MAS6004/472 Computational Inference

Module Information

1. **PLEASE NOTE:** These notes are deprecated. The slides (as opposed to these more verbose notes) contain the official version of the material for this module. These notes are 80% the same as the slides, but the slides don't contain all that is covered here, and not all of what is covered in the slides is covered here. I have made these available in case they are of use to you.

2. Getting help with this module
   My office is I14 in the Hicks Building. I am available for questions etc after every lecture, or you can make an appointment for some other time by contacting me via email `r.d.wilkinson@sheffield.ac.uk`.

3. Please use the discussion boards on MOLE for asking questions. This is so that everyone has the same information and receives the same level of help. If you email me with a question about the course, and it is a question that is relevant to others (e.g., *I don't understand page 2, what does x mean?, is there a typo on page 7?, the coursework question is ambigious* etc), then I will ask you to post your question to the MOLE discussion board before I answer it (note that you can post questions anonymously if you wish). I receive an email when posts appear on MOLE, and will try to answer all queries as soon as I can. Posting them on MOLE also means that other students can help to answer your questions.

4. Course materials
   All course materials are available on MOLE, including lecture slides.

5. Assessed coursework
   There are 3 pieces of coursework for both MAS472 and MAS6004 that contribute 5% each towards your mark for the module.

   - A integer mark will be awarded out of 5 for each piece of coursework.
   - Solutions will be provided.

   Deadlines for these are Tuesdays March 21st, April 4th, and May 9th.

6. The exam will a closed book two hour formal written examination worth 85% of the module mark.

7. R practicals
   There will be four R computer practical classes replacing lectures for undergraduates and residential MSc students. These will take place on Friday's in even weeks of the semester (weeks 2, 4, 6, 8 and 10) in the Diamond Building in computer lab 3.

Contents

0.1 Books and online resources

Materials for this course are available on MOLE. MOLE can either be accessed directly from

`http://vista.shef.ac.uk`

or via MUSE. Please contact me if you are unable to access the course page on MOLE.

Software

During this course we will be using the software package R. This is available on the managed XP network, and can be downloaded for free from `http://www.r-project.org/`
Rstudio (`http://www.rstudio.com/`) is the recommended way for all R work. It provides an excellent GUI that allows you to edit files, build packages, open multiple figures etc.

Textbooks

This course covers a range of different topics, and I am not aware of one book that is suitable for them all. One book suitable for R is

- Robert, C. and Casella, G. (2009) *Introducing Monte Carlo Methods with R*. Springer.

These books are recommended for material on Monte Carlo methods, randomisation/permutation tests, bootstrapping and random number generation:

- Gentle, J. E. (2002) *Elements of Computational Statistics*. Springer.
- Morgan, B. J. T. (1984) *Elements of Simulation*, Wiley.

These three books are suitable for likelihood methods, including the EM algorithm:

- Garthwaite, P. H., Joliffe I. T. & Jones B. (2002) *Statistical Inference*, 2nd edition. Oxford University Press.
- Little, R. J. A. & Rubin, D. B. (2002) *Statistical Analysis with Missing Data*. Wiley.
- Tanner, M. A. (1996) *Tools for Statistical Inference*. Springer.

1 Introduction

In this module we will be studying a range of computational tools for conducting statistical inference. We will learn how to use modern computing power to provide simple solutions to complex statistical problems. We will also see how these tools enable us to do inference even when the usual distributional assumptions are not valid.

In the first half of the module we will study the use of simulation in inference. We have already seen the key role simulation plays in applied Bayesian Statistics through MCMC methods. Here, we will learn how simulation can be used in frequentist inference. In the second half of the module we will study both theoretical and computational aspects of likelihood theory.

2 Monte Carlo methods

2.1 Introduction

The **Monte Carlo method** is an extremely versatile technique that can be used to give approximate solutions to a wide variety of problems in probability and statistics (though usually to a sufficient level of accuracy for practical purposes). Monte Carlo methods generally require implementation on a computer, and during this course we will be using the software package R.

2.2 Example problems

We will begin by considering a variety of example problems. Suppose you were confronted by any of these in a statistics exam. How might you go about tackling them?

1. A particular site is being considered for a wind farm. At that site, the log of the wind speed in m/s on day t is known to follow an $AR(2)$ process:

$$Y_t = 0.6Y_{t-1} + 0.4Y_{t-2} + \varepsilon_t, \tag{1}$$

   with $\varepsilon_t \sim N(0, 0.01)$. If $Y_1 = Y_2 = 1.5$, what is the probability that the wind speed $\exp(Y_t)$ will be below 15 kmh for more than 10 days in a 100 day period?

2. Given a sample of 5 standard normal random variables $X_1, \ldots, X_5$, what is the variance of $max_i\{X_i\} - min_i\{X_i\}$?

3. The following simple model describes the concentration of pollutant at any point in a region following a release from a point source

$$C(x, y, z) = \frac{Q}{2\pi u_{10}\sigma_z\sigma_y} \exp\left[-\frac{1}{2}\left\{\frac{y^2}{\sigma_y^2} + \frac{(z-h)^2}{\sigma_z^2}\right\}\right], \tag{2}$$

   where C is air concentration of the pollutant, Q is the rate of emission of the pollutant, u_{10} is the wind speed at $10m$ above ground, σ_y and σ_z give the diffusion in the horizontal and vertical directions respectively, h is the release height, and (x, y, z) are the coordinates along the wind direction, cross wind and above ground respectively.

   The pollutant is released from a source 50m above ground, with an emission rate of 100 units. However, the wind speed and diffusion parameters σ_y and σ_z are not known exactly, and are considered to be random variables with lognormal distributions:

$$\begin{aligned} \log u_{10} &\sim N(2, .1) \\ \log \sigma_y^2 &\sim N(10, 0.2) \\ \log \sigma_z^2 &\sim N(5, 0.05) \end{aligned}$$

   What is the 95th percentile of $C(100, 100, 40)$?

4. A hospital ward has 8 beds. The number of patients arriving each day is uniformly distributed between 0 and 5 inclusive. The length of stay for each patient is also uniformly distributed between 1 and 3 days inclusive. If all 8 beds are free initially, what is the expected number of days before there are more patients than beds?

5. Calculate the probability of dealing yourself a winning hand in the card game patience / solitaire.

These problems are all very difficult/impossible to tackle analytically. However, a single technique, the Monte Carlo method, can be used to obtain approximate answers to all of them (and as stated at the beginning, usually to a sufficient level of accuracy for all practical purposes).

In the example problems, you were asked for quantities such as means, variance, percentiles or probabilities (which can be thought of as proportions). In general, if you were asked to estimate one of these summaries of a particular population, you might expect that this would involve taking a *sample* from the population, and reporting the sample mean, variance etc. In these problems, it's not possible/practical to physically go out and *observe* members of the population, but nonetheless it's still possible to obtain a sample. In the Monte Carlo method, the sample of data is *simulated*. We can do this, because for each problem, the data generating process is known completely (for example, in question 1, the log wind speeds are known to follow an autoregressive process with all the parameters specified). Once we have the sample, we estimate the population mean, variance etc in the usual way. And because (most of the time) we can generate very large samples, we expect our estimates to be very accurate.

2.3 Some useful results

Before describing Monte Carlo solutions to each of these problems, we will review some helpful theory. Firstly, note that in the example problems, in most cases you were asked to give an *expectation*; either the expectation of some random variable X directly, of the expectation of some function of a random variable. For example, a variance can be expressed as two expectations:

$$Var(X) = E(X^2) - E(X)^2. \tag{3}$$

A probability $P(X < a)$ can also be expressed as an expectation: the expectation of an indicator function of X. An indicator function $I(E)$ of an event E is defined as

$$I(E) = \begin{cases} 1 & E \text{ is true} \\ 0 & E \text{ is false} \end{cases} \tag{4}$$

Then we have

$$\begin{aligned} E\{I(X < a)\} &= 1 \times P(X < a) + 0 \times P(X \geq a) & (5) \\ &= P(X < a). & (6) \end{aligned}$$

Now, given a sample $X_1, \ldots, X_n$ of independently drawn values from the distribution of X, we know that

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{7}$$

is an unbiased estimator of $E(X)$. Additionally if $Var(X)$ is finite and equal to σ^2, then,

$$Var(\bar{X}) = \frac{1}{n}\sigma^2, \tag{8}$$

so that increasing n decreases the variance of the estimator $\bar{X}$. If μ is the true value of $E(X)$, then from the strong law of large numbers, $\bar{X} \to \mu$ with probability one as $n \to \infty$. Finally, we can use the central limit theorem to construct confidence intervals for the estimator $\bar{X}$: for suitably large n we have

$$\bar{X} \sim N\left(\mu, \frac{\sigma^2}{n}\right). \tag{9}$$

Typically, σ^2 will be unknown, but we can of course estimate σ^2 in the usual way:

$$\hat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2. \tag{10}$$

If instead of $E(X)$ we require $E\{f(X)\}$, random observations from the distribution of $f(X)$ can be generated by generating $X_1, \ldots, X_n$ from the distribution of X, and then evaluating $f(X_1), \ldots, f(X_n)$. The preceding results can then be applied when estimating variances or probabilities of events.

A percentile of the distribution of a random variable X can be estimated by taking the sample percentile from the generated sample of values $X_1, \ldots, X_n$. Informally, we would expect the estimate to be more accurate as n increases. Determining a percentile is equivalent to inverting the distribution function; if for example we wish to know the 95th percentile, we must find ν such that

$$P(X \leq \nu) = 0.95, \tag{11}$$

so the more accurately we estimate the distribution function $F(X)$, the more accurate we would expect the estimate of any particular percentile to be.

2.4 Generating random variables in R

The Monte Carlo method requires the ability to generate random numbers, and we will study this topic in more detail later on in the course. For the time being, we will simply use functions R. (Broadly speaking, computers generate 'psuedo-random' uniform numbers between zero and one, i.e., numbers that can be assumed to be random even though they are in fact generated deterministicaly. These uniform random numbers are then transformed to give random numbers from other distributions).

Uniform$[a, b]$	`runif(n,min=a,max=b)`
Normal(m, s^2)	`rnorm(n,mean=m,sd=s)`
Binomial(N, p)	`rbinom(n,N,p)`
Poisson(r)	`rpois(n,r)`
Exponential(r)	`rexp(n,rate=r)`
Beta(a, b)	`rbeta(n,a,b)`
Gamma(a, b)	`rgamma(n,a,rate=b)`
lognormal: $\log x \sim N(m, s^2)$	`rlnorm(n,meanlog=m,sdlog=s)`

Rcan generate pseudo-random observations from various distributions. The following commands can be used to generate n random variables from the corresponding distributions:

So, for example, to generate 1000 uniform random numbers between 0 and 1, and store them as a vector x, use the command `x<-runif(1000,min=0,max=1)`. For more details on any command, e.g `rnorm`, type `?rnorm` in R.

2.5 Monte Carlo solutions to the example problems

Later on in the course we will obtain numerical answers to some of these problems using R. Here we just give outline descriptions of the solutions. Each solution is presented in the form of **pseudo code**. While not an actual programming language in itself, pseudo code will specify a sequence of tasks a computer program will need to perform in order to obtain the solution.

1. Define E to be the event of interest: the event that in 100 days the wind speed is below 15kmh for more than 10 days. To estimate $P(E)$, we can simply generate lots of individual time series, and count the proportion of times E occurs for each single time series.

   For $i = 1, 2, \ldots, N$:

   (a) Generate ith realisation of the time series process:
   For $t = 3, 4, \ldots, 100$:
   - Sample ε_t from $N(0, 0.01)$
   - Set $Y_t \leftarrow 0.6Y_{t-1} + 0.4Y_{t-2} + \varepsilon_t$

   (b) Count number of elements of $\{Y_1, \ldots, Y_{100}\}$ less than $\log 4.167$:
   - Set $X_i \leftarrow \sum_{t=1}^{100} I\{Y_t < \log 4.167\}$

   (c) Determine if event E has occured for time series i:
   - Set $E_i \leftarrow I\{X_i > 10\}$

   Estimate $P(E)$ by $\frac{1}{N}\sum_{i=1}^{N} E_i$

2. Define Z to be the difference between the maximum and minimum of 5 standard normal random variables. It is straightforward to generate random observations from the distribution of Z, and the variance can be estimated as follows:

> For $i = 1, 2, \ldots, N$:
>
> (a) Generate the ith random value of Z:
>
> - Sample $X_1, \ldots, X_5$ independently from $N(0, 1)$
> - Set $Z_i \leftarrow \max\{X_1, \ldots, X_5\} - \min\{X_1, \ldots, X_5\}$
>
> Estimate $Var(Z)$ by $\frac{1}{N-1}\sum_{i=1}^{N}(Z_i - \bar{Z})^2$

3. Effectively, we are dealing with a transformation of a random variable; given random variables $\mathbf{X} = (X_1, \ldots, X_d)$ we want to know the distribution of $Y = f(\mathbf{X})$. This can be determined analytically for fairly simple functions $f(.)$, but a Monte Carlo approach can be used for any function $f(.)$: we sample the unknown input parameters from their distributions, then evaluate the function to obtain an output value from its distribution. Given a suitably large sample, the 95th percentile from the distribution of $C(100, 100, 40)$ can be estimated by the 95th percentile from the sample of simulated values of $C(100, 100, 40)$. This is summarised in the following algorithm:

> For $i = 1, 2, \ldots, N$:
>
> (a) Sample a set of input values:
>
> - Sample $u_{10,i}$ from $\log N(2, .1)$
> - Sample $\sigma^2_{y,i}$ from $\log N(10, 0.2)$
> - Sample $\sigma^2_{z,i}$ from $\log N(5, 0.05)$
>
> (b) Evaluate the model output C_i:
>
> - Set $C_i \leftarrow \frac{100}{2\pi u_{10,i}\sigma_{z,i}\sigma_{y,i}} \exp\left[-\frac{1}{2}\left\{\frac{100^2}{\sigma^2_{y,i}} + \frac{(40-50)^2}{\sigma^2_{z,i}}\right\}\right]$
>
> Return the 95th percentile of $C_1, C_2, \ldots, C_N$.

The situation described here frequently occurs in practice; mathematical models are used various applications such as the environment, engineering or finance, and invariably there is uncertainty surrounding values that the model parameters should take. Investigating the effect of this parameter uncertainty is sometimes referred to as *uncertainty analysis* or *probabilistic sensitivity analysis*. A particularly important example of this is in the study of climate change, where climate forecasts are based on large mathematical models with uncertain model parameters (and where a failure to investigate parameter uncertainty properly gives ammunition to climate change deniers!) Note also that for large models, evaluating $f(\mathbf{X})$ can take considerable computing time (unlike the Gaussian plume model

in which we can calculate $f(\mathbf{X})$ very quickly). This motivates the need for *efficient* Monte Carlo procedures that give accurate answers for small values of N. We will return to the issue of efficiency in Monte Carlo methods later in the course.

4. Define W to be the number of days before the first patient arrives to find no available beds. The question has asked us to give $E(W)$. If we can generate $W_1, \ldots, W_n$ from the distribution of W, we can then estimate $E(W)$ by $\bar{W}$. All we need therefore is an algorithm for generating the W_is. This is essentially done by simulating the overall process of arrivals and duration of stays for the patients, and then observing when a patient arrives when all the beds are in use. The algorithm for generating a single W_i is as follows:

Define

B_j : number of beds available at beginning of day j

N_j : number of arrivals at beginning of day j

F_j : number of occupied beds that will become free at end of day j

(a) Initialisation:

- Set $B_1 \leftarrow 8$
- Set $F_k \leftarrow 0$ for all k
- Set $j \leftarrow 1$

(b) Determine how many patients arrive on day j

- Sample N_j from discrete uniform distribution on $\{0, 1, 2, 3, 4, 5\}$
- If $N_j > B_j$, there are more patients than beds; report $W_i = j - 1$ and stop.

(c) Generate the length of stay for the N_j arrivals and determine when they leave:
For $k = 1, \ldots, N_j$:

- Sample the length of stay S_k for arrival k from the discrete uniform distribution on $\{1, 2, 3\}$.
- Set $F_{j-1+S_k} \leftarrow F_{j-1+S_k} + 1$.

(d) Determine number of beds available next day:

- Set $B_{j+1} \leftarrow B_j + F_j - N_j$.

Set $j \leftarrow j + 1$ and return to step (b).

This process can then be repeated to give the desired sample $W_1, \ldots, W_n$.

5. I have included this problem for historical reasons though actually programming a solution would be outside the scope of this course.
One of the first users of the Monte Carlo method was the Polish mathematician Stanislaw Ulam. He is quoted in Eckhardt (1987):

"The first thoughts and attempts I made to practice [the Monte Carlo Method] were suggested by a question which occurred to me in 1946 as I was convalescing from an illness and playing solitaires [patience]. The question was what are the chances that a Canfield solitaire laid out with 52 cards will come out successfully? After spending a lot of time trying to estimate them by pure combinatorial calculations, I wondered whether a more practical method than abstract thinking might not be to lay it out say one hundred times and simply observe and count the number of successful plays. This was already possible to envisage with the beginning of the new era of fast computers, and I immediately thought of problems of neutron diffusion and other questions of mathematical physics, and more generally how to change processes described by certain differential equations into an equivalent form interpretable as a succession of random operations. Later [in 1946, I] described the idea to John von Neumann, and we began to plan actual calculations."

Another claim to fame by Ulam is as co-designer of the Hydrogen bomb.

2.6 Exercises

1. Using R, estimate $E(\cos X^2)$ where $X \sim Gamma(5, 2)$. Provide a 95% confidence interval to indicate the accuracy of your estimate.

2. [Hard!] Consider a more complex version of the hospital beds problem given in the examples: Patients may arrive at any time during the day. The arrival of patients is described by a Poisson process with rate 3 per day (so that the time between patients arriving in days is given by an exp(3) distribution.) The length of stay for each patient is described by an exp(0.5) distribution. Using simulation, give an outline description of how you would estimate the expected number of days until there are more patients than beds.

3 Monte Carlo Integration

3.1 Introduction

In this section we will look at an application of simulation methods in numerical integration. Integration is of course a commonly used operation in statistics, as means, variances and probabilities for continuous random variables are all defined as integrals. Monte Carlo integration is particularly useful in Bayesian statistics.

Consider the probability p that a standard normal random variable will lie in the interval $[0, 1]$. This can be written as an integral

$$p = \int_0^1 \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) dx. \tag{12}$$

Even for this simple example, we cannot evaluate this integral analytically. Three methods for estimating/evaluating this probability would be:

1. numerical integration
2. looking up $\Phi(1)$ (and $\Phi(0)$) in statistical tables
3. given a sample of standard normal random variables $Z_1, \ldots, Z_n$, look at the proportion of Z_is occurring in the interval $[0, 1]$.

Here we will consider an alternative approach, based on simulation:

1. Let Y be a random variable, and consider an arbitrary function $f(Y)$. To generate a random value from the distribution of $f(Y)$, we simply have to generate a a random Y from the distribution of Y, and then evaluate $f(Y)$.
2. Providing $E\{f(Y)\}$ exists, given a sample $f(Y_1), \ldots, f(Y_n)$, the estimator $\frac{1}{n}\sum_{i=1}^{n} f(Y_i)$ is an unbiased estimator of $E\{f(Y)\}$:
$$E\left\{\frac{1}{n}\sum_{i=1}^{n} f(Y_i)\right\} = \frac{1}{n}\sum_{i=1}^{n} E\{f(Y_i)\} = \frac{1}{n} \times nE\{f(Y)\} = E\{f(Y)\} \tag{13}$$
3. Let X be a random variable with a $U[0, 1]$ distribution. For an arbitrary function $f(X)$, what is the expectation of $f(X)$? This is given by the integral
$$E\{f(X)\} = \int_0^1 f(x)1dx, \tag{14}$$
(as the density function of a $U[0, 1]$ random variable is equal to 1 for $X \in [0, 1]$).
4. Now we'll choose f to be the function $f(X) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{X^2}{2}\right)$. Of course, remembering that we have defined X to be a random variable uniformly distributed on $[0, 1]$, it still holds that
$$E\{f(X)\} = \int_0^1 \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) 1dx \tag{15}$$
But given a sample $f(X_1), \ldots, f(X_n)$ from the distribution of $f(X)$, we can estimate $E\{f(X)\}$ by $\frac{1}{n}\sum_{i=1}^{n} f(X_i)$. This must also be an unbiased estimate of p, as from equations (12) and (15), we have $p = E\{f(X)\}$. So in this example, the **Monte Carlo** estimate $\hat{p}$ of the integral p is therefore given by
$$\hat{p} = \frac{1}{n}\sum_{i=1}^{n} f(X_i), \tag{16}$$
where X_i is drawn randomly from the $U[0, 1]$ distribution.

Note the key idea of this method: we re-expressed the integral of interest (12) as an *expectation*. The expectation (and therefore the integral) could then be estimated by the mean of a sample of random variables.

3.2 The general framework

In general, let the integral of interest be

$$R = \int f(x)dx \tag{17}$$

As we have seen, Monte Carlo integration works by re-expressing this integral as an expectation. Now suppose that $g(x)$ is some density function that we can easily produce a sample of values $X_1, \ldots, X_n$ from. (In the previous example $g(x)$ was the uniform density function on $[0, 1]$). So how do we re-write (17) as an expectation of a function of a random variable X with density function $g(x)$? Simply as

$$\begin{aligned} R &= \int \frac{f(x)}{g(x)} g(x)dx & (18)\\ &= \int h(x)g(x)dx, & (19) \end{aligned}$$

with $h(x) = \frac{f(x)}{g(x)}$. So we now have

$$R = E\{h(X)\}, \tag{20}$$

where X has the density function $g(x)$. If we now sample $X_1, \ldots, X_n$ from $g(x)$, then evaluate $h(X_1), \ldots, h(X_n)$, then

$$\hat{R} = \frac{1}{n} \sum_{i=1}^{n} h(X_i) \tag{21}$$

is an unbiased estimator of R.

3.3 Example

Use Monte Carlo integration to estimate

$$R = \int_{-1}^{1} \exp(-x^2)dx. \tag{22}$$

We'll consider two different choices for $g(x)$.

1. A uniform density on $[-1, 1]$: $g(x) = 0.5$ for $x \in [-1, 1]$. We sample $X_1, \ldots, X_n$ from $U[-1, 1]$, and estimate R by

$$\hat{R} = \frac{1}{n} \sum_{i=1}^{n} \frac{\exp(-X_i^2)}{g(X_i)} = \frac{1}{n} \sum_{i=1}^{n} 2\exp(-X_i^2). \tag{23}$$

   Below we tabulate values of $\hat{I}$ for various n:

   Evaluating the integral using Simpson's rule gives $R = 1.494$.

n	10	100	1000	100000
$\hat{R}$	1.845	1.505	1.493	1.492

2. A normal density function $N(0, 0.5)$. In this case, a sampled value X from $g(x)$ is not constrained to lie in $[-1, 1]$. To allow for this, we re-write R as

$$R = \int_{-\infty}^{\infty} I\{-1 \leq x \leq 1\} \exp(-x^2) dx, \tag{24}$$

where $I\{\}$ denotes the indicator function. We now sample $X_1, \ldots, X_n$ from $N(0, 0.5)$ and estimate R by

$$\hat{R} = \frac{1}{n} \sum_{i=1}^{n} \frac{I\{-1 \leq X_i \leq 1\} \exp(-X_i^2)}{g(X_i)} = \frac{1}{n} \sum_{i=1}^{n} \pi^{1/2} I\{-1 \leq X_i \leq 1\}. \tag{25}$$

Below we tabulate values of $\hat{I}$ for various n:

n	10	100	1000	100000
$\hat{R}$	1.595	1.559	1.514	1.491

3.4 Choice of $g(x)$

The density $g(x)$ needs to mimic the function $f(x)$ as closely as possible. To illustrate why, we return to the previous example of estimating

$$R = \int_{-1}^{1} \exp(-x^2) dx. \tag{26}$$

We will now consider two poor choices of g:

1. A uniform density on $[0, 1]$: $g(x) = 1$ for $x \in [0, 1]$. Recall that we can write R as

$$R = \int_{-\infty}^{\infty} I\{-1 \leq x \leq 1\} \exp(-x^2) dx, \tag{27}$$

so that the for the integrand we have $f(x) = I\{-1 \leq x \leq 1\} \exp(-x^2)$. But for $x \in [-1, 0)$, we have $f(x) > 0$ and $g(x) = 0$. This means that we will never sample values of x in the interval $[-1, 0)$ even though the integrand $f(x)$ is non-zero here. The Monte Carlo estimate of R is given by

$$\hat{R} = \frac{1}{n} \sum_{i=1}^{n} I\{-1 \leq X_i \leq 1\} \exp(-X_i^2). \tag{28}$$

In the following table we see that R does not converge to the true value:

Unsurprisingly, $\hat{R}$ converges to half the true value of R. The point here is that we must have $g(x) > 0$ for all x where $f(x) > 0$.

n	10	100	1000	100000
$\hat{R}$	0.766	0.734	0.754	0.746

2. A normal density $N(0, 0.09)$. In this case, we have $g(x) > 0$ for $x \in [-1, 1]$, but we when we sample x from g, we expect around 95% of the values to lie in the range $(-0.6, 6)$. The Monte Carlo estimate of R is given by

$$\hat{R} = \frac{1}{n}\sum_{i=1}^{n} I\{-1 \leq X_i \leq 1\}\frac{\exp(-X_i^2)\sqrt{0.18\pi}}{\exp(-5.56X_i^2)}. \tag{29}$$

Larger samples are needed to estimate R accurately:

n	10	100	1000	100000
$\hat{R}$	0.857	2.115	1.538	1.489

3.5 Convergence

Provided $f(x) > 0 \Rightarrow g(x) > 0$, the estimate $\hat{R}$ will converge to R as $n \to \infty$. In addition, we can use the central limit theorem to derive a confidence interval for $\hat{R}$; for suitably large n, we have

$$\hat{R} \sim N\left(R, \frac{\sigma^2}{n}\right), \tag{30}$$

where $\sigma^2 = Var\{h(X_i)\}$. Of course, σ^2 will be unknown, but we can estimate it by

$$\hat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left\{h(X_i) - \hat{R}\right\}^2 \tag{31}$$

We can then report the confidence interval as

$$\hat{R} \pm Z_{1-\alpha/2}\sqrt{\hat{\sigma}^2/n}, \tag{32}$$

where $Z_{1-\alpha/2}$ is the upper $100(1-\alpha/2)$ % point of the standard normal distribution. This gives us a means of determining how large n should be to achieve a desired accuracy, and a way comparing different choices of g. (Note that it usually won't be necessary to use the t_{n-1} distribution instead of the normal, as n is likely to be very large).

In the following table we show estimates of σ^2 for the three (valid) functions used in the example:
confirming the result that $N(0, 0.09)$ was a poor choice for $g(.)$.

g	$\hat{\sigma}^2$
$U[-1,1]$	0.16
$N(0,0.5)$	0.42
$N(0,0.09)$	6.81

3.6 Monte Carlo or numerical integration?

In the example we considered, it was not actually necessary to use Monte Carlo; the integral could be estimated accurately based on a small number of evaluations of $f(x)$ using Simpson's rule. Where Monte Carlo outperforms numerical integration is in high dimensional problems; for example in evaluating expectations such as

$$\int f(\mathbf{x})g(\mathbf{x})d\mathbf{x}, \tag{33}$$

where $\mathbf{x}$ is a vector of random variables with joint density function $g(\mathbf{x})$. As the dimension of $\mathbf{x}$ increases, the number of evaluations of $f(\mathbf{x})$ needed for numerical integration increases rapidly, and can make numerical integration impractical. Monte Carlo can often deal with high dimensional integrals without difficulty.

3.7 Monte Carlo integration in Bayesian statistics

In Bayesian statistics, we typically use Monte Carlo integration when we are unable to work with conjugate prior distributions. Consider first a random variable X, with density function $f(x)$. If we can sample $X_1, \ldots, X_n$ from $f(X)$, we can estimate $E(X)$ by $\frac{1}{n}\sum_{i=1}^{n} x_i$. Note that this is a special case of Monte Carlo integration:

$$E(X) = \int_{-\infty}^{\infty} xf(x)dx = \int_{-\infty}^{\infty} \frac{xf(x)}{g(x)}g(x)dx, \tag{34}$$

where we choose $g(x) = f(x)$ and so $h(x) = \{xf(x)\}/\{g(x)\} = x$. Now suppose we want to compute the posterior mean of a particular variable θ_i from the joint posterior density $f(\boldsymbol{\theta}|\mathbf{x})$. Writing $\boldsymbol{\theta}_{-i}$ as all elements of $\boldsymbol{\theta}$ except θ_i, the posterior mean of θ_i is

$$E(\theta_i|\mathbf{x}) = \int_{\theta_i}\int_{\boldsymbol{\theta}_{-i}} \theta_i f(\boldsymbol{\theta}|\mathbf{x})d\boldsymbol{\theta}_{-i}d\theta_i = \int_{\boldsymbol{\theta}} \theta_i f(\boldsymbol{\theta}|\mathbf{x})d\boldsymbol{\theta}. \tag{35}$$

We saw in the first semester how, typically, we cannot solve these integrals analytically (and we may only know $f(\boldsymbol{\theta}|\mathbf{x})$ up to proportionality in any case). We can, however, use Monte Carlo integration, with $g(\boldsymbol{\theta}) = f(\boldsymbol{\theta}|\mathbf{x})$ and $h(\boldsymbol{\theta}) = \theta_i$. We then only need samples $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots$ (or more precisely, samples $\theta_{i,1}, \theta_{i,2}, \ldots$) from $g(\boldsymbol{\theta}) = f(\boldsymbol{\theta}|\mathbf{x})$, which could be obtained using Markov chain Monte Carlo.

3.8 Exercises

1. Use Monte Carlo integration to estimate

$$R = \int_{-1}^{1} e^x x^2 \cos x dx.$$

   Use both a uniform density function and a normal density function for $g(x)$. In each case, estimate the sample size required to ensure that the width of a 95% confidence interval for R is no greater than 0.1.

2. Use Monte Carlo integration to estimate

$$\int_{-\infty}^{\infty} \frac{\cos x}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) dx.$$

4 Simulation methods in inference

In the example problems in section 2.2, we were given a complete description of the data generating process and we used Monte Carlo methods to learn about the implications of that process, by simulating data. This idea can be used in hypothesis testing, specifically to derive distributions of "non-standard" test statistics.

4.1 The hypothesis testing framework

We will now review the framework for hypothesis testing. The idea here will be to gain an appreciation for when hypothesis testing through analytical means is likely to prove difficult.

We have collected data $X_1, \ldots, X_n$ and wish to test some hypothesis regarding the process that generated this data. We proceed as follows;

1. An assumption H_0 is made about the data generating process. For example, H_0 may state that $X_1, \ldots, X_n$ are iid $N(0, 1)$ random variables.

2. Some function of the data $T(X_1, \ldots, X_n)$, the test statistic, is calculated. Continuing the example, for a one-sided test we may have

$$T(X_1, \ldots, X_n) = \frac{1}{n} \sum_{i=1}^{n} X_i. \tag{36}$$

   Note that increasing values of $T(X_1, \ldots, X_n)$ should be less likely if H_0 is true.

3. Assuming that H_0 is true, the theoretical sampling distribution of the test statistic T is derived. In this case we would have

$$T \sim N\left(0, \frac{1}{n}\right). \tag{37}$$

4. The observed value of the test statistic, T_{obs}, is compared with the distribution of T *if* H_0 *were true*. How we report the conclusion depends on whether we are using Fisher's or the Neyman-Pearson approach to hypothesis testing:

   (a) Neyman-Pearson: if T_{obs} is greater than some pre-specified value c, we reject H_0. The value c is chosen such that the probability of rejecting H_0 when H_0 is true is α, the size of the test (also pre-specified), i.e. $P(T \geq c|H_0) = \alpha$.

   (b) Fisher: we compute the p-value, a measure of the strength of evidence against H_0. This is given by $P(T \geq T_{obs}|H_0)$. In practice, we often choose to state that we reject H_0 if the p-value is sufficiently small, but it is not necessary to do so.

Clearly this process is dependent on being able to derive the sampling distribution of T under the hypothesis H_0; otherwise we will not know which values of T are plausible and which are not. Some reasons why we may not know the sampling distribution of the test statistic are as follows:

1. If T is not some fairly simple function of the data, we may not be able to derive its distribution analytically

2. The data $X_1, \ldots, X_n$ may not have been drawn randomly from the population of interest. Even if the hypothesis were true, we still may not know what the distribution of T is for a non-random sample.

3. Certain distributional assumptions regarding the data $X_1, \ldots, X_n$ may not hold. For example, the hypothesis may be a statement about the population mean only, but additional assumptions may be made such as equal variance, normality etc. If these do not hold, then we may be assuming the wrong sampling distribution for T.

4.2 Monte Carlo Testing

We'll begin with a very simple example:
Suppose we have an observation X with $X \sim N(\mu, 1)$, and with μ unknown. We wish to test the hypothesis

$$H_0 : \mu = 0,$$

at the 5% level, and observe $X = 2$. The test statistic in this case for a two-sided test is just $T = |X|$, and we need to determine the 95% critical value, denoted by c, where

$$P(T \leq c) = 0.95, \tag{38}$$

in other words, we need the 95th percentile from the distribution of T.

From statistical tables, we would see that $c = 1.96$, and so we would reject the null hypothesis. Now suppose instead we had to perform the hypothesis test without using statistical tables. What could we do?

We have already seen how simulation can be used to estimate percentiles. Consequently, we can estimate c by randomly generating sample test statistics $T_1, \ldots, T_n$, by randomly sampling $Z_1, \ldots, Z_n$, and setting $T_1 = |Z_1|, \ldots, T_n = |Z_n|$, and then taking the sample 95th percentile of $T_1, \ldots, T_n$. This would be done in R with $n = 1000$ as follows:

```
z<-rnorm(1000)
t<-abs(z)
c<-quantile(t,0.95)
```

Informally, this idea of using simultation to estimate critical values forms the basis of the Monte Carlo test procedure.

We will now construct a Monte Carlo hypothesis test of size α. Note that the test procedure must be constructed in such a way that $P(\text{reject } H_0 | H_0 \text{ true}) = \alpha$. We proceed as follows:

Monte Carlo Testing

1. Generate $n-1$ sample test statistics $T_1, \ldots, T_{n-1}$ according to the null hypothesis H_0
2. Define $m = n\alpha$ (with n sufficiently large for m to be an integer). If T_{obs} is one of the m largest values in the set $\{T_1, \ldots, T_{n-1}, T_{obs}\}$, then reject H_0.

By symmetry, if $\{T_1, \ldots, T_{n-1}, T_{obs}\}$ have all been generated according to the distribution specified by the null hypothesis, then the probability of T_{obs} being the ith largest value in this set is $\frac{1}{n}$. Consequently, the probability of T_{obs} being any one of the m largest values is $\frac{m}{n} = \alpha$. Hence the probability of rejecting the null hypothesis if H_0 is true is α, as required.

> Why do we make a point of generating $n-1$ test statistics rather than n? This is because in designing the test, we have specified that we wish the size to be α. The size of the test *will* be exactly equal to α, as long as $n\alpha$ is an integer. Had we generated n test statistics, we would reject H_0 if T_{obs} is one of the $(n+1)\alpha$ largest values, and so we have the slightly more awkward requirement that $n\alpha + \alpha$ is an integer, when choosing n.

4.2.1 Example 2: Testing for randomness in spatial patterns

Figure 1 shows the spatial location of 50 subjects in a study. Were the locations of these subjects generated randomly in the unit square, or is there evidence of clustering?

Firstly, we'll state the null hypothesis:

H_0: The spatial locations of each subject are randomly distributed over the unit square; both

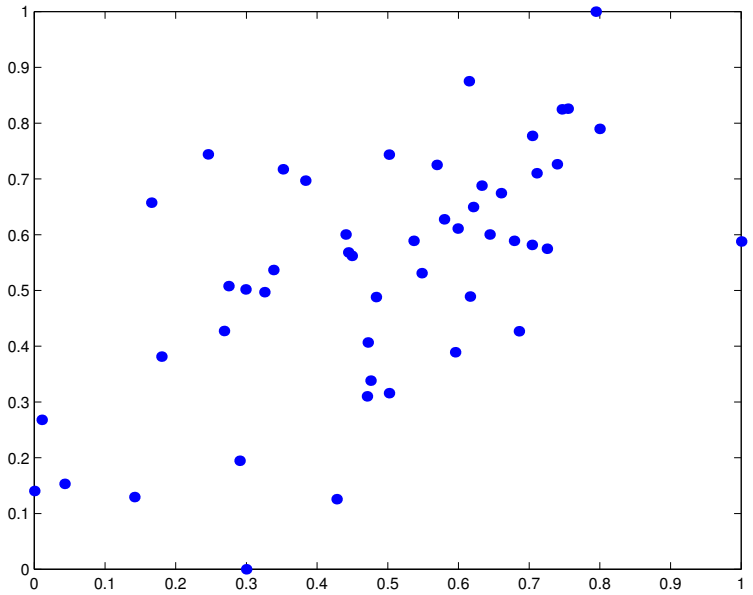

Figure 1: Observed spatial locations of 50 subjects

coordinates have $U[0,1]$ distributions.

Now we need a test statistic. There are various possibilities here, and the one we will use involves the *nearest-neighbour* of each subject. Let d_i denote the distance from subject i to the next closest subject, and define the test statistic T to be

$$T = \left(\sum_{i=1}^{50} d_i \right)^{-1} . \tag{39}$$

The idea here is that within a unit square, clustered patterns of 50 points should have smaller nearest-neighbour distances than patterns where points are scattered randomly across the entire square. Consequently, T is expected to increase as the degree of clustering increases. The value of T for the observed pattern is 0.307.

Under H_0, we don't know what the theoretical sampling distribution of T is. However, it is relatively straightforward to simulate values of T under H_0, and if we can do this, we can estimate the critical values (such as the 95th percentile) we need for the hypothesis test. A random value T_i is generated as follows:

1. Generate locations (x, y) of each subject by sampling x and y independently from $U[0,1]$. An example is plotted below:

2. For each subject, find the closest observation and measure the distance to it to obtain the nearest-neighbour distance for that observation

3. Take the reciprocal of the sum of the 50 nearest-neighbour distances to get T_i. For the pattern shown, the value of T_i is 0.235.

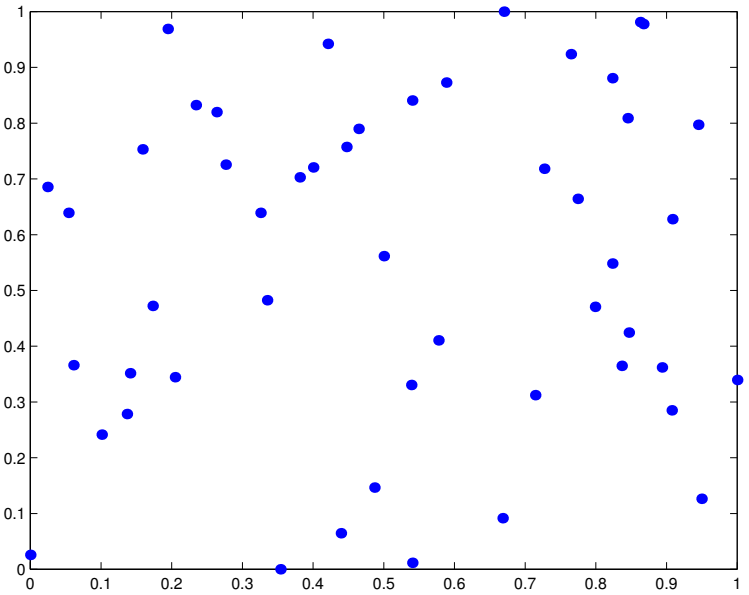

Figure 2: Randomly generated spatial locations of 50 subjects under H_0

Given a sample $T_1, \ldots, T_{n-1}$, we then rank $T_1, \ldots, T_{n-1}$ and the observed T_{obs} in order to give $T_{(1)}, \ldots, T_{(n)}$. For a test of size 5%, if T_{obs} is one of the $0.05 \times n$ largest values, then H_0 is rejected.

In this example, suppose we generate 99 tests statistics $T_1, \ldots, T_{99}$ under H_0, and find that 4 of them are greater than T, and 95 of them are less than T_{obs}. Since T_{obs} is the 5th largest value out of 100, H_0 is rejected at the 5% level.

4.2.2 Example 3: Chi-squared tests

Exam grades are to be compared between 16 boys and 19 girls in a single class. Data are given in the following table:

	A	B	C	D
boys	3	4	5	4
girls	8	8	3	0

The null hypothesis H_0 is that there is no difference between boys and girls in exam performance; a girl and boy chosen at random have the same probability of obtaining any particular grade. If we conduct a standard chi-squared test of independence, the observed test statistic is 7.907, with a p-value of 0.048. However, as a rule of thumb, to use the chi-squared test, the expected number of counts in each cell should be at least 5. In this example, 4 of the 8 expected number of counts are less than 5. Consequently, we may have doubts that the test-statistic

$$T = \sum \frac{(O_i - E_i)^2}{E_i} \tag{40}$$

has a χ^2 distribution with 3 degrees of freedom. We could merge cells (grades C and D), though this a similar result (the p-value is 0.041). Monte Carlo methods can be used to perform the hypothesis test without having to merge cells. As in the previous example, we generate sample values of the test statistic $T_1, \ldots, T_{n-1}$ under the null hypothesis. A random T_j is generated as follows:

1. Under H_0, probabilities of obtaining each grade are given by the column total over the grand total

grade	A	B	C	D
probability	$\frac{11}{35}$	$\frac{12}{35}$	$\frac{8}{35}$	$\frac{4}{35}$

2. We then generate a new set of results for boys and girls; the boys' results are sampled from the $multinomial(16, \frac{11}{35}, \frac{12}{35}, \frac{8}{35}, \frac{4}{35})$ distribution, and the girls' results are sampled from the $multinomial(19, \frac{11}{35}, \frac{12}{35}, \frac{8}{35}, \frac{4}{35})$ distribution. An example is shown below:

	A	B	C	D
boys	3	5	6	2
girls	4	5	7	3

3. Calculate $T_j = \sum \frac{(O_{i,j}-E_i)^2}{E_i}$, which for the example data is 5.323.

Then, as before, we rank $T_1, \ldots, T_{n-1}$ together with the observed value of the test statistic T_{obs}. In R we generate 99 test statistics $T_1, \ldots, T_{99}$ and find 75 to be less than the observed T_{obs}, and 24 to be greater. In this case the null hypothesis is not rejected at the 5% level.

We see that the two approaches have produced quite different results. Clearly, the Monte Carlo test is to be preferred, as we are working with the true distribution of the test statistic and not an approximation. (In Figure 3 we see that the χ^2_3 approximation is not very accurate). In general, when conducting hypothesis tests we do not have to be so reliant on distributional approximations, and we should always consider the option of working with exact distributions via simulation.

4.2.3 P-values

The subject of $p-$values in Monte Carlo tests can be slightly confusing! Two equivalent definitions of a $p-$value are as follows:

1. $P(T \geq T_{obs} | H_0 \text{ true})$

2. The smallest size α of the hypothesis test for which the null hypothesis would be rejected.

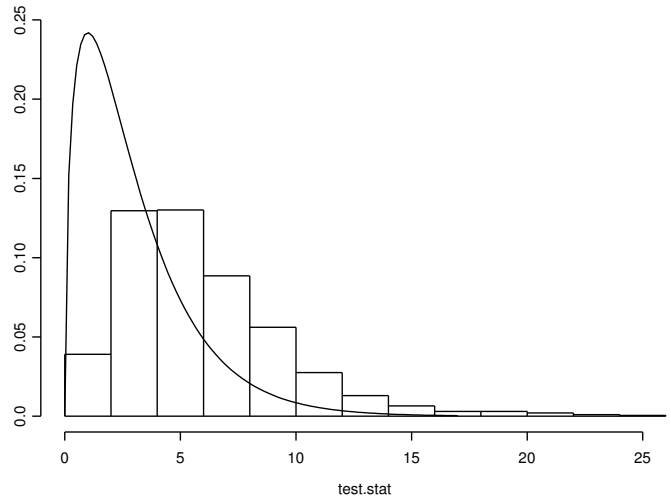

Figure 3: True distribution of T under H_0 (histogram) and χ^2_3 density function (solid line). The χ^2_3 distribution is not a good approximation of the true distribution in this case.

Using the first definition, the natural estimate of the $p-$value would be

$$\frac{1}{n-1}\sum_{i=1}^{n-1} I\{T_i \geq T_{obs}\}, \tag{41}$$

as this is an unbiased estimate of the probability given in the first definition. However, in Monte Carlo testing, the convention is to report the significance of the observed T according to the second definition. This results in a *biased* estimate of the true $p-$value. The observed level of significance is given by

$$\frac{1}{n}\left(\sum_{i=1}^{n-1} I\{T_i \geq T_{obs}\} + 1\right), \tag{42}$$

Now if we denote the true $p-$value by p^*, then the term $\sum_{i=1}^{n-1} I\{T_i \geq T_{obs}\}$ has the $Binomial(n-1, p^*)$ distribution, and so its expected value is $(n-1)p^*$. Consequently, although we are reporting the correct significance level of T_{obs}, the estimate of the $p-$value is biased; the expected value of (42) is

$$p^* + \frac{1-p^*}{n}, \tag{43}$$

though the bias will be small for large n.

4.2.4 How many test statistics should we generate?

Given α, and a sample size (for the sample $T_1, \ldots, T_{n-1}$), if $m = n\alpha$ is an integer, then we know that the size of the test is α. This could suggest making n as small as possible. However, we also know that due to the random variation in the sample $T_1, \ldots, T_{n-1}$, we may get a poor

estimate of the appropriate critical value if n is small. The motivation for choosing n is as follows:

For a test at the 5% level, given n we reject H_0 if the observed test statistic T_{obs} is one of the $0.05n$ largest values in the set $\{T_1, \ldots, T_{n-1}, T_{obs}\}$. If the true $p-$value for T_{obs} is given by p, then the probability that the Monte Carlo test rejects H_0 is given by

$$P(\text{reject}) = \sum_{i=0}^{0.05n-1} {}^{n-1}C_i \, p^i \, (1-p)^{n-1-i}. \tag{44}$$

If the Monte Carlo test behaves like a conventional test (with the true sampling distribution of T known), then for p-values smaller than 0.05 we would want a high rejection probability in (44), and for p-values larger than 0.05 we would want a low rejection probability. In Figure 4 we plot the rejection probability for various n and p.

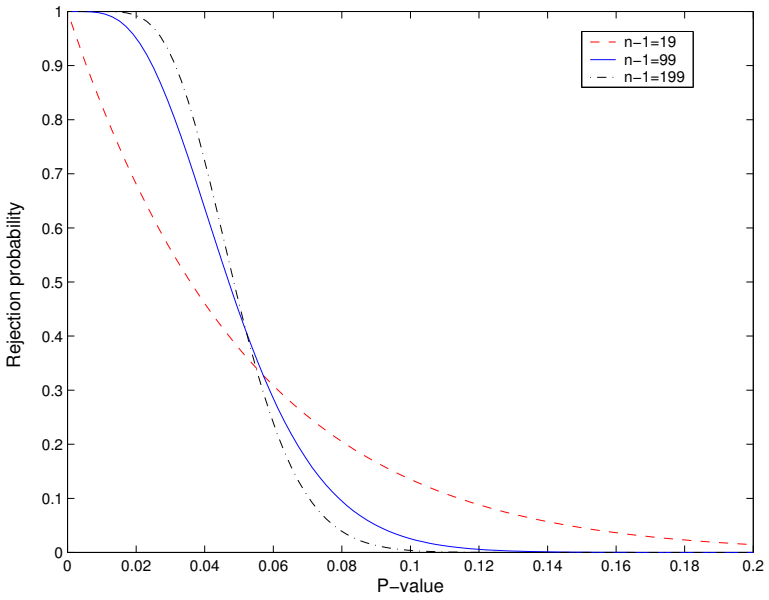

Figure 4: Rejection probability of Monte Carlo test

In conclusion, a sample size of 99 is considered to be acceptable, if the results are not interpreted too rigidly. Of course, this is only an issue if generating test statistics requires substantial computational effort. If it is trivial to generate sample test statistics, then a much larger sample size can be used.

4.2.5 Exercise

A random variable X_i is defined by

$$X_i = \max\{Z_{i,1}, \ldots, Z_{i,5}\} - \min\{Z_{i,1}, \ldots, Z_{i,5}\}, \tag{45}$$

where $Z_{i,1}, \ldots, Z_{i,5}$ are iid $N(\mu, \sigma^2)$ random variables, with both μ and σ^2 unknown. Given a sample of observations $X_1, \ldots, X_{10}$, describe a one-sided Monte Carlo test procedure to test

the hypothesis

$$H_0 : \sigma^2 = 1.,$$

where the alternative is of the form $H_1 : \sigma^2 < 1$. Hint: when choosing a suitable test statistic, consider what you would expect to happen to X_i as σ^2 gets smaller.

4.3 Randomisation tests

We will now consider a second technique for deriving the sampling distribution of the test statistic. No distributional assumptions about the data are required, and additionally, the sample of data does not need to have been drawn randomly from the population of interest.

The general scenario under consideration is that of an investigation into whether or not a particular treatment/covariate/factor has an effect on some response. We will illustrate the concept of the randomisation test with an example.

4.3.1 Example: Cholesterol data

A small study was conducted to investigate the effect of diet on cholesterol levels. Volunteers were randomly allocated to one of two diets, and cholesterol levels were recorded at the end of trial period. Data are given in the following table:

Diet A	233	291	312	250	246	197	268	224
Diet B	185	263	246	224	212	188	250	148

Table 1: Cholesterol data

The interest is in whether or not there is a significant difference between the mean cholesterol levels for the two groups. The null hypothesis is

$$H_0 : \text{ mean cholesterol levels with the two diets are equal .}$$

A standard analysis of this data could involve a two sample t-test, based on the statistic

$$T = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{s^2/8 + s^2/8}}, \tag{46}$$

for a two-sided test with

$$s^2 = \frac{\sum_j (X_{1,j} - \bar{X}_1)^2 + \sum_j (X_{2,j} - \bar{X}_2)^2}{8 + 8 - 2}. \tag{47}$$

Then under H_0, the test statistic T has a t distribution with 14 degrees of freedom. For this data, the observed test statistic T_{obs} is 2.0034, with a p-value of 0.0649 for a two-sided test.

In this instance, it may be desirable to analyse the data without assuming normality, particularly as the sample sizes are fairly small. We will now consider how to derive a distribution

of the test statistic T without reference to the t distribution:

Suppose the following three statements are all true:

1. The two diets have the same effect on cholesterol.
2. Subjects were allocated randomly to one of the two diets
3. The two group means were observed to be 'significantly' different.

Given the first two statements, how do we explain the third? How likely were we to observe significantly different group means?

If H_0 is true, then any difference in the two sample means would be solely due to how the 16 individuals were assigned to the two groups. So if H_0 is true, what is the probability of observing a sizeable difference between the two group means? It must be equal to the probability of assigning the individuals to the two groups in such a way that the imbalance occurs, as long as the individuals were assigned to the two groups at random in the actual study. This is the principle idea behind randomisation tests. The test is performed as follows:

Randomisation Test

1. Suppose the 16 individuals in the study have been labeled:

Diet A	233(1)	291(2)	312(3)	250(4)
	246(5)	197(6)	268(7)	224(8)
Diet B	185(9)	263(10)	246(11)	224(12)
	212(13)	188(14)	250(15)	148(16)

2. Randomly re-assign the sixteen individuals to the two groups, for example:

Diet A	250(15)	291(2)	185(9)	197(6)
	246(5)	250(4)	246(11)	188(14)
Diet B	312(3)	263(10)	268(7)	224(12)
	212(13)	224(8)	233(1)	148(16)

3. Re-calculate the test-statistic for this permuted data
4. Repeat steps 2 and 3 to obtain N sampled test-statistics, denoted by $T_1, \ldots, T_N$.
5. For a two-sided test, the estimated p-value of the observed test statistic T_{obs} is

$$\frac{1}{N}\sum_{i=1}^{N} I\{|T_i| \geq |T_{obs}|\}. \tag{48}$$

Using 10000 random permutations gave a p-value of 0.063, very similar to that achieved using the parametric test.

4.3.2 Equivalent test statistics

The significance level of T_{obs} was determined using (48). Now suppose we were to multiply both T_{obs} and all the T_is by some constant. Clearly, this would have no effect on the value of (48); the ordering would be preserved. Any alternative test statistic that leaves preserves the ordering is known as an *equivalent* test statistic. In the example, an equivalent test statistic would be

$$T = \bar{X}_1 - \bar{X}_2. \tag{49}$$

4.3.3 Why use randomisation tests?

There are arguments both for and against the use of randomisation tests, in favour of more conventional statistical tests. Some argue that they should always be used, as samples of data are never truly randomly drawn the population of interest; some members of the population are always going to be more accessible than others. On the other side, there is no theory to show that the result of a randomisation test can be generalised to the whole population; evidence against the null hypothesis is obtained for the observed sample only. Consequently, in either case, a 'non-statistical' judgment has to be made; that the sample can be treated as effectively random for a conventional test, or that the results can be generalised to the population for a randomisation test.

Two (perhaps more genuine) advantages of randomisation tests are that they can be used for *any* test statistic (i.e. in cases when it is not possible to analytically derive a distribution of the test statistic), and that it is certainly desirable not to have to assume a particular distribution for the data. Note that in the example, almost identical results were obtained using the two methods. In this case, the randomisation test could be seen as a means of supporting the results from the parametric test.

The requirement for the randomisation test to be valid is that the subjects are assigned randomly to each treatment. If random allocation is not explicitly part of the experimental procedure then there needs to be the belief that the actual allocation was as likely to occur as any other.

4.3.4 Exact randomisation tests

If it assumed that each possible assignment of individuals to each of the two treatment groups is equally likely, then the significance level of the observed test statistic could in principle be

determined exactly, by evaluating the test statistic systematically for *every* possible permutation of the data. Computationally, it is usually simpler to estimate the significance level based on a sample of random permutations. In addition, if the sample of random permutations is large enough, then the estimate should be sufficiently accurate. Randomisation tests in which all permutations are evaluated are known as *exact* randomisation tests, or *permutation* tests.

4.3.5 Example 2: outliers

The presence of an outlying observation in the data can cause problems for the conventional two sample t test. An outlier will increase the between group difference $\bar{X}_1 - \bar{X}_2$, but it will also inflate the within group variance. Consequently, the true significance of the test statistic may be *under*estimated.

In a randomisation test, you are comparing the *relative* size of the observed test statistic to its value under alternative random permutations, the outlier will not have the same effect. This is illustrated in some data from a study reported in Ezinga (1976):

treatment A	0.33	0.27	0.44	0.28	0.45	0.55	0.44	0.76	0.59	0.01
treatment B	0.28	0.80	3.72	1.16	1.00	0.63	1.14	0.33	0.26	0.63

The sample group means are $\bar{X}_A = 0.412$ and $\bar{X}_B = 0.995$, and the observed test statistic for a two sample t test is $T = 1.78$. For a two-tailed test this is significant at the 9.2% level, from statistical tables. Using a randomisation test, T is significant at the 2.6% level.

4.3.6 Example 3: Analysis of variance

Randomisation tests are applicable in many different contexts. Analysis of variance is another example. Below are responses measured on four treatment groups:

$$\begin{aligned}
\text{Group A} &: -0.10, -1.10, 0.74, -3.80 \\
\text{Group B} &: 0.94, -0.30, 0.67, 0.86, 1.19 \\
\text{Group C} &: -0.25, 0.84, 0.04, 0.25 \\
\text{Group D} &: 0.99, 0.08, 0.98, 0.75, 0.53
\end{aligned}$$

To test the null hypothesis

$$H_0 : \text{ All four group means are equal}$$

a conventional F test could be used: the ratio of the between-group mean square to the within-group mean square is compared with the $F_{3,14}$ distribution. The p-value for the observed F statistic is 0.08. Alternatively, a randomisation test could be applied as follows:

1. Randomly re-assign the observations to the four treatments, keeping the numbers in each treatment the same. e.g:

$$\begin{aligned} \text{Group A} &: 0.84, 0.53, 0.74, 0.94 \\ \text{Group B} &: -3.80, 0.98, 0.67, 0.04, 0.99 \\ \text{Group C} &: -0.30, -0.10, 0.86, 0.25 \\ \text{Group D} &: 1.19, 0.08, -0.25, 0.75, -1.10 \end{aligned}$$

2. Evaluate the test statistic

$$F = \frac{\{\sum_{i=1}^{4} n_i(\bar{x}_i - \bar{x})^2\}/3}{\{\sum_{i=1}^{4}\sum_{j=1}^{n_i}(x_{i,j} - \bar{x}_i)^2\}/14}$$

   for the permuted data

3. Repeat steps 1 and 2 N times to obtain sampled test statistics $F_1, \ldots, F_N$.

4. Estimate the significance level of F_{obs} by

$$\frac{1}{N}\sum_{i=1}^{N} I\{F_i \geq F_{obs}\} \tag{50}$$

Based on a sample size $N = 10000$, the estimated p-value for F_{obs} was 0.03, suggesting slightly stronger evidence against the null hypothesis.

4.3.7 Confidence intervals

The randomisation test procedure can additionally be used to construct confidence intervals for treatment differences. As an intermediate step, we will again consider the scenario of two treatments A and B, with unknown population means μ_A and μ_B. Data are as follows:

treatment A	130	119	119	168	130
treatment B	154	115	169	137	186

Now suppose that, perhaps following some previous study, there is an expectation that there is going to be a difference between the two population means of some specified size. Consider for example the null hypothesis:

$$H_0 : \mu_B - \mu_A = 20.$$

How would we test this using a randomisation test? Simply permuting the data and evaluating the difference between the sample means $\bar{X}_A - \bar{X}_B$ each time will not work here, as that would give the distribution of $\bar{X}_A - \bar{X}_B$ under the null hypothesis of equal population group means. However, suppose we were to add 20 to each observation in group A. Under H_0, what is the

treatment A+20	150	139	139	188	150
treatment B	154	115	169	137	186

expectation of 20+ a response in group A? If H_0 is true, then this expectation is $20+\mu_A = \mu_B$. So, if we first add 20 to each response in group A: then under the null hypothesis, both groups now have the same population mean. Any additional discrepancy in the data is now due to how the individuals were assigned to the two treatment groups, and so a randomisation test can be performed in exactly the same way as usual.

Now we will return to the question of constructing a confidence interval for the difference in population means, $\mu_B - \mu_A$. Firstly, recall that for a parameter θ, a 95% confidence interval for θ, given by (l, u) can be interpreted as follows: $a \in (l, u)$ if and only if the hypothesis

$$H_0 : \theta = a$$

is **not** rejected at the 5% level. So, if we denote the 95% confidence interval for $\mu_B - \mu_A$ by (l, u), then l will be the smallest value of k, and u will be the largest value of k such that the hypothesis

$$H_0 : \mu_B - \mu_A = k$$

is not rejected.

Unfortunately, there is no quick way to find l and u. However, as a starting point, we can try the lower and upper limits of a conventional 95% confidence interval based on the t_8 distribution: in this case $(-16.4, 54.3)$. We can then experiment with different values of k, and choose a final interval using interpolation:

k	-18	-14	52	56
$p-$value	0.03	0.08	0.07	0.03

Using linear interpolation then gives $(-15.6, 54)$ as an approximate 95% confidence interval, very similar to the parametric interval.

4.3.8 One-sample randomisation tests

Randomisation tests can be used for one-sample problems, but under stricter assumptions. This is demonstrated with the following example:

Given observations $\{10.61, 9.46, 7.02, 11.68, 9.58, 11.96, 11.28, 7.63, 6.42, 8.85\}$ drawn from some population with mean μ, test the null hypothesis

$$H_0 : \mu = 10.$$

It is not immediately obvious what can be permuted here. However, supposing the two following assumptions hold:

1. Each observation has been sampled randomly from its population.
2. The population distribution is symmetric about its mean.

Now suppose H_0 is true, and consider randomly sampling a value X from the population, and then evaluating $Y = X - 10$. If the population distribution is symmetric about 10, then Y must have an equal probability of being positive or negative.

In this example, subtracting 10 from each observation and taking the resulting mean gives a sample mean of -0.551. We will use the absolute value of this sample mean as the test statistic, so $T_{obs} = 0.551$ (for a two-sided test). If H_0 is true, and both assumptions hold, then the observed sample mean could simply be to due to an imbalance of positive and negative Y values. This can be tested as follows:

One-sample Randomisation Test (Fisher's Randomisation test)

1. Subtract hypothesised population mean from each observation:

$$\{0.61, -0.54, -2.98, 1.68, -0.42, 1.96, 1.28, -2.37, -3.58, -1.15\}$$

2. Calculate observed test statistic: $T_{obs} = 0.551$
3. With probability 0.5 for **each** observation, change the sign of $X - \mu$. E.g:

$$\{-0.61, -0.54, -2.98, -1.68, 0.42, 1.96, -1.28, -2.37, -3.58, 1.15\}$$

4. Calculate the test statistic for the new simulated observations: $T = 0.951$.
5. Repeat steps 3 and 4 to obtain sampled test statistics $T_1, \ldots, T_N$
6. Estimate the significance of T_{obs} by

$$\frac{1}{N} \sum_{i=1}^{N} I\{T_i \geq T_{obs}\}. \tag{51}$$

With $N = 10000$, the estimated significance of T_{obs} is 0.4021. Using a conventional $t-$test, the significance of T_{obs} is 0.3982, so there is close agreement between the two methods in this example.

4.3.9 Exercises

1. In the example in section 4.3.1, how may test statistics would you have to compute if you were conducting an exact randomisation test?

2. In section 4.3.6, the conventional ratio of between groups mean square to within groups mean square was used as the test statistic. Give a simpler, equivalent statistic.

3. Given observations $\{x_i, y_i\}$ for $i = 1, \ldots, n$, the following regression model is proposed:

$$y_i = \alpha + \beta x_i + \varepsilon_i, \tag{52}$$

where the ε_is are iid noise terms. How could a randomisation test be used to test the hypothesis $H_0 : \beta = 0$?

4.4 Bootstrap Methods

We will now consider a third computational technique known as **bootstrapping**. Bootstrapping can be also be used to tackle problems in inference without having to make distributional assumptions about the data.

4.4.1 Example: heart attack study

A controlled, randomized, double-blind study was carried out to investigate whether or not aspirin reduces the risk of heart attacks in healthy middle-aged men. Data from the study is given in table 2.

	heart attacks (fatal plus non-fatal)	subjects
aspirin	104	11037
placebo	189	11034

Table 2: heart attack study data (from Efron and Tibshirani, 1993)

Define θ_h to be the true ratio of proportions of heart attacks in those with aspirin to those with a placebo, the relative risk. From the data, the estimate of θ_h suggests that aspirin lowers the risk of a heart attack:

$$\hat{\theta}_h = \frac{104/11037}{189/11034} = 0.55. \tag{53}$$

Furthermore, it is possible to determine a 95% confidence interval for θ from theory. We assume that the log relative risk is normally distributed, with the standard error given by

$$\left\{ \frac{11037 - 104}{104 \times 11037} + \frac{11034 - 189}{189 \times 11034} \right\}^{0.5} = 0.121. \tag{54}$$

The 95% confidence interval is given by $\{\exp(\log 0.55 - 1.96 \times 0.121), \exp(\log 0.55 + 1.96 \times 0.121)\}$, i.e. (0.43,0.7). This further supports the conclusion that aspirin lowers the risk of a heart attack.

In addition to monitoring heart attacks, the study recorded the number of subjects who suffered a stroke. These data are given in table 3.

	strokes	subjects
aspirin	119	11037
placebo	98	11034

Table 3: stroke study data

Estimating the true ratio θ_s of the two proportions suggests that aspirin may *increase* the risk of a stroke:

$$\hat{\theta}_s = \frac{119/11037}{98/11034} = 1.21 \tag{55}$$

However, the 95% confidence interval for θ_s, (0.93,1.59) weakens this conclusion; we would not reject the null hypothesis $H_0 : \theta_s = 1$ at the 5% level.

The technique of bootstrapping enables us to derive these confidence intervals without assuming that the log relative risks are normally distributed. When computing a $(1-\alpha)\%$ confidence interval for a parameter θ, one interpretation of the process that we go through is as follows:

1. First assume that the true value of θ is given by our observed estimator $\hat{\theta}$
2. Now consider a future experiment which will produce a new value $\hat{\theta}^*$ of our estimator of θ
3. Derive an interval such that the probability that $\hat{\theta}^*$ will take a value in this interval is $(1-\alpha)$.

In bootstrapping, we simulate new experimental data, using the observed data to estimate the sampling distribution. In the example, this is done as follows:

1. Estimate the probability p_1 of a patient with aspirin having a stroke:
$$\hat{p}_1 = \frac{119}{11037} = 0.0105$$
2. Estimate the probability p_2 of a patient with a placebo having a stroke:
$$\hat{p}_1 = \frac{98}{11034} = 0.0087$$
3. Simulate data for a new experiment: sample r_1 from $Binomial(11037, 0.0105)$ and r_2 from $Binomial(11034, 0.0087)$. The new data is known as a *bootstrap* sample.

4. Obtain a new estimate of the ratio:

$$\hat{\theta}_s^* = \frac{r_1/11037}{r_2/11034}$$

Steps 2 and 3 are then repeated a large number of times, to obtain a sample $\{\hat{\theta}_s^*(1), \ldots, \hat{\theta}_s^*(N)\}$. We can then use the 2.5th and 97.5th percentiles of this sample as a 95% confidence interval for θ_s. With $N = 10000$, performing this procedure in R gave a 95% interval of (0.44,0.71) for θ_h and a 95% interval of (0.93,1.58) for θ_s. In both cases these are very close to the theoretically derived intervals.

4.4.2 The bootstrap estimate of standard error

We will now generalise the bootstrap procedure, starting with a discussion of how to estimate standard errors. We wish to estimate some parameter θ of a distribution given a sample of data $\mathbf{X} = \{X_1, \ldots, X_n\}$ from that distribution, and will use an estimator

$$\hat{\theta} = s(\mathbf{X}), \tag{56}$$

where $s(\mathbf{X})$ is some function of the data $\mathbf{X}$. The standard error of $\hat{\theta}$, denoted by $s.e.(\hat{\theta})$ is simply defined as

$$s.e.(\hat{\theta}) = \sqrt{Var\{s(\mathbf{X})\}} \tag{57}$$

Suppose we know the distribution F of the data X. How could we estimate (57)? By generating samples of data $\mathbf{X}_1, \ldots, \mathbf{X}_N$, and then using the estimate

$$\widehat{s.e.}(\hat{\theta}) = \left[\frac{1}{N-1}\sum_{i=1}^{N}\{s(\mathbf{X}_i) - \bar{s}(\mathbf{X})\}^2\right]^{0.5}, \tag{58}$$

with $\bar{s}(\mathbf{X}) = \sum_{i=1}^{N} s(\mathbf{X}_i)/N$. Of course, F is unknown, and in this section we are not going to make any parametric assumptions about F. However, if we had a good *approximation* of F, which we will denote by $\hat{F}$, we could then sample data from $\hat{F}$ and proceed as before, (hopefully) without too much loss of accuracy.

The idea at the heart of bootstrapping it to approximate F given data $\{X_1, \ldots, X_n\}$ by its **empirical distribution function**. This simply estimates a probability $P(X \leq x)$ by the unbiased estimate

$$\frac{1}{n}\sum_{i=1}^{n} I\{X_i \leq x\} \tag{59}$$

Suppose we have data $\{-0.4326, -1.6656, 0.1253, 0.2877, -1.1465\}$ (as it happens, randomly generated from a $N(0,1)$ distribution). The empirical distribution function is plotted in Figure 5, together with the true distribution function of a standard normal random variable for comparison. As the number of data points increases, the closer the empirical c.d.f becomes to

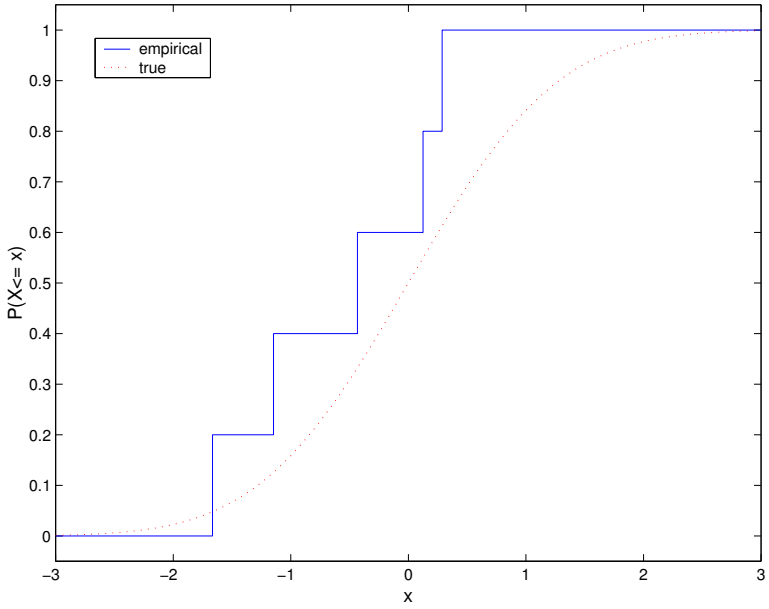

Figure 5: Empirical distribution function based on 5 data points

the true c.d.f. This can be seen in Figure 6, where the empirical c.d.f. is plotted based on a sample of 100 observations.

So in general, given data $\{X_1, \ldots, X_n\}$, we will estimate the true distribution of X by the empirical c.d.f., and denote this estimated distribution function by $\hat{F}$. Assuming this distribution $\hat{F}$ for X, the implication is that X is a discrete random variable, taking any of the n values in the set $\{X_1, \ldots, X_n\}$ with probability $1/n$, since

$$\begin{aligned} P(X = x) &= \frac{1}{n}\left[\sum_{i=1}^{n} I\{X_i \leq x\} - \sum_{i=1}^{n} I\{X_i < x\}\right] \\ &= \begin{cases} \frac{1}{n} & x \in \{X_1, \ldots, X_n\} \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

To generate N values from $\hat{F}$ we just sample N values from the set $\{X_1, \ldots, X_n\}$ with replacement. We can now describe the bootstrap algorithm for estimating standard error:

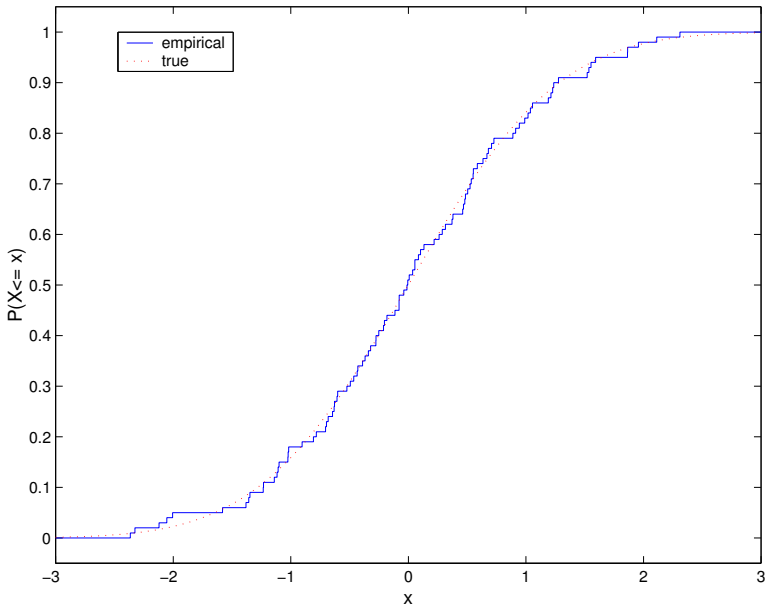

Figure 6: Empirical distribution function based on 100 data points

The bootstrap estimate of standard error

Given data $\mathbf{X} = \{X_1, \ldots, X_n\}$ the aim is to estimate the standard error of some estimator $\hat{\theta} = s(\mathbf{X})$ of an unknown parameter θ.

1. Generate new data $\mathbf{X}^*$ by sampling n values from $\{X_1, \ldots, X_n\}$ with replacement.
2. Evaluate the estimator given the bootstrap sample $\mathbf{X}^*$:
$$\hat{\theta}^* = s(\mathbf{X}^*)$$
3. Repeat steps 1 and 2 N times to obtain $\hat{\theta}^*(1), \ldots, \hat{\theta}^*(N)$.
4. Estimate the standard error of $\hat{\theta}$ by
$$\left[\frac{1}{N-1}\sum_{i=1}^{N}\left\{\hat{\theta}^*(i) - \bar{\hat{\theta}}^*\right\}^2\right]^{0.5},$$
with $\bar{\hat{\theta}}^* = \frac{1}{N}\sum_{i=1}^{N}\hat{\theta}^*(i)$

4.4.3 Example: Law school data

A sample of 15 law schools was taken, and two measurements were made for each school:

x_i : LSAT, average score for the class on a national law test

y_i : GPA, average undergraduate grade-point average for the class

The data is plotted in Figure 7 We are interested in the correlation coefficient between these

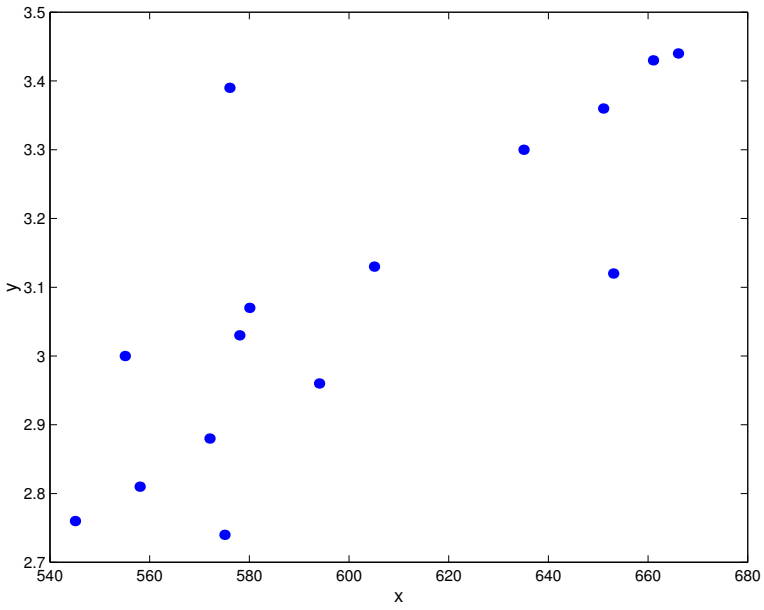

Figure 7: Law school data (from Efron and Tibshirani, 1993)

two quantities, which we will denote by θ. The sample correlation coefficient $\hat{\theta}$ for this data is 0.776. To consider how good the estimate $\hat{\theta}$ might be, we can now use bootstrapping to estimate the standard error of $\hat{\theta}$:

1. Sample 15 data points with replacement from the observed data $\mathbf{z} = \{(x_1, y_1), \ldots, (x_{15}, y_{15})\}$ to obtain new data $\mathbf{z}^*$.
2. Evaluate the sample correlation coefficient $\hat{\theta}^*$ for the newly sampled data $\mathbf{z}^*$.
3. Repeat steps 1 and 2 to obtain $\hat{\theta}^*(1), \ldots, \hat{\theta}^*(N)$
4. Estimate the standard error of the sample correlation coefficient by the sample standard deviation of $\hat{\theta}^*(1), \ldots, \hat{\theta}^*(N)$.

With $N = 1000$, the estimated standard error of $\hat{\theta}$ is 0.137. It can be worthwhile to plot a histogram of the sampled estimators $\hat{\theta}^*(1), \ldots, \hat{\theta}^*(N)$, as this will give more information about the distribution of $\hat{\theta}$. A histogram is plotted in Figure 8.

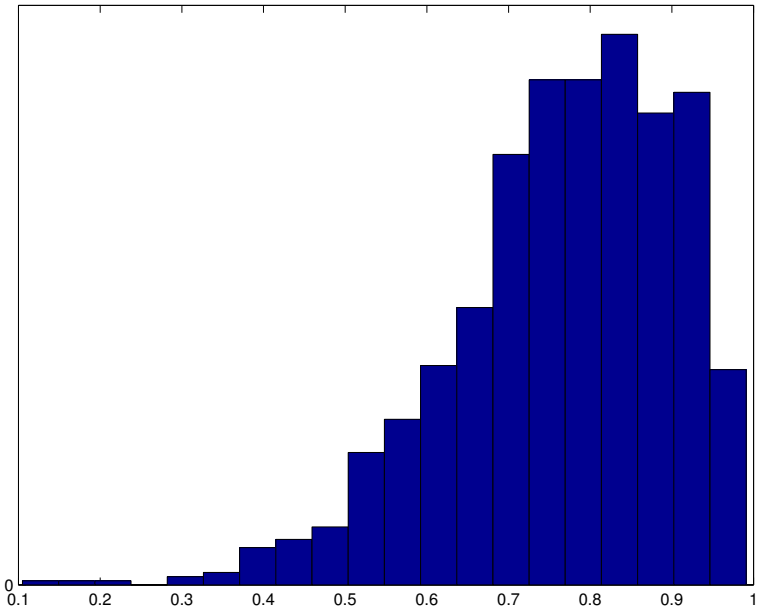

Figure 8: Histogram of sampled values of $\hat{\theta}$.

4.4.4 The parametric bootstrap

In parametric bootstrapping, we carry out exactly the same procedure as before, with one exception. Instead of approximating the distribution F by the empirical distribution function, we use a parametric approximation, where the parameters are estimated by the data.

4.4.5 Example: Law school data re-visited

Each observation $\{x_i, y_i\}$ in the data consists of two means (average results for a single class). It may then be reasonable to assume that the distribution of $\{x_i, y_i\}$ is bivariate normal:

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} \sim N(\mathbf{m}, V) \tag{60}$$

We can estimate the parameters $\mathbf{m}$ and V of this bivariate normal distribution from the data:

$$\hat{\mathbf{m}} = \begin{pmatrix} \frac{1}{15}\sum_{i=1}^{15} x_i \\ \frac{1}{15}\sum_{i=1}^{15} y_i \end{pmatrix} \tag{61}$$

$$\hat{V} = \frac{1}{14}\begin{pmatrix} \sum_{i=1}^{15}(x_i-\bar{x})^2 & \sum_{i=1}^{15}(x_i-\bar{x})(y_i-\bar{y}) \\ \sum_{i=1}^{15}(x_i-\bar{x})(y_i-\bar{y}) & \sum_{i=1}^{15}(y_i-\bar{y})^2 \end{pmatrix} \tag{62}$$

We then proceed as before:

1. Sample 15 data points from $N(\hat{\mathbf{m}}, \hat{V})$ to obtain new data $\mathbf{z}^*$.
2. Evaluate the sample correlation coefficient $\hat{\theta}^*$ for the newly sampled data $\mathbf{z}^*$.
3. Repeat steps 1 and 2 to obtain $\hat{\theta}^*(1), \ldots, \hat{\theta}^*(N)$

4. Estimate the standard error of the sample correlation coefficient by the sample standard deviation of $\hat{\theta}^*(1), \ldots, \hat{\theta}^*(N)$.

With $N = 1000$ the estimated standard error of $\hat{\theta}$ for the parametric bootstrap is 0.116, slightly different from the non-parametric estimate.

4.4.6 Confidence Intervals

In the literature, two methods for deriving confidence intervals using bootstrapping are presented, **bootstrap confidence intervals** and **percentile confidence intervals**.

- Bootstrap confidence intervals.
  Given an estimate of the standard error of $\hat{\theta}$, if we assume that the distribution of $\hat{\theta}$ is approximately normal, then an approximate 95% confidence interval is given by

$$\hat{\theta} \pm 2\widehat{s.e.}(\hat{\theta}).$$

- Percentile confidence intervals.
  For a 95% confidence interval, we need to find the two values l and u, with

$$P(\hat{\theta} < u) = 0.975, \tag{63}$$
$$P(l < \hat{\theta}) = 0.025, \tag{64}$$

  i.e., we need to identify the 2.5th and 97.5th percentiles from the distribution of $\hat{\theta}$. If we have a good approximation $\hat{F}$ to the true distribution F, then we should also have a good approximation of the distribution of $\hat{\theta}$. Hence to obtain the confidence interval, we just report the 2.5th and 97.5th percentiles of the sample

$$\{\hat{\theta}^*(1), \ldots, \hat{\theta}^*(N)\}.$$

  This method does not require a distribution assumption about $\hat{\theta}$, though a larger value of N may be required, to get accurate estimates of the 2.5th and 97.5th percentiles from $\hat{F}$. (The bootstrap interval will not usually need such a large N, as the standard error can be estimated reasonably accurately with a smaller sample. However, with a fast computer, the size of N may not be such an issue.)

In the law school data example, clearly the percentile confidence interval approach is to be preferred. From Figure 8, it we can see that the normality assumption is not valid.

4.4.7 Exercise

Given data $\{X_1, \ldots, X_{30}\}$, suppose we wish to construct a confidence interval for $\mu = E(X)$. Using the normal approximation, a 95% confidence interval would be

$$\bar{X} \pm 2\frac{\hat{\sigma}}{\sqrt{30}},$$

with $\bar{X}$ the sample mean and $\hat{\sigma}^2$ the sample variance. Now consider using bootstrapping to obtain a 95% confidence interval for μ. If X^* is a single randomly chosen value from the set $\{X_1, \ldots, X_{30}\}$, obtain expressions for $E(X^*)$ and $Var(X^*)$ in terms of $\bar{X}$ and $\hat{\sigma}^2$. If $\bar{X}^*$ is a bootstrapped sample mean, derive the approximate distribution of $\bar{X}^*$ based on the central limit theorem, and hence give expressions for the 95% bootstrap and percentile confidence intervals.

4.4.8 Hypothesis testing with the bootstrap

In this section we will see how bootstrapping can be used to carry out nonparametric one-sample and two-sample hypothesis/significance tests. The methodology is explained with an example:

4.4.9 Example: mice survival times

In an experiment, 7 out of 16 mice were randomly chosen to receive a new treatment, and the remaining 9 were assigned to a control group. Survival times are given in table 4

Treatment	94	197	16	38	99	141	23		
Control	52	104	146	10	50	31	40	27	46

Table 4: Mice survival times

The interest is in whether or not there is a difference between the two group means. Denote the 7 treatment observations by $\mathbf{x} = \{x_1, \ldots, x_7\}$, and the 9 control observations by $\mathbf{y} = \{y_1, \ldots, y_9\}$. We can of course perform a two-sample t test, assuming normally distributed responses and equal variances in the two groups. If we define μ_X to be the population treatment mean, and μ_Y to be the population control mean, then for a one-sided test of the hypothesis

$$H_0 : \mu_X = \mu_Y,$$

the observed $p-$value is 0.1405.

Bootstrapping can be used to perform this hypothesis test without the assumptions of normality. If we denote F_X to be the distribution of a treatment of survival time, and F_Y to be the distribution of a control survival time, we can write the hypothesis of no treatment effect as

$$H_0 : F_X = F_Y = F,$$

where F is the single common distribution of all the responses. We now estimate F by $\hat{F}$, the empirical cdf constructed from all 16 observations. The bootstrap significance test is performed as follows:

Bootstrap two-sample significance test

1. Sample 16 values with replacement from $\{x_1, \ldots, x_7, y_1, \ldots, y_9\}$.
2. Set $\{x_1^*, \ldots, x_7^*\}$ to be the first 7 sampled values, and $\{y_1^*, \ldots, y_9^*\}$ to be the remaining 9 sampled values.
3. Calculate the bootstrap test statistic
$$T^* = \frac{\bar{x}^* - \bar{y}^*}{\hat{\sigma^*}\sqrt{1/7 + 1/9}}$$
for the sampled data.
4. Repeat steps 1 to 3 N times to obtain $T^*(1), \ldots, T^*(N)$.
5. Estimate the significance of the observed T_{obs} by
$$\frac{1}{N}\sum_{i=1}^{N} I\{T^*(i) \geq T_{obs}\} \tag{65}$$

With $N = 10000$, the estimated significance is 0.145, very similar to the $p-$value from the two-sample t-test.

4.4.10 One sample hypothesis tests

Continuing with the mice example, suppose there is the belief from previous studies that for the treatment group, $\mu_X = 100$. We could test this with a one sample t test. A two-sided t-test of the hypothesis

$$H_0 : \mu_X = 100$$

gives a $p-$value of 0.6212 for the observed test statistic

$$T_{obs} = \frac{86.9 - 100}{\hat{\sigma}/\sqrt{7}}, \tag{66}$$

with $\hat{\sigma}$ the sample standard deviation, equal to 66.8.

So far, when performing bootstrap procedures, we have approximated the distribution F by the empirical cdf. The difficulty here is that if under the empirical cdf, the expectation of any single observation is the sample mean 86.9, and not the assumed mean of 100 under

the null hypothesis. If we are going to assume H_0 is true, we can't then simply generate new data under the empirical cdf, because this data wont have the correct expectation under H_0. Consequently, we first transform the data so that the sample mean does equal 100. For each of the 7 observations we set

$$x_i' = x_i - 86.9 + 100. \tag{67}$$

The assumption here is that the distribution F will have the same *shape* for any value of the true mean μ_X. (Strictly speaking this will not hold, as X has to be positive. A log-transformation may be preferable, but we will ignore this issue for this example). Then, rather than assuming a particular shape for the distribution, e.g., the bell-shaped curve of the normal distribution, we estimate the shape based on the empirical cdf. Finally, we suppose that this approximated shape is reasonable for any μ_X, even if μ_X doesn't agree with the sample mean.

The one-sample significance test is then performed as follows:

Bootstrap one-sample significance test

1. Transform the data such that the sample mean equals the mean under H_0:

$$x_i' = x_i - \bar{x} + \mu_X$$

2. Sample 7 values with replacement from {107.1,210.1,29.1,51.1,112.1,154.1,36.1} to obtain new data $\mathbf{x}'^*$

3. Re-evaluate the test statistic for the new data:

$$T^* = \frac{\bar{x'}^* - 100}{\hat{\sigma}'^*/\sqrt{7}},$$

where $\hat{\sigma}'^*$ is the sample standard deviation of the new data

4. Repeat steps 2 and 3 N times to obtain $T^*(1), \ldots, T^*(N)$.

5. Estimate the significance of T_{obs} by

$$\frac{1}{N}\sum_{i=1}^{N} I\{T^*(i) \geq T_{obs}\}$$

With $N = 10000$, the estimate p-value was 0.69, so there is some discrepancy with the t-test, although as both p-values are large this would not matter.

4.4.11 An example of bootstrap failure

We now consider an example where the bootstrap gives poor results.

We have observations $X_1, \ldots, X_{50}$ from a uniform distribution on $[0, \theta]$, where θ is unknown. The maximum likelihood estimate $\hat{\theta}$ for θ is simply the largest observed value. We will now try an experiment with the true $\theta = 1$: we sample 50 random $U[0, 1]$ and observe the largest as 0.98. We can use bootstrapping to sample from the distribution of $\hat{\theta}$ by re-sampling groups of 50 observations from $X_1, \ldots, X_{50}$ and setting $\hat{\theta}^*$ to be the largest in each case. A histogram based on 10,000 sampled values of $\hat{\theta}^*$ is shown in Figure 9, plot (a). We can also sample from the true distribution of $\hat{\theta}$, by repeatedly sampling groups of 50 observations from $U[0, 1]$. A histogram based on 10,000 sampled values of $\hat{\theta}$ is plotted in figure 9, plot(b). Note the poor agreement between the two histograms. This is because the empirical distribution function is a poor estimate of the true cdf of X in the tails. However, in this example, it is the tails of the distribution that we are interested in, as we want the distribution of the largest value of X in a sample of 50 points. To improve the estimate of the distribution, we would either have to consider some parametric fit to the data, or possibly smoothing out the empirical cdf.

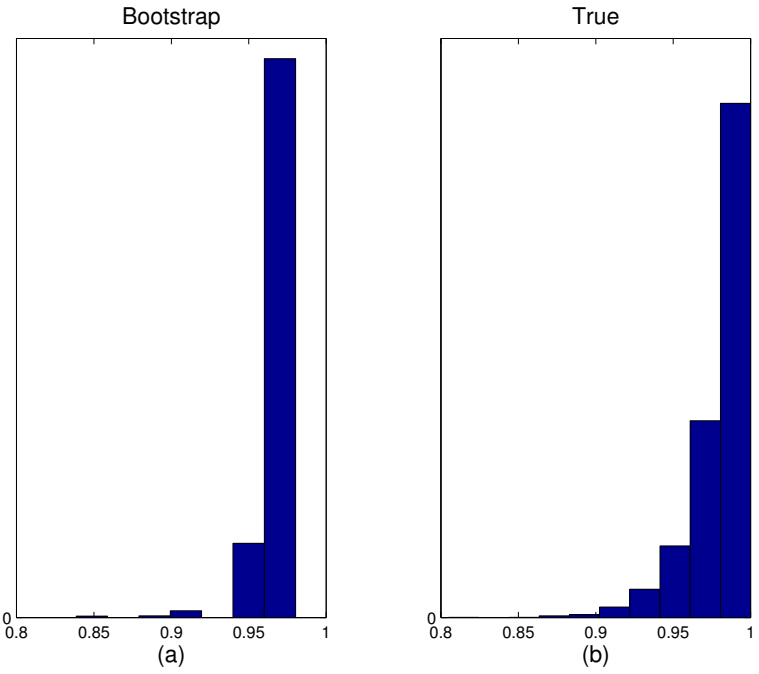

Figure 9: Estimated and true distributions of $\hat{\theta}$

4.4.12 Exercise

In the one-sample bootstrap significance test, we first transformed the data

$$x'_i = x_i - \bar{x} + \mu_X$$

before obtaining bootstrap samples of the test statistic

$$T^* = \frac{\bar{x'}^* - 100}{\hat{\sigma'}^*/\sqrt{7}}.$$

Prove that equivalently, we could re-sample the original, untransformed data, evaluating for each re-sampled data set the test statistic

$$T^* = \frac{\bar{x}^* - \bar{x}}{\hat{\sigma}^*/\sqrt{7}},$$

4.5 Summary

We have now studied three computational techniques for frequentist inference, each with its own advantages and disadvantages. A brief summary is given below:

1. **Monte Carlo tests**
    - Will work with any test statistic and hypothesis, but requires specification of the distribution of the data under the hypothesis in question.
    - Only procedure out of the three that produces 'completely new' data.
2. **Randomisation tests**
    - Can generally only handle tests of no treatment effect between different treatment groups. One sample tests can be performed, but under stricter assumptions.
    - No distribution is required/assumed for the data, only that allocation of subjects to treatment groups is random.
3. **Bootstrapping**
    - Arguably the most widely applicable method of the three.
    - Dependent on the empirical cdf being a good approximation to the true distribution.
    - Accuracy ultimately depends on size of **original** sample.

5 Prediction errors and cross-validation

We will now look at a useful computational tool for assessing the performance of a model in terms of its *predictive* ability. This is generally in the context of regression (predicting a continuous response y given some variable x) or classification (predicting a class membership C given some variable x). Prediction error can be assessed within the framework of bootstrapping, although here we will only consider an alternative method, known as **cross-validation**.

5.1 Cross-validation in regression

Given data $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, then (assuming n distinct x values), we can always fit this data perfectly with a polynomial of degree $n - 1$:

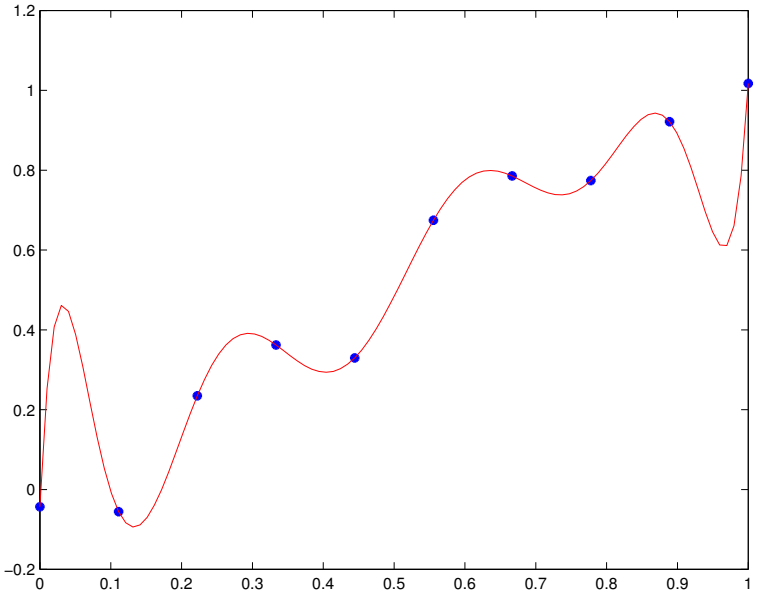

Figure 10: 9th degree polynomial fitted to regression data

Of course, we know in general that fitting high order polynomials to regression data is not a sensible thing to do, but how can we demonstrate this? With more data, we could check the predictions from this model against the new observations. However, it's not always going to be possible to obtain new data just to check the performance of a model.

The idea of cross-validation is to divide the data into two parts, a *training set* to fit the model to, and a *test set* to test the predictive performance. The algorithm is fairly simple:

Cross-validation

For $i = 1, \ldots, n$:

1. Fit the regression model to the reduced data set (or training set) $\{(x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, , (x_n, y_n)\}$
2. Obtain from the fitted model the predicted value $\hat{y}_i$ at x_i.
3. Compute the squared error $\varepsilon_i = (\hat{y}_i - y_i)^2$

An average prediction can then be reported as $\frac{1}{n}\sum_{i=1}^{n} \varepsilon_i$

The term $\frac{1}{n}\sum_{i=1}^{n} \varepsilon_i$ could be described as an expected prediction error for a future obser-

vation, but this would assume that the x values are also random. Additionally, it is not the expected prediction error of the actual model, though it should be a close if n is sufficiently large (so that the model fitted to $n-1$ points is very similar to the model fitted to all n points). A variant of the approach is to remove subsets of size k from the data each time, so that the training set has $n-k$ observations and the test set has k observations. This is known as $k-$fold cross validation.

We can now demonstrate the poor predictive results that can follow from fitting high degree polynomials to regression data. In the cross-validation procedure, we remove a single point from the data set in turn, and fit an 8th degree polynomial to the nine remaining observations. The predicted 10th value given x is then compared to the actual 10th value. It can be very instructive to plot the predicted values $\hat{y}$ against true values y. This is shown in Figure 11. For comparison, we also show prediction errors that result from fitting a linear model in x to each training set. It is clear from the diagram that the linear regression model is much better

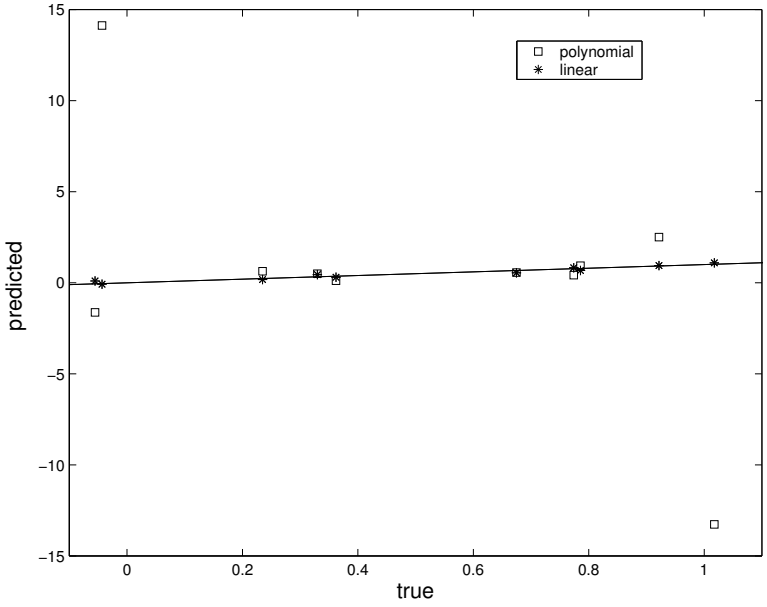

Figure 11: Cross-validation for polynomial and linear models

in terms of its predictive performance.

In a classification setting, we are predicting which class out of a set $\{C_1, \ldots, C_m\}$ a subject belongs to given a covariate observation x. Cross-validation can be applied in exactly the same way, except that now we would be counting the number of miss-classifications rather than the size of the prediction errors.

6 Programming exercises

In this section we will primarily be working through the examples in the lecture notes, using R to perform the computations required. First get the R workspace `MAS472.RData` from MOLE,

and load it into R. This contains data for questions 3 and 6, and example solutions for question 1.

1. Section 2.2 example problems.
   Write functions in R to obtain approximate solutions to problems 1,2 and 3. If you don't know where to start, a full solution to problem 1 is given at the end of this section. You will find techniques used in the solution helpful for the other problems in this section.

2. Section 4.2.5
   Use a Monte Carlo test procedure to test the hypothesis $H_0 : \sigma^2 = 1$, given data $\{1.77, 2.47, 2.16, 0.44, 0.96, 1.25, 2.10, 1.46, 2.92, 1.55\}$

3. Section 4.2.1 testing for randomness in spatial patterns.
   Locations of 50 points are stored in the vector `spatial1`. The coordinates are arranged in a 50 x 2 matrix `x`.

   Use a Monte Carlo test to test the null hypothesis that the distribution of points is uniform over the unit square
   Hint: The hardest part to this problem is computing the nearest neighbour distances for any particular pattern of points. If you are unable to write your own function to do this, a function is given at the end of this section, and is stored in the `MAS472.RData` workspace under the name `nnsum`.

4. Section 4.3.5 skewed data example.
   Use a randomization test to test the hypothesis that the two population means are equal.
   Hint: Given a vector of data `x`, you can randomly permute the order of the elements in `x` with the command
   `sample(x)`

5. Section 4.3.6 analysis of variance.
   Use a randomization test to test the hypothesis that the four group means are equal.

6. Section 4.3.9, exercise 2.
   A regression dataset is stored under the name `regression1` from the same website. The x values are stored in the first column, and the y values are stored in the second.

   Use a randomization test to test the hypothesis that $\beta = 0$. On a graph, plot the fitted line from the actual data, and on the same axes, plot 100 fitted lines from randomized data. (Use a different colour for the actual fitted line: `plot(x,y,type="l",col=8)`.

7. The law school data from section 4.4.3 are as follows: {(576,3.39), (635,3.30), (558,2.81), (578,3.03),(666,3.44), (580,3.07), (555,3.00), (661,3.43), (651,3.36), (605,3.13), (653,3.12), (575,2.74), (545,2.76), (572,2.88), (594,2.96)}

Write a function that will use bootstrapping to estimate the standard error of the sample correlation coefficient between x and y.

1. Solution to Q1, part 1.
   The following function can be used to estimate the required probability. This function is stored under the name `windprob.slow`.

   Comments `(#)` have been added to explain how the function works, but if you are still unsure of what a particular command does, you could either try the command yourself in the command window, or you could add in a `print` command in the function. e.g., to see the effect of
   `y<-vector("numeric",100)`
   add in the command
   `print(y)`
   on the following line.

```
windprob<-function(n){
# n is the number of time series to be generated
# Define x and y initially as vectors of zeros.  Individual elements of x and
y can then be changed later.
x<-vector("numeric",n)
y<-vector("numeric",100)
# y is a 100 element vector containing a single time series
# x is an n element vector which will store the number of days the wind speed
is below the target level for each simulate time series.
# the first two values of the time series are fixed at 1.5:
y[1]<-1.5
y[2]<-1.5
# set up a for loop to generate n time series in turn
for(i in 1:n){
# set up a for loop to generate the remaining 98 elements of a single time
series:
for(j in 3:100){
# simulate the next value of the time series:
y[j]<-0.6*y[j-1]+0.4*y[j-2]+rnorm(1,0,0.1)
}
# Count the number of elements of the time series that are
less than log(4.167):
x[i]<-sum(y<1.427)
# for a vector (or matrix) y, the command z<- y<1.427 produces another vector
(or matrix) z where z[i] is 1 if y[i]<1.427, and z[i] is 0 otherwise.  This
```

```
is essentially how we program indicator functions.
}
# Find the proportion of the n series that have more than 10 elements
less than log(4.167):
return(mean(x>10))
# Use the return command to specify what the output of your function is.
}
```

Typing `windprob(n)` will generate n time series of 100 observations, and count the proportion of times the event E occurs for each time series. Note that this not an efficiently written function; `for` loops tend to be executed rather slowly and should be avoided whenever possible. Here, we used a loop to generate the elements of `x` in turn. This can be avoided, because each element of `x` is independent of the others, i.e., it is possible to generate all `n` elements of `x` simultaneously.

The following function generates all n times series at once, and so runs much faster. This function is stored under the name `windprob.fast`.

```
windprob<-function(n){
# Set up a single matrix of all n time series
# Each column of the matrix represents a single time series
y<-matrix(1.5,100,n)
# Generate a single matrix of all the noise terms required to
# generate all the time series (98 * n in total)
noise<-matrix(rnorm(98*n,0,0.1),98,n)
# Set a single loop to generate all n time series at once:
for(i in 3:100){
# y[i,] gives the ith row of the matrix, i.e., the ith
# element of all n time series
y[i,]<-0.6*y[i-1,]+0.4*y[i-2,]+noise[i-2,]
}
# Apply the function sum to each column of the matrix formed by y<1.427
x<-apply((y<1.427),2,sum)
# (The command x<-apply((y<1.427),1,sum) would apply the function sum to each
row of the matrix formed by y<1.427)
return(mean(x>10))
}
```

2. Function for computing sum of nearest neighbour distances. This function is stored under the name `nnsum`.

```
nnsum<-function(x){
```

```
# the input x is a n by 2 matrix of the spatial coordinates
# Determine how many observations there are:
n<-length(x[,1])
# The next three commands produce a n by n matrix of all the pairwise
distances.  If you can't see what's going on here, try these commands in the
command window with a matrix x of say 3 points.
mx1<-matrix(x[,1],n,n,byrow=F)
mx2<-matrix(x[,2],n,n,byrow=F)
# t(mx1) gives the transpose of the matrix mx1.
distances<-((mx1-t(mx1))^2+(mx2-t(mx2))^2)^0.5
# We're about to get the nearest neighbour for each point by taking the min
of each column (or row).  However, we need to ignore the diagonal, because
each element on the diagonal is the distance from a point to itself.  The next
command will ensure that the diagonal values are not the smallest, by adding
100 along the diagonal.
distances<-distances+diag(100,n,n)
# Find the nearest neighbour for each point, and sum them.
return(sum(apply(distances,2,min)))
}
```

7 Generating Random Variables

We have considered several computational techniques for inference that either involve generating new observations at random, or resamping/permuting existing observations at random. In this section we will look at how to generate a random variable X from an arbitrary density function $f_X(x)$.

In semester 1, Bayesian Statistics, we studied one technique that could be used to simulate from *any* density function: Markov chain Monte Carlo. However, to do MCMC sampling we need to be able to generate random variables from our choice of proposal density, hence the need for methods of generating random variables that are simpler than MCMC itself. For non-standard multivariate density functions, typically of the sort encountered in Bayesian statistics, MCMC is the most commonly used technique. For scalar density functions there are often more efficient alternative methods.

The starting point for generating a random variable from any density function involves generating variables from the $U[0,1]$ distribution. Given this sample $U_1, \ldots, U_n$ we can then consider a transformation $g(U)$, possibly combined with some other procedures, to obtain a random draw from $f_X(x)$.

7.1 Generating random numbers from a $U[0,1]$ distribution

A computer cannot actually generate random numbers, only 'pseudo-random' numbers, a deterministic sequence $X_1, X_2, \ldots$ that can, for all practical purposes, be treated as random. This is done through the use of **congruential generators**. A sequence of numbers $X_1, X_2, \ldots$ is produced using the formula

$$N_i = (aN_{i-1} + c) \text{mod } M \tag{68}$$

$$X_i = \frac{N_i}{M}, \tag{69}$$

for integers N_i. An example is $N_i = (8404997N_{i-1} + 1) \text{ mod } 2^{35}$. Some points to note:

- The numbers generated are restricted to the values $0, \frac{1}{M}, \frac{2}{M}, \ldots, \frac{M-1}{M}$, and the sequence of random numbers must be periodic with period M.
- As long as n is small relative to M, and M is sufficiently large, the sequence $X_1, \ldots, X_n$ will 'behave' like a sample of independent values from a $U[0,1]$ distribution.
- The starting value N_1 is known as the **seed**. This can be reset so that you can reproduce the same set of uniform random numbers. In R, use the command: `set.seed(i)`, where i is an integer between 0 and 1023.

7.2 Obtaining non-uniform random numbers with the inversion method

Let X be any random variable and define $Y = F_X(X)$, where $F_X(x) = P(X \leq x)$. Note that we are defining Y as a *random variable* here; Y is defined as a function of the random variable X, where the function in this case happens to be the distribution function of X. Now $Y \in [0,1]$, and the distribution function of Y is

$$F_Y(y) = P(Y \leq y) \tag{70}$$

$$= P(F_X(X) \leq y) \tag{71}$$

$$= P(X \in A), \tag{72}$$

where A is the set $\{x : F_X(x) \leq y\}$.

From Figure 12 we can see that

$$P(X \in A) = y \tag{73}$$

$$\Rightarrow F_Y(y) = y \tag{74}$$

which is the distribution function of a uniform random variable on $[0,1]$. Irrespective of the distribution of X, the random variable $Y = F_X(X)$ is uniformly distributed on $[0,1]$. So if we want to generate a random value of $F_X(X)$, we just generate a $U[0,1]$ random variable. But given the value of $F_X(X)$, we can deduce the corresponding (unique) value of X, as long as we

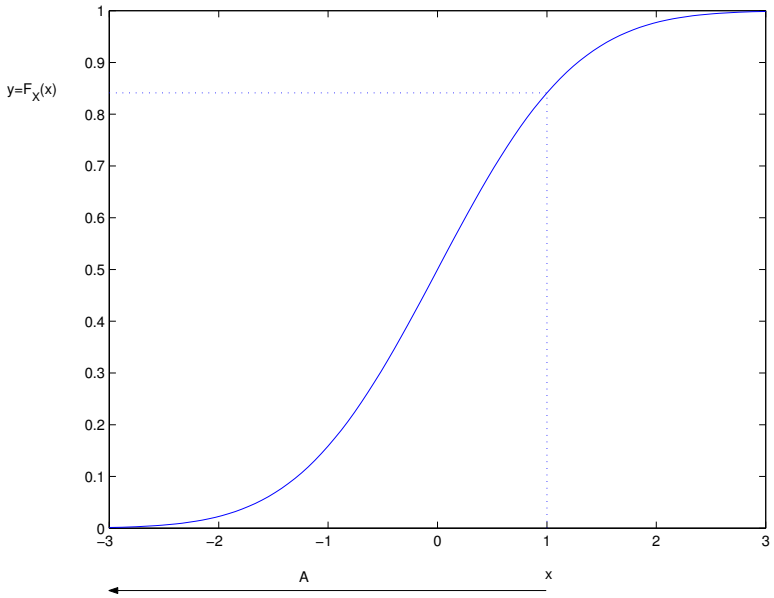

Figure 12:

can invert F. We already know how to generate uniform (psuedo) random variables, so if we can invert the distribution function, we can then generate random values from the distribution of X as follows:

$$\text{Let } U = Y = F_X(X). \text{ Then } X = F_X^{-1}(U).$$

i.e, if we generate a single uniform random variable u, calculate $F_X^{-1}(u) = x$, and set $X = x$, then X will have the required distribution. This is known as the inversion method.

7.2.1 Example: the exponential distribution

Let X have the exponential distribution with mean λ. i.e.

$$\begin{aligned} f_X(x) &= \frac{1}{\lambda}\exp\left(-\frac{x}{\lambda}\right) \quad \text{for } x \geq 0 && (75)\\ \Rightarrow F_X(x) &= \int_0^x \frac{1}{\lambda}\exp\left(-\frac{t}{\lambda}\right) dt && (76)\\ &= \left[-\exp\left(-\frac{t}{\lambda}\right)\right]_0^x && (77)\\ &= 1-\exp\left(-\frac{x}{\lambda}\right). && (78) \end{aligned}$$

Now set $u = 1 - \exp\left(-\frac{x}{\lambda}\right)$ and solve for x.

$$x = -\lambda \ln(1-u), \tag{79}$$

so generate a uniform random number u on $[0, 1]$, and plug u into equation (79) to get a random number from the *exponential*(λ) distribution.

• Discrete distributions.

We can also use the inversion method to sample from discrete distributions. Let X be discretely distributed with possible values x_i having probability p_i, so that $P(X = x_i) = p_i$, and $\sum_i p_i = 1$. Then the distribution function $F_X(x) = \sum_{x_i \leq x} p_i$ is a step function (Figure 13).

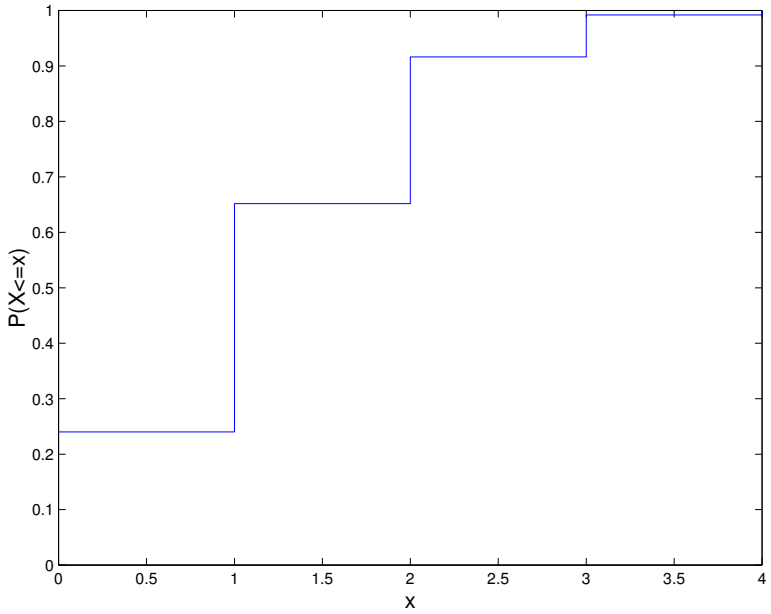

Figure 13:

We generate u from $U[0, 1]$, then $X = x$ if

$$\sum_{x_i<x} p_i < u \leq \sum_{x_i \leq x} p_i \tag{80}$$

7.2.2 Example: binomial distribution

Let $X \sim Binomial(n = 4, p = 0.3)$. Then $P(X = 0) = 0.2401$, $P(X = 1) = 0.4116$, $P(X = 2) = 0.2646$, $P(X = 3) = 0.0756$, $P(X = 4) = 0.0081$. The inversion algorithm then tells you to generate u from $U[0, 1]$ and then set

$$\begin{aligned} x &= 0 \text{ if } 0 \leq u \leq 0.2401 \\ x &= 1 \text{ if } 0.2401 < u \leq 0.6517 \\ x &= 2 \text{ if } 0.6517 < u \leq 0.9163 \\ x &= 3 \text{ if } 0.9163 < u \leq 0.9919 \\ x &= 4 \text{ if } 0.9919 < u \leq 1.0 \end{aligned}$$

7.2.3 Generating normal random variables

We can't easily use the inversion method to generate normal random variables, because the cumulative distribution function $\Phi(.)$ cannot be written in closed form. A favoured approach for generating normal random variables is the **Box-Muller Method**:
Take two independent $U[0,1]$ random variables U_1, U_2, then calculate:

$$X_1 = \sqrt{-2\ln U_1}\cos(2\pi U_2) \tag{81}$$

$$X_2 = \sqrt{-2\ln U_1}\sin(2\pi U_2). \tag{82}$$

Then X_1 and X_2 are independent $N(0,1)$ variables.

7.3 The rejection method

The rejection method is a general purpose method for sampling from univariate distributions. Suppose we wish to sample X from $f_X(x)$, but cannot do so directly. In the rejection method, we find an alternative density function $g_Y(y)$ that we can sample from. The function $g_Y(y)$ is known as the **envelope function** and needs to satisfy the condition

$$\frac{f_X(x)}{g_Y(x)} \text{ bounded } \forall x, \tag{83}$$

so that we can find a constant c with

$$c\, g_Y(x) \geq f_X(x) \,\forall\, x, \tag{84}$$

i.e $c \geq \sup_x \frac{f_X(x)}{g_Y(x)}$. The rejection method can then be stated as follows:

1. Generate y from density $g_Y(y)$, and a uniform u from $U[0,1]$.
2. If $u \leq \frac{f_X(y)}{c\, g_Y(y)}$, state $X = y$, otherwise return to step 1.

Step 2 can be written as accept $X = y$ if $u\, c\, g_Y(y) \leq f_X(y)$, reject otherwise.

We keep generating pairs (y, u) from $g_Y(y)$ and $U[0,1]$ until the condition is satisfied.
To see why it works, we consider $P(X \leq x)$ using this method, and show that this probability is given by the desired distribution function $F_X(x)$.

$$P(X \leq x) = P\left(Y \leq x | U \leq \frac{f_X(Y)}{c\, g_Y(Y)}\right) \tag{85}$$

$$= \frac{P\left(Y \leq x,\, U \leq \frac{f_X(Y)}{c\, g_Y(Y)}\right)}{P\left(U \leq \frac{f_X(Y)}{c\, g_Y(Y)}\right)} \tag{86}$$

$$= \frac{\int_{-\infty}^{x} \int_0^{\frac{f_X(y)}{c\, g_Y(y)}} g_Y(y)\, 1\, du\, dy}{\int_{-\infty}^{\infty} \int_0^{\frac{f_X(y)}{c g_Y(y)}} g_Y(y)\, 1\, du\, dy} \tag{87}$$

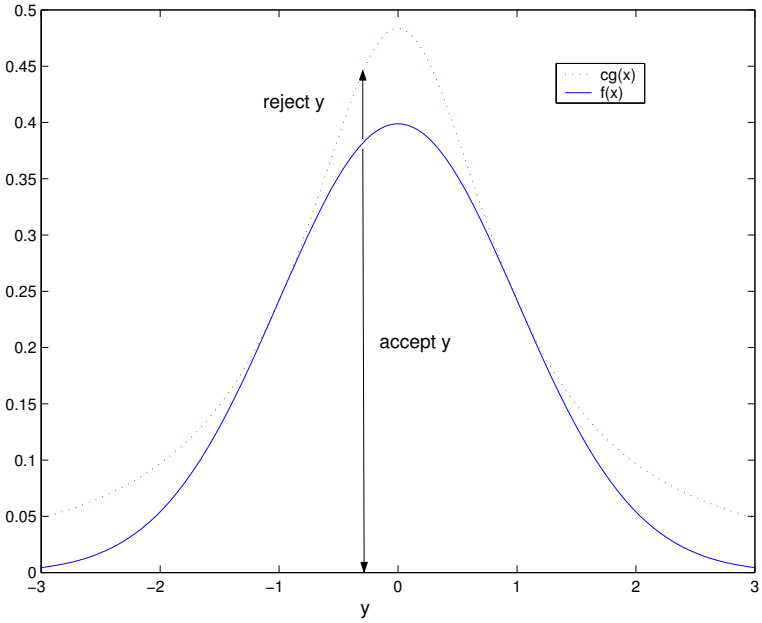

Figure 14:

The numerator is

$$\int_{-\infty}^{x} g_Y(y) \left[u\right]_0^{\frac{f_X(y)}{c\, g_Y(y)}} dy = \int_{-\infty}^{x} \frac{f_X(y)}{c} dy \tag{88}$$

$$= \frac{1}{c} F_X(x) \tag{89}$$

and the denominator is $\frac{1}{c}$, so $P(X \leq x) = F_X(x)$ as required.

7.3.1 Efficiency of the rejection method

To generate a single X from $f_X(x)$ we may have to generate many (Y, U) pairs until a particular Y is accepted. The probability of rejection for any single pair (Y, U) is

$$P\left(U \geq \frac{f_X(Y)}{c\, g_Y(Y)}\right) = 1 - \frac{1}{c}. \tag{90}$$

The number of attempts required for Y to be accepted therefore has a geometric distribution with expectation c. For maximum efficiency we want c as small as possible, i.e. we want $\sup_x \frac{f_X(x)}{g_Y(x)}$ as small as possible. This means we must find a density function g that is both easy to sample from, and mimics f as closely as possible.

> You will note that we have defined $c \geq \sup_x \frac{f_X(x)}{g_Y(x)}$ rather than $c = \sup_x \frac{f_X(x)}{g_Y(x)}$. This is because in some cases identifying the *least* upper bound of $\frac{f_X(x)}{g_Y(x)}$ may be difficult, whereas identifying *an* upper bound may be more straightforward. Clearly the rejection method is valid if we choose $c > \sup_x \frac{f_X(x)}{g_Y(x)}$, though we can see that it will not be as efficient.

7.3.2 Example

The following example may offer some more insight into how the rejection method works: Suppose $X \in [0,1]$, and has density function $f_X(x)$. We will choose $g_Y(y)$ to be the uniform density function on $[0,1]$. Then we want

$$c \geq \sup_x \frac{f_X(x)}{g_Y(x)} = \sup_x f_X(x), \tag{91}$$

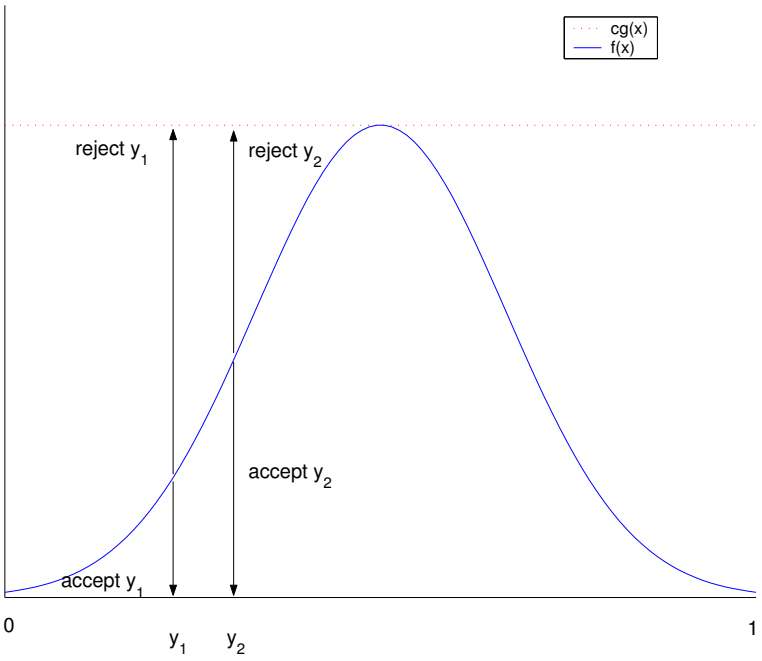

Figure 15:

Now consider two points y_1 and y_2, with $f_X(y_2) = 2f_X(y_1)$ (Figure 15). In a large sample $x_1, \ldots, x_N$ from $f_X(x)$, you would expect there to be twice as many points in the interval $(y_2, y_2 + \varepsilon)$ (for small, non-zero ε) as there are in the interval $(y_1, y_1 + \varepsilon)$, because

$$\begin{aligned} P\{X \in (y_2, y_2 + \varepsilon)\} &\approx \varepsilon f_X(y_2) && (92) \\ &= 2\varepsilon f_X(y_1) && (93) \\ &\approx 2P\{X \in (y_1, y_1 + \varepsilon)\}. && (94) \end{aligned}$$

Since we are generating points from a uniform density $g_Y(y)$, in a large sample $y_1, \ldots, y_M$, we will have roughly the same number of values of y in the two intervals $(y_1, y_1 + \varepsilon)$ and $(y_2, y_2 + \varepsilon)$. From the rejection algorithm, any point $y \in (y_1, y_1 + \varepsilon)$ will be accepted with probability approximately equal to $\frac{f_X(y_1)}{c}$. Any point $y \in (y_2, y_2 + \varepsilon)$ will be accepted with probability approximately equal to $\frac{f_X(y_2)}{c}$. This is twice the value of $\frac{f_X(y_1)}{c}$, so twice as many points in $(y_2, y_2 + \varepsilon)$ will be accepted, as required.

7.3.3 Example: generating normal random variables from cauchy random variables

Suppose we can easily generate cauchy random variables Y. How do we use the rejection method to sample a normal random variable X? We have

$$g_Y(y) = \frac{1}{\pi(1+y^2)}. \tag{95}$$

We need to find

$$\sup_x \frac{f_X(x)}{g_Y(x)} = \sup_x \frac{\exp(-0.5x^2)\pi(1+x^2)}{\sqrt{2\pi}} \tag{96}$$

By differentiating with respect to x and equating to zero, we can show that the maximum occurs at $x = \pm 1$, and is $\sqrt{\frac{2\pi}{e}}$, and so $c = \sqrt{\frac{2\pi}{e}}$. So we generate u from a $U[0,1]$ distribution, and y from the cauchy distribution, and accept y as the normal random variable if

$$u \leq \frac{f_X(y)}{c\, g_Y(y)} = \frac{\sqrt{e}}{2}(1+y^2)\exp(-y^2/2). \tag{97}$$

7.3.4 Truncated distributions

Suppose we wish to sample X from the following distribution:

$$f_X(x) \propto \begin{cases} g_X(x) & \text{for } x \in A \\ 0 & \text{otherwise} \end{cases} \tag{98}$$

where $g_X(x)$ is a known density that we can sample from. An example would be if $g_X(x)$ was the $N(0,1)$ density, and $A = [0, \infty)$. In fact, we can write

$$f_X(x) = \begin{cases} k\, g_X(x) & \text{for } x \in A \\ 0 & \text{otherwise} \end{cases} \tag{99}$$

where k is a normalising constant, given by

$$k^{-1} = \int_A g_X(x)dx \tag{100}$$

Now consider using the rejection method to sample X from $f_X(x)$. We will sample Y from the full (non-truncated) density $g_X(x)$. Note that

$$\frac{f_X(x)}{g_X(x)} = \begin{cases} k & \text{if } x \in A \\ 0 & \text{otherwise} \end{cases} \tag{101}$$

So $c = \sup_x \frac{f_X(x)}{g_X(x)} = k$. Now the standard rejection algorithm is to sample u from $U[0,1]$ and y from $g_Y(y)$, and accept $X = y$ if $u \leq \frac{f_X(y)}{c\, g_Y(y)}$. But since

$$\frac{f_X(x)}{c\, g_X(x)} = \begin{cases} \frac{f_X(x)}{k\, g_X(x)} = 1 & \text{if } x \in A \\ 0 & \text{otherwise} \end{cases} \tag{102}$$

we will always have $u \leq \frac{f_X(y)}{c\, g_Y(y)}$ if $y \in A$, and $u \geq \frac{f_X(y)}{c\, g_Y(y)}$ if $y \notin A$. So we don't actually need to sample u. We can just do the following:

1. generate y from $g_Y(y)$
2. if $y \in A$, accept $X = y$
3. otherwise, return to step 1.

As usual, the acceptance probability will be high if c is small, i.e $\int_A g_Y(y)\,dy$ is near 1. So if the truncated region is large, rejection sampling will be inefficient.

7.4 Exercises

1. The $Beta(a, b)$ density function is given by

$$f(\theta) = \begin{cases} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\theta^{a-1}(1-\theta)^{(b-1)} & \theta \in (0,1) \\ 0 & \text{otherwise,} \end{cases}$$

   with $\Gamma(x) = (x-1)!$ for integer x. Explain how you would generate a single random $Beta(2, 2)$ random variable using rejection sampling with a uniform density as the envelope function, giving the expected number of uniform random variables required.

2. Is it possible to simulate Cauchy random variables using a normal density as the envelope function?

7.5 Adaptive rejection sampling

A difficulty with rejection sampling lies in identifying a suitable choice of envelope function $g_Y(y)$. It may then also be non-trivial to find the maximum of the ratio

$$\frac{f_X(x)}{g_Y(x)}.$$

A very useful technique that can be applied to density functions that are **log-concave** is that of adaptive rejection sampling. We will define

$$h(x) = \log f_X(x). \tag{103}$$

The density function $f_X(x)$ is log-concave if the first derivative $h'(x)$ of $h(x)$ decreases monotonically with x. This means that any tangent to $h(x)$ will not intersect the function h at any other point. The idea of adaptive rejection sampling is then to work on the log scale, and construct an envelope function from a set of tangents to $h(x)$ at various values of x.

Working on the log scale, we will directly construct

$$g^*(y) = log(cg_Y(y)) \tag{104}$$

to be a *piece-wise linear upper hull* of $h(x)$ (see figure 16). This is done as follows. Define

$$S_k = \{x_1, \ldots, x_k\}, \tag{105}$$

and evaluate both $h(x_1), \ldots, h(x_k)$ and $h'(x_1), \ldots, h'(x_k)$. Now, for $i = 1, \ldots, k-1$, the tangents to $h(x)$ at the points x_i and x_{i+1} intersect at

$$z_i = \frac{h(x_{i+1}) - h(x_i) - x_{i+1}h'(x_{i+1}) + x_i h'(x_i)}{h'(x_i) - h'(x_{j+1})}. \tag{106}$$

We can now construct the piece-wise linear upper hull $g^*(x)$: for $x \in [z_{i-1}, z_i]$ with $i = 1, \ldots, k$ we state $g^*(x) = g_i^*(x)$ with

$$g_i^*(x) = h(x_i) + (x - x_i)h'(x_i). \tag{107}$$

(z_0 and z_k are the lower and upper bounds of the random variable X, and may be $-\infty$ and $+\infty$ respectively). Given $g^*(x)$, we can then obtain a the envelope density function $g_Y(y)$ as the piece-wise exponential density

$$g_Y(y) = g_{Y,i}(y) \text{ for } y \in [z_{i-1}, z_i], \tag{108}$$

with

$$g_{Y,i}(y) = \frac{\exp\{g_i^*(y)\}}{\int_{z_0}^{z_1} \exp\{g_1^*(x)\}dx + \ldots + \int_{z_{k-1}}^{z_k} \exp\{g_k^*(x)\}dx} \tag{109}$$

We can analytically determine the distribution function of $g_Y(y)$, which we can then invert, so that we can generate random values from $g_Y(y)$ using inversion. Note that the greater the value of k, the more tangents are used to construct the envelope function, and so the closer $g^*(x)$ mimics the shape of $h(x)$.

The full adaptive-rejection algorithm can then be stated as follows:

1. Choose an initial value of k and initial values $S_k = \{x_1, \ldots, x_k\}$. A possibility is $k = 2$, and if X is unbounded then we would require $h'(x_1) > 0$ and $h'(x_2) < 0$.

2. Evaluate $h(x_1), \ldots, h(x_k)$ and $h'(x_1), \ldots, h'(x_k)$.

3. Determine $z_1, \ldots, z_{k-1}$ and construct the piecewise-linear upper hull $g^*(x)$

4. From $g^*(x)$ determine the envelope density $g_Y(y)$

5. Sample Y from $g_Y(y)$ and U from $U[0, 1]$.

6. If
$$U \leq \exp\{h(Y) - g^*(Y)\},$$
then accept $X = Y$.

7. Otherwise, arrange $\{x_1, \ldots, x_k, Y\}$ to obtain a new set $S_{k+1} = \{x_1, \ldots, x_{k+1}\}$ and return to step 2.

To see what's going on on the original scale, we plot $f_X(x)$ and $cg_Y(x)$ in Figure 18.

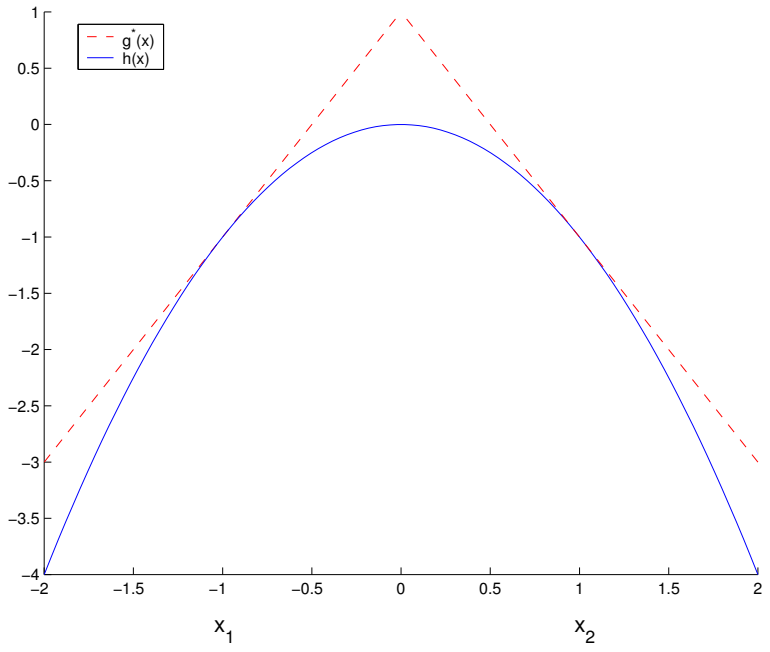

Figure 16: $h(x)$ and the piece-wise linear upper hull $g^*(x)$

7.6 Multivariate Generators

We will now consider how to generate a random vector $\mathbf{X} = \{X_1, \ldots, X_d\}$ from a density $f_{\mathbf{X}}(\mathbf{x})$. Some general points to note:

1. The simplest case is when $\{X_1, \ldots, X_d\}$ are independent, so that we have:

$$f_{\mathbf{X}}(\mathbf{x}) = f_{X_1}(x_1) \ldots f_{X_d}(x_d). \tag{110}$$

   In this case we can just generate X_1 from $f_{X_1}(x_1)$, then generate X_2 from $f_{X_2}(x_2)$ etc., perhaps using inversion or rejection in each case (different uniforms will be needed each time).

2. We cannot use inversion based on a single uniform, i.e., we cannot invert $U = F_{\mathbf{X}}(\mathbf{x})$. (Why?)

3. Rejection can still be used (even if $\{X_1, \ldots, X_d\}$ are not independent). We generate $\mathbf{Y}$ from $g_{\mathbf{Y}}(\mathbf{y})$, generate U from $U[0, 1]$, and as before accept $\mathbf{X} = \mathbf{Y}$ if

$$Ucg_{\mathbf{Y}}(\mathbf{Y}) \leq f_{\mathbf{X}}(\mathbf{Y}), \tag{111}$$

   with

$$c \geq \sup_x \frac{f_{\mathbf{X}}(\mathbf{x})}{g_{\mathbf{Y}}(\mathbf{x})}. \tag{112}$$

   $g_{\mathbf{Y}}(\mathbf{y})$ may be the product of d independent density functions, though in some cases this may lead to the RHS of (112) increasing as d increases, and we know that larger c implies a less efficient sampling algorithm.

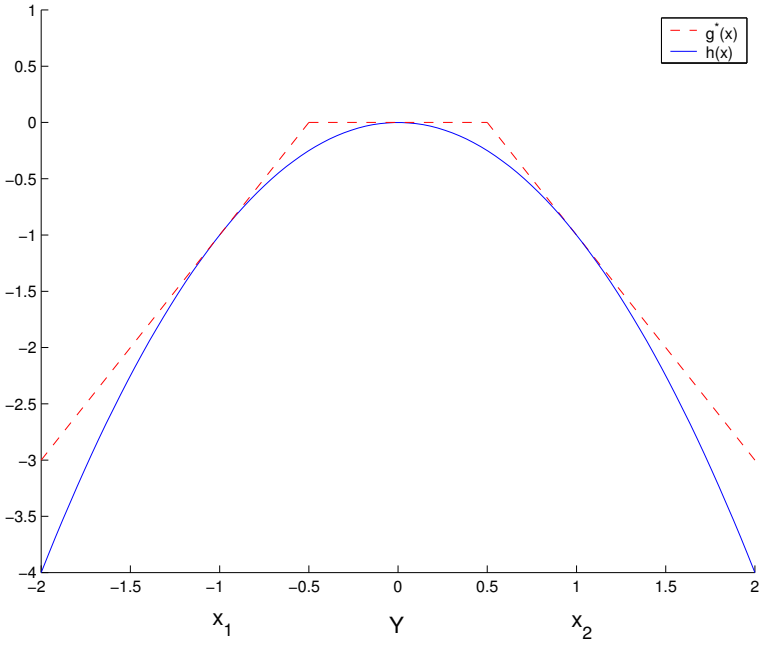

Figure 17: If Y has been rejected, a new tangent is added at Y so that $g^*(x)$ will fit $h(x)$ more closely

7.6.1 Sequential methods

Any multivariate density function $f_{\mathbf{X}}(\mathbf{x})$ can be written in the form

$$f_{\mathbf{X}}(\mathbf{x}) = f_{X_1}(x_1)\, f_{X_2|X_1}(x_2|x_1)\, f_{X_3|X_2,X_1}(x_3|x_2,x_1)\ldots \tag{113}$$

and so we can sample from X_1 from $f_{X_1}(x_1)$, then sample X_2 from $f_{X_2|X_1}(x_2|x_1)$ etc.

7.6.2 Example

We wish to sample $\{\theta, \phi\}$ from the density function

$$f(\theta,\phi) \propto \phi^3 \exp\{-1(1+5\phi)\theta^2 + 40\phi\theta - 81\phi\}. \tag{114}$$

Firstly, it can be shown that the marginal density of ϕ is

$$f(\phi) \propto \phi^3(1+5\phi)^{-1/2} \exp\left\{-\phi\left(1+\frac{80}{1+5\phi}\right)\right\}. \tag{115}$$

Additionally, the conditional distribution of $\theta|\phi$ is normal:

$$\theta|\phi \sim N\left(\frac{20\phi}{1+5\phi}, \frac{1}{2(1+5\phi)}\right). \tag{116}$$

To sample from $f(\theta,\phi)$ we could first sample ϕ from $f(\phi)$, using rejection sampling, then generate Z from $N(0,1)$, and finally set

$$\theta = \frac{20\phi}{1+5\phi} + \frac{1}{\sqrt{2(1+5\phi)}} Z. \tag{117}$$

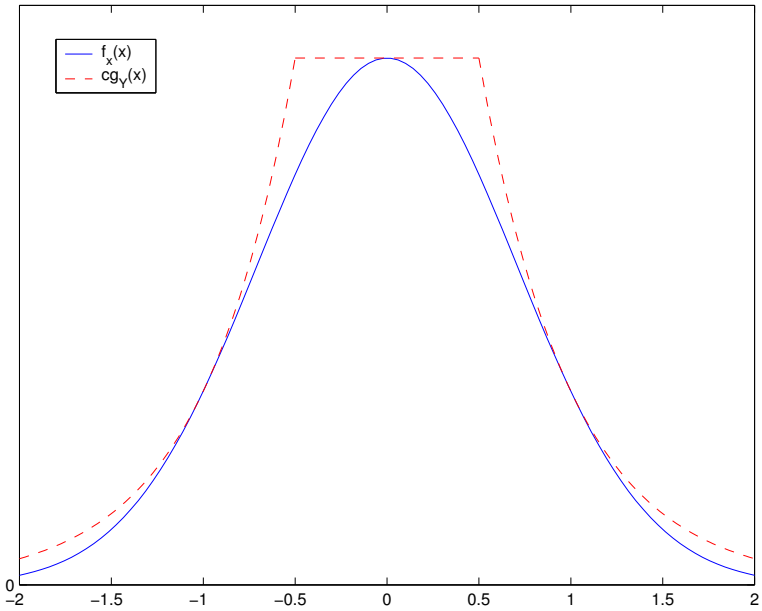

Figure 18: Target density $f_X(x)$ and envelope $cg_Y(x)$

7.6.3 Multivariate normal distributions

Suppose we wish to generate $\mathbf{X}$ where

$$\mathbf{X} \sim N(\mathbf{m}, V), \tag{118}$$

for some non-diagonal matrix V, given a sample of independent standard normal random variables $Z_1, Z_2, \ldots$. One technique for doing so involves the use of the **Cholesky square root** of the matrix V:

For any (symmetric, square) positive definite matrix V, we can find a square root U, such that

$$U^T U = V. \tag{119}$$

There is no unique square root matrix, but one particular square root is an upper triangular matrix, and is known as the Cholesky square root. (Note that some authors define the Cholesky square root to be a *lower* triangular matrix. However, this definition is consistent with R). To find the Cholesky square root of a matrix V in R, simply type
`chol(V)`.
Now, define $\mathbf{Z}$ to be vector of independent standard normal random variables, of equal dimension to $\mathbf{X}$, and consider the transformation $\mathbf{Y} = \mathbf{m} + U^T\mathbf{Z}$. We then have $\mathbf{Y}$ normally distributed, as each element of $\mathbf{Y}$ is a linear combination of the elements of $\mathbf{Z}$, and

$$\begin{aligned} E(\mathbf{m} + U^T\mathbf{Z}) &= \mathbf{m}, & (120)\\ Var(\mathbf{m} + U^T\mathbf{Z}) &= U^T I U & (121)\\ &= V, & (122) \end{aligned}$$

(with I the identity matrix, the variance of $\mathbf{Z}$). Hence to generate $\mathbf{X}$, we generate independent standard normal random variables $\mathbf{Z}$, and then transform them by $\mathbf{m} + U^T\mathbf{Z}$ to obtain $\mathbf{X}$.

7.6.4 Exercise

In R, generate 100000 random values from the $N\left\{\begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}\right\}$ distribution. Verify using the `var` command that your simulated data has the correct covariance properties.

7.7 Importance sampling

Another technique for sampling from probability distributions is that of importance sampling. The main application of importance sampling is in Bayesian statistics, where we wish to sample from non-standard (posterior) distributions. In low dimensions it can be simpler to implement than MCMC, and more efficient. As in MCMC, it is only necessary to know the density function up to proportionality (i.e. the product of prior and likelihood).

Suppose we wish to sample a random variable X from some density function $f(x)$, but are unable to do so directly. Note that X may be a scalar or a vector here. One possibility is to consider sampling $X_1, \ldots, X_n$ from an alternative density $g(x)$ to obtain a **weighted sample** $\{X_i, w_i\}_{i=1}^n$, with

$$w_i = \frac{f(X_i)}{g(X_i)}. \tag{123}$$

The density $g(x)$ is known as the **importance density**. If the purpose of sampling from $f(x)$ is to estimate a quantity such as $E(X)$, then from our study of Monte Carlo integration, we already know that this weighted sample will be sufficient, as

$$\hat{R} = \frac{1}{n}\sum_{i=1}^{n} w_i X_i \tag{124}$$

is an unbiased estimator of $E(X)$. This weighted sample can be used to estimate any moment of X, and any probability $P(X \in A)$ with the estimator

$$\hat{R} = \frac{1}{n}\sum_{i=1}^{n} I\{X_i \in A\} w_i \tag{125}$$

since

$$\begin{aligned} E\left[I\{X \in A\}w\right] &= E\left[I\{X \in A\}\frac{f(X)}{g(X)}\right] && (126)\\ &= \int I\{x \in A\}\frac{f(x)}{g(x)}g(x)dx && (127)\\ &= \int_{x\in A} f(x)dx && (128)\\ &= P(X \in A) && (129) \end{aligned}$$

The usual considerations of choosing $g(x)$ to mimic $f(x)$ as closely as possible apply.

We can go one step further to obtain an un-weighted (or equal-weighted) sample from $f(x)$ by **re-sampling**. Given $\{X_i, w_i\}_{i=1}^n$, set

$$X_i^* = X_j \text{ with probability proportional to } w_j.$$

The sample $X_1^*, \ldots, X_n^*$ is then a sample from $f(x)$, and can be used for example for estimating percentiles from the distribution of X. This technique is known as **sampling importance re-sampling (SIR)**. The re-sampling step should only be used when the weighted sample is not sufficient for the task in hand, as re-sampling will reduce the diversity in your sample (and so requires a larger sample size).

7.7.1 Importance sampling with unnormalised density functions

Suppose the density function $f(x)$ is only known up to proportionality, i.e., we know $\tilde{f}(x)$ with

$$f(x) = \frac{\tilde{f}(x)}{\int \tilde{f}(x)dx}, \tag{130}$$

but the value of the integral $K = \int \tilde{f}(x)dx$ is unknown. This integral can, of course, be estimated using Monte Carlo integration. As usual, we sample $X_1, \ldots, X_n$ from $g(x)$, then estimate K by

$$\hat{K} = \frac{1}{n}\sum_{i=1}^{n} \tilde{w}(X_i), \tag{131}$$

with $\tilde{w}(X) = \tilde{f}(X)/g(X)$. We could then approximate $f(x)$ by

$$f(x) \simeq \frac{\tilde{f}(x)}{\frac{1}{n}\sum_{i=1}^n \tilde{w}(X_i)}. \tag{132}$$

Now consider using importance sampling to sample from $f(x)$. If we sample $X_1, \ldots, X_n$ from $g(x)$, using the approximation in (132), our weighted sample will be $\{X_i, w(X_i)\}_{i=1}^n$, with

$$w(X_i) = \frac{f(X_i)}{g(X_i)} = \frac{\tilde{f}(X_i)}{g(X_i)\frac{1}{n}\sum_{i=1}^n \tilde{w}(X_i)} = \frac{n\tilde{w}(X_i)}{\sum_{i=1}^n \tilde{w}(X_i)}. \tag{133}$$

If, for example, we wish to estimate $E(X)$, then we would use the estimate

$$\hat{E}(X) = \frac{1}{n}\sum_{i=1}^{n} X_i W(X_i) = \frac{\sum_{i=1}^n X_i\tilde{w}(X_i)}{\sum_{i=1}^n \tilde{w}(X_i)}. \tag{134}$$

Note that this is not an unbiased estimate of $E(X)$ due to the estimate of $\int \tilde{f}(x)dx$ used in the denominator. However, it is possible to prove that (under weak assumptions) as $n \to \infty$, $\hat{E}(X) \to E(X)$ almost surely.

In Bayesian statistics, we may only know the posterior density up to proportionality, i.e., we cannot obtain analytically the normalising constant $f(\mathbf{x}) = \int f(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})d\boldsymbol{\theta}$. By writing

$\tilde{f}(\boldsymbol{\theta}|\mathbf{x}) = f(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})$, we can immediately see how, in prinicple, importance sampling can be used to obtain weighted samples from $f(\boldsymbol{\theta}|\mathbf{x})$ without knowing the value of $f(\mathbf{x})$. If we wish to know the posterior mean, we can estimate it by

$$\hat{E}(\boldsymbol{\theta}|\mathbf{x}) = \frac{\sum_{i=1}^{n} \boldsymbol{\theta}_i \tilde{w}(\boldsymbol{\theta}_i)}{\sum_{i=1}^{n} \tilde{w}(\boldsymbol{\theta}_i)}, \tag{135}$$

with $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n$ sampled from some other density $g(\boldsymbol{\theta})$.

7.7.2 Choice of g and the normal approximation

The main issue in importance sampling is, of course, how to go about choosing a suitable density function g for sampling the random variables from. If we wish to sample from a posterior density function $f(\boldsymbol{\theta}|\mathbf{x})$, then one possibility is to use the prior distribution, i.e. set $g(\boldsymbol{\theta}) = f(\boldsymbol{\theta})$. If we are not able to obtain $f(\mathbf{x})$ directly then the weights will be given by

$$w(\boldsymbol{\theta}_i) = \frac{nf(\mathbf{x}|\boldsymbol{\theta}_i)}{\sum_{i=1}^{n} f(\mathbf{x}|\boldsymbol{\theta}_i)}. \tag{136}$$

This may not be very efficient if the prior is not very informative relative to the data (likelihood). In this case, we are very likely to sample $\boldsymbol{\theta}_i$ in regions where the likelihood is very small, so that the corresponding weights $w(\boldsymbol{\theta}_i)$ will also be very small. Typically we will find that in the set $\{w(\boldsymbol{\theta}_1), \ldots, w(\boldsymbol{\theta}_n)\}$, a small number of weights will be large (when the likelihood is large), and most will be small, so that estimates of, e.g., $E(\boldsymbol{\theta}|\mathbf{x})$ will be dominated by one or two of the sampled $\boldsymbol{\theta}$ values.

An alternative is to use a normal approximation of f as the sampling density g. The approximation is surprisingly easy to obtain! We first define $h(\boldsymbol{\theta}) = \log f(\boldsymbol{\theta}|\mathbf{x})$. Now define $\mathbf{m}$ to be the posterior mode of $\boldsymbol{\theta}$, so that $\mathbf{m}$ maximises both $f(\boldsymbol{\theta}|\mathbf{x})$ and $h(\boldsymbol{\theta})$. Typically, we will have to use numerical optimisation to find $\mathbf{m}$, but note that $\mathbf{m}$ can be found without knowing the normalising constant $f(\mathbf{x})$. We now do a Taylor series expansion of $h(\boldsymbol{\theta})$

$$h(\boldsymbol{\theta}) = h(\mathbf{m}) + (\boldsymbol{\theta} - \mathbf{m})^T \mathbf{h}'(\mathbf{m}) + \frac{1}{2}(\boldsymbol{\theta} - \mathbf{m})^T M(\boldsymbol{\theta} - \mathbf{m}) + \ldots \tag{137}$$

with $h'(\mathbf{m})$ the vector of first derivatives of $h(\boldsymbol{\theta})$, and M the matrix of second derivatives of $h(\boldsymbol{\theta})$, both evaluated at $\boldsymbol{\theta} = \mathbf{m}$.

Since $\mathbf{m}$ maxmises $h(\mathbf{m})$ we have $h'(\mathbf{m}) = \mathbf{0}$. Hence

$$f(\boldsymbol{\theta}|\mathbf{x}) = \exp\{h(\boldsymbol{\theta})\} \simeq \exp\{h(\mathbf{m})\} \exp\left\{-\frac{1}{2}(\boldsymbol{\theta} - \mathbf{m})^T V^{-1}(\boldsymbol{\theta} - \mathbf{m})\right\}, \tag{138}$$

where $-V^{-1} = M$. From (138) we see that an approximation of $f(\boldsymbol{\theta}|\mathbf{x})$ is given by a multivariate normal density function with mean vector $\mathbf{m}$ and variance matrix $-M^{-1}$. The approximation will be good for $\boldsymbol{\theta}$ close to $\mathbf{m}$, and so will be a good approximation to $f(\boldsymbol{\theta}|\mathbf{x})$ if there is sufficient

data (or prior information) such that most of the posterior mass is concentrated around the mode. Note that it is not necessary to know the normalising constant to obtain M, since

$$h(\boldsymbol{\theta}) = \log f(\boldsymbol{\theta}|\mathbf{x}) = \log f(\boldsymbol{\theta}) + \log f(\mathbf{x}|\boldsymbol{\theta}) - \log f(\mathbf{x}), \tag{139}$$

and so the $\log f(\mathbf{x})$ will disappear when we differentiate $h(\boldsymbol{\theta})$. Note that if (138) is a good approximation to $f(\boldsymbol{\theta}|\mathbf{x})$ everywhere, then it can be used directly to give approximate summaries (means, percentiles etc.) from $f(\boldsymbol{\theta}|\mathbf{x})$.

7.7.3 Assessing convergence

Suppose we wish to estimate the posterior mean of some function $r(\boldsymbol{\theta})$. If the normalising constant is known, then our estimate of $E\{r(\boldsymbol{\theta})|\mathbf{x}\}$ is given by

$$\hat{E}\{r(\boldsymbol{\theta})|\mathbf{x}\} = \frac{1}{n}\sum_{i=1}^{n} r(\boldsymbol{\theta}_i)w(\boldsymbol{\theta}_i), \tag{140}$$

and we can use the central limit theorem to obtain a confidence interval for $E\{r(\boldsymbol{\theta})|\mathbf{x}\}$, exactly as in (32). Unfortunately, if the normalising constant is unknown, so that our estimate of $E\{r(\boldsymbol{\theta})|\mathbf{x}\}$ is given by

$$\hat{E}\{r(\boldsymbol{\theta})|\mathbf{x}\} = \frac{\sum_{i=1}^{n} r(\boldsymbol{\theta}_i)\tilde{w}(\boldsymbol{\theta}_i)}{\sum_{i=1}^{n} \tilde{w}(\boldsymbol{\theta}_i)}, \tag{141}$$

then the central limit theorem can only be applied under stricter conditions to derive a confidence interval, which we will not consider in this course. Consequently, we will consider two informal approaches for checking convergence:

1. Increase the sample size n to check stability of any estimate.

2. Increase the standard deviation in the importance density, to check stability of any estimate to the choice of g. Using the normal approximation, we could try multiplying the variance matrix V by 4, so that individual standard deviations are doubled.

7.7.4 Example: leukaemia data

Patients suffering from leukaemia are given a drug, 6-mercaptopurine (6-MP), and the number of days x_i until freedom from symptoms is recorded of patient i:

$$6^*, 6, 6, 6, 7, 9^*, 10^*, 10, 11^*, 13, 16, 17^*, 19^*, 20^*, 22, 23, 25^*, 32^*, 32^*, 34^*, 35^*.$$

A * denotes an observation censored at that time. One possible model for this data is to suppose that the time x to the event of interest follows a *Weibull* distribution:

$$f(x|\alpha, \beta) = \alpha\beta(\beta x)^{\alpha-1}\exp\{-(\beta x)^{\alpha}\} \tag{142}$$

for $x > 0$. When $\alpha = 1$, this reduces to the exponential distribution. Regarding a censored observation, we have

$$P(x > t|\alpha, \beta) = \exp\{-(\beta t)^\alpha\}. \tag{143}$$

Denote d to be the number of uncensored observations, and $\sum_u \log x_i$ to be the sum of the logs of all the uncensored observations. Writing $\boldsymbol{\theta} = (\alpha, \beta)^T$, the log likelihood is given by

$$\log f(\mathbf{x}|\boldsymbol{\theta}) = d \log \alpha + \alpha d \log \beta + (\alpha - 1) \sum_u \log x_i - \beta^\alpha \sum_{i=1}^n x_i^\alpha. \tag{144}$$

There is no conjugate prior distribution for $\boldsymbol{\theta}$, so it will not be possible to obtain the posterior distribution of $\boldsymbol{\theta}$ for any choice of prior. For illustration, we consider proper, but vague priors for both α and β:

$$f(\alpha) = 0.001 \exp(-0.001\alpha), \tag{145}$$

$$f(\beta) = 0.001 \exp(-0.001\beta). \tag{146}$$

We now consider the use of importance sampling to simulate from the posterior distribution $f(\boldsymbol{\theta}|\mathbf{x})$, using the normal approximation to obtain $g(\boldsymbol{\theta})$. This will be implemented in R(see the Rscript `weibull-importance-sampling.R` on the course website for details).

1. **Obtain the posterior mode of $\boldsymbol{\theta}$.**
   We maximise the log posteior, i.e.

   $$h(\boldsymbol{\theta}) = d \log \alpha + \alpha d \log \beta + (\alpha - 1) \sum_u \log x_i - \beta^\alpha \sum_{i=1}^n x_i^\alpha - 0.001\alpha - 0.001\beta + K, \tag{147}$$

   for some constant K. In R, we find the mode to be $\mathbf{m} = (1.354, 0.030)$.

2. **Derive the matrix of second derivatives of** h($\boldsymbol{\theta}$).
   We need

   $$M = \begin{pmatrix} \frac{\partial^2}{\partial\alpha^2} h(\boldsymbol{\theta}) & \frac{\partial^2}{\partial\alpha\partial\beta} h(\boldsymbol{\theta}) \\ \frac{\partial^2}{\partial\alpha\partial\beta} h(\boldsymbol{\theta}) & \frac{\partial^2}{\partial\beta^2} h(\boldsymbol{\theta}) \end{pmatrix}, \tag{148}$$

   evaluated at $\boldsymbol{\theta} = \mathbf{m}$.

   The required derivatives are

   $$\frac{\partial^2}{\partial\alpha^2} h(\boldsymbol{\theta}) = -\frac{d}{\alpha^2} - \beta^\alpha \left\{ (\log\beta)^2 \sum_{i=1}^n x_i^\alpha + 2\log\beta \sum_{i=1}^n x_i^\alpha \log x_i + \sum_{i=1}^n x_i^\alpha (\log x_i)^2 \right\}, \tag{149}$$

   $$\frac{\partial^2}{\partial\beta^2} h(\boldsymbol{\theta}) = \frac{1}{\beta^2} \left\{ \beta^\alpha \alpha(1-\alpha) \sum_{i=1}^n x_i^\alpha - d\alpha \right\}, \tag{150}$$

   $$\frac{\partial^2}{\partial\alpha\partial\beta} h(\boldsymbol{\theta}) = \frac{1}{\beta} \left[d - \beta^\alpha \left\{ \alpha \log\beta \sum_{i=1}^n x_i^\alpha + \sum_{i=1}^n x_i^\alpha + \alpha \sum_{i=1}^n x_i^\alpha \log x_i \right\} \right], \tag{151}$$

   and so

   $$M = \begin{pmatrix} -31.462 & 175.900 \\ 175.900 & -18828.553 \end{pmatrix}. \tag{152}$$

3. **Obtain the normal approximation to use as $g(\boldsymbol{\theta})$.**
   We have $g(\boldsymbol{\theta})$ a bivariate normal density with mean vector $\mathbf{m}$ and variance matrix $V = -M^{-1}$:
   $$\boldsymbol{\theta} \sim N\left\{ \begin{pmatrix} 1.354 \\ 0.030 \end{pmatrix}, \begin{pmatrix} 0.0335 & 0.0003 \\ 0.0003 & 0.00006 \end{pmatrix} \right\} \tag{153}$$

4. **Sample $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n$ from $g(\boldsymbol{\theta})$ and compute the importance weights $w(\boldsymbol{\theta}_1), \ldots, w(\boldsymbol{\theta}_n)$.**
   The weights are given by
   $$w(\boldsymbol{\theta}_i) = \frac{n\tilde{w}(\boldsymbol{\theta}_i)}{\sum_{i=1}^{n} \tilde{w}(\boldsymbol{\theta}_i)}, \tag{154}$$
   with
   $$\tilde{w}(\boldsymbol{\theta}_i) = \frac{f(\boldsymbol{\theta}_i) f(\mathbf{x}|\boldsymbol{\theta}_i)}{g(\boldsymbol{\theta}_i)}. \tag{155}$$
   Note that is we may sample some negative $\boldsymbol{\theta}$ from $g(\boldsymbol{\theta})$. Since both α and β must be positive, these can simply be discarded. Effectively we then have a truncated normal density for $g(\theta)$. When computing $w(\boldsymbol{\theta}_i)$, it is not necessary to rescale the normal density function $g(\theta)$ so that it integrates to 1, as any normalising constant in $g(\theta)$ will cancel in (154).

5. **Estimate the posterior mean of $\boldsymbol{\theta}$**
   We compute the estimate
   $$\hat{E}(\boldsymbol{\theta}|\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\theta}_i w(\boldsymbol{\theta}_i). \tag{156}$$
   In R, with $n = 100000$, this gives $\hat{E}(\boldsymbol{\theta}|\mathbf{x}) = (1.346, 0.031)^T$.

6. **Check for convergence**
   We repeat steps 4 and 5 with more dispersion in the importance density:

$g(\boldsymbol{\theta})$	$\hat{E}(\boldsymbol{\theta}\|\mathbf{x})$
$N(\mathbf{m}, V)$	$(1.346, 0.031)^T$
$N(\mathbf{m}, 4V)$	$(1.384, 0.031)^T$
$N(\mathbf{m}, 16V)$	$(1.380, 0.031)^T$

   This suggests that $N(\mathbf{m}, 4V)$ is more suitable as the importance density. Finally, we double the sample size to 200,000, but this does not have a significant effect.

If we wish to obtain percentiles from the distribution of $\boldsymbol{\theta}$, we need to resample, where the probability of resampling $\boldsymbol{\theta}_i$ is proportional to $w(\boldsymbol{\theta}_i)$. This is very simple to do in R, using the `sample` command. Suppose we have `theta` as a 100000×2 matrix of 100000 sampled (α, β) pairs, and `W` a vector of weights. We use the commands

```
index<-sample(c(1:100000),100000,replace=T,prob=(W/sum(W)))
newtheta<-theta[index,]
```

The argument `prob=W/sum(W)` specifies the probabilities of choosing each $\boldsymbol{\theta}$ value in the original sample. The resampled set of $\boldsymbol{\theta}$ values can now be thought of as a draw directly from $f(\boldsymbol{\theta}|\mathbf{x})$, and so we can just find the sample percentiles to estimate posterior intervals for α and β. For α, we get

```
> quantile(newtheta[, 1], c(0.025, 0.975))
      2.5%    97.5%
 0.7425045 2.184742
```

and for β we get

```
> quantile(newtheta[, 2], c(0.025, 0.975))
       2.5%      97.5%
 0.01490134 0.04649747
```

7.7.5 Exercises

1. In R, use importance sampling to obtain a weighted sample from the $Beta(10, 15)$ distribution, using the $U[0, 1]$ distribution as the importance distribution. Use resampling to estimate the 5th and 95th percentiles of this distribution.

2. Suggest an alternative choice of importance density based on the normal approximation in section 7.7.2.

7.8 ⋆ MCMC and convergence diagnostics ⋆

MCMC sampling was introduced in semester 1, Bayesian Statistics. To recap, we sample from the *target* density $f_X(x)$ by generating a Markov chain $X_1, X_2, \ldots$ whose stationary distribution is $f_X(x)$. The general algorithm (the Metropolis-Hastings algorithm) is as follows:

1. Generate a *candidate point* Y from a *proposal density* $q(Y|X_t)$.

2. Set $X_{t+1} = Y$ with probability $\alpha(X_t, Y)$, or set $X_{t+1} = X_t$ with probability $1 - \alpha(X_t, Y)$, where
$$\alpha(X_t, Y) = \min\left(1, \frac{f_X(y)q(X_t|Y)}{f_X(X_t)q(Y|X_t)}\right) \tag{157}$$
To do this step, we sample U from the $U[0, 1]$ distribution. If $U \leq \alpha(X_t, Y)$, we set $X_{t+1} = Y$, and if $U > \alpha(X_t, Y)$ we set $X_{t+1} = X_t$.

If the Markov chain has converged to its stationary distribution by time T, then $X_{T+1}, X_{T+2}, \ldots$ is a sample from $f_X(x)$. A practical difficulty in implementing MCMC is identifying whether or not convergence has been achieved. In semester 1, we used an informal approaches such as visually inspecting long runs of the chain, and trying different starting values for X_1. Various *convergence diagnostics* have been proposed in the literature, as a more formal way of checking for convergence. Convergence diagnostics will not *prove* whether a Markov chain has converged

or not, but can sometimes be more reliable than simple visual checks. In the next section, we consider one such diagnostic.

7.8.1 The Brooks, Gelman and Rubin (BGR) diagnostic

There are two versions of this diagnostic. The original Gelman and Rubin (1992) diagnostic is described in Bayesian Data Analysis (chapter 11). A modified version, proposed by Brooks and Gelman (1998), is implemented in WinBUGS (if you have used that program).

Recall that when a Markov chain has reached equilibrium, it has 'forgotten' its starting value, and so samples obtained from two Markov chains that have both reached equilibrium should have the same properties, even if the starting values were very different. In the BGR diagnostic, we generate several parallel chains with *overdispersed* starting values, and compare between-chain and within-chain variability. By overdispersed, we mean that if the starting values of each chain are sampled randomly, their sampling variance should be larger than that of $Var(X)$. This can be achieved through the use of an approximation to $f_X(x)$, but here we just choose some suitably varied starting values, perhaps chosen following an initial run of the Markov chain.

Suppose we generate m parallel chains, and discard the first half of each chain as burn-in. For the remainder of chain j, let X_{ij} denote the i-th simulated value, for $i = 1, \ldots, n$. Now define B to be the between-sequence variance:

$$B = \frac{n}{m-1} \sum_{j=1}^{m} \left(\bar{X}_{\bullet j} - \bar{X}_{\bullet\bullet} \right)^2, \tag{158}$$

where

$$\bar{X}_{\bullet j} = \frac{1}{n} \sum_{i=1}^{n} X_{ij}, \tag{159}$$

$$\bar{X}_{\bullet\bullet} = \frac{1}{m} \sum_{j=1}^{m} X_{\bullet j}, \tag{160}$$

and W to be the within-sequence variance:

$$W = \frac{1}{m} \sum_{j=1}^{m} \frac{1}{n-1} \sum_{i=1}^{n} \left(X_{ij} - \bar{X}_{\bullet j} \right)^2. \tag{161}$$

Note the similarity with one-way ANOVA, where we consider the ratio of between-group and within-group variances. Now consider the following weighted estimate of $Var(X)$:

$$\widehat{Var}^{+}(X) = \frac{n-1}{n} W + \frac{1}{n} B. \tag{162}$$

If the starting value for each chain is itself a draw from $f_X(x)$, or in the limit as $n \to \infty$, the estimator $\widehat{Var}^{+}(X)$ is an unbiased estimator of $Var(X)$. If instead we choose the starting

values to be overdispersed, then $\widehat{Var}^+(X)$ will overestimate $Var(X)$ (we would expect the between-chain variability to be too large). Now consider the ratio

$$R = \sqrt{\frac{\widehat{Var}^+(X)}{Var(X)}}. \tag{163}$$

Given the properties of $\widehat{Var}^+(X)$, this will tend to 1 as $n \to \infty$. For finite n, suppose we use $\sqrt{\widehat{Var}^+(X)}$ as our estimate of $\sqrt{Var(X)}$. We can then see that R tells us by what factor our estimated standard deviation would shrink, were we to continue running the chains indefinitely. If this factor is large, then our values X_{ij} are not representative of a sample from $f_X(x)$.

We cannot actually evaluate R, as we do not know $Var(X)$. Instead, we monitor

$$\widehat{R} = \sqrt{\frac{\widehat{Var}^+(X)}{W}}. \tag{164}$$

For finite n we would expect W to underestimate $Var(X)$, as any individual Markov chain may not have fully explored the target distribution $f_X(x)$. Consequently, we would expect $\hat{R}$ to overestimate R, so that $\hat{R}$ provides a conservative assessment of convergence.

Gelman et al (2004) suggest that a value of $\hat{R}$ below 1.1 should be acceptable in most cases, though you should consider how much precision is required in your inferences, and whether smaller values of $\hat{R}$ may be necessary. Fairly small values of m are used in practice, e.g. $m = 5$ or 10.

The modified Brooks and Gelman (1998) version uses an alternative measure of spread, specifically, the width of a $100(1-\alpha)\%$ credible interval rather than a variance, but the principle is the same. Define p_δ to be δ quantile of the pooled sampled values $\{X_{ij}\}$, $i = 1, \ldots, n$, $j = 1, \ldots, m$, and $p_{j,\delta}$ to be δ quantile of the j-th chain. We then monitor

$$\hat{R} = \frac{p_{1-\alpha/2} - p_{\alpha/2}}{\frac{1}{m}\sum_{j=1}^{m} p_{j,1-\alpha/2} - p_{j,\alpha/2}}. \tag{165}$$

The numerator gives a pooled measure of variability, based on the combined data from all m chains. The denominator gives an average measure of variability within each chain. Large values of R indicate more variability between the chains than within them, suggesting that the chains have not all converged to their stationary distribution.

7.8.2 Exercises

1. (Hard). If the starting value of each chain has been sampled from $f_X(x)$, prove that $\widehat{Var}^+(X)$ is an unbiased estimator of $Var(X)$.

7.9 Reducing Variance in Monte Carlo Simulation

Consider estimating the expectation of some function g of a random variable X:

$$E\{g(\mathbf{X})\} = \int g(\mathbf{x}) f(\mathbf{x}) d\mathbf{x}. \tag{166}$$

Given a sample $\mathbf{X}_1, \ldots, \mathbf{X}_n$ from $f(\mathbf{x})$, we can estimate $E\{g(\mathbf{X})\}$ in the usual way by $\frac{1}{n}\sum_{i=1}^{n} g(\mathbf{X}_i)$. However, in some cases, we may be limited to a fairly small value of n, specifically if $g(\mathbf{x})$ takes considerable computing time to evaluate. In this case, it can be desirable to reduce the variance in the estimator $\frac{1}{n}\sum_{i=1}^{n} g(\mathbf{X}_i)$ that is caused by variation in the sample $\mathbf{X}_1, \ldots, \mathbf{X}_n$. In this section we will study three variance reduction techniques: Latin hypercube sampling, antithetic variables, and control variables.

7.9.1 Latin Hypercube Sampling

Consider two independent random variables X and Y, with $X \sim N(0,1)$ and $Y \sim U(0,1)$. A simple random sample of size n from the joint distribution of (X, Y) would be obtained by sampling $X_1, \ldots, X_n$ independently from $N(0,1)$, sampling $Y_1, \ldots, Y_n$ independently from $U(0,1)$, and then pairing the sampled values together to get $\{X_i, Y_i\}_{i=1}^{n}$.

A Latin Hypercube Sample (LHS) is a type of stratified sample. We first divide the sample space of X into n regions of equal probability, and then sample one value at random from each region, to get $X_1, \ldots, X_n$. We then do likewise for Y to get $Y_1, \ldots, Y_n$. Finally, we randomly permute the order of $Y_1, \ldots, Y_n$ before pairing the X and Y values, so that each X_i will be randomly paired with one value from the set $\{Y_1, \ldots, Y_n\}$. The idea is that by stratifying into regions of equal probability, we can obtain a small sample that is 'more representative' of the distribution of X and Y. Note that this procedure guarantees that the marginal distributions of both X and Y are 'covered' by the sample. An illustration for an LHS of size 5 is given in Figure 19.

Note that given the 5 regions for X, the X values are not sampled uniformly from each region. For example, X_2 is sampled from a $N(0,1)$ distribution truncated to the interval $(-0.843, -0.253)$. This could be done using the rejection method in section 7.3.4, or alternatively using inversion and the `qnorm` function in R: we first sample U from $U[0.2, 0.4]$, and then invert the normal cdf (in R) to obtain X_2.

This process can be extended to any number of dimensions. Suppose we have $\mathbf{X} = (X_1, \ldots, X_d)$. For $i = 1, \ldots, d$:

1. Divide the sample space of X_i into n regions of equal probability.
2. Sample one random value from each region to get $\{X_{i,1}, \ldots, X_{i,n}\}$.
3. Randomly permute the n random values to get $\{X^*_{i,1}, \ldots, X^*_{i,n}\}$

The j-th random value of $\mathbf{X}$ in the LHS is then given by $(X^*_{1,j} \ldots, X^*_{d,j})$. Note that it is not actually necessary to perform step 3 for $i = 1$, only for $i = 2, \ldots, d$.

If we have some scalar function $Y = h(\mathbf{X})$ and an LHS $\mathbf{X}_1, \ldots, \mathbf{X}_n$, then it possible to prove that $\frac{1}{n}\sum_{i=1}^{n} g(Y_i)$ is an unbiased estimator of $E\{g(Y)\}$, where $Y_i = h(\mathbf{X}_i)$. Additionally, if h is monotone with respect to each element of $\mathbf{X}$, and g is a monotone function of Y, then it also

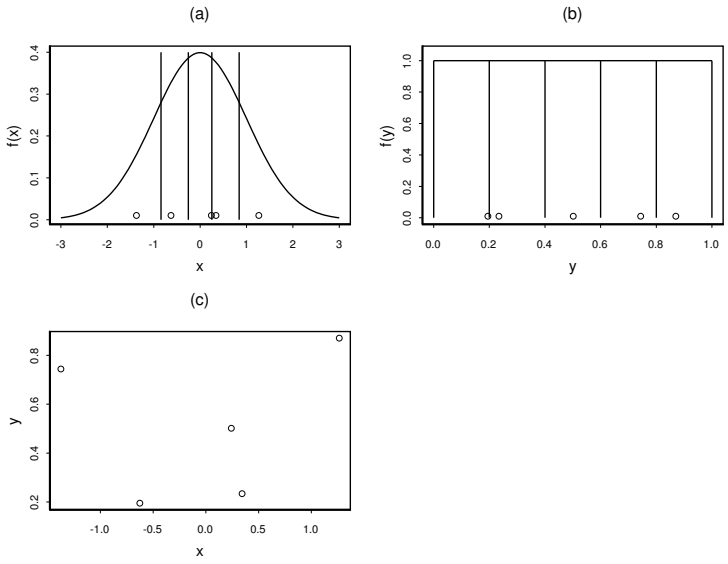

Figure 19: Panels (a) and (b) show 5 sampled X and Y values respectively, each sampled from a region of equal probability. In panel (c), the variables $Y_1, \ldots, Y_5$ before being paired off with $X_1, \ldots, X_5$, to obtain the LHS from the joint distribution of X and Y.

possible to prove that this estimator has smaller variance than the usual estimator based on a simple random sample of the same size. Note that even if these conditions do not hold, Latin hypercube may still be more efficient than simple Monte Carlo sampling.

• **Example**

Consider again the third problem in section 2.2. We have a function of several known and unknown variables.

$$C(x, y, z) = \frac{Q}{2\pi u_{10}\sigma_z\sigma_y} \exp\left[-\frac{1}{2}\left\{\frac{y^2}{\sigma_y^2} + \frac{(z-h)^2}{\sigma_z^2}\right\}\right], \tag{167}$$

with

$$\begin{aligned} \log u_{10} &\sim N(2, .1), \\ \log \sigma_y^2 &\sim N(10, 0.2), \\ \log \sigma_z^2 &\sim N(5, 0.05), \end{aligned}$$

and we wish to know the distribution of $C(100, 100, 40)$, when $h = 50$ and $Q = 100$. We will change the notation here and write this uncertain concentration as $g(\boldsymbol{\theta})$ as a function of the three uncertain variables $\boldsymbol{\theta} = (u_{10}, \sigma_y^2, \sigma_z^2)$. We will compare Latin hypercube sampling with sample random sampling to see which is more efficient for estimating both the mean and distribution function of $g(\boldsymbol{\theta})$.

In this experiment, we generate 1000 simple random samples of size 100, and 1000 Latin hypercube samples of size 100. For each sample, we compute the sample mean of $g(\theta)$ and the empirical cdf $\hat{F}\{g(\boldsymbol{\theta})\}$. In Figure 20 we plot pointwise 95% intervals for the empirical cdf, and

in figure 21 we show histograms for the sample means, obtained from both sampling methods. In both cases we can see that the variability of the estimators is noticeably smaller using Latin hypercube sampling.

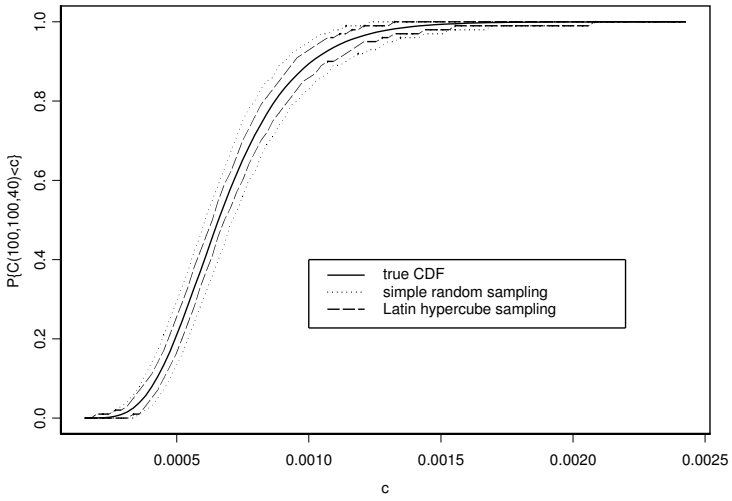

Figure 20: The true distribution function of $g(\boldsymbol{\theta}) = C(100, 100, 40)$, together with pointwise 95% intervals for the empirical cdf obtained from both simple random sampling and Latin hypercube sampling

7.9.2 Antithetic Variables

So far, we have considered generating *independent* variables $X_1, \ldots, X_n$ for use in the estimator $\bar{X}$. But what would happen if the X_is were not independent? For the case $n = 2$, suppose we have $E(X_1) = E(X_2) = \mu$, $Var(X_1) = Var(X_2) = \sigma^2$, and $Cov(X_1, X_2) = \rho\sigma^2 \neq 0$. We still have $E(\bar{X}) = \mu$, so $\bar{X}$ is still an unbiased estimator of μ, but now we have

$$\begin{aligned} Var(\bar{X}) &= \frac{1}{4}Var(X_1 + X_2) && (168)\\ &= \frac{1}{4}\{Var(X_1) + Var(X_2) + 2Cov(X_1, X_2)\} && (169)\\ &= \frac{\sigma^2}{2}(1 + \rho). && (170) \end{aligned}$$

If X_1 and X_2 are independent, then we would have $Var(\bar{X}) = \frac{\sigma^2}{2}$, as before. If instead X_1 and X_2 are negatively correlated, then we would have $\rho < 0$, and so $Var(\bar{X}) < \frac{\sigma^2}{2}$. The idea of **antithetic variables** is to generate X_is in pairs such that the X_is in each pair are negatively correlated.

If X is a uniform random variable, this is straightforward. Given U from the $U[0, 1]$ distribution, we set $X_i = U$ and $X_j = 1-U$. Then the pair $\{X_i, X_j\}$ both have uniform distributions,

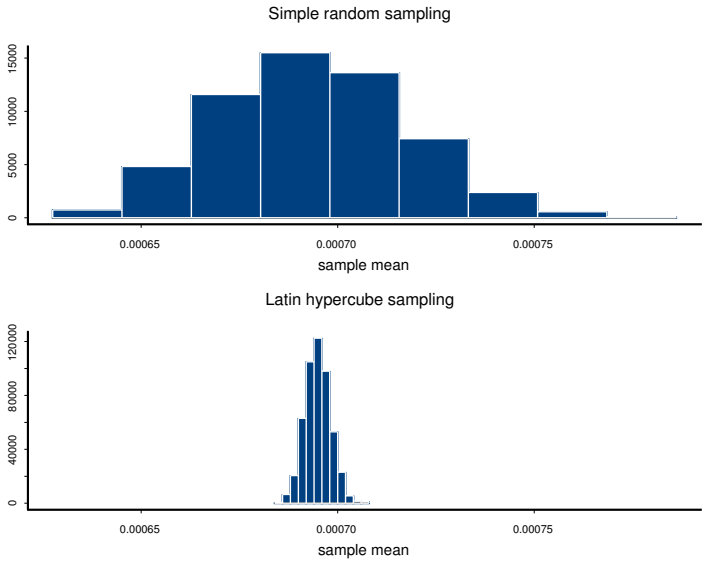

Figure 21: The distribution of the sample mean (for a sample of size 100) of $g(\boldsymbol{\theta})$, obtained by using simple random sampling (top graph) and Latin hypercube sampling (bottom graph). Notice the considerably larger dispersion in the top graph.

and it is easy to see that they are negatively correlated. We can extend this to other distributions by using the inversion method.

7.9.3 ⋆ Control variables ⋆

Suppose that in addition to generating $X_1, \ldots, X_n$, it is also possible to generate variables $Z_1, \ldots, Z_n$, with $E(Z_i) = m$ known. Now we define

$$Y_i = X_i - Z_i, \tag{171}$$

so that

$$E(\bar{Y}) = \mu - m, \tag{172}$$

with the result that $\bar{Y} + m$ is an unbiased estimator of μ. The variance of this estimator is given by

$$\begin{aligned} Var(\bar{Y} + m) &= Var(\bar{Y}) && (173)\\ &= \frac{1}{n}\{Var(X) + var(Z) - 2Cov(X, Z)\} && (174)\\ &= \frac{1}{n}\{\sigma^2 + Var(Z) - 2Cov(X, Z)\} && (175) \end{aligned}$$

We can now see that if X and Z are *positively* correlated to the extent that $2Cov(X, Z) > Var(Z)$, then $\bar{Y} + m$ has a smaller variance than the estimator $\bar{X}$. We call Z the **control variable**. In the control variable approach, we are making use of the correlation between X and Z, and the fact that the expectation of Z is known exactly, to estimate the expectation of

X more efficiently. However, identifying a suitable control variable for use in any given problem is far from straightforward!

7.9.4 Exercise

Generate and plot a Latin hypercube sample of size 100 from the distribution of $\mathbf{X} = (X_1, X_2)$ where $X_1 \sim \Gamma(3, 4)$ and $\log X_2 \sim N(0, 1)$.

8 Likelihood-based inference

In this section of the course we will consider both theoretical and computational aspects of inference based on the likelihood function. We begin with a review of point estimation, interval estimation and hypothesis testing based on the likelihood.

8.1 The likelihood function

We have data $\mathbf{x} = \{x_1, \ldots, x_n\}$, where the joint distribution of $\mathbf{x}$ depends on some unknown parameter θ, which may be a scalar or vector. The likelihood is the joint density (or probability if x_i is discrete) of the data x conditional on the parameter θ, i.e.

$$f(\mathbf{x}|\theta). \tag{176}$$

The likelihood function is thought of as a function of θ *for fixed* $\mathbf{x}$, so to make this explicit we denote the likelihood function by $L(\theta; \mathbf{x})$, with

$$L(\theta; \mathbf{x}) = f(\mathbf{x}|\theta). \tag{177}$$

If $x_1, \ldots, x_n$ are independent, then $f(\mathbf{x}|\theta) = f(x_1|\theta) \times f(x_2|\theta) \times \ldots \times f(x_n|\theta)$, and so

$$L(\theta; \mathbf{x}) = \prod_{i=1}^{n} f(x_i|\theta). \tag{178}$$

Note that the likelihood should *not* be interpreted as a probability density function. While it is certainly true that $\int f(\mathbf{x}|\theta) d\mathbf{x} = 1$, in most cases

$$\int L(\theta; \mathbf{x}) d\theta = \int f(\mathbf{x}|\theta) d\theta \neq 1. \tag{179}$$

8.1.1 Examples

1. Binomial data.
   Consider $x|\theta \sim Binomial(n, \theta)$. Then

   $$L(\theta; x) = f(x|\theta) = {}^{n}C_{x}\theta^{x}(1-\theta)^{n-x}. \tag{180}$$

   (Strictly, as x is discrete we should write $L(\theta; x) = P(x|\theta)$). Note that we have just written $L(\theta; x)$ instead of $L(\theta; x, n)$; we have not included the constant n in the notation.

2. Exponential data.
   We have $x_i|\theta \sim Exp(rate = \theta)$ for $i = 1, \ldots, n$, with $x_1, \ldots, x_n$ independent. (Note: from a Bayesian perspective, we would say $x_1, \ldots, x_n$ are *conditionally* independent *given* θ). Then
   $$L(\theta; \mathbf{x}) = \prod_{i=1}^{n} f(x_i|\theta) = \prod_{i=1}^{n} \theta \exp(-\theta x_i) = \theta^n \exp\left(-\theta \sum_{i=1}^{n} x_i\right). \tag{181}$$

3. The simple linear regression model.
   Suppose we have observations (x_i, y_i) with $i = 1, \ldots, n$ and
   $$y_i = \alpha + \beta x_i + \varepsilon_i, \tag{182}$$
   with $\varepsilon_i \sim N(0, \sigma^2)$ and $\varepsilon_1, \ldots, \varepsilon_n$ independent. We treat $x_1, \ldots, x_n$ as constants and consider the data here to be $y_1, \ldots, y_n$. We define $\theta = (\alpha, \beta, \sigma^2)$ to be the vector of the three unknown parameters. To obtain the likelihood function for θ, we first note that
   $$y_i|\theta, x_i \sim N(\alpha + \beta x_i, \sigma^2). \tag{183}$$
   Hence the likelihood is given by
   $$\begin{aligned} L(\theta; y_1, \ldots, y_n) &= \prod_{i=1}^{n} f(y_i|\theta, x_i) && (184)\\ &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(y_i - \alpha - \beta x_i)^2\right\} && (185)\\ &= (2\pi\sigma^2)^{-n/2} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \alpha - \beta x_i)^2\right\}. && (186) \end{aligned}$$

8.2 Maximum likelihood estimation

The maximum likelihood estimate (m.l.e.) of a parameter θ given data $\mathbf{x}$ is simply the value of θ that maximises the likelihood function $L(\theta; \mathbf{x})$. We denote the m.l.e. of θ by $\hat{\theta}$. The m.l.e. is an intuitively appealing estimator, in that the observed data are 'more probable' when $\theta = \hat{\theta}$ than they are for any other value of θ. Note that for a uniform prior distribution (proper or improper), the m.l.e. is equal to the mode of the posterior distribution $f(\theta|\mathbf{x})$.

It is often more convenient to work with the log-likelihood instead of the likelihood. We denote the log-likelihood by $l(\theta; \mathbf{x})$. Since the transformation from $L(\theta; \mathbf{x})$ to $l(\theta; \mathbf{x})$ is monotonic the same value $\hat{\theta}$ will maximise both functions. Note that for independent observations
$$l(\theta; \mathbf{x}) = \log L(\theta; \mathbf{x}) = \sum_{i=1}^{n} \log f(x_i|\theta). \tag{187}$$

8.2.1 Example - binomial data

From (180) we have

$$l(\theta; x) = \log {}^nC_x + x \log(\theta) + (n - x) \log(1 - \theta). \tag{188}$$

Then

$$\frac{\partial}{\partial \theta} l(\theta; x) = \frac{x}{\theta} - \frac{n - x}{1 - \theta}, \tag{189}$$

which must equal 0 at $\theta = \hat{\theta}$, as $\hat{\theta}$ maximises both $L(\theta; x)$ and $l(\theta; x)$. Hence

$$0 = \frac{x}{\hat{\theta}} - \frac{n - x}{1 - \hat{\theta}} \tag{190}$$

$$\Rightarrow \hat{\theta} = \frac{x}{n}. \tag{191}$$

(We should check that $\hat{\theta}$ is a *maximum* by checking that the second derivative is negative at $\theta = \hat{\theta}$

Note that the constant ${}^nC_x$ disappeared when we differentiated the log-likelihood function. This will happen for any multiplicative constant (function of x only, not θ) in the likelihood, and so for maximum likelihood estimation, it is only necessary to consider the likelihood up to proportionality. Hence we would just write

$$L(\theta; x) \propto \theta^x (1 - \theta)^{n-x}. \tag{192}$$

8.2.2 Exercises

Refer to the examples in section 8.1.1.

1. For the exponential data, prove that the m.l.e. of θ is $\hat{\theta} = \frac{1}{\bar{x}}$.

2. For the simple linear regression model,

$$y_i = \alpha + \beta x_i + \varepsilon_i, \tag{193}$$

with $\varepsilon_i \sim N(0, \sigma^2)$, prove that the m.l.e.s of α and β are the same as the least squares estimates of α and β.

8.3 Some properties of maximum likelihood estimates

8.3.1 Bias

Maximum likelihood estimates are not necessarily unbiased. For the binomial data in section 8.2.1, we have

$$E(\hat{\theta}) = \frac{E(x)}{n} = \frac{n\theta}{n} = \theta. \tag{194}$$

However, for normally distributed data $x_1, \ldots, x_n \sim N(\mu, \sigma^2)$, the m.l.e. of σ^2 is

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2, \tag{195}$$

(you should verify this as an exercise), though

$$\begin{aligned} E\left\{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2\right\} &= E\{(x_i - \bar{x})^2\} && (196)\\ &= E(x_i^2) + E(\bar{x}^2) - \frac{2}{n}E(x_i \times x_i) - \frac{2}{n}\sum_{j\neq i}E(x_i x_j) && (197)\\ &= \sigma^2 + \mu^2 + \frac{\sigma^2}{n} + \mu^2 - \frac{2}{n}(\sigma^2 + \mu^2) - \frac{2}{n}(n-1)\mu^2 && (198)\\ &= \sigma^2\left(\frac{n-1}{n}\right), && (199)\end{aligned}$$

though the bias will tend to 0 as $n \to \infty$.

8.3.2 Invariance property

Maximum likelihood estimates are invariant to transformations. Consider a likelihood function $L(\theta; \mathbf{x})$, and let $\phi = h(\theta)$ be a bijective (one-to one and onto) transformation of θ. We then have $L(\theta; \mathbf{x}) = L(\phi; \mathbf{x})$, as this re-parameterisation does not affect the distribution of x, i.e

$$f(x|\theta = \theta_1) = f(x|\phi = h(\theta_1)) \tag{200}$$

Additionally, if $\hat{\phi}$ maximises $f(x|\phi)$, then $\hat{\theta} = h^{-1}(\hat{\phi})$ will maximise $f(x|\theta)$. To verify this, consider example 2 in section 8.1.1, and show that the m.l.e. of ϕ is $\bar{x}$, where $\phi = 1/\theta$.

8.4 Asymptotic normality of the maximum likelihood estimator

In this section we will derive an important result that can be used to derive confidence intervals for parameters given reasonably large sample sizes. This is the result that the distribution of the m.l.e. tends to a a normal distribution as the sample size tends to infinity. A rigorous proof of is outside the scope of this course, so we will just consider some informal arguments.

8.4.1 Score statistics, Fisher information and the Cramer-Rao minimum variance bound

The score statistic is defined as

$$\frac{\partial}{\partial\theta}l(\theta; \mathbf{x}) = \frac{\partial}{\partial\theta}\log f(\mathbf{x}|\theta). \tag{201}$$

Now let $\mathbf{X}$ be the unobserved value of the data vector $\mathbf{x}$. We now define the *random variable*

$$\frac{\partial}{\partial\theta}l(\theta; \mathbf{X}) = \frac{\partial}{\partial\theta}\log f(\mathbf{X}|\theta). \tag{202}$$

We think of (202) as a transformation of a random variable $\mathbf{X}$, where the transformation is given by the derivative, with respect to θ, of the log of the density of $\mathbf{X}$. This interpretation can be a little confusing, as here we are thinking of $l(\theta; \mathbf{X})$ as a function of the random data $\mathbf{X}$, *evaluated at the true value of* θ, rather than a function of the parameter θ for fixed data $\mathbf{x}$. We will shortly derive the mean and variance of this random variable, but first note that

$$\frac{\partial}{\partial\theta} l(\theta; \mathbf{X}) = \frac{\partial}{\partial\theta} \log L(\theta; \mathbf{X}) = \left\{ \frac{\partial}{\partial\theta} L(\theta; \mathbf{X}) \right\} \times \frac{1}{L(\theta; \mathbf{X})} = \left\{ \frac{\partial}{\partial\theta} f(\mathbf{X}|\theta) \right\} \times \frac{1}{f(\mathbf{X}|\theta)}. \tag{203}$$

For the mean we have

$$\begin{aligned} E\left\{ \frac{\partial}{\partial\theta} l(\theta; \mathbf{X}) \right\} &= \int \left\{ \frac{\partial}{\partial\theta} l(\theta; \mathbf{x}) \right\} f(\mathbf{x}|\theta) d\mathbf{x} && (204) \\ &= \int \left\{ \frac{\partial}{\partial\theta} f(\mathbf{x}|\theta) \right\} \times \frac{1}{f(\mathbf{x}|\theta)} f(\mathbf{x}|\theta) d\mathbf{x} && (205) \\ &= \frac{\partial}{\partial\theta} \int f(\mathbf{x}|\theta) d\mathbf{x} && (206) \\ &= \frac{\partial}{\partial\theta} 1 = 0. && (207) \end{aligned}$$

This shows that the expected value of the derivative of the log-likelihood at the true value of θ is 0. To help understand this, consider the example of $X \sim exp(rate = \theta)$. Then $l(\theta; X) = \log\theta - \theta X$ and

$$\frac{\partial}{\partial\theta} l(\theta; \mathbf{X}) = \frac{1}{\theta} - X, \tag{208}$$

so

$$\begin{aligned} E\left\{ \frac{\partial}{\partial\theta} l(\theta; \mathbf{X}) \right\} &= \int \left(\frac{1}{\theta} - x \right) \theta \exp(-\theta x) dx && (209) \\ &= \frac{1}{\theta} \int \theta \exp(-\theta x) dx - \int x\theta \exp(-\theta x) dx && (210) \\ &= \frac{1}{\theta} - \frac{1}{\theta} = 0. && (211) \end{aligned}$$

However, the expected value of the derivative of the log-likelihood evaluated at the *wrong* value of θ, say θ^*, is not 0. For example,

$$\frac{\partial}{\partial\theta} l(\theta; \mathbf{X}) \bigg|_{\theta=\theta^*} = \frac{1}{\theta^*} - X, \tag{212}$$

with

$$\begin{aligned} E\left\{ \frac{\partial}{\partial\theta} l(\theta; \mathbf{X}) \bigg|_{\theta=\theta^*} \right\} &= \int \left(\frac{1}{\theta^*} - x \right) \theta \exp(-\theta x) dx && (213) \\ &= \frac{1}{\theta^*} - \frac{1}{\theta}, && (214) \end{aligned}$$

which is non-zero for $\theta^* \neq \theta$.

Note that although our expectation is that the likelihood is flat at the true value of θ, this does not imply that the expected value of the m.l.e. $\hat{\theta}$ is the true value of θ, as we have seen in section 8.3.1.

To derive an expression for the variance of $\frac{\partial}{\partial\theta}l(\theta;\mathbf{X})$, we note that

$$0 = \int \frac{\partial}{\partial\theta} l(\theta;\mathbf{x}) f(\mathbf{x}|\theta) d\mathbf{x} \tag{215}$$

$$\Rightarrow \quad 0 = \frac{\partial}{\partial\theta} \int \frac{\partial}{\partial\theta} l(\theta;\mathbf{x}) f(\mathbf{x}|\theta) d\mathbf{x} \tag{216}$$

$$\Rightarrow \quad 0 = \int \left\{ \frac{\partial^2}{\partial\theta^2} l(\theta;\mathbf{x}) \right\} f(\mathbf{x}|\theta) d\mathbf{x} + \int \left\{ \frac{\partial}{\partial\theta} l(\theta;\mathbf{x}) \right\} \left\{ \frac{\partial}{\partial\theta} f(\mathbf{x}|\theta) \right\} d\mathbf{x} \tag{217}$$

$$\Rightarrow \quad 0 = \int \left\{ \frac{\partial^2}{\partial\theta^2} l(\theta;\mathbf{x}) \right\} f(\mathbf{x}|\theta) d\mathbf{x} + \int \left\{ \frac{\partial}{\partial\theta} l(\theta;\mathbf{x}) \right\} \left\{ \frac{\partial}{\partial\theta} l(\theta;\mathbf{x}) \right\} f(\mathbf{x}|\theta) d\mathbf{x} \tag{218}$$

$$\Rightarrow \quad E\left[\left\{ \frac{\partial}{\partial\theta} l(\theta;\mathbf{X}) \right\}^2 \right] = -E\left\{ \frac{\partial^2}{\partial\theta^2} l(\theta;\mathbf{X}) \right\}. \tag{219}$$

Since $E\{\frac{\partial}{\partial\theta}l(\theta;\mathbf{X})\} = 0$, we have

$$Var\left\{ \frac{\partial}{\partial\theta} l(\theta;\mathbf{X}) \right\} = -E\left\{ \frac{\partial^2}{\partial\theta^2} l(\theta;\mathbf{X}) \right\}. \tag{220}$$

The term $-E\left\{\frac{\partial^2}{\partial\theta^2}l(\theta;\mathbf{X})\right\}$ is known as the **Fisher information** which we will denote by $I_E(\theta)$:

$$I_E(\theta) \equiv -E\left\{ \frac{\partial^2}{\partial\theta^2} l(\theta;\mathbf{X}) \right\}. \tag{221}$$

Fisher information is a measure of the amount of information a sample size of n contains about θ. Note that for independent observations $X_1, \ldots, X_n$,

$$l(\theta;\mathbf{X}) = \sum_{i=1}^{n} \log f(X_i|\theta), \tag{222}$$

and so

$$I_E(\theta) = -nE\left\{ \frac{\partial^2}{\partial\theta^2} l(\theta;X_i) \right\}, \tag{223}$$

hence Fisher information is proportional to sample size.

• Example

Suppose we have $X_1, \ldots, X_n \sim N(\theta, \sigma^2)$ with σ^2 known. Then

$$-E\left\{ \frac{\partial^2}{\partial\theta^2} l(\theta;\mathbf{X}) \right\} = -E\left\{ \frac{\partial^2}{\partial\theta^2} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (X_i - \theta)^2 \right\} \tag{224}$$

$$= -E\left\{ -\frac{n}{\sigma^2} \right\}, \tag{225}$$

and so the Fisher information is n/σ^2. As σ^2 decreases, the observations become more likely to be close to θ, and so the data are more informative about θ.

Fisher information can be used to give a bound on the variance of an estimator. Let $T(\mathbf{X})$ be an unbiased estimator, with $X_1, \ldots, X_n$ independent. Then it is possible to prove that

$$Var(T) \geq \frac{1}{I_E(\theta)}. \tag{226}$$

This is known as the **Cramer-Rao minimum variance bound**.

8.4.2 Consistency of the m.l.e.

We now show that the m.l.e is constant, i.e., with probability 1, $\hat{\theta} \to \theta$ as $n \to \infty$. Informally, we demonstrate this by proving

$$\frac{L(\theta^*; \mathbf{X})}{L(\theta; \mathbf{X})} \to 0 \text{ as } n \to \infty, \tag{227}$$

where θ^* is some alternative (incorrect) value of the parameter. Hence as the sample size increases, the likelihood function will become larger at the true value of θ compared to any other value of θ, and so the maximum of the likelihood function must occur at the true value of θ.

We first note that

$$E\left\{\frac{f(X|\theta^*)}{f(X|\theta)}\right\} = \int \frac{f(x|\theta^*)}{f(x|\theta)} f(x|\theta) d\mathbf{x} = 1. \tag{228}$$

We now apply a form of Jensen's inequality, which states that for a strictly convex function g,

$$E\{g(X)\} > g\{E(X)\}. \tag{229}$$

Now $-\log(x)$ is a strictly convex function on $(0, \infty)$, and so

$$E\left\{-\log \frac{f(X|\theta^*)}{f(X|\theta)}\right\} > -\log E\left\{\frac{f(X|\theta^*)}{f(X|\theta)}\right\} = 0 \tag{230}$$

$$\Rightarrow E\left\{\log \frac{f(X|\theta^*)}{f(X|\theta)}\right\} < 0. \tag{231}$$

From the strong law of large numbers, with probability 1,

$$\log \frac{L(\theta^*; \mathbf{X})}{L(\theta; \mathbf{X})} = \sum_{i=1}^{n} \log \frac{f(X_i|\theta^*)}{f(X_i|\theta)} \to nE\left\{\log \frac{f(X|\theta^*)}{f(X|\theta)}\right\}, \tag{232}$$

as $n \to \infty$, i.e.

$$\log \frac{L(\theta^*; \mathbf{X})}{L(\theta; \mathbf{X})} \to -\infty \text{ as } n \to \infty, \tag{232}$$

with probability 1, and so

$$\frac{L(\theta^*; \mathbf{X})}{L(\theta; \mathbf{X})} \to 0 \text{ as } n \to \infty, \tag{233}$$

with probability 1, as required.

8.4.3 Asymptotic normality

The consistency result tells us that for large sample sizes, $\hat{\theta}$ is very likely to be close to the true value θ. Consider a Taylor series expansion of $\frac{\partial}{\partial\theta}l(\theta;\mathbf{X})$:

$$\frac{\partial}{\partial\theta}l(\theta;\mathbf{X})\bigg|_{\theta=\hat{\theta}} = \frac{\partial}{\partial\theta}l(\theta;\mathbf{X}) + (\hat{\theta}-\theta)\frac{\partial^2}{\partial\theta^2}l(\theta;\mathbf{X}) + \ldots \tag{234}$$

Since $\frac{\partial}{\partial\theta}l(\theta;\mathbf{X}) = 0$ at $\theta = \hat{\theta}$, and for large n, θ is likely to be close to $\hat{\theta}$, we have

$$\hat{\theta} - \theta \simeq \frac{\frac{\partial}{\partial\theta}l(\theta;\mathbf{X})}{-\frac{\partial^2}{\partial\theta^2}l(\theta;\mathbf{X})}. \tag{235}$$

Informally, we can argue that for large n, the denominator is likely to be close to its expected value, $I_E(\theta)$, as defined in (221), and the numerator is approximately normally distributed with mean 0 and variance $I_E(\theta)$, as given in (220), by the central limit theorem. Hence for large n, the distribution of the m.l.e $\hat{\theta}$ is approximately normal, with

$$\hat{\theta} \sim N\{\theta, I_E(\theta)^{-1}\}. \tag{236}$$

Thus for large n, the m.l.e. $\hat{\theta}$ is *approximately* unbiased, and achieves the minimum variance bound given in (226).

In the multivariate case with $\theta = (\theta_1, \ldots, \theta_d)$ we have

$$I_E(\theta) = \begin{pmatrix} e_{1,1}(\theta) & \cdots & e_{1,d}(\theta) \\ \vdots & & \vdots \\ e_{d,1}(\theta) & \cdots & e_{d,d}(\theta) \end{pmatrix}, \tag{237}$$

with

$$e_{i,j}(\theta) = E\left\{-\frac{\partial^2}{\partial\theta_i\,\partial\theta_j}l(\theta)\right\}. \tag{238}$$

So for large n, the distribution of the m.l.e of θ is approximately multivariate normal:

$$\hat{\theta} \sim N_d(\theta, I_E(\theta)^{-1}), \tag{239}$$

8.4.4 Example: normally distributed data with unknown variance

Suppose we have independent observations $X_1, \ldots, X_n$ with $X_i \sim N(\theta_1, \theta_2)$, with both θ_1 and θ_2 unknown. We write $\theta = (\theta_1, \theta_2)^T$. Then

$$l(\theta;\mathbf{x}) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log\theta_2 - \frac{1}{2\theta_2}\sum_{i=1}^{n}(x_i-\theta_1)^2, \tag{240}$$

with

$$\hat{\theta}_1 = \frac{1}{n}\sum_{i=1}^{n}x_i \tag{241}$$

$$\hat{\theta}_2 = \frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2 \tag{242}$$

From the definition in (238) we obtain

$$I_E(\theta) = \begin{pmatrix} \frac{n}{\theta_2} & 0 \\ 0 & \frac{n}{2\theta_2^2} \end{pmatrix}. \tag{243}$$

(Verify this as an exercise). So for large n, the approximate distribution of $\hat{\theta} = (\hat{\theta}_1, \hat{\theta}_2)^T$ is

$$\begin{pmatrix} \hat{\theta}_1 \\ \hat{\theta}_2 \end{pmatrix} \sim N\left\{ \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}, \begin{pmatrix} \frac{\theta_2}{n} & 0 \\ 0 & \frac{2\theta_2^2}{n} \end{pmatrix} \right\} \tag{244}$$

Note that the approximation is exact for $\hat{\theta}_1$, since we know that $\bar{X} \sim N(\theta_1, \theta_2/n)$. We can investigate the accuracy of the approximation $\hat{\theta}_2$, since the true distribution of $\hat{\theta}_2$ can be derived from the result

$$\frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{\theta_2} \sim \chi^2_{n-1}. \tag{245}$$

For illustration, in Figure 22 we compare the true and approximate density functions for different sample sizes when $\theta_2 = 1$.

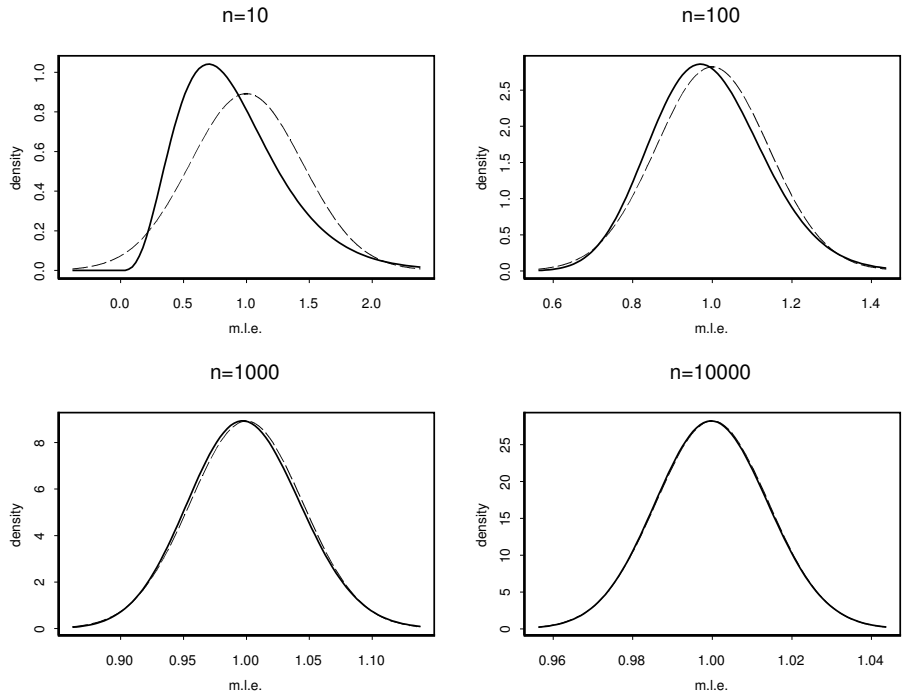

Figure 22: The true (solid line) and approximate (dashed line) density functions of the m.l.e. of $\hat{\theta}_2$

We see that the approximation is fairly poor for $n = 10$, but is reasonable for $n = 100$ and very accurate for larger sample sizes.

8.5 Confidence intervals based on asymptotic normality

Now suppose we want to construct a $100(1-\alpha)\%$ confidence interval for any particular element of θ, say θ_j. For suitably large n, we have

$$\hat{\theta_j} \sim N(\theta_j, \gamma_{j,j}), \tag{246}$$

where we $\gamma_{j,j}$ is the $\{j,j\}$ element of $I_E(\theta)^{-1}$. This then gives us an approximate interval as

$$(\hat{\theta_j} - z_{1-\frac{\alpha}{2}}\sqrt{\gamma_{j,j}}, \hat{\theta_j} + z_{1-\frac{\alpha}{2}}\sqrt{\gamma_{j,j}}), \tag{247}$$

with $z_{1-\frac{\alpha}{2}}$ the appropriate percentage point from the standard normal distribution.

We will not be able to calculate this interval in practice, as we do not know the true value of θ, which we would need to evaluate $I_E(\theta)$. Instead, we approximate $I_E(\theta)$ by the observed information matrix

$$I_O(\theta) = \begin{pmatrix} -\frac{\partial^2}{\partial\theta_1^2}l(\theta) & \cdots & -\frac{\partial^2}{\partial\theta_1\partial\theta_d}l(\theta) \\ \vdots & & \vdots \\ -\frac{\partial^2}{\partial\theta_d\partial\theta_1}(\theta) & \cdots & -\frac{\partial^2}{\partial\theta_d^2}l(\theta) \end{pmatrix}, \tag{248}$$

evaluated at $\theta = \hat{\theta}$. Then denoting $\tilde{\gamma}_{i,j}$ as the i,jth element of the inverse of $I_O(\theta)$, we use

$$(\hat{\theta_j} - z_{1-\frac{\alpha}{2}}\sqrt{\tilde{\gamma}_{j,j}}, \hat{\theta_j} + z_{1-\frac{\alpha}{2}}\sqrt{\tilde{\gamma}_{j,j}}), \tag{249}$$

as an approximate confidence interval. Since we know that $\hat{\theta} \to \theta$ as $n \to \infty$, with probability 1, we would expect $I_O(\theta)$ to be similar to $I_E(\theta)$ for large sample sizes.

An example application of this approximation method is in obtaining confidence intervals for parameters in generalised linear models. For example, suppose we are modelling the relationship between a binary response variable X and covariates $\mathbf{z}$ via

$$P(X = 1|\theta) = \frac{\exp(\mathbf{z}^T\theta)}{1 + \exp(\mathbf{z}^T\theta)}, \tag{250}$$

and wish to obtain confidence intervals for the elements of θ. Although $\hat{\theta}$ will have to be obtained numerically, it is relatively straightforward to obtain approximate CIs based on asymptotic normality, via differentiating the likelihood to obtain $I_O(\theta)$. Note however that the normal approximation may not always be accurate, particularly for small sample sizes.

8.6 Hypothesis testing

Likelihood functions play an important role in hypothesis testing. Hypothesis test can be conducted using **likelihood ratios**.

8.6.1 Simple hypotheses and the Neymann-Pearson Lemma

Suppose we wish to test simple hypotheses of the form

$$\begin{aligned} H_0 &: \theta = \theta_0, \\ H_1 &: \theta = \theta_1. \end{aligned}$$

The likelihood ratio test takes the form reject H_0 if

$$\frac{L(\theta_1; x)}{L(\theta_0; x)} > k, \tag{251}$$

for some critical value k. Values of the ratio $\frac{L(\theta_1;x)}{L(\theta_0;x)}$ greater than 1 imply that the data are 'more probable' under H_1 than H_0, and so it is intuitively sensible that we would reject H_0 if the ratio was sufficiently large. For test of size α, the value k must be chosen such that

$$P\left(\frac{L(\theta_1; x)}{L(\theta_0; x)} > k | H_0 \text{ true}\right) = \alpha, \tag{252}$$

i.e., the probability of rejecting H_0 when H_0 is true is α. The Neyman-Pearson Lemma states that this test is optimal, in the sense that for a given α it maximises the power, i.e. the probability of rejecting H_0 when H_0 is false.

8.6.2 Composite hypotheses and the generalised likelihood ratio test

Now consider hypotheses of the form

$$\begin{aligned} H_0 &: \theta \in \Theta_0, \\ H_1 &: \theta \in \Theta_1. \end{aligned}$$

This can be tested with generalised likelihood ratio (GLR) test, which takes the form reject H_0 if

$$\lambda = \frac{\sup_{\theta \in \Theta_1} L(\theta; x)}{\sup_{\theta \in \Theta_0} L(\theta; x)} > k, \tag{253}$$

again, with k chosen such that

$$P\left(\lambda > k | H_0 \text{ true}\right) = \alpha. \tag{254}$$

Though this intuitively sensible, the test has no optimality, with regard to power. A test is **uniformly most powerful** if it maximises $P(\text{reject } H_0 | H_1 \text{ true})$ for all $\theta \in \Theta_1$. If the hypotheses take the form

$$\begin{aligned} H_0 &: \theta = \theta_0, \\ H_1 &: \theta \neq \theta_0, \end{aligned}$$

then no hypothesis test will be uniformly most powerful.

Standard hypothesis tests such as the t-test and the F-test can be derived from the GLR test.

8.6.3 Asymptotic distribution of the GLR test statistic

Let θ be a vector of k parameters, and write $\theta = (\theta_r, \theta_k)$, where θ_r is a subvector of r parameters. Now consider a hypothesis of the form

$$\begin{aligned} H_0 &: \theta_r = \theta_0, \\ H_1 &: \theta_r \neq= 0. \end{aligned}$$

We write the GLR test statistic as

$$\lambda = \frac{L(\theta_0, \hat{\theta^*}_s; \mathbf{x})}{L(\hat{\theta}; \mathbf{x})}, \tag{255}$$

where $\hat{\theta^*}_s$ maximises the likelihood subject to the constraint $\theta_r = \theta_0$, and $\hat{\theta}$ is the usual, unconstrained, m.l.e. It is then possible to prove that as the sample size tends to infinity, the distribution of $-2\log\lambda$ tends to the χ^2_r distribution, when H_0 is true. Hence for a test of size α, we would reject H_0 if $-2\log\lambda$ is greater than the $100(1-\alpha)$ percentile of the χ^2_r distribution.

All the techniques described in this section involve being able to maximise the likelihood function $L(\theta)$ or log-likelihood $l(\theta)$. In simple cases, this can be done analytically. If this is not possible, numerical optimisation techniques can be used. In the next section, we will consider a computational technique for a particular class of problems.

9 Maximum likelihood estimation using the E-M algorithm

The E-M (Expectation-Maximisation) algorithm is a technique used for maximising likelihood functions when the data is in some sense incomplete. It may be the case that some observations are censored, or missing altogether, or alternatively there may be additional, unrecorded information that may greatly simplify the analysis. We will look at examples of both scenarios.

Note that in some situations, missing data may simply be ignored. Consider for example a survey which asks for people to state their incomes, and suppose it is discovered that some members of the survey have not responded. If it is the case that the reason for non-response is entirely unrelated to income, then we could ignore the fact that some data is missing. Alternatively, if it is believed that the likelihood of someone replying depends on their income level, then if we simply ignore the missing values then our data may be biased.

9.1 Motivating example

Suppose we have random variables $X_1, \ldots, X_{n+k}$ independently observed from an $Exponential(\theta)$ distribution, with some observations censored at different times t_j. (For convenience suppose that $X_1, \ldots, X_n$ will be observed, $X_{n+1}, \ldots, X_{n+k}$ are censored.)

Given data $X_i = x_i$ for $i = 1, \ldots, n$ and $X_i \geq t_i$ for $i = n+1, \ldots, n+k$, the likelihood function is

$$f(\mathbf{x}|\theta) = \prod_{i=1}^{n} \theta \exp(-\theta x_i) \prod_{i=n+1}^{n+k} P(X_i \geq t_i) \quad (256)$$

$$= \prod_{i=1}^{n} \theta \exp(-\theta x_i) \prod_{i=n+1}^{n+k} \exp(-\theta t_i) \quad (257)$$

If there were no censored observations, the likelihood would be given by

$$f(\mathbf{x}|\theta) = \prod_{i=1}^{n+k} \theta \exp(-\theta x_i), \quad (258)$$

with m.l.e given by $\hat{\theta} = (n+k)/\sum_{i=1}^{n+k} x_i$. For the purpose of this example we'll suppose we cannot find the m.l.e from the censored likelihood function analytically, i.e., we'll suppose that we don't know how to maximise (257), but we do know how to maximise (258).

The key idea behind the E-M algorithm is to consider what additional data would simplify the analysis.

In the censored exponential example, we *augment* the data with the values of the censored observations, $x_{n+1}, \ldots, x_{n+k}$. We can carry out this augmentation by calculating the expected values of the censored observations, conditional on θ:

If $X_j \sim Exp(\theta)$, then

$$E(X_j|X_j > t_j) = t_j + \frac{1}{\theta}, \quad (259)$$

due to the lack of memory property of the exponential distribution. The E-M algorithm in this case can be summarised as follows:

1. Start with an initial estimate θ_{old} of the m.l.e.
2. Set $x_j = t_j + \theta_{old}^{-1}$ for $j = n+1, \ldots, n+k$. (expectation)
3. Re-estimate the m.l.e as $\theta_{new} = \frac{n+k}{\sum_{i=1}^{n+k} x_i}$. (maximisation)
4. Return to step 2, replacing θ_{old} with θ_{new}. Repeat until convergence.

9.2 The general framework

In general, we can't simply plug in expectations of missing values. We must look at how the data enter the function being maximised.

We observe data $X = x$. Suppose we then wish to maximise the log-likelihood

$$\log f(x|\theta) \tag{260}$$

but cannot do so directly. We then consider additional data Y, such that it would be easier to maximise the **augmented log-likelihood**

$$\log f(x, y|\theta) \tag{261}$$

if we knew that $Y = y$. Of course, we don't know what Y is, so somehow we are going to have to maximise

$$\log f(x, Y|\theta). \tag{262}$$

The way we get round this problem is to take the *expectation* of $\log f(x, Y|\theta)$ with respect to Y:

$$E_Y\{\log f(x, Y|\theta)\} = \int \log f(x, y|\theta) f_Y(y) dy \tag{263}$$

and find θ to maximise this expected log-likelihood. The final difficulty is that without knowing θ, we don't know what the density of Y is. Consequently, we simply choose a value θ_{old}, condition on $\theta = \theta_{old}$ and replace $f_Y(y)$ by $f_{Y|x,\theta_{old}}(y|\theta_{old}, x)$. The value θ_{old} will be a guess at the m.l.e of θ in (260) So given this guessed value θ, we then find a new θ to maximise

$$\begin{aligned} Q(\theta|\theta_{old}) &= E_{old}\{\log f(x, Y|\theta_{new})|x, \theta_{old}\} & (264) \\ &= \int \log f(x, y|\theta_{new}) f_{Y|x,\theta_{old}}(y|x, \theta_{old}) dy & (265) \end{aligned}$$

Once we have found θ to maximise (265), we can use this as a new guess for the m.l.e. in (260), and start again. The E-M algorithm can then be stated as follows:

1. Choose a starting value θ_{old}.
2. Expectation: calculate $Q(\theta|\theta_{old})$.
3. Maximisation: choose θ_{new} to be the value of θ that maximises $Q(\theta|\theta_{old})$.
4. Replace θ_{old} by θ_{new} and repeat until convergence.

It can be proved that

$$f(x|\theta_{new}) \geq f(x|\theta_{old}), \tag{266}$$

with equality only achieved at a maximum (though not necessarily a global maximum) of $f(x|\theta)$.

9.3 Example re-visited

We have data $\mathbf{x} = \{x_1, \ldots, x_n\}$ and $\mathbf{t} = \{t_{n+1}, \ldots, t_{n+k}\}$
We augment the data with $Y = \{X_{n+1}, \ldots, X_{n+k}\}$
Then

$$\log f(\mathbf{x}, Y|\theta) = (n+k)\log\theta - \theta\left(\sum_{i=1}^{n} x_i + \sum_{i=n+1}^{n+k} X_i\right) \tag{267}$$

To calculate $Q(\theta|\theta_{old})$ we must take the expectation of (267) with respect to $Y = \{X_{n+1}, \ldots, X_{n+k}\}$, where Y has density function $f(Y|\mathbf{x}, \mathbf{t}, \theta_{old})$. Doing so gives

$$\begin{aligned} Q(\theta|\theta_{old}) &= (n+k)\log\theta - \theta\left\{\sum_{i=1}^{n} x_i + \sum_{i=n+1}^{n+k} E(X_i|X_i > t_j, \theta_{old})\right\} &(268)\\ &= (n+k)\log\theta - \theta\left\{\sum_{i=1}^{n} x_i + \sum_{i=n+1}^{n+k} (t_i + \theta_{old}^{-1})\right\}. &(269)\end{aligned}$$

The value of θ that maximises this function is then given by

$$\theta_{new} = \frac{n+k}{\sum_{i=1}^{n} x_i + \sum_{i=n+1}^{n+k}(t_i + \theta_{old}^{-1})}. \tag{270}$$

9.4 How it works

We'll now consider in a little more detail how the E-M algorithm works. To make things slightly easier to understand, we will work with discrete data X and Y (though you should be able to see how this will generalise to the continuous case). The algorithm produces a sequence of values $\theta_1, \theta_2, \theta_3 \ldots$ which will converge to a local maximum of $P(x|\theta)$. Here we will simply look at why $P(x|\theta_{i+1}) \geq P(x|\theta_i)$.

Denote $l(\theta)$ to be the log-likelihood $\log P(x|\theta)$. Suppose the current estimate of the m.l.e is θ_i and consider choosing θ_{i+1}. We have

$$\begin{aligned} l(\theta) - l(\theta_i) &= \log P(x|\theta) - \log P(x|\theta_i) &(271)\\ &= \log\left(\frac{P(x|\theta)}{P(x|\theta_i)}\right). &(272)\end{aligned}$$

Now we introduce the 'missing' data Y:

$$l(\theta) - l(\theta_i) = \log\left(\frac{\sum_y P(x|y,\theta)P(y|\theta)}{P(x|\theta_i)}\right). \tag{273}$$

Next, we make use of Jensen's inequality, which states that if $\sum_j w_j = 1$, then

$$\log\sum_j w_j z_j \geq \sum_j w_j \log z_j. \tag{274}$$

The trick is then to make the RHS of (273) look like $\log \sum_j w_j z_j$, with $\sum_j w_j = 1$. We can do this by writing

$$l(\theta) - l(\theta_i) = \log \left(\sum_y \frac{P(x|y,\theta)P(y|\theta)P(y|x,\theta_i)}{P(x|\theta_i)P(y|x,\theta_i)} \right). \tag{275}$$

Now, looking at Jensen's equality, remembering that y is discrete we see that we have $w_j = P(y_j|x,\theta_i)$, and since the probabilities sum to 1, we have

$$l(\theta) - l(\theta_i) \geq \sum_y P(y|x,\theta_i) \log \left(\frac{P(x|y,\theta)P(y|\theta)}{P(x|\theta_i)P(y|x,\theta_i)} \right), \tag{276}$$

and so expanding the log term, we have

$$l(\theta) \geq l(\theta_i) + \sum_y P(y|x,\theta_i) \log\{P(x|y,\theta)P(y|\theta)\} - \sum_y P(y|x,\theta_i) \log\{P(x|\theta_i)P(y|x,\theta_i)\}. \tag{277}$$

Note that the terms $l(\theta_i)$ and $\sum_y P(y|x,\theta_i) \log\{P(x|\theta_i)P(y|x,\theta_i)\}$ are not functions of θ, and so are constant for all possible values of θ. This means that we should look for θ_{i+1} to maximise the term

$$Q(\theta|\theta_i) = \sum_y P(y|x,\theta_i) \log\{P(x|y,\theta)P(y|\theta)\} \tag{278}$$

By choosing θ to maximise $Q(\theta|\theta_i)$, we must be maximising the difference $l(\theta) - l(\theta_i)$. Finally, note that since

$$P(x|y,\theta)P(y|\theta) = P(x,y|\theta), \tag{279}$$

we have

$$Q(\theta|\theta_i) = \sum_y P(y|x,\theta_i) \log\{P(x,y|\theta)\}, \tag{280}$$

the expectation of the complete data log-likelihood, where the missing data Y has probability mass function $P(y|x,\theta_i)$.

9.5 Exercise: multinomially distributed data

In a dataset, 197 animals are distributed multinomially into four categories. The observed data are given by

$$\mathbf{x} = (x_1, x_2, x_3, x_4) = (125, 18, 20, 34). \tag{281}$$

The probabilities of membership in each category for any animal are given by a genetic model:

$$\left\{ \frac{1}{2} + \frac{\pi}{4}, \frac{1}{4}(1-\pi), \frac{1}{4}(1-\pi), \frac{\pi}{4} \right\}, \tag{282}$$

for some unknown value of $\pi \in (0,1)$. The likelihood function for π is then given by

$$f(\mathbf{x}|\pi) = \frac{(x_1+x_2+x_3+x_4)!}{x_1!x_2!x_3!x_4!} \left(\frac{2+\pi}{4} \right)^{x_1} \left(\frac{1-\pi}{4} \right)^{x_2} \left(\frac{1-\pi}{4} \right)^{x_3} \left(\frac{\pi}{4} \right)^{x_4}. \tag{283}$$

To maximise this likelihood with the E-M algorithm, we again have to consider what 'missing' data might simplify the analysis. In this case, we can convert the likelihood into the more familiar binomial form $\pi^n(1-\pi)^m$ by supposing that there are in fact five categories instead of four, so that the complete data would be given by $\mathbf{y} = (y_1, y_2, y_3, y_4, y_5)$, with $x_1 = y_1 + y_2$, $x_2 = y_3$, $x_3 = y_4$ and $x_4 = y_5$. We then suppose that the probabilities for the five new categories are

$$\left\{\frac{1}{2}, \frac{\pi}{4}, \frac{1}{4}(1-\pi), \frac{1}{4}(1-\pi), \frac{\pi}{4}\right\}, \tag{284}$$

This gives the likelihood

$$\begin{aligned} f(\mathbf{y}|\pi) &= \frac{(y_1+y_2+y_3+y_4+y_5)!}{y_1!y_2!y_3!y_4!y_5}\left(\frac{1}{2}\right)^{y_1}\left(\frac{\pi}{4}\right)^{y_2}\left(\frac{1-\pi}{4}\right)^{y_3}\left(\frac{1-\pi}{4}\right)^{y_4}\left(\frac{\pi}{4}\right)^{y_5} && (285)\\ &\propto \pi^{y_2+y_5}(1-\pi)^{y_3+y_4}. && (286)\end{aligned}$$

Given a current estimate π_{old} of the m.l.e., how would you apply one iteration of the EM algorithm to derive an improved estimate?

9.6 The E-M algorithm within the exponential family

When presenting the general framework for the E-M algorithm, it was commented that we can't simply plug in the expected value of the missing data (given a guessed value θ_{old}) into the likelihood to get the complete data log-likelihood, though in practice, this is often what happens. More accurately, when the complete data likelihood $f(x, y|\theta)$ is a member of the **exponential family**, we shall see shortly that the E-M algorithm takes on a simplified form, that involves replacing **sufficient statistics** in the complete data log-likelihood by their expectations (conditional on the guessed value θ_{old}).

9.6.1 The exponential family

Let θ be a k dimensional vector $\theta = \{\theta_1, \ldots, \theta_k\}$. A density function $f(x|\theta)$ is said to belong to the **exponential family** if we can write

$$f(x|\theta) = \exp\{\sum_{i=1}^{k} A_i(\theta)B_i(x) + C(x) + D(\theta)\}. \tag{287}$$

Some distributions that we commonly work with are members of the exponential family. For example:

- Normal distribution : $\theta = (\mu, \sigma^2)$.

$$\begin{aligned} A_1(\theta) &= \sigma^{-2} && (288)\\ A_2(\theta) &= \mu/\sigma^2 && (289)\\ B_1(x) &= -x^2/2 && (290)\\ B_2(x) &= x && (291)\\ C(x) &= 0 && (292)\\ D(\theta) &= -(\log(2\pi\sigma^2) + \mu^2/\sigma^2)/2 && (293) \end{aligned}$$

- Exponential distribution

$$\begin{aligned} A_1(\theta) &= -\theta && (294)\\ B_1(x) &= x && (295)\\ C(x) &= 0 && (296)\\ D(\theta) &= \log\theta && (297) \end{aligned}$$

- Binomial distribution

$$\begin{aligned} A_1(\theta) &= \log\left(\frac{\theta}{1-\theta}\right) && (298)\\ B_1(x) &= x && (299)\\ C(x) &= \log\left({}^nC_x\right) && (300)\\ D(\theta) &= n\log(1-\theta) && (301) \end{aligned}$$

9.6.2 Sufficient statistics

Informally, a sufficient statistic is some function of the data that contains all the relevant information required for estimating a particular parameter. For example, when estimating the mean of a normal distribution, given data $X_1, \ldots, X_n$, the sample mean $\bar{x}$ is a sufficient statistic; knowing the individual values $X_1, \ldots, X_n$ will not give us any additional information about the population mean.

Formally a statistic $S(X)$ given data $X_1, \ldots, X_n$ is a sufficient statistic for a parameter θ if and only if the conditional distribution of $X_1, \ldots, X_n$ given $S(X)$ is independent of θ.

Sufficient statistics can be identified using the *factorization theorem*, which states that a statistic $S(X)$ given data $X_1, \ldots, X_n$ is a sufficient statistic for a parameter θ if and only if the joint density or distribution of $X_1, \ldots, X_n$ can be factored so that

$$f(X_1, \ldots, X_n|\theta) = g(s(X), \theta)h(X_1, \ldots, X_n), \tag{302}$$

where $g(s(X), \theta)$ depends on $s(X)$ and θ only, and $h(X_1, \ldots, X_n)$ does not depend on θ.

Note that whenever a maximum likelihood estimator exists, it is always a function of a sufficient statistic. For the purposes of this course, you will find the simplest way to identify a sufficient statistic is simply to obtain the maximum likelihood estimator.

9.6.3 The E-M algorithm for the exponential family likelihoods

Given all the data (X, Y), (i.e. both observed and missing), define $S(X, Y)$ to be the sufficient statistic for the unknown parameter θ. The $E - M$ algorithm in this case is summarised as follows:

1. Start with an initial value θ_{old}.
2. Let $S_{old} = E\{S(X, Y)|X = x, \theta_{old}\}$.
3. Choose θ_{new} to be the solution θ to the equation
$$E\{S(X, Y)|\theta\} = S_{old}.$$
4. Replace θ_{old} by θ_{new} and repeat until convergence.

We'll return to the first example with censored exponential observations, and see how this version of the algorithm fits in.

The complete data (both fully observed and censored observations is given by $\{X = (X_1, \ldots, X_n), Y = (X_{n+1}, \ldots, X_{n+k}), \mathbf{t} = (t_{n+1}, \ldots, t_{n+k})\}$. The sufficient statistic for θ is given by

$$S(X, Y) = X_1 + \ldots + X_{n+k}. \tag{303}$$

So, we begin with an initial guess θ_{old}. Now we compute

$$\begin{aligned} S_{old} &= E\{S(X, Y)|X_1 = x_1, \ldots, X_n = x_n, \mathbf{t}, \theta_{old}\} && (304)\\ &= x_1 + \ldots, x_n + t_{n+1} + \frac{1}{\theta_{old}} + \ldots + t_{n+k} + \frac{1}{\theta_{old}}. && (305)\end{aligned}$$

In computing S_{old}, we can than see that we have simply plugged in the expectations of the missing values, conditional on a guessed value θ_{old}, into the expression for the sufficient statistic $S(X, Y)$. Now,

$$E\{S(X, Y)|\theta\} = \frac{1}{\theta} + \ldots + \frac{1}{\theta} = \frac{n + k}{\theta}. \tag{306}$$

Finally, we find θ_{new} by solving

$$E\{S(X, Y)|\theta\} = S_{old}, \tag{307}$$

i.e., by solving

$$\frac{n + k}{\theta} = x_1 + \ldots, x_n + t_{n+1} + \frac{1}{\theta_{old}} + \ldots + t_{n+k} + \frac{1}{\theta_{old}}. \tag{308}$$

This gives us

$$\theta_{new} = \frac{n + k}{x_1 + \ldots, x_n + t_{n+1} + \frac{1}{\theta_{old}} + \ldots + t_{n+k} + \frac{1}{\theta_{old}}} \tag{309}$$

9.6.4 How the E-M algorithm works in the exponential family case

We will begin with a couple of general points. Let x have likelihood function $f(x|\theta)$, and define $\phi = h(\theta)$, where the transformation $h(\theta)$ is bijective (one-to one and onto). We can then write the likelihood instead as $f(x|\phi)$. This re-parameterisation does not affect the distribution of x, i.e

$$f(x|\theta = \theta_1) = f(x|\phi = h(\theta_1)) \tag{310}$$

Additionally, if $\hat{\phi}$ maximises $f(x|\phi)$, then $\hat{\theta} = h^{-1}(\hat{\phi})$ will maximise $f(x|\theta)$.

For this section we will modify the notation slightly to simplify the algebra. We will use x_{obs} to denote the observed data, x_{mis} to denote the missing data, and y to be the complete data, so that $y = (x_{obs}, x_{mis})$. We will write the full likelihood as $f(y|\phi)$, and the likelihood for the observed data as

$$g(x_{obs}|\phi) = \int f(y|\phi)\, dx_{mis}. \tag{311}$$

The objective is then to find ϕ to maximise $g(x_{obs}|\phi)$. As before $f(y|\phi)$ is a member of the exponential family. However to make life slightly easier later on in this section, we are going to write this density function in a slightly different form to (287); we will write it in the **regular exponential family form**:

$$f(y|\phi) = \frac{b(y)\exp\{\phi t(y)\}}{a(\phi)}. \tag{312}$$

Written in this form, ϕ is the **natural** parameter in the distribution of y. Given a particular distribution, the natural parameter ϕ is typically a non-linear transformation of the **conventional** parameter θ. For example, suppose y has a binomial distribution with probability of success θ. The natural parameter ϕ is then given by

$$\phi = \log\left(\frac{\theta}{1-\theta}\right) \tag{313}$$

We then have

$$\theta = \frac{e^{\phi}}{1+e^{\phi}}, \tag{314}$$

and in the regular exponential family form, $t(y) = y$, $b(y) = {}^nC_y$ and

$$a(\phi) = \left(1 - \frac{e^{\phi}}{1-e^{\phi}}\right)^{-n}. \tag{315}$$

In this section, we will show how the E-M algorithm gives the m.l.e for the natural parameter ϕ, since we know that this can then be transformed to give the m.l.e of the conventional parameter θ.

Considering the log-likelihood $\log f(y|\phi)$, we can see that the sufficient statistic for ϕ in the complete data log-likelihood will be $t(y)$. Consequently, we can write the E-M algorithm here as

1. Start with an intial guess ϕ_{old}.

2. Calculate $t_{old}(y) = E\{t(y)|x_{obs}, \phi_{old}\}$
3. Set ϕ_{new} to be the solution of
$$E\{t(y)|\phi\} = t_{old}(y). \tag{316}$$
4. Take ϕ_{new} as the new guess for the m.l.e, and repeat until convergence, i.e, until $\phi_{new} = \phi_{old}$.

Firstly, define
$$l(\phi) = \log g(x_{obs}|\phi). \tag{317}$$
Now define $k(y|x_{obs}, \phi)$ to be the conditional density of y given both x_{obs} and ϕ. From the definition of conditional densities, we have
$$k(y|x_{obs}, \phi) = \frac{f(y|\phi)}{g(x_{obs}|\phi)}. \tag{318}$$
It then follows that
$$l(\phi) = \log f(y|\phi) - \log k(y|x_{obs}, \phi) \tag{319}$$
The next step is to note that from (311) and (312), we have
$$g(x_{obs}|\phi) = \frac{1}{a(\phi)} \int b(y) \exp\{\phi t(y)\}\, dx_{mis}, \tag{320}$$
(remembering that $y = (x_{obs}, x_{mis})$). Now, if we substitute this expression for g back into (318), we get
$$k(y|x_{obs}, \phi) = \frac{b(y) \exp\{\phi t(y)\}}{a(\phi|x_{obs})}, \tag{321}$$
with
$$a(\phi|x_{obs}) = \int b(y) \exp\{\phi t(y)\}\, dx_{mis}. \tag{322}$$
Now, substituting (312) and (321) into (319), we get
$$\begin{aligned} l(\phi) &= \log \left\{ \frac{b(y) \exp\{\phi t(y)\}}{a(\phi)} \right\} - \log \left\{ \frac{b(y) \exp\{\phi t(y)\}}{a(\phi|x_{obs})} \right\} && (323) \\ &= -\log a(\phi) + \log a(\phi|x_{obs}). && (324) \end{aligned}$$
Since (312) is a density function in y, it must integrate to 1 (when integrated with respect to y), and so we can deduce that
$$a(\phi) = \int b(y) \exp\{\phi t(y)\}\, dy. \tag{325}$$
Now differentiate both $a(\phi)$ and $a(\phi|x_{obs})$ with respect to ϕ:
$$\begin{aligned} \frac{\partial}{\partial \phi} \log a(\phi) &= \frac{1}{a(\phi)} \frac{\partial}{\partial \phi} a(\phi) && (326) \\ &= \frac{1}{a(\phi)} \int t(y)\, b(y) \exp\{\phi t(y)\}\, dy && (327) \\ &= \int t(y) f(y|\phi)\, dy && (328) \\ &= E\{t(y)|\phi\}, && (329) \end{aligned}$$

and

$$\begin{aligned}
\frac{\partial}{\partial\phi}\log a(\phi|x_{obs}) &= \frac{1}{a(\phi|x_{obs})}\frac{\partial}{\partial\phi}a(\phi|x_{obs}) && (330)\\
&= \frac{1}{a(\phi|x_{obs})}\int t(y)\,b(y)\exp\{\phi t(y)\}\,dx_{mis} && (331)\\
&= \int t(y)k(y|x_{obs},\phi)\,dy && (332)\\
&= E\{t(y)|x_{obs},\phi\} && (333)
\end{aligned}$$

Hence,

$$\frac{\partial}{\partial\phi}l(\phi) = E\{t(y)|x_{obs},\phi\} - E\{t(y)|\phi\} \tag{334}$$

Now, if $\hat{\phi}$ is a maximum of the log-likelihood $l(\phi)$, we must have

$$\frac{\partial}{\partial\phi}l(\phi) = 0 \tag{335}$$

at $\phi = \hat{\phi}$. Additionally, the E-M algorithm will claim to have found a maximum when $\phi_{new} = \phi_{old}$. This means that the solution of

$$E\{t(y)|\phi\} = t_{old}(y) \tag{336}$$

is $\phi = \phi_{old}$, i.e.,

$$E\{t(y)|x_{obs},\phi_{old}\} = E\{t(y)|\phi_{old}\}, \tag{337}$$

and so (334) is zero as required.

9.7 Mixture distributions and the E-M algorithm

The distribution of some random variable X in a population can be represented by a *mixture distribution*, when the population can be split into subgroups, with each subgroup having a different distribution for X. For example, the distribution of heights of twenty-year olds could be modelled as a mixture distribution; a combination of two distributions, one representing males and one representing females. In ANOVA type scenarios, we may for example have data $x_{i,j}$ corresponding to the jth member of the ith group. A model for the data would then be

$$x_{i,j} = \mu_i + \varepsilon_{i,j}, \tag{338}$$

with $\varepsilon_{i,j} \sim N(0,\sigma^2)$. If we were to now consider the distribution of a future observation x, but *we do not know which group the observation will come from*, we might then consider the distribution of x to be a mixture of normal distributions, as the expectation of x varies according to group membership.

9.7.1 Example: mixture of two normal distributions

Let $X_1, \ldots, X_n$ be independent observations drawn from a mixture of $N(\mu, \sigma^2)$ with probability ϕ, and $N(\nu, \tau^2)$ with probability $1 - \phi$. Both variances σ^2 and τ^2 are known. Denote the observed values of $X_1, \ldots, X_n$ by $\mathbf{x} = (x_1, \ldots, x_n)$. Use the E-M algorithm to obtain the m.l.e. of μ, ν and ϕ.

What additional data would simplify the analysis?

In this case, knowing which normal distribution each observation X_i came from.

For each observation X_i we introduce an indicator variable Y_i, where Y_i identifies which normal distribution X_i was actually drawn from:

$$Y_i = \begin{cases} 1 & \text{if } X_i \text{ drawn from } N(\mu, \sigma^2) \\ 0 & \text{if } X_i \text{ drawn from } N(\nu, \tau^2) \end{cases} \tag{339}$$

Writing $\mathbf{y} = (y_1, \ldots, y_n)$ as the values of the missing data $Y_1, \ldots, Y_n$. We will first derive the complete data likelihood function. We need to consider the joint density of $\mathbf{x}$ and $\mathbf{y}$, but we first write

$$f(\mathbf{x}, \mathbf{y}|\phi, \mu, \nu) = f(\mathbf{x}|\mathbf{y}, \phi, \mu, \nu) f(\mathbf{y}|\phi, \mu, \nu). \tag{340}$$

The two densities on the RHS are easier to work with; $f(\mathbf{y}|\phi, \mu, \nu)$ will be in the form of the standard binomial likelihood function, and conditional on $\mathbf{y}$, we know which of the two normal distributions each element of $\mathbf{x}$ comes from, so $f(\mathbf{x}|\mathbf{y}, \phi, \mu, \nu)$ will simply be a product of normal density functions. We have

$$f(\mathbf{y}|\phi, \mu, \nu) = \phi^{\sum y_i}(1-\phi)^{n - \sum y_i} \tag{341}$$

$$f(\mathbf{x}|\mathbf{y}, \phi, \mu, \nu) \propto \exp\left\{ -\frac{1}{2\sigma^2} \sum_{y_i=1} (x_i - \mu)^2 - \frac{1}{2\tau^2} \sum_{y_i=0} (x_i - \nu)^2 \right\}, \tag{342}$$

so the complete data log-likelihood is given by

$$\begin{aligned} \log f(\mathbf{x}, \mathbf{y}|\phi, \mu, \nu) &= K + \sum y_i \log \phi + (n - \sum y_i) \log(1-\phi) - \frac{1}{2\sigma^2} \sum_{y_i=1} (x_i - \mu)^2 \\ &\quad - \frac{1}{2\tau^2} \sum_{y_i=0} (x_i - \nu)^2, \end{aligned} \tag{343}$$

for some constant K. This density is a member of the exponential family and the sufficient statistics for $\theta = (\phi, \mu, \nu)$ are $S = (\sum Y_i, \sum_{Y_i=1} X_i, \sum_{Y_i=0} X_i)$. This follows from the factorization theorem. We can write

$$\begin{aligned} \log f(\mathbf{x}, \mathbf{y}|\phi, \mu, \nu) &= K + \sum y_i \log \phi + (n - \sum y_i) \log(1-\phi) - \frac{1}{2\sigma^2} \sum_{y_i=1} (\mu^2 - 2\mu x_i) \\ &\quad - \frac{1}{2\tau^2} \sum_{y_i=0} (\nu^2 - 2\nu x_i) - \frac{1}{2\sigma^2} \sum_{y_i=1} x_i^2 - \frac{1}{2\tau^2} \sum_{y_i=0} x_i^2. \end{aligned} \tag{344}$$

Referring back to the notation in (302), we have

$$\begin{aligned}\log g(s(X,Y),\phi,\mu,\nu) &= \sum y_i \log\phi + (n-\sum y_i)\log(1-\phi) \\ &\quad -\frac{1}{2\sigma^2}\sum_{y_i=1}(\mu^2-2\mu x_i) - \frac{1}{2\tau^2}\sum_{y_i=0}(\nu^2-2\nu x_i) \qquad (345)\\ \log h(X,Y) &= -\frac{1}{2\sigma^2}\sum_{y_i=1}x_i^2 - \frac{1}{2\tau^2}\sum_{y_i=0}x_i^2. \qquad (346)\end{aligned}$$

Starting with an initial guess $\theta_{old} = (\phi_{old}, \mu_{old}, \nu_{old})$, we must now calculate $S_{old} = E\{S(X,Y)|X = \mathbf{x}, \theta_{old}\}$.
Consider first $E(Y_i|x_i, \theta_{old})$. We have

$$\begin{aligned}E(Y_i|x_i,\theta_{old}) &= P(Y_i=1|x_i,\theta_{old}) \qquad (347)\\ &= \frac{P(Y_i=1|\theta_{old})f(x_i|Y_i=1,\theta_{old})}{f(x_i|\theta_{old})} \qquad (348)\end{aligned}$$

From Bayes' theorem. Now

$$\begin{aligned}P(Y_i=1|\theta_{old}) &= \phi_{old} \qquad (349)\\ f(x_i|Y_i=1,\theta_{old}) &= \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{1}{2\sigma^2}(x_i-\mu_{old})^2\right\}. \qquad (350)\end{aligned}$$

Deriving the density $f(x_i|\theta_{old})$ is a little harder. We again use the same conditioning idea as in (340): conditional on the value of Y_i, we know the density of X_i. We write

$$f(x_i|\theta_{old}) = f(x_i|Y_i=1,\theta_{old})P(Y_i=1|\theta_{old}) + f(x_i|Y_i=0,\theta_{old})P(Y_i=0|\theta_{old}), \qquad (351)$$

i.e. we integrate out Y_i from the joint distribution of X_i and Y_i. We can now write

$$\begin{aligned}f(x_i|\theta_{old}) &= \phi_{old}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{1}{2\sigma^2}(x_i-\mu_{old})^2\right\} \\ &\quad +(1-\phi_{old})\frac{1}{\sqrt{2\pi\tau^2}}\exp\left\{-\frac{1}{2\tau^2}(x_i-\nu_{old})^2\right\} \qquad (352)\end{aligned}$$

and we write that $P(Y_i = 1|x_i, \theta_{old}) = p_i$, where p_i is obtained by substituting these results into (348). Now we need

$$E\{\sum_{Y_i=1} X_i|X = \mathbf{x}, \theta_{old}\}.$$

Firstly, note that

$$\sum_{Y_i=1} X_i = \sum_{i=1}^{n} Y_i X_i. \qquad (353)$$

Now

$$\begin{aligned}E\{Y_iX_i|Xi = x_i,\theta_{old}\} &= x_iE(Y_i|Xi=x_i,\theta_{old}) \qquad (354)\\ &= p_ix_i \qquad (355)\end{aligned}$$

Additionally,

$$\sum_{Y_i=0} X_i = \sum_{i=1}^{n} (1 - Y_i) X_i, \tag{356}$$

and so

$$\begin{aligned} E\{(1 - Y_i)X_i | Xi = x_i, \theta_{old}\} &= x_i E\{(1 - Y_i)|Xi = x_i, \theta_{old}\} & (357) \\ &= (1 - p_i)x_i & (358) \end{aligned}$$

We now have the expected value of the sufficient statistics conditional on θ_{old} and $X = \mathbf{x}$:

$$S_{old} = (\sum p_i, \sum p_i x_i, \sum (1 - p_i)x_i). \tag{359}$$

Next, we need to derive

$$E\{S(X, Z)|\theta\}. \tag{360}$$

Firstly,

$$E\{\sum Y_i | \theta = (\phi, \mu, \nu)\} = n\phi. \tag{361}$$

Also,

$$\begin{aligned} E(Y_i X_i|\theta) &= \int 1.x.P(Y_i = 1|\theta) f_{X_i}(x|\theta, Y_i = 1)dx \\ &\quad + \int 0.x.P(Y_i = 0|\theta) f_{X_i}(x|\theta, Y_i = 0)dx & (362) \\ &= \phi \int x f_{X_i}(x|\theta, Y_i = 1)dx & (363) \\ &= \phi\mu & (364) \end{aligned}$$

Similarly,

$$E\{(1 - Y_i)X_i|\theta\} = (1 - \phi)\nu \tag{365}$$

This gives us

$$\begin{aligned} n\phi &= \sum p_i & (366) \\ n\phi\mu &= \sum p_i x_i & (367) \\ n(1 - \phi)\nu &= \sum (1 - p_i)x_i & (368) \end{aligned}$$

So, for θ_{new} we have

$$\begin{aligned} \phi_{new} &= \frac{\sum p_i}{n} & (369) \\ \mu_{new} &= \frac{\sum p_i x_i}{\sum p_i} & (370) \\ \nu_{new} &= \frac{\sum (1 - p_i)x_i}{\sum (1 - p_i)} & (371) \end{aligned}$$

9.8 Exercise

Data $X_1, \ldots, X_n$ are independent and identically distributed, with the distribution of each X_i being a mixture of Poisson distributions, i.e.,

$$\begin{aligned} X_i &\sim Poisson(\lambda) \text{ with probability } p, \\ X_i &\sim Poisson(\gamma) \text{ with probability } 1-p \end{aligned}$$

The parameters λ, γ and p are all unknown, and for any X_i, it is not known which of the two Poisson distributions X_i was actually sampled from. Given initial guesses for the maximum likelihood estimates of λ, γ and p given $X_1, \ldots, X_n$, applying one iteration of the E-M algorithm, what are the new estimates of the maximum likelihood estimates of λ, γ and p?

10 Profile Likelihood

In this section we will be concentrating on handling multivariate likelihood functions. We suppose a random variable X has density function f with a vector of parameters $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_d\}$. One possible scenario is that given data $\mathbf{x} = (x_1, \ldots, x_n)$, we are only interested in making inferences about a *subset* of the unknown parameters. We partition $\boldsymbol{\theta}$ into $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ with $\boldsymbol{\theta}_1$ the parameters of direct interest. $\boldsymbol{\theta}_2$, the parameters not of direct interest are known as **nuisance parameters**. As an example, we could have $X \sim N(\mu, \sigma^2)$ with both μ and σ^2 unknown, though we may only be interested in the mean parameter μ.

In section 8.5 we used asymptotic normality of the m.l.e. to derive approximate confidence intervals. We will now consider an alternative form of the likelihood function which in some cases can produce more accurate confidence intervals. Again, partitioning $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$, the **profile** log-likelihood function for $\boldsymbol{\theta}_1$ is defined by

$$l_p(\boldsymbol{\theta}_1; \mathbf{x}) = \max_{\boldsymbol{\theta}_2} l(\boldsymbol{\theta}; \mathbf{x}). \tag{372}$$

So, to get the profile log-likelihood function for θ_1, you first treat $\boldsymbol{\theta}_1$ as a constant, and find the maximum likelihood estimate $\hat{\boldsymbol{\theta}}_2$ in terms of the data $\mathbf{x}$ *and* $\boldsymbol{\theta}_1$. You then plug in this expression for $\hat{\boldsymbol{\theta}}_2$ into the full log-likelihood $l(\boldsymbol{\theta}; \mathbf{x})$ to get the profile log-likelihood $l_p(\boldsymbol{\theta}_1; \mathbf{x})$. If we have $\boldsymbol{\theta} = (\theta_i, \boldsymbol{\theta}_{-i})$, then plotting $l_p(\theta_i; \mathbf{x})$ would give us the profile of the log-likelihood surface viewed from the θ_i axis.

Clearly, if $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2)$ maximises $l(\boldsymbol{\theta}; \mathbf{x})$, then $\hat{\boldsymbol{\theta}}_1$ will maximise $l_p(\boldsymbol{\theta}_1; \mathbf{x})$ and $\hat{\boldsymbol{\theta}}_2$ will maximise $l_p(\boldsymbol{\theta}_2; \mathbf{x})$. Two reasons for considering profile likelihood are that it can be useful exploratory tool, by allowing you to plot a likelihood $l_p(\theta_i; \mathbf{x})$ for a single parameter θ_i, and that it can be used to derive more accurate confidence intervals.

10.1 Example 1

Suppose we have $X_1, \ldots, X_n \sim N(\mu, \sigma^2)$ independently distributed. The log likelihood for μ and σ^2 (ignoring constant terms) given data $x_1, \ldots, x_n$ is then

$$l(\mu, \sigma^2; \mathbf{x}) = -n \log \sigma - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (x_i - \mu)^2. \tag{373}$$

What are the profile log-likelihood functions $l_p(\mu; \mathbf{x})$ and $l_p(\sigma^2; \mathbf{x})$?

Fixing μ, the MLE of σ^2 is $\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2$. Substituting this back into the full log-likelihood $l(\mu, \sigma^2; \mathbf{x})$, we get

$$l_p(\mu; \mathbf{x}) = -\frac{n}{2} \log\{\frac{1}{n} \sum (x_i - \mu)^2\} - \frac{n}{2}. \tag{374}$$

Fixing σ^2, the MLE of μ is $\bar{x}$. The profile log-likelihood for σ^2 is therefore

$$l_p(\sigma^2; \mathbf{x}) = -n \log \sigma - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (x_i - \bar{x})^2. \tag{375}$$

10.2 Inference using the deviance function

The asymptotic normality of the maximum likelihood estimator can be used to quantify the uncertainty in the MLE $\hat{\boldsymbol{\theta}}$, and hence construct confidence intervals for $\boldsymbol{\theta}$. An alternative approach to deriving a confidence interval is based on the **deviance function**. For an arbitrary value of $\boldsymbol{\theta}$, say $\boldsymbol{\theta}^*$, this is defined as

$$D(\boldsymbol{\theta}^*) = 2\{l(\hat{\boldsymbol{\theta}}; \mathbf{x}) - l(\boldsymbol{\theta}^*; \mathbf{x})\}. \tag{376}$$

Since $\hat{\boldsymbol{\theta}}$ maximises the log-likelihood, this function is always positive. If $D(\boldsymbol{\theta}^*)$ is small, then $l(\boldsymbol{\theta}^*; \mathbf{x})$ must be close to $l(\hat{\boldsymbol{\theta}}; \mathbf{x})$, which suggests that $\boldsymbol{\theta}^*$ is a plausible estimate for the true unknown value of $\boldsymbol{\theta}$. A confidence interval (or more formally a *region* if $\boldsymbol{\theta}$ is a vector) could then be of the form

$$C = \{\boldsymbol{\theta}^* : D(\boldsymbol{\theta}^*) \leq c\}, \tag{377}$$

for some suitable value of c.

With data $x_1, \ldots, x_n$, for sufficiently large n, from section 8.6.3 we know that at the true value of $\boldsymbol{\theta}$,

$$D(\boldsymbol{\theta}) \sim \chi^2_d, \tag{378}$$

(where d is the dimensionality of $\boldsymbol{\theta}$). An approximate $(1 - \alpha)$ confidence region for $\boldsymbol{\theta}$ is then given by

$$C_\alpha = \{\boldsymbol{\theta}^* : D(\boldsymbol{\theta}^*) \leq c_\alpha\}, \tag{379}$$

with c_α the $(1 - \alpha)$ percentage point of the χ^2_d distribution. This approximation is usually more accurate than the asymptotic normality approximation, though it may require greater computational effort.

10.3 Profile likelihood and the deviance function

Profile likelihood can be used to construct a confidence interval for any single parameter θ_i. Firstly, consider the partition $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$, with $\boldsymbol{\theta}_1$ a k-dimensional subset of $\boldsymbol{\theta}$. Now define the **profile deviance**

$$D_p(\boldsymbol{\theta}_1^*) = 2\{l(\hat{\boldsymbol{\theta}}; \mathbf{x}) - l_p(\boldsymbol{\theta}_1^*; \mathbf{x})\}, \tag{380}$$

with $\hat{\boldsymbol{\theta}}$ the maximum likelihood estimator of $\boldsymbol{\theta}$. Based on a sample of size n, with n sufficiently large, it can be shown that at the true value of $\boldsymbol{\theta}_1$,

$$D_p(\boldsymbol{\theta}_1) \sim \chi_k^2. \tag{381}$$

Hence we can obtain a confidence interval for any element θ_i as

$$C_\alpha = \{\theta_i^* : D_p(\theta_i^*) \leq c_\alpha\}, \tag{382}$$

again, with c_α the $(1-\alpha)$ percentage point of the χ_1^2 distribution. This will often be more accurate than the interval

$$\hat{\theta}_i \pm z_{\frac{\alpha}{2}} \sqrt{\psi_{i,i}} \tag{383}$$

stated earlier.

10.4 Example: leukaemia data re-visited

We return to the example in section 7.7.4, with data

$$6^*, 6, 6, 6, 7, 9^*, 10^*, 10, 11^*, 13, 16, 17^*, 19^*, 20^*, 22, 23, 25^*, 32^*, 32^*, 34^*, 35^*,$$

and a * denoting an observation censored at that time. We have the Weibull model

$$f_T(t) = \alpha\beta(\beta t)^{\alpha-1} \exp\{-(\beta t)^\alpha\} \tag{384}$$

for $t > 0$. Regarding a censored observation, we have

$$P(T > t) = \exp\{-(\beta t)^\alpha\}. \tag{385}$$

Denote d to be the number of uncensored observations, and $\sum_u \log t_i$ to be the sum of the logs of all the uncensored observations. Then

$$l(\alpha, \beta; \mathbf{x}) = d \log \alpha + \alpha d \log \beta + (\alpha - 1) \sum_u \log t_i - \beta^\alpha \sum_{i=1}^n t_i^\alpha. \tag{386}$$

We will now derive the profile log-likelihood and a confidence interval for α.

We first treat α as fixed, and find the MLE of β as a function of the data and α. Differentiating $l(\alpha, \beta)$ with respect to β and equating to zero gives

$$\hat{\beta} = \left(\frac{d}{\sum_{i=1}^n t_i^\alpha} \right)^{\frac{1}{\alpha}}. \tag{387}$$

The profile log-likelihood of α is then given by

$$l_p(\alpha;\mathbf{x}) = l(\alpha,\hat{\beta}) \tag{388}$$

$$= d\log\alpha + \alpha d\log\left(\frac{d}{\sum_{i=1}^n t_i^\alpha}\right)^{\frac{1}{\alpha}} + (\alpha-1)\sum_u \log t_i - d \tag{389}$$

The profile log-likelihood is plotted in Figure 23.

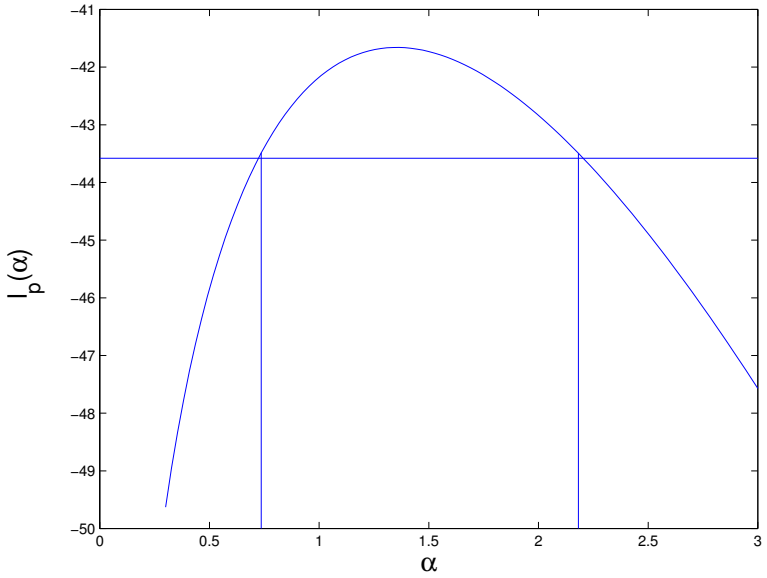

Figure 23: profile log-likelihood $l_p(\alpha)$

Finding the full MLE $(\hat{\alpha},\hat{\beta})$ cannot be done analytically, so numerical methods have to be used. In R, if you define a function that takes a single (vector) input `x`:
`f<-function(x){...}`
you can numerically minimise the function given a starting value `z` with the command:
`nlmin(f,z)`
Type `?nlmin` for more details.

To construct the confidence interval, you only need to find the value $\hat{\alpha}$ that maximises $l_p(\hat{\alpha};\mathbf{x})$, as

$$l_p(\hat{\alpha};\mathbf{x}) = l(\hat{\alpha},\hat{\beta};\mathbf{x}). \tag{390}$$

For a 95% confidence interval, the 95th percentage point of the χ^2_1 distribution is 3.841. The confidence interval is then given by

$$C_{0.05} = \{\alpha^* : D_p(\alpha^*) \le 3.841\} \tag{391}$$

$$= [\alpha^* : 2\{l_p(\hat{\alpha}) - l_p(\alpha^*)\} \le 3.841] \tag{392}$$

$$= \{\alpha^* : l_p(\alpha^*) > l_p(\hat{\alpha}) - 3.841/2\}. \tag{393}$$

Numerically, we estimate the MLE $\hat{\alpha}$ to be 1.35, with $l_p(\hat{\alpha}) = -41.66$. From the graph, we can then read off the 95% confidence interval for α as (0.73,2.2). This contains the value 1, so the

simpler exponential distribution is plausible for this dataset. (To obtain the numerical value of the MLE $\hat{\beta}$, we simply substitute $\alpha = 1.35$ into equation (387)

10.5 Example: machine component failure

In an experiment, the level of corrosion w in a machine component is recorded and the component is tested until a failure is observed, at time t. The level of corrosion varies within a sample of components used in the experiment. The data are plotted in Figure 24.

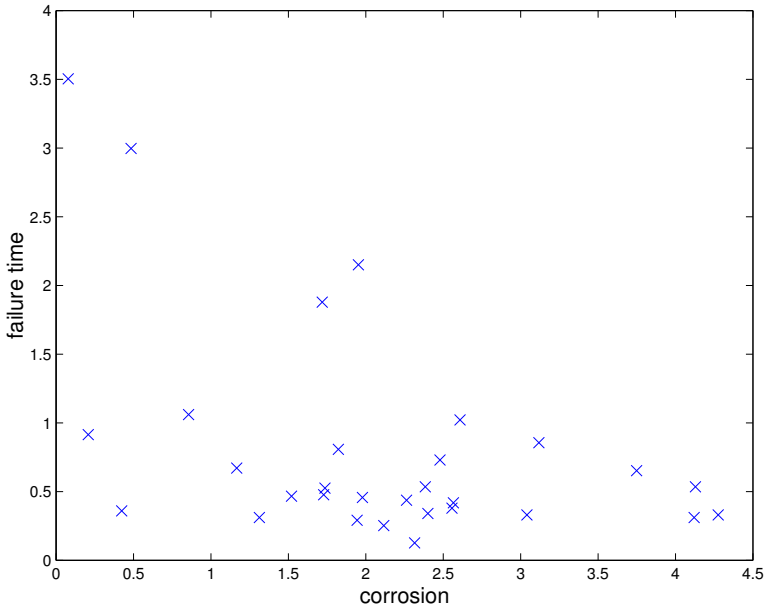

Figure 24: machine component failure data

Denote each observation by (w_i, t_i), where w_i is the level of corrosion, and t_i is the failure time. A possible model for the data is to suppose that a failure time T has the $Exponential(\lambda)$ distribution, with λ a function of the corrosion level w:

$$\lambda = \alpha w^{\beta}. \tag{394}$$

The corrosion w is not treated as random variable here, so that we are considering the distribution of the failure time conditional on the corrosion. With $\beta = 0$, we would have the same expected time to failure, α^{-1} for all components, regardless of the corrosion level w. The density of a single observation (w, t) is given by

$$f_T(t) = \alpha w^{\beta} \exp\{-\alpha w^{\beta} t\}. \tag{395}$$

The log-likelihood for α and β is then given by

$$l(\alpha, \beta; \mathbf{x}) = n \log \alpha + \beta \sum_{i=1}^{n} \log w_i - \alpha \sum_{i=1}^{n} w_i^{\beta} t_i. \tag{396}$$

We can derive an expression for the profile log-likelihood of β:
Treating β as fixed, we obtain the MLE of α as

$$\hat{\alpha} = \frac{n}{\sum_{i=1}^{n} w_i^{\beta} t_i}. \tag{397}$$

We then substitute this expression for α in the full log-likelihood $l(\alpha, \beta)$ to get the profile log-likelihood for β:

$$l_p(\beta; \mathbf{x}) = n \log \left(\frac{n}{\sum_{i=1}^{n} w_i^{\beta} t_i} \right) + \beta \sum_{i=1}^{n} \log w_i - n. \tag{398}$$

The profile log-likelihood is plotted in Figure 25. Numerically, we estimate the MLE $\hat{\beta}$ to be

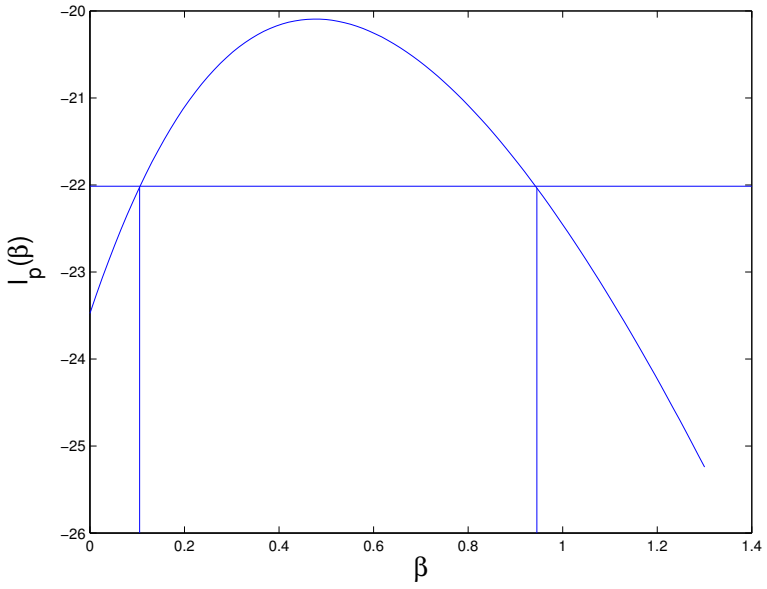

Figure 25: profile log-likelihood $l_p(\beta)$

0.473, with $l_p(\hat{\beta}) = -20.01$. From the graph, we can then read off the 95% confidence interval for β as (0.11,0.95). This doesn't contain zero, and so there is clear evidence that $\beta \neq 0$, and failure time is dependent on corrosion.

For comparison, we can also compute a confidence interval for β using the asymptotic normal approximation. The observed information matrix is given by

$$\begin{pmatrix} -\frac{\partial^2 l}{\partial \alpha^2} & -\frac{\partial^2 l}{\partial \alpha \partial \beta} \\ -\frac{\partial^2 l}{\partial \alpha \partial \beta} & -\frac{\partial^2 l}{\partial \beta^2} \end{pmatrix} = \begin{pmatrix} n\alpha^{-2} & \sum w_i^{\beta} t_i \log w_i \\ \sum w_i^{\beta} t_i \log w_i & \alpha \sum w_i^{\beta} t_i (\log w_i)^2 \end{pmatrix} \tag{399}$$

We can obtain $\hat{\alpha}$ by substituting $\beta = 0.473$ into equation (397). This gives $\hat{\alpha} = 1.099$. We now substitute $\alpha = 1.099$ and $\beta = 0.473$ into (399) and invert to get the variance covariance matrix V:

$$V = \begin{pmatrix} 0.0534 & -0.0241 \\ -0.0241 & 0.0442 \end{pmatrix}. \tag{400}$$

The confidence interval for β using the asymptotic normality approximation is then

$$\hat{\beta} \pm 1.96 \times 0.0442^{0.5}, \tag{401}$$

which gives (0.0611,0.8849). This is of a similar width to the interval derived previously, but is shifted to the left slightly. From the skewness in figure 3, this is to be expected.