based on approximations that may fail if predicted proportions are close to 0 or 1 and/or the sample size is small.

14.2.2 How important are the linearity assumptions?

The linearity assumptions are restrictive, even allowing for the use of regression spline terms to model non-linear effects. It is not obvious how to choose the appropriate degree for each of a number of terms. The attempt to investigate and allow for interaction effects adds further complications. In order to make progress with the analysis, it may be expedient to rule out any but the most obvious interaction effects. These issues affect regression methods (including GLMs) as well as discriminant methods.

14.2.3 Low-dimensional Graphical Representation

In linear discriminant analysis, discriminant scores in as many dimensions as seem necessary are used to classify the points. These scores can be plotted. Each pair of dimensions gives a two-dimensional projection of the data. If there are three groups and at least two explanatory variables, the two-dimensional plot is a complete summary of the analysis. Even where higher numbers of dimensions are required, two dimensions may capture most of the information. This can be checked.

With most other methods, a low-dimensional representation does not arise so directly from the analysis. An approach that will be demonstrated with random forests, can be adapted for use with other methods.

14.3 A further example – cuckoo egg lengths

To illustrate linear and quadratic discriminant analysis, we will use the data set `cuckoos` (*DAAG* package), in the first instance limiting attention to hedge sparrow and wren nests. This dataset provides measurements on the length and breadth of eggs of each of six host species. Because there are just two measurements, a two-dimensional representation provides a complete description of the results of the analysis. Any plot of scores will be a rotated version of the plot of `length` versus `breadth`.

```
> library(DAAG); library(MASS); library(latticeExtra)
> cuckoos.lda <- lda(species ~ length + breadth, data=cuckoos)
> cuckoos.lda

Call:
lda(species ~ length + breadth, data = cuckoos)

Prior probabilities of groups:
hedge.sparrow  meadow.pipit  pied.wagtail         robin    tree.pipit
    0.1166667     0.3750000     0.1250000     0.1333333     0.1250000
         wren
    0.1250000

Group means:
                length  breadth
hedge.sparrow 23.11429 16.76429
meadow.pipit  22.29333 16.74000
pied.wagtail  22.88667 16.50000
robin         22.55625 16.45000
tree.pipit    23.08000 16.66667
wren          21.12000 15.83333

Coefficients of linear discriminants:
              LD1       LD2
length  -0.634933  1.018737
breadth -1.428932 -2.037959

Proportion of trace:
   LD1    LD2
0.7442 0.2558

> print(cuckoos.lda$means)

                length  breadth
hedge.sparrow 23.11429 16.76429
meadow.pipit  22.29333 16.74000
pied.wagtail  22.88667 16.50000
robin         22.55625 16.45000
tree.pipit    23.08000 16.66667
wren          21.12000 15.83333

> print(cuckoos.lda$scaling)

              LD1       LD2
length  -0.634933  1.018737
breadth -1.428932 -2.037959
```

Examining `cuckoos.lda$scaling`, the entries in the column headed `LD1` are the coefficients of `length` and `breadth` that give the first set of discriminant scores. Those in the column headed `LD2` give the second set of discriminant scores. These scores can be obtained directly from the calculation `predict(cuckoos.lda)$x`

The following uses leave-one-out cross-validation to give an assessments of the accuracy for `lda()`

```
> ## Leave-one-out cross-validation
> ## Accuracies for linear discriminant analysis
> cuckooCV.lda <- lda(species ~ length + breadth,
                      data=cuckoos, CV=TRUE)
> confusion(cuckoos$species, cuckooCV.lda$class,
            gpnames=abbreviate(levels(cuckoos$species), 10))

Overall accuracy = 0.433
This assumes the following prior frequencies:
hedg.sprrw meadow.ppt pied.wagtl      robin tree.pipit       wren
     0.117      0.375      0.125      0.133      0.125      0.125

Confusion matrix
            Predicted (cv)
Actual       hedg.sprrw meadow.ppt pied.wagtl robin tree.pipit  wren
  hedg.sprrw      0.000      0.571      0.143 0.071      0.143 0.071
  meadow.ppt      0.000      0.867      0.067 0.000      0.022 0.044
  pied.wagtl      0.067      0.467      0.200 0.067      0.067 0.133
  robin           0.000      0.625      0.188 0.000      0.062 0.125
  tree.pipit      0.067      0.667      0.200 0.067      0.000 0.000
  wren            0.000      0.267      0.000 0.067      0.000 0.667
```

The following uses leave-one-out cross-validation to give assessments of the accuracy for `qda()`:

```
> ## Accuracies for quadratic discriminant analysis
> cuckooCV.qda <- qda(species ~ length + breadth,
                      data=cuckoos, CV=TRUE)
> acctab <-confusion(cuckoos$species, cuckooCV.qda$class,
                     gpnames=abbreviate(levels(cuckoos$species), 10),
                     printit=FALSE)
> tab <- table(cuckoos$species)
> ##
> ## Overall accuracy
> sum(diag(acctab)*tab)/sum(tab)

[1] 0.425

> ## Confusion matrix
> round(acctab, 3)

            Predicted (cv)
Actual       hedg.sprrw meadow.ppt pied.wagtl robin tree.pipit  wren
  hedg.sprrw      0.214      0.429      0.143 0.071      0.000 0.143
  meadow.ppt      0.000      0.822      0.044 0.000      0.044 0.089
  pied.wagtl      0.067      0.533      0.067 0.067      0.133 0.133
  robin           0.000      0.688      0.188 0.000      0.000 0.125
  tree.pipit      0.200      0.600      0.133 0.067      0.000 0.000
  wren            0.067      0.133      0.000 0.133      0.000 0.667
```

The calculations that follow will require `lda()` and `qda()` fits with `CV=FALSE`, which is the default:

```
> cuckoos.lda <- lda(species ~ length + breadth, data=cuckoos)
> cuckoos.qda <- qda(species ~ length + breadth, data=cuckoos)
```

Figure 16A plots `length` versus `breadth`, with the axes for the discriminant scores added.

Figure **??** shows contours for distinguishing `wren` from not-wren, both for the `lda()` analysis (solid line) and for the `qda()` analysis (gray line). The contours are very different. These different contours lead in each case to a cross-validated accuracy of 66.7% for correctly predicting wren eggs as wren – a close agreement that may seem surprising.

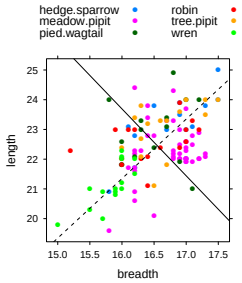


Figure 42: Length versus breadth, compared between cuckoo eggs laid in hedge sparrow and those laid in wren nests. Axes for the scores are overlaid.

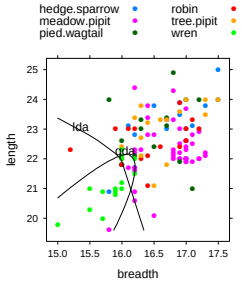


Figure 43: Length versus breadth, compared between cuckoo eggs laid in hedge sparrow and those laid in wren nests. The boundary lines for distinguishing `wren` from not-wren are shown, both for the `lda()` analysis and for the `qda()` analysis.

The following creates a graphics object that plots the points:

```
> gph <- xyplot(length ~ breadth, groups=species, data=cuckoos,
                type=c("p"), auto.key=list(columns=2), aspect=1,
                scales=list(tck=0.5), par.settings=simpleTheme(pch=16))
```

The code for Figure 42 is then:

```
> library(latticeExtra)  # This package has the function layer()
> LDmat <- cuckoos.lda$scaling
> ld1 <- LDmat[,1]
> ld2 <- LDmat[,2]
```

```
> library(DAAGxtras)
> gm <- sapply(cuckoos[, c("length", "breadth")], mean)
> av1 <- gm[1] + ld1[2]/ld1[1]*gm[2]
> av2 <- gm[1] + ld2[2]/ld2[1]*gm[2]
> gphA <- gph + layer(panel.abline(av1, -ld1[2]/ld1[1], lty=1),
                      panel.abline(av2, -ld2[2]/ld2[1], lty=2))
```

The code for Figure **??** is:

```
> x <- pretty(cuckoos$breadth, 20)
> y <- pretty(cuckoos$length, 20)
> Xcon <- expand.grid(breadth=x, length=y)
> cucklda.pr <- predict(cuckoos.lda, Xcon)$posterior
> cuckqda.pr <- predict(cuckoos.qda, Xcon)$posterior
> m <- match("wren", colnames(cucklda.pr))
> ldadiff <- apply(cucklda.pr, 1, function(x)x[m]-max(x[-m]))
> qdadiff <- apply(cuckqda.pr, 1, function(x)x[m]-max(x[-m]))
> gphB <- gph + as.layer(contourplot(ldadiff ~ breadth*length,
                                     at=c(-1,0,1), labels=c("", "lda",""),
                                     label.style="flat",
                                     data=Xcon), axes=FALSE)  +
                as.layer(contourplot(qdadiff ~ breadth*length,
                                     at=c(-1,0,1), labels=c("", "qda",""),
                                     label.style="flat",
                                     data=Xcon), axes=FALSE)
> gphB
```

For quadratic discriminant analysis, use `qda()` in place of `lda()`.

15 Accuracy comparisons

The function `compareModels()` (*DAAGxtras*) can be used to compare the accuracies of alternative model fits, checking for consistency over the data as a whole. Three model fits will be compared – the `lda()` fit above, the `qda()` fit above, and a variation on the `lda()` fit that includes terms in `length`$\hat{2}$, `breadth`$\hat{2}$ and `length*breadth`

```
> cucklda.pr <- cuckooCV.lda$posterior
> cuckqda.pr <- cuckooCV.qda$posterior
> cucklda.pr2 <- lda(species ~ length + breadth + I(length^2)
                     + I(breadth^2) + I(length*breadth), CV=TRUE,
                     data=cuckoos)$posterior
> compareModels(groups=cuckoos$species,
                estprobs=list(lda=cucklda.pr, qda=cuckqda.pr,
                              "lda plus"=cucklda.pr2))

[1] "Average accuracies for groups:"
  WinF  WinNF    Veh    Con   Tabl   Head
0.1703 0.5113 0.1467 0.1497 0.1574 0.5780
Approx sed
    0.0271
[1] "Average accuracies for methods:"
     lda      qda lda plus
  0.3342   0.3402   0.3510
Approx sed
    0.0049
```

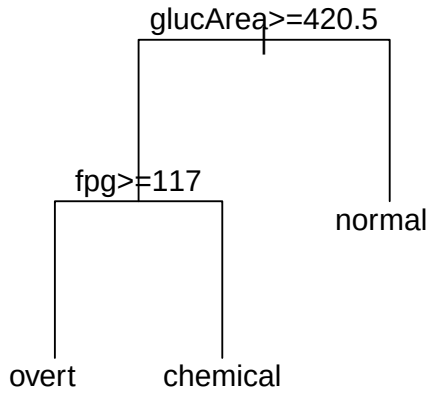


Figure 44: Can the clinical diagnosis be derived directly from the five available clinical measures? The graph shows the classification rule that is given by a tree-based classification.

16 Tree-based methods and random forests

On a scale in which highly parametric methods lie at one end and highly non-parametric methods at the other, linear discriminant methods lie at the parametric end, and tree-based methods and random forests at the non-parametric extreme. An attraction of tree-based methods and random forests is that model choice can be pretty much automated.

Figure 44 is a visual summary of results from the use of tree-based classification. The three classes are from a clinical classification of Diabetes – `overt` (overt diabetic), `chemical` (chemical diabetic), and `normal` (normal).

The clinical measures (explanatory variables) are `relwt` (relative weight), `fpg` (fasting plasma glucose), `glucArea` (glucose area), `Insulin` (insulin area), and `SSPG` (steady state plasma glucose).

Tree-based classification proceeds by constructing a sequence of decision steps. At each node, the split is used that best separates the data into two groups. Here (Figure 44) tree-based regression does unusually well (CV accuracy = 97.2%), perhaps because it is well designed to reproduce a simple form of sequential decision rule that has been used by the clinicians.

How is 'best' defined? Splits are chosen so that the Gini index of "impurity" is minimized. Other criteria are possible, but this is how `randomForest()` constructs its trees.

16.0.1 Random forests

The random forest methodology will usually improve (but not here), sometimes quite dramatically, on tree-based classification. Figure 45 shows trees that have been fitted to different bootstrap samples of the diabetes data. Typically 500 or more trees are fitted, without a stopping rule. Individual trees are likely to overfit. As each tree is for a different random sample of the data, there is no overfitting overall.

Figure 46 is a visual summary of the random forest classification result. The proportion of trees in which any pair of points appear together at the same node may be used as a measure of the "proximity" between that pair of points. Then, subtracting proximity from one to obtain a measure of distance, an ordination method is used to find a representation of those points in a low-dimensional space.

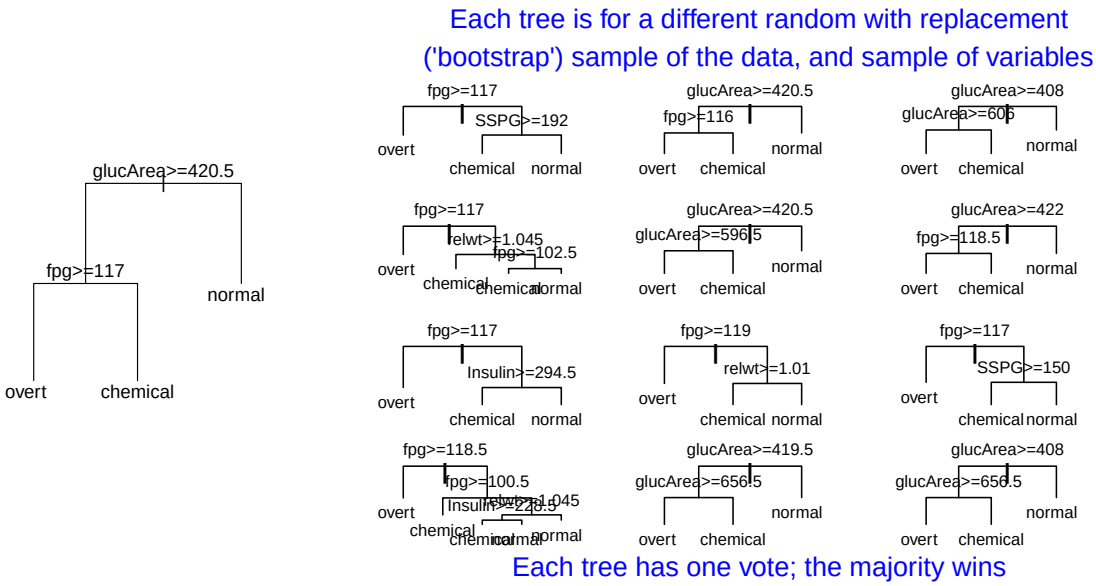


Figure 45: The left panel is a classification tree that was derived from tree-based classification. Each tree in the right panel is for a different bootstrap sample of the diabetes data. Additionally, a different random sample of variables is used for each different tree. The final classification is determined by a random vote over all trees.

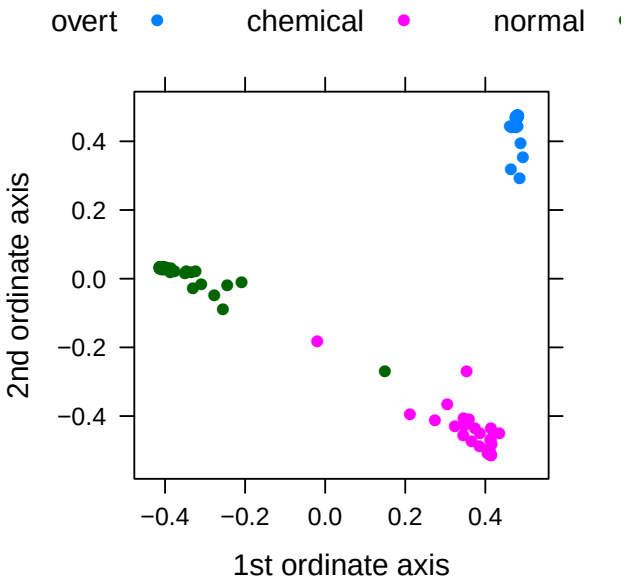


Figure 46: The plot is an attempt to represent, in two dimensions, the random forest result. This plot tries hard to reflect probabilities of group membership assigned by the analysis. It does not result from a 'scaling' of the feature space.

16.1 The `randomForests()` Function

A good first check on the adequacy of "linear" methods in the style of `lda()` and `qda()` adequate is comparison with the highly nonparametric analysis of the function `randomForest()` (*randomForest* package). Random Forests may do well when complex interactions are required to explain the dependence.

The `randomForest()` function can be used in a manner that is highly automatic. There is relatively limited scope for tuning. Such tuning as is possible will often make a very limited improvement.

The random forests methodology takes many (the `randomForest()` default is 500) different bootstrap random samples from the data, each with the same number of observations as the original data. For each such random sample, it takes a random sample of variables, and builds a tree. Splitting of trees usually to the fullest possible extent. The class to which an observation will be assigned is determined by taking a vote between trees.

For each bootstrap sample, predictions can be made for the observations that were not included – i.e., for the out-of-bag data. This is done for each bootsrap sample. Comparison with the actual group assignments then provides an unbiased estimate of accuracy.

16.2 Prior probabilities

In the randomForest() implementation, there is no direct provision for varying prior probabilities from the relative group frequencies. It is unclear what the argument `classwt` does in the R implementation, but the effect is not equivalent to the specifying of prior probabilities.

The effect of specifying prior probabilities can however be achieved by varying the sample size (`sampsize` between groups. As an example, consider the dataset `Pima.tr`. The 200 sample points divide up as follows:

```
> table(Pima.tr$type)

 No Yes
132  68
```

The default is to take a bootstrap sample of size 200 from the total data. On average each boostrap sample will have 34% of type `Yes`, i.e., these are diabetics, but this proportion will vary from sample to sample. Here are error rates for the default settings:

```
> set.seed(41)     # This seed should reproduce the result given here
> (Pima.rf <- randomForest(type ~ ., data=Pima.tr))

Call:
 randomForest(formula = type ~ ., data = Pima.tr)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 29.5%
Confusion matrix:
     No Yes class.error
No  108  24   0.1818182
Yes  35  33   0.5147059
```

The overall error rate is:

```
> 0.66*0.1818182+0.34*0.5147059

[1] 0.295
```

This varies quite a bit from run to run. Thus, in one set of five different runs, the variation was between 27% and 30%. Note the much greater accuracy for classifying the larger number in the `No` group, as opposed to the `Yes` group.

The following alternative insists on choosing bootstrap samples separately for the two groups, so that each bootstrap sample has exactly 132 that are `No` and 68 that are `Yes`.

```
> Pima.rf<- randomForest(type ~ ., data=Pima.tr, sampsize=c(132,68))
```

One source of variation, i.e., in the relative numbers of `Yes` and `No` in the bootstrap samples, has been removed. Thus, one might expect less variability in the error rates, with the average rate the same as before. Detecting the difference in variability would require quite a large number of runs with each of the two settings of `sampsize`.

Smaller samples are possible, e.g.

```
> Pima.rf <- randomForest(type ~ ., data=Pima.tr, sampsize=c(66,34))
> Pima.rf

Call:
 randomForest(formula = type ~ ., data = Pima.tr, sampsize = c(66,      34))
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 26.5%
Confusion matrix:
     No Yes class.error
No  111  21   0.1590909
Yes  32  36   0.4705882
```

Sample sizes however not allowed to be larger than the numbers in the respective groups, i.e., 132 for `No` and 68 for `Yes`. Taking smaller samples can actually increase accuracy, ore may not reduce it very much. Reduced accuracy for individual trees is traded off against reduced correlation between trees.

Finally, note that we can vary the sample sizes so that they give the effect of some desired prior relative probability, For equal sample sizes, we can reduce the first sample size to be the same size as the smaller of the two groups, i.e., `sampsize=c(68,68)`:

```
> Pima.rf <- randomForest(type ~ ., data=Pima.tr, sampsize=c(68,68))
```

16.2.1 Varying prior probabilities – an example

The error rate for a target population should be lowest when the proportions in the target population are the same as those used in building thye model. More generally, costs can be applied, for example an error in classifying `Yes` in the Pima data might be treated as twice as serious (costly) as an error in classifying a `No`.

For this purpose, consider the `statlog` dataset from the website `http://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29`. The variable in column 18, here identified as `V18`, identifies a customer as good (=1) or bad (=2). Variables that are categorical had values that were prefixed with an A, so that on reading them into R they became factors.

It will be useful to have a function that, once a model has been fitted, makes it straightforward to check how the error rate varies between populations with different relative numbers that are `Yes` and `No`:

```
> compareTargets <-
    function(rfobj, prior1, prior2){
        nam1 <- deparse(substitute(prior1))
        nam2 <- deparse(substitute(prior2))
```

```
  print(c(nam1,nam2))
  err <- rfobj$confusion[,3]
  err1 <- sum(err*prior1)/sum(prior1)
  err2 <- sum(err*prior2)/sum(prior2)
  errvec <- c(err, err1,err2)
  names(errvec) <- c("error-good", "error-bad", nam1, nam2)
  errvec
}
```

The numbers of 'good' and 'bad' customers are:

```
> statlog$V18 <- factor(statlog$V18, labels=c("good","bad"))
> table(statlog$V18)

good  bad
 845  155
```

First fit a model in which the prior probabilities are the relative proportions in the two samples, then comparing how they perform on populations with `good:bad` proportions of (1) 855:134 (as in the data), and (2) 134:134

```
> set.seed(41)     # Use this seed to get the result shown
> germ.rf <- randomForest(V18 ~ ., data=statlog, sampsize=c(845,155))
> round(compareTargets(germ.rf, prior1 = c(845,155), prior2 = c(155,155)), 4)

[1] "c(845, 155)" "c(155, 155)"
 error-good   error-bad c(845, 155) c(155, 155)
     0.0083      0.8581      0.1400      0.4332
```

As always with such calculations, repeating the calculation several times will give a better basis fro assessment than can be obtained from a sigle run.

Next repeat the calculations, now with equal prior probabilities for non-spam and spam:

```
> set.seed(41)     # Use this seed to get the result shown
> germ.rf <- randomForest(V18 ~ ., data=statlog, sampsize=c(155,155))
> round(compareTargets(germ.rf, prior1 = c(845,155), prior2 = c(155,155)), 4)

[1] "c(845, 155)" "c(155, 155)"
 error-good   error-bad c(845, 155) c(155, 155)
     0.1964      0.3871      0.2260      0.2918
```

The error is indeed smaller for the 845,155 target when the prior also has a 845:155 ratio. It is smaller for the 155:155 target when the prior is also 155:155.

Part VI
Ordination

Ordination is a generic name for methods for providing a low-dimensional view of points in multi-dimensional space, such that "similar" objects are near each other and dissimilar objects are separated. The plot(s) from an ordination in 2 or 3 dimensions may provide useful visual clues on clusters in the data and on outliers. The methods described help all use some form of multi-dimensional scaling (MDS)

Distances may be already given, or it may be necessary to start by calculating distances between points. In either case, the distances are the starting point for an ordination. Similarities will be transformed into distances before starting the ordination calculations.

Examples are:

1. From Australian road travel distances between cities and larger towns, can we derive a plausible "map" showing the relative geographic locations?

2. Starting with genomic data, various methods are available for calculating genomic "distances " between, e.g., different insect species. The distance measures are based on evolutionary models that aim to give distances between pairs of species that are a monotone function of the time since the two species separated.

3. Given a matrix $\mathbf{X}$ of n observations by p variables, a low-dimensional representation is required, i.e., the hope is that a major part of the information content in the data can be summarized in a small number of constructed variables. There is typically no good model, equivalent to the evolutionary models used by molecular biologists, that can be used to motivate distance calculations. There is then a large element of arbritariness in the distance measure used.

If data can be separated into known classes that should be reflected in any ordination, then the scores from classification using `lda()` may be a good basis for an ordination. Plots in 2 or perhaps 3 dimensions may then reveal additional classes and/or identify points that may be misclassified and/or are in some sense outliers. It may indicate whether the classes that formed the basis for the ordination seem real and/or the effectiveness of the discrimination method in choosing the boundaries between classes.

The function `randomForest()` is able to return "proximities" that are measures of the closeness of any pair of points. These can be turned into rough distance measures that can then form the basis for an ordination. With Support Vector Machines, decision values are available from which distance measures can be derived and used as a basis for ordination.

16.3 Distance measures

16.3.1 Euclidean distances

Treating the rows of $\mathbf{X}$ (n by p) as points in a p-dimensional space, the squared Euclidean distance d_{ij}^2 between points i and j is

$$d_{ij}^2 = \sum_{k=1}^{p} (x_{ik} - x_{jk})^2$$

The distances satistfy the triangle inequality

$$d_{ij} \leq d_{ik} + d_{kj}$$

The columns of $\mathbf{X}$ can be arbitrarily transformed before calculating the d_{ij}. Where all elements of a column are positive, use of the logarithmic transformation is common. A logarithmic scale makes sense for biological morphometric data, and for other data that has similar characteristics. For morphometric data, the effect is to focus attention on relative changes in the various body proportions, ignoring the overall magnitude.

The columns may be standardized before calculating distances, i.e., scaled so that the standard deviation is one. The columns may be weighted differently. Use of an unweighted measure with all columns scaled to a standard deviation of one is equivalent to working with the unscaled columns and calculating d_{ij}^2 as

$$d_{ij}^2 = \sum_{k=1}^{p} w_{ij}(x_{ik} - x_{jk})^2$$

where $w_{ij} = (s_i s_j)^{-1}$ is the inverse of the product of the standard deviations for columns i and j. Results may depend strongly on the distance measure.

16.3.2 Non-Euclidean distance measures

Euclidean distance is one of many possible choices of distance measures, still satisfying the triangle inequality. As an example of a non-Euclidean measure, consider the Manhattan distance. This has

$$d_{ij} = \sum_{k=1}^{p} \mid x_{ik} - x_{jk} \mid$$

The Manhattan distance is the shorest distance for a journey that always proceeds along one of the co-ordinate axes. In Manhattan in New York, streets are laid out in a rectangular grid. This is then (with $k = 2$) the walking distance along one or other street. For other choices, see the help page for the function `dist()`.

The function `daisy()` in the *cluster* package offers a still wider range of possibilities, including distance measures that can be used when columns that are factor or ordinal. It has an argument `stand` that can be used to ensure standardization when distances are calculated. Unless measurements are comparable (e.g., relative growth, as measured perhaps on a logarithmic scale, for different body measurements), then it is usually desirable to standardize before using ordination methods to examine the data.

Irrespective of the method used for the calculation of the distance measure, ordination methods yield a representation in Euclidean space. Depending on the distance measure and the particular set of distances, an exact representation may or may not be possible.

See Gower & Legendre (1986) for a detailed discussion of the netric and Euclidean propoerties of a wide variety of similarity coefficients.

16.4 From distances to a configuration in Euclidean space

Given a set of "distances" d_{ij} that satisfy the triangle inequality, there is in general no guarantee that it will be possible to derive a configuration X in Euclidean that exactly reproduces those distances. Where Euclidean distances are calculated between the rows of a matrix $\mathbf{X}$, clearly the matrix $\mathbf{X}$ is itself one possible configuration in Euclidean space. So also is $\mathbf{XP}$, where $\mathbf{P}$ is an orthogonal matrix.

Suppose however that non-metric distances are derived from a matrix $\mathbf{X}$. For example, they may be Manhattan distances. For some distance measures, it is always possible to find a configuration X in Euclidean space (an embedding) that exactly reproduces those distances. For other choices of distance this is not always possible.

It is however always possible to find a configuration X in Euclidean space in which the distances are approximated, perhaps rather poorly. This is true whether ot not the triangle inequality is satisfied. It will become apparent in the course of seeking the configuration whether an exact embedding (matrix X) is possible, and how accurate this embedding is.

Given such a matrix X if it exists, we can write

$$\begin{aligned} d_{ij}^2 &= \sum_{k=1}^{p} (\mathsf{x}_{ik} - \mathsf{x}_{jk})^2 \\ &= \sum_{k=1}^{p} \mathsf{x}_{ik}^2 + \sum_{k=1}^{p} \mathsf{x}_{jk}^2 - 2\sum_{k=1}^{p} \mathsf{x}_{ik}\mathsf{x}_{jk} \end{aligned}$$

Thus

$$d_{ij}^2 = q_{ii} + q_{jj} - 2q_{ij}$$

where $q_{ii} = \sum_{k=1}^{p} x_{ik}^2$; $q_{ij} = \sum_{k=1}^{p} x_{ik}x_{jk}$. Observe that q_{ij} is the (i, j)th element of the matrix $\mathbf{Q} = \mathsf{X}\mathsf{X}^T$. Thus, the matrix $\mathbf{Q}$ has all the information needed to derive the distances. Because $\mathbf{Q} = \mathsf{X}\mathsf{X}^T$, it is positive semidefinite.

The mapping from the q_{ij} to the d_{ij} is one to one. Given distances, it is possible to find such a matrix $\mathbf{Q}$, if it exists. The detailed derivation is in Section 23. This shows that it is always possible to derive a symmetric matrix $\mathbf{Q}$. If and only if $\mathbf{Q}$ is positive definite, there is an exact embedding X in Euclidean space.

Having thus recovered a symmetric matrix $\mathbf{Q}$, the spectral decomposition yields

$$\mathbf{Q} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$$

where $\mathbf{\Lambda}$ is a diagonal matrix. The diagonal elements λ_i are ordered so that

$$\lambda_1 \geq \lambda_2 \ldots \geq \lambda_n$$

Providing $\lambda_i \geq 0$, choose

$$\mathsf{X} = \mathbf{U}\mathbf{\Lambda}^{\frac{1}{2}}$$

As the rows and columns of $\mathbf{Q}$ sum to zero, $\mathbf{Q}$ is singular. Hence if $\mathbf{Q}$ is positive definite, as required for exact embedding in Euclidean space, $\lambda_i \geq 0$ for all i and $\lambda_n = 0$.

Important points are:

- Often, most of the information will be in the first few dimensions. We may for example be able to approximate $\mathbf{Q}$ by replacing $\mathbf{\Lambda}$ in $\mathbf{Q} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ by a version of $\mathbf{\Lambda}$ in which diagonal elements after the kth have been set to zero. If `cmdscale()` is called with `eig=TRUE`, it returns both the eigenvalue information (the λ_i) and a goodness of fit statistic, by default (assuming at least two non-zero λ_i) for the configuration with $k = 2$.
- If $\mathbf{Q}$ is not positive semidefinite, the ordination can still proceed. However one or more eigenvalues λ_i will now be negative. If relatively small, it may be safe to ignore dimensions that correspond to negative eigenvalues. It is then more than otherwise desirable to check that the ordination reproduces the distances with acceptable accuracy.

16.4.1 The connection with principal components

Let $\mathbf{X}$ be a matrix that is the basis for the calculation of Euclidean distances, after any transformations and/or weighting. Then metric p-dimensional ordination, applied to Euclidean distances between the rows of $\mathbf{X}$, yields an orthogonal transformation of the space spanned by the columns of $\mathbf{X}$. If the successive dimensions are chosen to "explain" successively larger proportions of the trace of $\mathbf{X}\mathbf{X}^T$, it is equivalent to the principal components transformation. Thus `cmdscale()` yields, by a different set of matrix manipulations, a principal components decomposition.

16.5 Non-metric scaling

These methods all start from "distances", but allow greater flexibility in their use to create an ordination. The aim is to represent the "distances" in some specified number of dimensions, typically two dimensions. As described here, a first step is to treat the distances as Euclidean, and determine a configuration in Euclidean space. These Euclidean distances are then used as a starting point for a representation in which the requirement that these are Euclidean distances, all determined with equal accuracy, is relaxed. The methods that will be noted here are:

Sammon scaling: A configuration with distances $\tilde{d}$ is chosen to minimize a weighted squared "stress"

$$\frac{1}{\sum_{i\neq j} d_{ij}} \sum_{i\neq j} \frac{(d_{ij} - \tilde{d}_{ij})^2}{d_{ij}}$$

Kruskal's non-metric multidimensional scaling: This aims to minimize

$$\frac{\sum_{i\neq j}(\theta(d_{ij}) - \tilde{d}_{ij})^2}{\sum_{i\neq j}\tilde{d}_{ij}^2}$$

with respect to the configuration of points and an increasing function θ of the distance d_{ij}.

Often, it makes sense to give greater weight to small distances than to large distances. The distance scale should perhaps not be regarded as rigid. Larger distances may not be measured on the same Euclidean scale as shorter distances. The ordination should perhaps preserve relative rather than absolute distances.

16.6 Examples

16.6.1 Australian road distances

The distance matrix that will be used is in the matrix `audists`, in the image file **audists.Rdata**. Consider first the use of classical multi-dimensional scaling, as implemented in the function `cmdscale()`:

```
> library(DAAGxtras)
> aupoints <- cmdscale(audists)
> plot(aupoints)
> text(aupoints, labels=paste(rownames(aupoints)))
```

An alternative to `text(aupoints, labels=paste(rownames(aupoints)))`, allowing better placement of the labels, is `identify(aupoints, labels=rownames(aupoints))`. We can compare the distances in the 2-dimensional representation with the original road distances:

```
> audistfits <- as.matrix(dist(aupoints))
> misfit <- as.matrix(dist(aupoints)) - as.matrix(audists)
> for (j in 1:9)for (i in (j+1):10){
    lines(aupoints[c(i,j), 1], aupoints[c(i,j), 2], col="gray")
    midx <- mean(aupoints[c(i,j), 1])
    midy <- mean(aupoints[c(i,j), 2])
    text(midx, midy, paste(round(misfit[i,j])))
    }
> colnames(misfit) <- abbreviate(colnames(misfit),6)
> print(round(misfit))
```

```
          Adelad Alice Brisbn Broome Cairns Canbrr Darwin Melbrn Perth Sydney
Adelaide       0   140   -792   -156    366     20     11     82   482   -273
Alice        140     0  -1085   -175    -41     76   -118    106   -26   -314
Brisbane    -792 -1085      0    198    319    -25   -233   -471   153    -56
Broome      -156  -175    198      0    527     -7      6    -65   990     70
Cairns       366   -41    319    527      0    277    -31    178     8    251
Canberra      20    76    -25     -7    277      0     -1   -241   372     -8
Darwin        11  -118   -233      6    -31     -1      0    -12    92    -58
Melbourne     82   106   -471    -65    178   -241    -12      0   301   -411
Perth        482   -26    153    990      8    372     92    301     0    271
Sydney      -273  -314    -56     70    251     -8    -58   -411   271      0
```

The graph is a tad crowded, and for detailed information it is necessary to examine the table.

It is interesting to overlay this "map" on a physical map of Australia.

```
> library(oz)
> oz()
> points(aulatlong, col="red", pch=16, cex=1.5)
```

```
> comparePhysical <- function(lat=aulatlong$latitude, long=aulatlong$longitude,
                              x1=aupoints[,1], x2 = aupoints[,2]){
    ## Get best fit in space of (latitude, longitude)
    fitlat <- predict(lm(lat ~ x1+x2))
    fitlong <- predict(lm(long ~ x1+x2))
    x <- as.vector(rbind(lat, fitlat, rep(NA,10)))
    y <- as.vector(rbind(long, fitlong, rep(NA,10)))
    lines(x, y, col=3, lwd=2)
    }
> comparePhysical()
```

An objection to `cmdscale()` is that it gives long distances the same weight as short distances. It is just as prepared to shift Canberra around relative to Melbourne and Sydney, as to move Perth. It makes more sense to give reduced weight to long distances, as is done by `sammon()` (*MASS*).

```
> aupoints.sam <- sammon(audists)

Initial stress        : 0.01573
stress after  10 iters: 0.00525, magic = 0.500
stress after  20 iters: 0.00525, magic = 0.500

> oz()
> points(aulatlong, col="red", pch=16, cex=1.5)
> comparePhysical(x1=aupoints.sam$points[,1], x2 = aupoints.sam$points[,2])
```

Notice how Brisbane, Sydney, Canberra and Melbourne now maintain their relative positions much better.

Now try full non-metric multi-dimensional scaling (MDS). This preserves only, as far as possible, the relative distances. A starting configuration of points is required. This might come from the configuration used by `cmdscale()`. Here, however, we use the physical distances.

```
> oz()
> points(aulatlong, col="red", pch=16, cex=1.5)
> aupoints.mds <- isoMDS(audists, as.matrix(aulatlong))

initial  value 11.875074
iter   5 value 5.677228
iter  10 value 4.010654
final  value 3.902515
converged

> comparePhysical(x1=aupoints.mds$points[,1], x2 = aupoints.mds$points[,2])
```

Notice how the distance between Sydney and Canberra has been shrunk quite severely.

16.6.2 Genetic Distances – Hasegawa's selected primate sequences

Here, matching genetic DNA or RNA or protein or other sequences are available from each of the different species. Distances are based on probabilistic genetic models that describe how gene sequences change over time. The package *ape* implements a number of alternative measures. For details see `help(dist.dna)`.

Hasegawa's sequences were selected to have as little variation in rate, along the sequence, as possible. The sequences are available either from the *DAAGxtras* package or from the wepage `http://evolution.genetics.washington.edu/book`. They can be read into R as:

```
> ## Obtain data from the web page on the next line, calculate distances
> url <- "http://evolution.genetics.washington.edu/book/primates.dna"
> library(ape)
> primates.dna <- read.dna(url)
> ## Alternative - download and then read in data
> # download.file(webpage, destfile="primates.txt")   # Alternative
> # primates.dna <- read.dna("primates.txt")
> ## Now calculate distances, using Kimura's K80 model
> primates.dist <- dist.dna(primates.dna, model="K80")
```

The *DAAGxtras* package has the dataset `primateDNA`. These are the same data, but stored in character format. For use with `dist.dna()` use the function `dist.dna()` to convert the data to a binary format. The following is an alternative to the code given above:

```
> ## Use dataset primateDNA from the DAAGbio package
> library(DAAGbio)
> library(ape)
> ## Calculate distances, using Kimura's K80 model
> primates.dist <- dist.dna(as.DNAbin(primateDNA), model="K80")
```

We now try for a two-dimensional representation, using `cmdscale()` from the *MASS* package:

```
> primates.cmd <- cmdscale(primates.dist)
> eqscplot(primates.cmd, xlab="Axis 1", ylab="Axis 2")
> lefrt <- 2+2*(primates.cmd[,1] < mean(par()$usr[1:2]))
> text(primates.cmd[,1], primates.cmd[,2], row.names(primates.cmd), pos=lefrt)

initial  value 19.892084
iter   5 value 13.849956
iter  10 value 13.553589
final  value 13.527427
converged
```

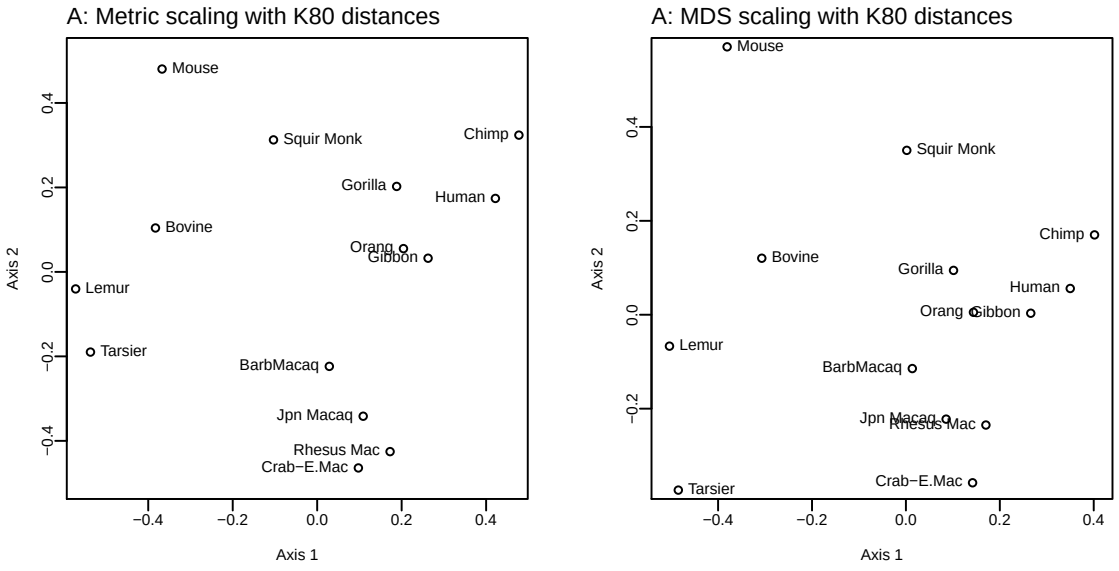


Figure 47: The plot on the left has used classical metric scaling, i.e., calculations seek a Euclidean space representation of the distances. The plot on the right has used the `isoMDS()` function to show results from Kruskal's non-metric multidimensional scaling, i.e., the "distances" provide an ordering in Euclidean space.

Now see how well Figure 47A reproduces the distances:

```
> d <- dist(primates.cmd)
> sum((d-primates.dist)^2)/sum(primates.dist^2)

[1] 0.101
```

With only around 5% of the sum of squared distances unaccounted for, it is hardly worth examining a 3-dimensional representation. Here, however, is the code:

```
> library(lattice)
> primates.cmd <- cmdscale(primates.dist, k=3)
> cloud(primates.cmd[,3] ~ primates.cmd[,1]*primates.cmd[,2])
> d <- dist(primates.cmd)
> sum((d-primates.dist)^2)/sum(primates.dist^2)
```

Now repeat the above with `sammon()` and `mds()`.

```
> primates.sam <- sammon(primates.dist, primates.cmd, k=2)
> eqscplot(primates.sam$points)
> text(primates.sam$points[,1], primates.sam$points[,2],
        row.names(primates.sam$points), pos=lefrt)
```

There is no harm in asking for three dimensions, even if only two of them will be plotted.

The following code is used to for the multidimensional scaling representation in Figure 47B:

```
> primates.mds <- isoMDS(primates.dist, primates.cmd, k=2)
> eqscplot(primates.mds$points, xlab="Axis 1", ylab="Axis 2")
> text(primates.mds$points[,1], primates.mds$points[,2],
        row.names(primates.mds$points), pos=lefrt)
```

16.6.3 Pacific rock art

Here, the the 614 features were all binary – the presence or absence of specific motifs in each of 98 Pacific sites. (Actually, there were 103 sites, but 5 were omitted because they had no motifs in common with any of the other sites.) Data are from Meredith Wilson's PhD thesis at Australian National University.

The binary measure of distance was used – the number of locations in which only one of the sites had the marking, as a proportion of the sites where one or both had the marking. Here then is the calculation of distances:

```
> pacific.dist <- dist(x = as.matrix(rockArt[-c(47,54,60,63,92), 28:641]),
                       method = "binary")
> sum(pacific.dist==1)/length(pacific.dist)

[1] 0.631

> plot(density(pacific.dist, to = 1))
> ## Now check that all columns have some distances that are less than 1
> symmat <- as.matrix(pacific.dist)
> table(apply(symmat, 2, function(x) sum(x==1)))

13 21 27 28 29 32 33 35 36 38 40 41 42 43 44 45 46 47 48 49 51 52 53 54 55 56
 1  1  1  1  2  1  2  1  2  2  1  2  4  3  1  3  1  2  1  1  2  2  3  2  2  2
57 58 61 62 64 65 66 67 68 69 70 71 73 75 76 77 79 81 83 84 85 90 91 92 93 94
 1  3  3  1  2  1  1  1  3  3  1  1  4  1  2  1  1  1  2  1  1  3  1  1  3  1
95 96 97
 1  3  4
```

It turns out that 63% of the distances were 1. This has interesting consequences, for the plots we now do.

```
> pacific.cmd <- cmdscale(pacific.dist)
> plot(pacific.cmd)
> pacific.mds <- isoMDS(pacific.dist, pacific.cmd)

initial  value 54.388728
iter   5 value 40.556391
iter  10 value 37.297430
iter  15 value 36.120966
iter  20 value 35.291828
iter  25 value 34.785333
iter  30 value 34.259107
iter  35 value 33.771381
iter  35 value 33.739070
iter  35 value 33.723549
final  value 33.723549
converged

> plot(pacific.mds$points)
```

Part VII
*Some Further Types of Model

17 *Multilevel Models – Introductory Notions

Basic ideas of multilevel modeling will be illustrated using data on yields from packages on eight sites on the Caribbean island of Antigua. They are a summarized version of a subset of data given in Andrews and Herzberg 1985, pp.339-353.

Multilevel models break away from the assumption of independently and identically distributed observations. The dependence is however of a very specific form. Models for time series move away from those assumptions in a different way, typically allowing some form of sequential correlation.

Depending on the use that will be made of the results, it may be essential to correctly model the structure of the random part of the model. The analysis will use the abilities of the `lme()` function in the *nlme* package, though the example is one where it is easy, using modest cunning, to get the needed sums of squares from a linear model calculation. For these data, there is more than one type (or "level") of prediction or generalization, with very different accuracies for the different generalizations. The data give results for each of several packages at a number of different locations (sites). In such cases, a prediction for a new package at one of the existing locations is likely to be more accurate than a prediction for a totally new location. Multi-level models are able to account for such differences in predictive accuracy.

The multiple levels that are in view are multiple levels in the *noise* or *error* term, and are superimposed on any effects that are predictable. For example, differences in historical average annual rainfall may partly explain location to location differences in crop yield. The error term in the prediction for a new location will account for variation that remains after taking account of differences in the rainfall.

Examples abound where the intended use of the data makes a multi-level model appropriate. Examples of two levels of variability, at least as a first approximation, include: variation between houses in the same suburb, as against variation between suburbs; variation between different clinical assessments of the same patients, as against variation between patients; variation within different branches of the same business, as against variation between different branches; variations in the bacterial count between different samples from the same lake, as opposed to variation between different subsamples of the same sample; variation between the drug prescribing practices of clinicians in a particular specialty in the same hospital, as against variation between different clinicians in different hospitals; and so on. In all these cases, the accuracy with which predictions are possible will depend on the mix of the two levels of variability that are involved. These examples can all be extended in fairly obvious ways to include more than two levels of variability.

In all the examples just mentioned, one source of variability is *nested* within the other – thus packages of land are nested within locations. Variation can also be *crossed*. For example different years may be crossed with different locations. Years are not nested in locations, nor are locations nested in years. Examples of crossed error structures are beyond the scope of the present discussion.

17.1 The Antiguan Corn Yield Data

For the version of the Antiguan corn data presented here, the hierarchy has two levels of *random effects*. Variation between packages in the same site is at the lower of the two levels, and is called level 0 in the later discussion. Variation between sites is the higher of the two levels, and is called level 1 in the later discussion. A farmer who lived close to one of the experimental sites might take data from that site as indicative of what to expect. Other farmers may think it more appropriate to regard their farms as new sites, distinct from the experimental sites, so that the issue is one of generalizing to new sites.

The analysis will use the `lme()` function in the *nlme* package, though the example is one where it is easy, using modest cunning, to get the needed sums of squares from a linear model calculation.

The data that will be analyzed are in the second column of Table 8, which has means of packages of land for the Antiguan data. In comparing yields from different packages, there are two sorts of comparison. Packages on the same site should be relatively similar, while packages in different sites should be relatively more different. The figure that was given earlier suggested that this is indeed the case.

Site	Site means	Site effect		Residuals from site mean
DBAN	5.16, 4.8, 5.07, 4.51		+0.59	0.28, −0.08, 0.18, −0.38
LFAN	2.93, 4.77, 4.33, 4.8		−0.08	−1.28, 0.56, 0.12, 0.59
NSAN	1.73, 3.17, 1.49, 1.97		−2.2	−0.36, 1.08, −0.6, −0.12
ORAN	6.79, 7.37, 6.44, 7.07	(4.29)	+2.62	−0.13, 0.45, −0.48, 0.15
OVAN	3.25, 4.28, 5.56, 6.24		+0.54	−1.58, −0.56, 0.73, 1.4
TEAN	2.65, 3.19, 2.79, 3.51		−1.26	−0.39, 0.15, −0.25, 0.48
WEAN	5.04, 4.6, 6.34, 6.12		+1.23	−0.49, −0.93, 0.81, 0.6
WLAN	2.02, 2.66, 3.16, 3.52		−1.45	−0.82, −0.18, 0.32, 0.68
v		square, add, multiply by 4, divide by d.f.=7, to give ms		square, add, divide by d.f.=24, to give ms

Table 8: The leftmost column has harvest weights (`harvwt`), for the packages in each site, for the Antiguan corn data. Each of these harvest weights can be expressed as the sum of the overall mean (= 4.29), site effect (third column), and residual from the site effect (final column). This information that can be used to create the analysis of variance table. (Details of the analysis of variance approach to analysis of these data, although straightforward, get only passing mention in these notes.)

Note: In an analysis of variance formalization, the two-level structure of variation is handled by splitting variation, as measured by the total sum of squares about the grand mean, into two parts – variation within sites, and variation between site means. The final two columns in Table 8 indicate how to calculate the relevant sums of squares and (by dividing by degrees of freedom) mean squares. The division of the sum of squares into two parts mirrors two different types of predictions that can be based on these data. First, suppose that we are interested in another package on one of these same sites. Within what range of variation would we expect its yield to lie? Second, suppose that a trial were to be carried out on some different site, not one of the original eight. What is the likely range of variation of the mean yield, i.e., how accurate is the accuracy of prediction of the yield for that new site?

The model

The model that is used is:

$$\text{yield} = \text{overall mean} + \underset{\text{(random)}}{\text{site effect}} + \underset{\text{(random)}}{\text{package effect}}$$

In formal mathematical language:

$$y_{ij} = \mu + \underset{\text{(site, random)}}{\alpha_i} + \underset{\text{(package, random)}}{\beta_{ij}} \qquad (i = 1, \ldots, 8; j = 1, \ldots, 4)$$

with $\text{var}[\alpha_i] = \sigma^2_L$, $\text{var}[\beta_{ij}] = \sigma^2_B$.

The quantities σ^2_L and σ^2_B are known, technically, as *variance components*. (Those who are familiar with the analysis of variance breakdown may wish to note that the variance components analysis allows inferences that are not immediately available from the breakdown of the sums of squares in the analysis of variance table.) Importantly, the variance components provide information that can help design another experiment.

17.2 The variance components

Here is how the variance components should be interpreted, for the Antiguan data:

- Variation between packages at a site is due to one source of variation only. Denote this variance by σ_B^2. The variance of the difference between two such packages is $2\sigma_B^2$
  [Both packages have the same site effect α_i, so that var(α_i) does not contribute to the variance of the difference.]

- Variation between sites in different plots is partly a result of variation between packages, and partly a result of additional variation between sites. In fact, if σ_L^2 is the (additional) component of the variation that is due to variation between sites, the variance of the difference between two packages that are in different site is
$$2(\sigma_L^2 + \sigma_B^2)$$

- For s single package, the variance is $\sigma_L^2 + \sigma_B^2$. The variance of the estimate of the site mean is a mean over the four packages at the one site, and is
$$\sigma_B^2 + \frac{\sigma_L^2}{4}$$
  [Notice that while σ_L^2 is divided by four, σ_B^2 is not. This is because the site effect is the same for all four packages.]

18 *Survival models

Survival (or failure) analysis introduces features different from any of those encountered in the regression methods discussed in earlier chapters. It has been widely used for comparing the times of survival of patients suffering a potentially fatal disease who have been subject to different treatments. Computations can be handled in R using the *survival* package, written for S-PLUS by Terry Therneau, and ported to R by Thomas Lumley.

Section 5.4.1 discusses an example that is onveniently handled using survival models.

Other names, mostly used in non-medical contexts, are *Failure Time Analysis* and *Reliability*. Yet another term is *Event History Analysis*. The focus is on time to any event of interest, not necessarily failure. It is an elegant methodology that is too little known outside of medicine and industrial reliability testing.

Applications include:

- the failure time distributions of industrial machine components, electronic equipment, automobile components, kitchen toasters, light bulbs, businesses, etc. (failure time analysis, or reliability),
- the waiting time to germination of seeds, to marriage, to pregnancy, or to getting a first job,
- the waiting time to recurrence of an illness or other medical condition.

The outcomes are survival times, but with a twist. The methodology is able to handle data where failure (or another event of interest) has, for a proportion of the subjects, not occurred at the time of termination of the study. It is not necessary to wait till all subjects have died, or all items have failed, before undertaking the analysis! Censoring implies that information about the outcome is incomplete in some respect, but not completely missing. For example, while the exact point of failure of a component may not be known, it may be known that it did not survive more than 720 hours (= 30 days). In a clinical trial, there may for some subjects be a final time up to which they survived, but no subsequent information. Such observations are said to be right censored.

Thus, for each observation there are two items of information: a time, and censoring information. Commonly the censoring information indicates either right censoring denoted by 0, or failure denoted by 1.

Many of the same issues arise as in more classical forms of regression analysis. One important set of issues has to do with the diagnostics used to check on assumptions. Here there have been large advances in recent years. A related set of issues has to do with model choice and variable selection. There are close connections with variable selection in classical regression. Yet another set of issues has to do with the incomplete information that is available when there is censoring.

Yang & Letourneau (2005) is an interesting example of a data mining paper where survival methods could and should have been used. The methodology may be regarded as an unsatisfactory attempt to reinvent survival methods! Their methodology is tortuous and does not make the most effective use of the data.

Part VIII
Technical Mathematical Results

19 Linear models – matrix derivations & extensions

- $\boldsymbol{y}$ (n by 1) is a vector of observed values, $\mathbf{X}$ (n by p) is model matrix, and $\boldsymbol{\beta}$ (p by 1) is a vector of coefficients.
- The model is $\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, i.e. $\boldsymbol{y}_i = \mathbf{X}_i\boldsymbol{\beta} + \boldsymbol{\epsilon}_i$ where the vector $\boldsymbol{\epsilon}$ of residuals is n by 1 The classical theory assumes that $E[\boldsymbol{y}] = \boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$, i.e. $\mathrm{E}[\boldsymbol{\epsilon}] = \mathbf{0}$.
- Least squares *normal* equations are
$$\mathbf{X}'\mathbf{X}\boldsymbol{\beta} = \mathbf{X}'\boldsymbol{y}$$
(assuming $\boldsymbol{\epsilon}_i$ are iid normal, these are the maximum likelihood estimates)
- If variances are unequal, modify *normal* equations to
$$\mathbf{X}'\boldsymbol{W}\mathbf{X}\boldsymbol{\beta} = \mathbf{X}'\boldsymbol{W}\boldsymbol{y} \tag{2}$$
where $\boldsymbol{W}$ is a diagonal matrix with elements equal to the inverses of the variances (justification is from maximum likelihood, or argue that leverage should be independent of variance)
- More generally, if $\boldsymbol{\epsilon}$ is multivariate normal with known variance-covariance matrix $\boldsymbol{\Sigma}$, then ML theory gives the equation as above with $\boldsymbol{W} = \boldsymbol{\Sigma}^{-1}$.

19.0.1 Linear Models – correlated observations

Two observations with a high positive correlation contain, jointly, less information than two independent values. In the extreme case where the correlation is 1, the two carry the same information. Suppose for example that fruit trees are randomized to receive one of two different fertilizer treatments. Then repeat assessments of fruit load on the one tree, made perhaps by different technical staff, are likely to be highly correlated.

Use of a general variance-covariance matrix can in principle account for correlated data. Except in computer simulations, the variance-covariance matrix $\boldsymbol{\Sigma}$ is unlikely to be exactly known and must be estimated. Setting $\boldsymbol{W} = \hat{\boldsymbol{\Sigma}}^{-1}$ in equation 2 no longer gives, in general, estimates that are optimal. Hence the many different special methods that are available for specific types of correlation structure that have in practice proved useful. For example, time series models typically try to account for correlations that are highest between points that are close together in time. Spatial analysis models typically allow for correlations that are a function of separation in space. Hierarchical multi-level models allow for different variance-covariance structures at each of several levels of hierarchy.

19.0.2 Least squares computational methods

A separate set of notes describes the approach, based on the QR matrix decomposition, that is used in R and in most of the R packages. Where methods that are directly based on QR are too slow, there may be a specialized method that takes advantage of structure in $\mathbf{X}$ to greatly speed up computation. Sparse least squares is an important special case. See Bates (2006); Koenker and & Ng (2003).

20 Generalized Linear Models – theory & computation

Here, it is convenient to recast the equations in matrix form.

- As before, we have $\boldsymbol{\mu} = \mathrm{E}[\boldsymbol{y}]$ (n by 1), $\mathbf{X}$ (n by p), and $\boldsymbol{\beta}$ (p by 1).

- The model is now
$$f(\boldsymbol{\mu}) = \mathbf{X}\boldsymbol{\beta}, \quad \text{where } \mathrm{E}[\boldsymbol{y}] = \boldsymbol{\mu}$$
Here, $f()$, which must be monotonic, has the name *link* function. For example,
$$f(\boldsymbol{\mu}_i) = \log(\frac{\boldsymbol{\mu}_i}{N_i - \boldsymbol{\mu}_i})$$
- The distribution of y_i is a function of the predicted value μ_i, independently for different observations. The different y_i are from the same exponential family, but the distributions are not identical. Commonly used exponential family distributions are the normal, binomial and Poisson.
- An extension is to the quasi-exponential family, where the variance is a constant multiple of an exponential family variance. The multiplying constant is estimated as part of the analysis. Applications for models with quasibinomial or quasipoisson errors may if anything be more extensive than for their exponential family counterparts.
- Just as for linear models, spline or other terms that model nonlinear responses can be fitted.

20.1 Maximum likelihood parameter estimates

- Recall that the equation is
$$f(\boldsymbol{\mu}) = E(\boldsymbol{y}) = \mathbf{X}\boldsymbol{\beta}$$
where $\boldsymbol{\mu} = E[\boldsymbol{y}]$
- Assuming a distribution from the exponential family, the maximum likelihood estimates of the parameters are given by
$$\mathbf{X}'\boldsymbol{W}\boldsymbol{\mu} = \mathbf{X}'\boldsymbol{W}\boldsymbol{y}$$
where $f(\boldsymbol{\mu}_i) = \mathbf{X}_i\boldsymbol{\beta}$
- Note that the (diagonal) element $\boldsymbol{W}_{ii}$ of $\boldsymbol{W}$ are functions both of var$[\boldsymbol{y}_i]$ and of $f(\boldsymbol{\mu}_i)$
- The ML equations must in general be solved by iteration ($\boldsymbol{\beta}$ appears on both sides of the equation.) Iteratively reweighted least squares is used, i.e. Newton-Raphson. Each iteration uses a weighted least squares calculation. As the weights are inversely proportional to the variances, they depend on the fitted values. Starting values are required to initiate calculations. The weighted least squares calculation is repeated, with new weights at each new iteration, until the fitted values converge.

21 Least Squares Estimates

21.1 The mean is a least squares estimator

The `lm()` function uses the method of least square to find estimates. The following is the simplest possible example. Given sample values
$$y_1, y_2, \ldots, y_n$$
what choice of μ will minimize $\sum_{i=1}^{n}(x_i - \mu)^2$? Observe that
$$\begin{aligned}
\sum_{i=1}^{n}(x_i - \mu)^2 &= \sum_{i=1}^{n}[(x_i - \bar{x}) + (\bar{x} - \mu)]^2 \\
&= \sum_{i=1}^{n}[(x_i - \bar{x})^2 + 2(x_i - \bar{x})(\bar{x} - \mu) + (\bar{x} - \mu)^2] \\
&= \sum_{i=1}^{n}(x_i - \bar{x})^2 + 2(\bar{x} - \mu)\sum_{i=1}^{n}(x_i - \bar{x}) + n(\bar{x} - \mu)^2
\end{aligned}$$

As

$$\sum_{i=1}^{n}(x_i - \bar{x} = \sum_{i=1}^{n} x_i - n\bar{x} = 0$$

this equals

$$\sum_{i=1}^{n}(x_i - \bar{x})^2 + n(\bar{x} - \mu)^2$$

Then $n(\bar{x} - \mu)^2 >= 0$, with equality for $\mu = \hat{\mu} = \bar{x}$.

Because $\bar{x}$ is the least squares estimator of μ, it is possible to use a linear model to calculate the mean. For this, a model is specified in which the only term is the constant term. Thus, for the female Adelaide statistics students:

```
library(MASS)
y <- na.omit(survey[survey$Sex=="Female", "Height"])
lm(y ~ 1)
```

21.2 Least squares computations for linear models

Given the model

$$\mathbf{y} = \mathbf{X}\beta + \epsilon$$

the least squares estimate $\mathbf{b}$ of β is obtained by solving the normal equations

$$\mathbf{X}'\mathbf{X}\beta = \mathbf{X}'\mathbf{y}$$

In practice it is usually best not to solve this equation directly, but to work from the QR orthogonal decomposition of $\mathbf{X}$. For details, see the references that appear on the help page for R's function `qr()`.

21.3 Beyond Least Squares – Maximum Likelihood

Least squares may not work very well for non-normal data. Typically, statisticians then appeal to the maximum likelihood principle. For normal data, with independent and identically distributed errors, maximum likelihood gives the same parameter estimates as least squares. Attention will at various points be drawn to types of model where it really is necessary to work with maximum likelihood estimates. Thus, note the logistic regression models that are discussed in Section 10.

22 Variances of Sums and Differences

The needed results are most easily derived using expectation algebra. For present purposes, it will be adequate to define

$$\mathrm{E[g(X)]} = \int \mathrm{g(x)f(x)dx}$$

if X is a continuous random variable with density $f(x)$ at the point x, and

$$\mathrm{E[g(X)]} = \sum \mathrm{g(x)Pr(X = x)}$$

where the integral or sum is taken over the support of X. The key result from expectation algebra is that, for any two random variables X and Y, $\mathrm{E[c_1X + c_2X] = c_1E[X] + c_2E[Y]}$. The proof, for two special cases noted above, is left as an exercise.

The variance of a random variable X with mean $\mu = \mathrm{E[X]}$ is $\mathrm{E(y - \mu)^2}$. Then

$$\mathrm{var[X_1 + X_2] = var[X_1] + var[X_2] + 2cov[X_1, X_2]}$$

which equals $\mathrm{var[X_1] + var[X_2]}$ if and only if

$$\mathrm{cov[X_1, X_2] = E[(X_1 - E[X_1])(X_2 - E[X_2])] = 0}$$

A very similar argument shows that $\mathrm{var[X_1 - X_2] = var[X_1] + var[X_2]}$ if and only if $\mathrm{cov[X_1, X_2] = 0}$.

A sufficient condition for $\mathrm{cov[X_1, X_2] = 0}$ is that X_1 and X_2 are independent.

23 From Distances to a Representation in Euclidean Space

Given an embedding X in Euclidean space, if it exists, the squared Euclidean distance between points i and j can be written

$$\begin{aligned} d_{ij}^2 &= \sum_{k=1}^{p} (\mathsf{x}_{ik} - \mathsf{x}_{jk})^2 \\ &= \sum_{k=1}^{p} \mathsf{x}_{ik}^2 + \sum_{k=1}^{p} \mathsf{x}_{jk}^2 - 2 \sum_{k=1}^{p} \mathsf{x}_{ik}\mathsf{x}_{jk} \end{aligned}$$

Thus

$$d_{ij}^2 = q_{ii} + q_{jj} - 2q_{ij} \tag{3}$$

where $q_{ii} = \sum_{k=1}^{p} \mathsf{x}_{ik}^2$; $\quad q_{ij} = \sum_{k=1}^{p} \mathsf{x}_{ik}\mathsf{x}_{jk}$.

Observe that q_{ij} is the (i, j)th element of the matrix $\mathbf{Q} = \mathsf{X}\mathsf{X}'$. Thus, the matrix $\mathsf{X}\mathsf{X}'$ has all the information needed to construct distances.

Now require that columns of X are centered, i.e.

$$\sum_{i=1}^{n} \mathsf{x}_{ik} = 0, i = 1, \ldots p$$

This implies that

$$\begin{aligned} \sum_{i=1}^{n} q_{ij} &= \sum_{i=1}^{n} (\sum_{k=1}^{p} \mathsf{x}_{ik}\mathsf{x}_{jk}) \\ &= \sum_{k=1}^{p} (\sum_{i=1}^{n} \mathsf{x}_{ik}\mathsf{x}_{jk}) \\ &= \sum_{k=1}^{p} (\mathsf{x}_{jk} \sum_{i=1}^{n} \mathsf{x}_{ik}) \\ &= 0 \end{aligned}$$

i.e., that the rows and columns of $\mathbf{Q}$ sum to zero.

23.1 An exact representation?

It will now be shown that given distances d_{ij}, then equation 3 uniquely determines a matrix $\mathbf{Q}$ whose rows and columns sum to zero. The demand that the d_{ij} satisfy the triangle inequality is unfortunately not enough to guarantee that this matrix will be positive definite, as is required to yield a configuration that can be exactly embedded in Euclidean space.

Set $A = \sum_{i=1}^{n} q_{ii}$. Summing $d_{ij} = q_{ii} + q_{jj} - 2q_{ij}$ over i, it follows that

$$\sum_{i=1}^{n} d_{ij}^2 = A + nq_{jj} \tag{4}$$

$$\sum_{j=1}^{n} d_{ij}^2 = A + nq_{ii} \tag{5}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij}^2 = 2nA \tag{6}$$

From equation 6

$$A = \frac{1}{2n} \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij}^2 \tag{7}$$

From equation 3, substituting for q_{ii} and q_{jj} from equations 4 and 5 above, and then for A from equation 7 above

$$\begin{aligned} q_{ij} &= -\frac{1}{2}d_{ij}^2 + \frac{1}{2n}(\sum_{i=1}^{n} d_{ij}^2 + \sum_{j=1}^{n} d_{ij}^2 - 2A) \\ &= -\frac{1}{2}d_{ij}^2 + \frac{1}{2n}(\sum_{i=1}^{n} d_{ij}^2 + \sum_{j=1}^{n} d_{ij}^2 - \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}^2) \end{aligned}$$

Having thus recovered a symmetric matrix $\mathbf{Q}$, the spectral decomposition yields

$$\mathbf{Q} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$$

where $\mathbf{\Lambda}$ is a diagonal matrix. The diagonal elements λ_i are ordered so that

$$\lambda_1 \geq \lambda_2 \ldots \geq \lambda_n$$

An exact embedding is possible if and only if $\lambda_i \geq 0$ for all i. For this, set

$$\mathsf{X} = \mathbf{U}\mathbf{\Lambda}^{\frac{1}{2}}$$

24 References

References

Aldrich, 1995. Correlations genuine and spurious in Pearson and Yule. *Statistical Science*, 10:364–376.

Ambroise, C. and McLachlan, G.J. 2001. Selection bias in gene extraction on the basis of microarray gene-expression data. *Proceedings of the National Academy of Sciences USA*, **99** 6562-6566.

Bates, D. 2007. Comparing least squares calculations. *Vignette "Comparisons" accompanying the package "Matrix" for R.*

Berk, R. 2008. *Statistical Learning from a Regression Perspective.*
[Berk has insightful commentary that injects needed reality checks into the discussion of data mining and statistical learning.]

Bickel, P. J.; Hammel, E. A.; and O'Connell, J. W., 1975. Sex bias in graduate admissions: data from Berkeley. *Science*, 187:398–403.

Bishop, C. M. 2006. Pattern Recognition and Machine Learning. Springer.
[This is a comprehensive account, from a Bayesian decision theory perspective, of machine learning methods. The range of methodologies has a large overlap with what might be expected in a text on statistical learning. Models that account for dependence are discussed, but with an inattention to the statistical literature that is untypical of the book as a whole. Geometrical intuition is strongly emphasised.]

Blackard, Jock A. 1998. Comparison of Neural Networks and Discriminant Analysis in Predicting Forest Cover Types. Ph.D. dissertation. Department of Forest Sciences. Colorado State University. Fort Collins, Colorado.
[Data are available from `<URL:\http://www.ics.uci.edu/~mlearn/MLRepository.html>`]

Bland, M. & Altman, D. 2005. Do the left-handed die young? *Significance*, 2:166–170.

Breiman, L. 2001. Statistical modeling: the two cultures (with discussion). *Statistical Science* **16** 199- 231.
[This is a controversial paper whose major claims are, in my view and in that of at least one of the discussants, nonsense. It, and the subsequent discussion are a good read.]

Britton, A., & Marmot, M. 2004. Different measures of alcohol consumption and risk of coronary heart disease and all-cause mortality: 11-year follow-up of the Whitehall II Cohort Study. *Addiction* **99**:109–116.

Carroll, R.J. 2004. Measuring diet. Texas A & M Distinguished Lecturer series.
[Overheads are available from `<URL:http://stat.tamu.edu/~carroll/talks.php`]

Chambers, J.M. 2000. Users, Programmers, and Statistical Software. *ASA Journal of Computational and Graphical Statistics* 9:3 (September, 2000), pp. 404-422.
[Discusses issues that are of importance when software systems are used for data analysis, and how these should affect the design of statistical software systems. In the R project, John Chambers how has a number of very able statistical computing specialists involved with him in thinking through such issues, and to encoding in software the ideas that emerge.]

Cox, D.R. and Solomon, P.J. 2003. *Components of Variance.* Chapman and Hall.
[Multi-level models are, as usually formulated, components of variance models.]

Dalgaard, P. 2002. *Introductory Statistics with R.* Springer-Verlag, New York.
[This is an introductory account of the use of the R language for statistical analysis, with a slant towards biostatistical applications.]

Davey Smith, G. & Ebrahim, S. 2005. What can mendelian randomisation tell us about modifiable behavioural and environmental exposures? British Medical Journal **330**:1076 - 1079.

Economist Data, data everywhere. A special report on managing information. *Economist special report*, Feb. 27, 2010.

Farmer, C.H. 2005. Another look at Meyer and Finney's 'Who wants airbags?'. *Chance*, 19:15–22.

Fuller, W. A. 1987. *Measurement Error Models*. Wiley.

Gower, J. C. & Legendre, P. 1986. Metric and Euclidean properties of dissimilarity coefficients. *Journal of Classification* **3:** 5-48.

Han, J. and Kamber, M. 2001. *Data Mining: Concepts and Techniques.* Morgan Kaufmann.
[This is a widely used data mining text.]

Hand, J., Blunt, G., Kelly, M.G. and Adams, N.M. 2000. Data mining for fun and profit (with discussion). *Statistical Science* **15:** 111-131.
[This gives a statistical perspective on data mining.]

Hand, D.J. 2006. Classifier technology and the illusion of progress. *Statistical Science* **21: 15:** 1-14, and (comment) 15-34.

Hand, D., Mannila, H. and Smyth, P. 2001. *Principles of Data Mining.* MIT Press.
[While better than other treatments of statistical issues that I have seen in data mining texts, there are nevertheless serious gaps in its treatment.]

Hérnan MA, Alonso A, Logan R, Grodstein F, Michels KB, Willett WC, Manson JE, Robins JM. Observaational studies analyzed like randomized experiments: an application to postmenopausal hormone therapy and coronary heart disease (with discussion). *Epidemiology*, in press.

Hastie, T.; Tibshirani, R.; and Friedman, J. 2009. *Elements of Statistical Learning. Data Mining, Inference and Prediction*. Springer.
[This is a comprehensive account of statistical learning approaches, albeit staying strictly within an independent observations theoretical framework.]

Hyndman, R. J.; Koehler, A. B.; Ord, J. K.; and Snyder, R. D. 2008. *Forecasting with Exponential Smoothing: The State Space Approach*, 2^{nd} edn, Springer.

Izenman, A. J. 2008. Modern Multivariate Statistical Techniques. Regression, Classification and Manifold Learning. Springer.
[This is a near encyclopedic account of topics that come generally under the headings of regression, classification, cluster analysis and low-dimensional representation, albeit with a strong statistical learning perspective. Graphical displays are used to excellent effect. Readers are expected to have previous knowledge of probability, statistical theory and methods, multivariate calculus, and linear/matrix algebra. It stays strictly within an independent observations theoretical framework.]

Jackson, R., Broad, J., Connor, J. and Wells, S. 2001. Alcohol and ischaemic heart disease: probably no free lunch. The Lancet 366: 1911-1912.

Koenker, R and Ng, P 2003. SparseM: A sparse matrix package for R. *Journal of Statistical Software* 8(6).

Latter, O. H., 1902. The egg of *cuculus canorus*. an inquiry into the dimensions of the cuckoo's egg and the relation of the variations to the size of the eggs of the foster-parent, with notes on coloration, &c. *Biometrika*, 1:164–176.

Leavitt, S. D. and Dubner, S. J. 2005. *Freakonomics. A Rogue Economist Explores the Hidden Side of Everything*. William Morrow.

Leek JT and Storey JD 2007. Capturing heterogeneity in gene expression studies by surrogate variable analysis. PLoS Genet 3(9): e161. doi:10.1371/journal.pgen.0030161
[Analysis of expression array analyses from biological samples is an interesting and important instance where accounting for dependence between observations makes a large difference to the results. See Subsection 3.2.4.]

Maindonald, J. H. 2003. The role of models in predictive validation. Invited Paper.

Maindonald, J.H. 2004a. Computation and biometry. In Modern Biometry, from Encyclopedia of Life Support Systems (EOLSS), Developed under the Auspices of the UNESCO, Eolss Publishers, Oxford, UK, `http://www.eolss.net`

Maindonald, J.H. 2004b. Statistical Computing. In Modern Biometry, from Encyclopedia of Life Support Systems (EOLSS), Developed under the Auspices of the UNESCO, Eolss Publishers, Oxford, UK, `http://www.eolss.net`.
[This has my view of major current directions in statistical computing software development.]

Maindonald, J.H. 2005. Data, science, and new computing technology. New Zealand Science Review 62: 126-128.

Maindonald, J.H. 2006. Data Mining Methodological Weaknesses and Suggested Fixes. Proceedings of Australasian Data Mining Conference (Aus06), Sydney, Nov 29-30, 2006.
`http://www.maths.anu.edu.au/~johnm/dm/ausdm06/ausdm06-jm.pdf` (paper)
`http://wwwmaths.anu.edu.au/~johnm/dm/ausdm06/ohp-ausdm06.pdf` (overheads)

Maindonald, J. H. and Braun, W.J. 2010. *Data Analysis and Graphics Using R – An Example-Based Approach*, 3^{rd} edition, Cambridge University Press.
`<URL:http://www.maths.anu.edu.au/~johnm/r-book.html>`
[This is aimed at practicing scientists who have some modest statistical sophistication, and at statistical practitioners. It demonstrates the use of the R system for data analysis and for graphics.]

Maindonald, J.H., Waddell, B.C. and Petry, R.J. 2001. Apple cultivar effects on codling moth (Lepidoptera: Tortricidae) egg mortality following fumigation with methyl bromide. Postharvest Biology and Technology **22** 99-110.

Meyer, M.C. and Finney, T. 2005. Who wants airbags?. *Chance* **18**:3-16.

M.C. Meyer. Commentary on "Another look at Meyer and Finney's 'who wants airbags?'". *Chance*, 19:23–24, 2006.

Murrell, P. 2009. Introduction to Data Technologies. Chapman & Hall/CRC.
[Covers HTML, XML, Data formats (plain text, binary, and other), Databases, the SQL query language, Data Processing using R, and regular expressions.]

Newton, A. 1893-1896. Cuckoos. In A. Newton and H. Gadow, eds., *Dictionary of Birds*. A. and C. Black.

R Core Development Team. *An Introduction to* R. Supplied with most installations of R, and available also from CRAN sites (`<URL:http://cran.r-project.org>` gives the list of sites).

Ripley, B. D. 1996. *Pattern Recognition and Neural Networks*. Cambridge University Press.

Rosenbaum, P.R. 1999. Choice as an alternative to control in observational studies. *Statistical Science* **14** 259-278, with following discussion, pp. 279-304.

Rosenbaum, P.R. 2002. *Observational Studies*, 2nd edn. Springer-Verlag.
[This is an important recourse and source of insight for anyone who works with observational data.]

Schatzkin, A., Kipnis, V., Carroll R.J., Midthune, D., Subar, A.F., Bingham, S., Schoeller D.A., Troiano, R.P. and Freedman, L.S. 2003. A comparison of a food frequency questionnaire with a 24-hour recall for use in an epidemiological cohort study: results from the biomarker-based Observing Protein and Energy Nutrition (OPEN) study. *International Journal of Epidemiology* 32: 1054-1062.

Rossouw, J.E., Anderson, G.L., Prentice, RL, et al. Risks and benefits of estrogen plus progestin in healthy postmenopausal women: principal results From the Women's Health Initiative randomized controlled trial. *Journal of the American Medical Association* **2002**:288:321.

Senn, S., 2003. *Dicing with Death: Chance, Risk and Health*. Cambridge University Press.

Simpson, E. H., 1951. The interpretation of interaction in contingency tables. *Journal of the Royal Statistical Society, Series B*, 13:238–241.

Taleb, Naseem. 2004. *Fooled By Randomness: The Hidden Role Of Chance In Life And In The Markets.* Random House, 2ed.
[Has many insightful comments about the over-interpretation of phenomena in which randomness is likely to have a large role.]

Torgo, L. 2003. *Data Mining with R.* (Available from `<URL:http://www.liacc.up.pt/~ltorgo>`)
[This has a data mining flavour. There is a brief discussion of databases. The second of the data sets (a stock market time series) is available as a MySQL database. This may be a good way to start learning about the interface that the *RODBC* package offers to MySQL. The reliance on the R `source()` command for storage and entry of data is not a good idea, in general. Use image (**.RData**) files instead. Comments on statistical issues, and notably on the handling of missing data, suggest approaches that, while widely used in the past, are known to have serious potential problems.]

Venables, W. N. and Ripley, B. D. 2002. *Modern Applied Statistics with* S. Springer-Verlag, 4 edition. See also R Complements to Modern Applied Statistics with S.
`http://www.stats.ox.ac.uk/pub/MASS4/`
[This is a wide-ranging account of statistical methods, including statistical learning methods, with details of the S-PLUS and R code required to carry out the computations. Note especially pp.331–341 (lda and qda) and pp.187–198 (logistic and other GLMs).]

Wilkinson, G. N. & Rogers, C. E. 1973. *Symbolic description of models in analysis of variance.* Applied Statistics 22: 392-399.

Witten, I.H. and Frank, E. 2000. *Data Mining. Practical machine Learning Tools and Techniques with Java Implementations.* Morgan Kaufmann.
[This is a popular data mining text.]

Wood, S. N. 2006. *Generalized Additive Models.* An Introduction with R. Chapman & Hall/CRC.
[This has an elegant treatment of linear models and generalized linear models, as a lead-in to generalized additive models. It is an almost indispensible reference for use of the `mgcv` package for R. Refer to it for the theory of and practical use of regression splines, various types of penalized splines, thin plate splines, etc. A final chapter is devoted to Generalised Additive mixed models.]

Yang, C, & Letourneau, S.(2005) Learning to Predict Train Wheel Failures. The Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2005). Chicago, Illinois, USA. August 21-22, 2005. NRC 48130.
`iit-iti.nrc-cnrc.gc.ca/iit-publications-iti/docs/NRC-48130.pdf`

Young, G. and Smith, R. L. 2005. *Essentials of Statistical Inference*. Cambridge University Press.