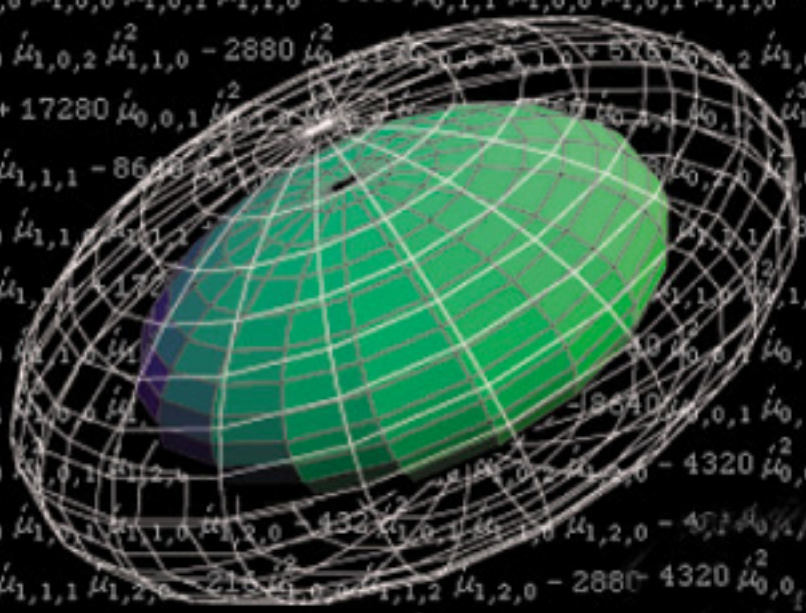
SPRINGER TEXTS IN STATISTICS

MATHEMATICAL STATISTICS

with *Mathematica*®



COLIN ROSE

MURRAY D. SMITH

Notes

Chapter 1 Introduction

1. ***Nota bene*** Take note.

Chapter 2 Continuous Random Variables

1. **Warning:** On the one hand, σ is often used to denote the standard deviation. On the other hand, some distributions use the symbol σ to denote a parameter, even though this parameter is not equal to the standard deviation; examples include the Lognormal, Rayleigh and Maxwell–Boltzmann distributions.

2. The textbook reference solution, as listed in Johnson *et al.* (1994, equation (18.11)), is incorrect.

3. Black and Scholes first tried to publish their paper in 1970 at the *Journal of Political Economy* and the *Review of Economics and Statistics*. Both journals immediately rejected the paper without even sending it to referees!

4. The assumption that investors are risk-neutral is a simplification device: it can be shown that the solutions derived are valid in all worlds.

Chapter 3 Discrete Random Variables

1. The more surface area a face has, the greater the chance that it will contact the table-top. Hence, shaving a face increases the chance that it, and its opposing face, will occur. Now, because the die was a perfect cube to begin with, shaving the 1-face is no different from shaving the 6-face. The chance of a 1 or 6 is therefore uniformly increased. To see intuitively what happens to the probabilities, imagine throwing a die that has been shaved to extreme — the die would be a disk with two faces, 1 and 6, and almost no edge, so that the chance of outcomes 2, 3, 4 or 5 drop (uniformly) to zero.

2. The interpretation of the limiting distribution is this: after the process has been operating for a long duration ('burnt-in'), the (unconditional) probability p_k of the process being in a state k is independent of how the process first began. For given states k and j, p_k is defined as $\lim_{t\to\infty} P(X_t = k \,|\, X_0 = j)$ and is invariant to the value of j. Other terms for the limiting distribution include 'stationary distribution', 'steady-state distribution', and 'equilibrium distribution'. For further details on Markov chains see,

for example, Taylor and Karlin (1998), and for further details on asymptotic statistics, see Chapter 8.

3. In the derivations to follow, it is easier to think in terms of draws being made one-by-one without replacement. However, removing at once a single handful of m balls from the urn is probabilistically equivalent to m one-by-one draws, if not physically so.

4. To see that $f(x)$ has the same probability mass under `domain[f]={x,0,n}` as under `domain[f]={x,0,Min[n,r]}`, consider the two possibilities: If $n \le r$, everything is clearly fine. If $n > r$, the terms added correspond to every $x \in \{r+1, \ldots, n\}$. In this range, $x > r$, and hence $\binom{T-n}{r-x}$ is always 0, so that the probability mass $f(x) = 0$ for $x > r$. Thus, the probability mass is not affected by the inclusion of the extra terms.

5. It is *not* appropriate to treat the component-mix X as if it is a weighted average of random variables. For one thing, the domain of support of a weighted average of random variables is more complicated because the values of the weights influence the support. To see this, consider two Bernoulli variables. The domain of support of the component-mix is the union $\{0, 1\} \bigcup \{0, 1\} = \{0, 1\}$, whereas the domain of support of the weighted average is $\{0, \omega_1, \omega_2, 1\}$.

6. The assumption of Normality is not critical here. It is sufficient that Y_i has a finite variance. Then approximate Normality for $Y = \sum_{i=1}^{t} Y_i$ follows by a suitable version of the Central Limit Theorem; see, for example, Taylor and Karlin (1998, p. 75).

7. When working numerically, the trick here is to ensure that the variance of the Normal pdf σ^2 matches the variance of the parameter-mix model given by `Expect[`$t\,\omega^2, g$`]` $= \omega^2 / p$. Then, taking say $\sigma^2 = 1$, we require $p = \omega^2$ for the variances to match. The values used in Fig. 11 ($\sigma = 1$, $\omega = \sqrt{0.1}$, $p = 0.1$) are consistent with this requirement.

8. Lookup tables are built by `DiscreteRNG` using *Mathematica*'s `Which` function. To illustrate, here is a lookup table for *Example 17*, where `u = Random[]`:

```
Which[ 0   < u < 0.1,   -1.  ,
       0.1 < u < 0.5,    1.5 ,
       0.5 < u < 0.8,    Pi  ,
       True,             4.4  ]
```

Chapter 4 Distributions of Functions of Random Variables

1. Notes:

   (i) For a more detailed proof, see Walpole and Myers (1993, Theorem 7.3).

   (ii) Observe that $J = \dfrac{dx}{dy} = \dfrac{1}{dy/dx}$.

2. Let $X \sim$ Exponential($\frac{1}{a}$) with pdf $h(x)$:

```
h = a ℮^-ax ;  domain[h] = {x, 0, ∞} && {a > 0} && {b > 0} ;
```

Then, the pdf of $Y = b\,e^X$ $(b > 0)$ is:

```
Transform[y == b e^x, h]
TransformExtremum[y == b e^x, h]
```

```
a b^a y^-1-a
```

```
{y, b, ∞} && {a > 0, b > 0}
```

3. The multivariate case follows analogously; see, for instance, Roussas (1997, p. 232) or Hogg and Craig (1995, Section 4.5).

Chapter 5 Systems of Distributions

1. The area defining *I(J)* in Fig. 1 was derived symbolically using *Mathematica*. A comparison with Johnson *et al*. (1994) shows that their diagram is actually somewhat inaccurate, as is Ord's (1972) diagram. By contrast, Stuart and Ord's (1994) diagram seems fine.

2. For somewhat cleaner results, note that:
   (i) §7.2 B discusses unbiased estimators of central moments calculated from sample data;
   (ii) The 'quick and dirty' formulae used here for calculating moments from grouped data assume that the frequencies occur at the mid-point of each interval, rather than being spread over the interval. A technique known as Sheppard's correction can sometimes correct for this effect: see, for instance, Stuart and Ord (1994, Section 3.18).

3. The reader comparing results with Stuart and Ord (1994) should note that there is a typographic error in their solution to μ_3.

4. Two alternative methods for deriving Hermite polynomials (as used in statistics) are `H1` and `H2`, where:

```
H1[j_] := 2^-j/2 HermiteH[j, z/Sqrt[2]] // Expand
```

and:

```
Clear[g];   g'[z] = -z g[z];

H2[j_] := (-1)^j D[g[z], {z, j}]/g[z] // Expand
```

`H1` makes use of the built-in `HermiteH` function, while `H2` notes that if density $g(z)$ is $N(0, 1)$, then $g'(z) = -z\,g(z)$. While both `H1` and `H2` are more efficient than `H`, they are somewhat less elegant in the present context.

5. The original source of the data is Schwert (1990). Pagan and Ullah then adjusted this data for calendar effects by regressing out twelve monthly dummies.

Chapter 6 Multivariate Distributions

1. In order to ascribe a particular value to the conditioning variable, say $f(x_1 \mid X_2 = \frac{1}{2})$, proceed as follows:

```
Conditional[x1, f] /. x2 → 1/2
```

```
– Here is the conditional pdf  f ( x1 | x2 ):
```

$$\frac{1}{2} + x_1$$

Do *not* use `Conditional[x1, f /. x2 → 1/2]`. In **mathStatica** functions, the syntax `f /. x2 → 1/2` may only be used for replacing the values of parameters (not variables).

2. Some texts refer to this as the Farlie–Gumbel–Morgenstern class of distributions; see, for instance, Kotz *et al*. (2000, p. 51).

3. More generally, if $Z \sim N(0, 1)$, its cdf is $\Phi(z) = \frac{1}{2}\left(1 + \text{Erf}\left[\frac{z}{\sqrt{2}}\right]\right)$. Then, in a zero correlation m-variate setting with $\vec{Z} = (Z_1, \ldots, Z_m) \sim N(\vec{0}, I_m)$, the joint cdf will be:

$$\left(\frac{1}{2}\right)^m \left(1 + \text{Erf}\left[\frac{z_1}{\sqrt{2}}\right]\right) \cdots \left(1 + \text{Erf}\left[\frac{z_m}{\sqrt{2}}\right]\right).$$

This follows because $(Z_1, \ldots, Z_m)$ are mutually stochastically independent (Table 3 (i)).

4. *Mathematica*'s `Multinormal` statistics package contains a special `CDF` function for the multivariate Normal density. Under *Mathematica* Version 4.0.x, this function does not work if any $\rho_{ij} = 0$, irrespective of whether the 0 is a symbolic zero (0) or a numerical zero (0.). For instance, $P(X \le -2, Y \le 0, Z \le 2)$ fails to evaluate under zero correlation:

```
CDF[dist3 /. ρ_ → 0, {-2, 0, 2}]
```

```
– Solve::svars :
    Equations may not give solutions for all "solve" variables.

– CDF::mnormfail : etc ...
```

Fortunately, this problem has been fixed, as of *Mathematica* Version 4.1.

5. Under *Mathematica* Version 4.0, the `CDF` function in *Mathematica*'s `Multinormal` statistics package has two problems: it is very slow, and it consumes unnecessarily large amounts of memory. For example:

```
G[1, -7, 3] // Timing
```

```
{7.25 Second, 1.27981 × 10^-12}
```

Rolf Mertig has suggested (in email to the authors) a fix to this problem that does not alter the accuracy of the solution in any way. Simply enter:

```
Unprotect[MultinormalDistribution];

UpValues[MultinormalDistribution] =
  UpValues[MultinormalDistribution] /.
   HoldPattern[NIntegrate[a_, b__]] :>
    NIntegrate[Evaluate[a], b];
```

and then the `CDF` function is suddenly more than 40 times faster, and it no longer hogs memory:

```
G[1, -7, 3] // Timing
```

```
{0.11 Second, 1.27981 × 10^-12}
```

Under *Mathematica* Version 4.1, none of these problems occur, so there is no need to fix anything.

6. A random vector $\vec{X}$ is said to be spherically distributed if its pdf is equivalent to that of $\vec{Y} = H\,\vec{X}$, for all orthogonal matrices H. The zero correlation bivariate Normal is a member of the spherical class, because its pdf

$$\frac{1}{2\pi}\exp\left(-\frac{\vec{x}^{\,\mathrm{T}}\,\vec{x}}{2}\right)$$

depends on $\vec{x}$ only through the value of the scalar $\vec{x}^{\,\mathrm{T}}\,\vec{x}$, and so $(H\,\vec{x})^{\mathrm{T}}\,(H\,\vec{x}) = \vec{x}^{\,\mathrm{T}}\,(H^{\mathrm{T}}\,H)\,\vec{x} = \vec{x}^{\,\mathrm{T}}\,\vec{x}$, because $H^{\mathrm{T}}\,H = I_2$. An interesting property of spherically distributed variables is that a transformation to polar co-ordinates yields mutually stochastically independent random variables. Thus, in the context of *Example 20* (Robin Hood) above, when $\rho = 0$, the angle Θ will be independent of the radius (distance) R (see density $g(r, \theta)$). For further details on the spherical family of distributions, see Muirhead (1982).

7. The multinomial coefficient

$$\binom{n}{x_1, x_2, \ldots, x_m} = \frac{n!}{x_1!\,x_2!\cdots x_m!}$$

is provided in *Mathematica* by the function `Multinomial[` $x_1, x_2, \ldots, x_m$ `]`. It gives the number of ways to partition n objects into m sets of size x_i.

8. Alternatively, one can find the solution 'manually' as follows:

$$\begin{aligned} E[e^{t_1 Y_1 + t_2 Y_2 + (t_1+t_2) Y_0}] &= E[e^{t_1 Y_1}]\,E[e^{t_2 Y_2}]\,E[e^{(t_1+t_2) Y_0}] \qquad \text{by Table 3 (ii)} \\ &= \exp\left((e^{t_1}-1)\,\lambda_1 + (e^{t_2}-1)\,\lambda_2 + (e^{t_1+t_2}-1)\,\lambda_0\right). \end{aligned}$$

The same technique can be used to derive the pgf.

Chapter 7 Moments of Sampling Distributions

1. Chapter 2 introduced a suite of converter functions that allow one to express any population moment ($\acute{\mu}$, μ, or κ) in terms of any other population moment ($\acute{\mu}$, μ, or κ). These functional relationships also hold between the sample moments. Thus, by combining the moment converter functions with equation (7.2), we can convert any sample moment (raw, central or cumulant) into power sums. For instance, to convert the fourth central sample moment m_4 into power sums, we first convert from central m to raw $\acute{m}$ moments using `CentralToRaw[4, m, ḿ]` (note the optional notation arguments m and $\acute{m}$), and then use (7.2) to convert the latter into power sums. Here is m_4 in terms of power sums:

```
CentralToRaw[4, m, ḿ] /. ḿi_ :> si/n
```

$$m_4 \to -\frac{3\,s_1^4}{n^4} + \frac{6\,s_1^2\,s_2}{n^3} - \frac{4\,s_1\,s_3}{n^2} + \frac{s_4}{n}$$

This is identical to:

```
SampleCentralToPowerSum[4]
```

$$m_4 \to -\frac{3\,s_1^4}{n^4} + \frac{6\,s_1^2\,s_2}{n^3} - \frac{4\,s_1\,s_3}{n^2} + \frac{s_4}{n}$$

2. Kendall's comment on the term 'polykays' can be found in Stuart and Ord (1994, Section 12.22).

3. Just as we can think of moments as being 'about zero' (raw) or 'about the mean' (central), one can think of cumulants as also being 'about zero' or 'about the mean'. The *moment of moment* functions that are expressed in terms of cumulants, namely:

```
RawMomentToCumulant
CentralMomentToCumulant
CumulantMomentToCumulant
```

… do their internal calculations *about the mean*. That is, they set $\mu_1 = \kappa_1 = 0$. As such, if `p = PolyK[{1,2,3}][[2]]`, then `RawMomentToCumulant[1, p]` will return 0, not $\kappa_1\,\kappa_2\,\kappa_3$. To force **mathStatica** to do its `___ToCumulant` calculations about zero rather than about the mean, add `Z` to the end of the function name: *e.g.* use `RawMomentToCumulant`**`Z`**. For example, given:

```
p = PolyK[{1, 2, 3}][[2]];
```

... compare:

```
RawMomentToCumulant[1, p]
```

```
0
```

with:

```
RawMomentToCumulantZ[1, p]
```

$$\kappa_1\ \kappa_2\ \kappa_3$$

Working 'about zero' requires greater computational effort than working 'about the mean', so the various ___CumulantZ functions are often significantly slower than their Z-less cousins.

4. PowerSumToAug, AugToPowerSum and MonomialToPowerSum are the only **mathStatica** functions that allow one to use shorthand notation such as $\{1^4\}$ to denote {1, 1, 1, 1}. This feature does not work with any other **mathStatica** function.

Chapter 8 Asymptotic Theory

1. The discussion of Calculus`Limit` has benefitted from detailed discussions with Dave Withoff of Wolfram Research.

2. Some texts (*e.g.* Billingsley (1995)) separate the definition into two parts: (i) terming (8.1) the weak convergence of $\{F_n\}_{n=1}^{\infty}$ to F, and (ii) defining convergence in distribution of $\{X_n\}_{n=1}^{\infty}$ to X only when the corresponding cdf's converge weakly.

3. van Beek improved upon the original version of the bounds referred to in the so-called Berry–Esseen Theorem; for details see, amongst others, Bhattacharya and Rao (1976).

4. Φ is the limiting distribution of W_* by the Lindeberg–Feller version of the Central Limit Theorem. This theorem is not discussed here, but details about it can be found in Billingsley (1995) and McCabe and Tremayne (1993), amongst others.

5. Under Version 4.0 of *Mathematica*, some platforms give the solution for μ_3^+ as

$$\frac{1}{(2+\theta)\ (4+\theta)\ \Gamma\left[\frac{\theta}{2}\right]}\ \left(e^{-\theta/2}\ \left(2^{4-\frac{\theta}{2}}\ \theta^{\frac{4+\theta}{2}}\ (2+\theta)\ -\right.\right.$$
$$2\ e^{\theta/2}\ \left(32\ (4+3\,\theta)\ \Gamma\left[1+\frac{\theta}{2}\right]+8\ (-4+\theta^2)\ \Gamma\left[3+\frac{\theta}{2}\right]-\right.$$
$$\left.\left.\left.\theta^4\ (6+\theta)\ \Gamma\left[\frac{\theta}{2}\right]-64\ \text{Gamma}\left[3+\frac{\theta}{2},\ \frac{\theta}{2}\right]\right)\right)\right)$$

Although this solution appears different to the one derived in the text, the two are nevertheless equivalent.

6. We emphasise that for any finite choice of n, this pseudo-random number generator is only approximately $N(0, 1)$.

7. For example, it makes no sense to consider the convergence in probability of $\{X_n\}_{n=1}^{\infty}$ to X, if all variables in the sequence are measured in terms of pounds of butter, when X is measured in terms of numbers of guns.

8. Letting MSE $= E\left[(\overline{X}_n - \theta)^2\right]$, write

$$\text{MSE} = E\left[\left(\frac{1}{n}\sum_{i=1}^{n}(X_i - \theta)\right)^2\right] = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}E\left[(X_i - \theta)(X_j - \theta)\right].$$

Of the n^2 terms in the double-sum there are n when the indices are equal, yielding expectations in the form of $E[(X_i - \theta)^2]$; the remaining $n(n-1)$ terms are of the form $E[(X_i - \theta)(X_j - \theta)]$. Due to independence, the latter expectation can be decomposed into the product of expectations: $E[X_i - \theta]\,E[X_j - \theta]$. Thus,

$$\text{MSE} = \frac{1}{n^2}\sum_{i=1}^{n}E[(X_i - \theta)^2] + \frac{1}{n^2}\sum_{\substack{i=1\\ i\neq j}}^{n}\sum_{j=1}^{n}E[X_i - \theta]\,E[X_j - \theta].$$

As each of the random variables in the random sample is assumed to be a copy of a random variable X, replace $E[(X_i - \theta)^2]$ with $E[(X - \theta)^2]$, as well as $E[X_i - \theta]$ and $E[X_j - \theta]$ with $E[X - \theta]$. Finally, then,

$$\begin{aligned}\text{MSE} &= \frac{1}{n^2}\sum_{i=1}^{n}E[(X - \theta)^2] + \frac{1}{n^2}\sum_{\substack{i=1\\ i\neq j}}^{n}\sum_{j=1}^{n}(E[X - \theta])^2\\ &= \frac{1}{n}E[(X - \theta)^2] + \frac{n-1}{n}(E[X] - \theta)^2.\end{aligned}$$

Chapter 9 Statistical Decision Theory

1. Sometimes, we do not know the functional form of $g\left(\hat{\theta};\theta\right)$; if this is the case then an alternative expression for risk involves the multiple integral:

$$R_{\hat{\theta}}(\theta) = \int\cdots\int L\left(\hat{\theta}(x_1, \ldots, x_n), \theta\right) f(x_1, \ldots, x_n; \theta)\, dx_1 \cdots dx_n$$

where we let $\hat{\theta}(X_1, \ldots, X_n)$ express the estimator in terms of the variables in the random sample $X_1, \ldots, X_n$, the latter having joint density f (here assumed continuous). For the examples encountered in this chapter, we shall assume the functional form of $g\left(\hat{\theta};\theta\right)$ is known.

2. The pdf of $X_{(r)}$ can be determined by considering the combinatorics underlying the re-arrangement of the random sample. In all, there are n candidates from $(X_1, \ldots, X_n)$ for $X_{(r)}$, and $n-1$ remaining places that fall into two classes: $r-1$ places below x (x represents values assigned to $X_{(r)}$), and $n-r$ places above x. Those that fall below x do so with probability $F(x)$, and those that lie above x do so with probability $1 - F(x)$, while the successful candidate contributes the value of the pdf at x, $f(x)$.

3. Johnson *et al.* (1995, equation (24.14)) give an expression for the pdf of $X_{(r)}$ which differs substantially to the (correct) output produced by **mathStatica**. It is not difficult to show that the former is incorrect. Furthermore, it can be shown that equations

(24.15), (24.17) and (24.18) of Johnson *et al*. (1995) are incorrectly deflated by a factor of two.

4. *Mathematica* solves many integrals by using a large lookup table. If the expression we are trying to integrate is not in a standard form, *Mathematica* may not find the expression in its lookup table, and the integral will fail to evaluate.

Chapter 10 Unbiased Parameter Estimation

1. Many texts use the term Fisher Information when referring to either measure. Sample Information may be viewed as Fisher Information per observation on a size n random sample $\vec{X} = (X_1, \ldots, X_n)$.

2. *Example 10* is one such example. See Theorem 10.2.1 in Silvey (1995), or Gourieroux and Monfort (1995, pp. 81–82) for the conditions that a given statistical model must meet in order that the BUE of a parameter exists.

3. If the domain of support of X depends on unknown parameters (*e.g.* θ in $X \sim \text{Uniform}(0, \theta)$), added care needs to be taken when using (10.13). In this book, we shall not concern ourselves with cases of this type; instead, for further details, we refer the interested reader to Stuart and Ord (1991, pp. 638–641).

4. This definition suffices for our purposes. For the full version of the definition, see, for example, Hogg and Craig (1995, p. 330).

5. Here, $E[T] = \big(0 \times P(X_n \le k)\big) + \big(1 \times P(X_n > k)\big) = P(X_n > k)$. Since X_n is a copy of X, it follows that T is unbiased for $g(\lambda)$.

Chapter 11 Principles of Maximum Likelihood Estimation

1. If θ is a vector of k elements, then the first-order condition requires the simultaneous solution of k equations, and the second-order condition requires establishing that the $(k \times k)$ Hessian matrix is negative definite.

2. It is conventional in the Normal statistical model to discuss estimation of the pair (μ, σ^2) rather than (μ, σ). However, because *Mathematica* treats σ^2 as a `Power` and not as a `Symbol`, activities such as differentiation and equation-solving involving σ^2 can not be undertaken. This can be partially overcome by entering `SuperD[On]` which invokes a **mathStatica** function that allows *Mathematica* to differentiate with respect to `Power` variables. Unfortunately, **mathStatica** does not contain a similar enhancement for equation-solving in terms of `Power` variables.

3. The following input generates an information message:

```
NSum::nslim: Limit of summation n is not a number.
```

This has no bearing on the correctness of the output so this message may be safely ignored. We have deleted the message from the text.

4. Of course, biasedness is just one aspect of small sample performance. Chapter 9 considers other factors, such as performance under Mean Square Error.

5. The mgf of the Gamma(n, $\frac{1}{n\theta}$) distribution may be derived as:

```
g = y^(a - 1) e^(-y/b) / Γ[a] b^a /. {a → n, b → 1/(n θ)};

domain[g] = {y, 0, ∞} && {n > 0, n ∈ Integers, θ > 0};

Expect[e^(t y), g]

(1 - t/(n θ))^-n
```

   Using simple algebra, this output may be re-written $(\theta/(\theta - \frac{t}{n}))^n$, which matches the mgf of $\overline{\log X}$.

6. Let $\{Y_n\}$ be a sequence of random variables indexed by n, and Y a random variable such that $Y_n \xrightarrow{d} Y$. Let g denote a continuous function (it must be independent of n) throughout the domain of support of $\{Y_n\}$. The Continuous Mapping Theorem states that $g(Y_n) \xrightarrow{d} g(Y)$; see, for example, McCabe and Tremayne (1993). In our case, we set $g(y) = y^{-1}$, and because convergence in distribution to a constant implies convergence in probability to the same constant (§8.5 A), the theorem may be applied.

7. Alternatively, the limiting distribution of $\sqrt{n}\left(\hat{\theta} - \theta\right)$ can be found by applying Skorohod's Theorem (also called the delta method). Briefly, let the sequence of random variables $\{Y_n\}$ be such that $\sqrt{n}\,(Y_n - c) \xrightarrow{d} Y$, where c is a constant and Y a random variable, and let a function g have a continuous first derivative with $G = \partial g(c)/\partial y$. Then $\sqrt{n}\,(g(Y_n) - g(c)) \xrightarrow{d} G\,Y$. In our case, we have $\sqrt{n}\,(\hat{\theta}^{-1} - \theta^{-1}) \xrightarrow{d} \theta^{-1}\,Z$. So $\{Y_n\} = \{\hat{\theta}^{-1}\}$, $c = \theta^{-1}$, $Y = \theta^{-1}\,Z$, where $Z \sim N(0, 1)$. Now set $g(y) = 1/y$, so $G = -1/c^2$. Applying the theorem yields:

$$\sqrt{n}\left(\hat{\theta} - \theta\right) \xrightarrow{d} (-\theta^2)\,\theta^{-1}\,Z \sim N(0, \theta^2).$$

8. The log-likelihood can be concentrated with respect to the MLE of α_0. Thus, if we let $((Y_1, X_1), \ldots, (Y_n, X_n))$ denote a random sample of size n on the pair (Y, X), the MLE of α_0 can, as a function of β, be shown to equal

$$\hat{\alpha} = \hat{\alpha}(\beta) = \log\left(\frac{1}{n}\sum_{i=1}^{n} Y_i\, e^{-\beta X_i}\right).$$

   The concentrated log-likelihood function is given by $\log L(\hat{\alpha}(\beta), \beta)$, which requires numerical methods to be maximised with respect to β (numerical optimisation is discussed in Chapter 12).

Chapter 12 Maximum Likelihood Estimation in Practice

1. Of course, elementary calculus may be used to symbolically maximise the observed log-likelihood, but our purpose here is to demonstrate `FindMaximum`. Indeed, from *Example 5* of Chapter 11, the ML estimator of λ is given by the sample mean. For the Nerve data, the ML estimate of λ is:

```
SampleMean[xdata]
```

```
0.218573
```

2. For commentary on the comparison between ML and OLS estimators in the Normal linear regression model see, for example, Judge *et al*. (1985, Chapter 2).

3. Just for fun, another (equivalent) way to construct `urules` is:

```
urules = MapThread[(u#1 → #2) &, {Range[n], uvec}];
Short[urules]
```

```
{u1 → 0., u2 → 0.13, <<236>>, u239 → -0.11}
```

4. `FindMaximum` / `FindMinimum` may sometimes work with subscript parameters if `Evaluate` is wrapped around the expression to be optimised (*i.e.* `FindMinimum[Evaluate[expr],{...}]`); however, this device will not always work, and so it is best to avoid using subscript notation with `FindMaximum` / `FindMinimum`.

5. In practice, of course, a great deal of further experimentation with different starting values is usually necessary. For space reasons, we will not pursue our search any further here. However, we do encourage the reader to experiment further using their own choices in the above code.

6. In general, if $X_n \xrightarrow{d} X$ and $Y_n \xrightarrow{p} c$, where c is a constant, then $X_n\, Y_n \xrightarrow{d} c\, X$. We can use this result by defining

$$Y_n = \sqrt{\frac{\alpha\,\beta^2}{\hat{\alpha}\,\hat{\beta}^2}}\,.$$

Because of the consistency property of the MLE, we have $Y_n \xrightarrow{p} c = 1$. Thus,

$$\sqrt{\frac{\alpha\,\beta^2}{\hat{\alpha}\,\hat{\beta}^2}}\ \sqrt{n}\ (\hat{\mu} - \mu) \xrightarrow{d} 1 \times N(0,\, \alpha\beta^2) \;=\; N(0,\, \alpha\beta^2).$$

Therefore, at the estimates of $\hat{\alpha}$ and $\hat{\beta}$,

$$\sqrt{\frac{\alpha\,\beta^2}{\hat{\alpha}\,\hat{\beta}^2}}\ \sqrt{n}\ (\hat{\mu} - \mu) \overset{a}{\sim} N(0,\, \alpha\beta^2).$$

Thus,

$$\sqrt{n}\ (\hat{\mu} - \mu) \overset{a}{\sim} N\left(0,\, \hat{\alpha}\,\hat{\beta}^2\right).$$

7. The inverse cdf of the $N(0, 1)$ distribution, evaluated at $1 - \omega/2$, is derived as follows:

```
f = e^(-x^2/2) / Sqrt[2 π];      domain[f] = {x, -∞, ∞};
Solve[y == Prob[x, f], x] /. y → (1 - ω/2)
{{x → Sqrt[2] InverseErf[0, -1 + 2 (1 - ω/2)]}}
```

8. *Mathematica*'s on-line help for `FindMinimum` has an example of this problem in a one-dimensional case; see also Wolfram (1999, Section 3.9.8).

9. If we used `count[[x + 1]]` instead of n_x, the input would fail. Why? Because the product,

$$\prod_{x=0}^{G} \left(\frac{e^{-\gamma}\, \gamma^{x}}{x!} \right)^{\text{count}[[x+1]]}$$

   is taken with `x` increasing from `0` to `G`, where `G` is a symbol (because it has not been assigned any numerical value). Since the numerical value of `G` is unknown, *Mathematica* can not evaluate the product. Thus, *Mathematica* must treat `x` as a symbol. This, in turn, causes `count[[x + 1]]` to fail.

10. In the previous input, it would not be advisable to replace `G` with 9, for then *Mathematica* would expand the product, and `SuperLog` would not take effect.

11. The log-likelihood is concave with respect to γ because:

```
Hessian[logLγ, γ]
-(Σ_{x=0}^{G} x n_x) / γ^2
```

   … is strictly negative.

12. For example, an estimate of the standard error of the ML estimator of γ, using the Hessian estimator given in Table 3, is given by:

```
Sqrt[1 / (-Hessian[logLγ, γ] /. {G → 9, n_x_ :> count[[x + 1]]})] /. solγ
0.0443622
```

13. It is a mistake to use the Newton–Raphson algorithm when the Hessian matrix is positive definite at points in the parameter space because, at these points, (12.12) must be positive-valued. This forces the penalty function / log-likelihood function to increase / decrease in value from one iteration to the next — the exact opposite of how a

gradient method algorithm is meant to work. The situation is not as clear if the Hessian matrix is indefinite at points in the parameter space, because (12.12) can still be negative-valued. Thus, the Newton–Raphson algorithm can work if the Hessian matrix happens to be indefinite, but it can also fail. On the other hand, the BFGS algorithm will work properly wherever it is located in parameter space, for (12.12) will always be negative.

In our example, it is easy to show that the Hessian matrix is *not* negative definite throughout the parameter space. For example, at $(a, b, c) = (0, 1, 2)$, the Hessian matrix is given by:

```
h = Hessian[obslogLλ, {a, b, c}] /. {a → 0, b → 1, c → 2} // N
```

$$\begin{pmatrix} -180.07 & -63.7694 & -374.955 \\ -63.7694 & -2321.03 & 75.9626 \\ -374.955 & 75.9626 & 489.334 \end{pmatrix}$$

The eigenvalues of this matrix are:

```
Eigenvalues[ h ]
```

```
{-2324.45, 660.295, -347.606}
```

Thus, `h` is indefinite since it has both positive and negative eigenvalues. Consequently, the Hessian matrix is not negative definite throughout the parameter space.

14. If `Method→QuasiNewton` or `Method→Newton` is specified, then it is unnecessary to supply the gradient through the option `Gradient→Grad[obslogLλ,{a,b,c}]`, since these methods calculate the gradient themselves. If `Method→QuasiNewton` or `Method→Newton` is specified, but *Mathematica* cannot find symbolic derivatives of the objective function, then `FindMaximum` will not work.

15. To illustrate, let the scalar function $f(x)$ be such that the scalar x_0 minimises f; that is, $f'(x_0) = 0$. Now, for a point x close to x_0, and for f quadratic in a region about x_0, a Taylor series expansion about x_0 yields $f(x) = f(x_0) + f''(x_0)(x - x_0)^2/2$. Point x will be numerically distinct from x_0 provided at least that $(x - x_0)^2$ is greater than precision. Therefore, if `$MachinePrecision` is equal to 16, it would not be meaningful to set tolerance smaller than 10^{-8}.

16. It is inefficient to include a check of the positive-definiteness of $W_{(j)}$. This is because, provided $W_{(0)}$ is positive definite, BFGS will force all $W_{(j)}$ in the sequence to be positive definite.

17. Our analysis of this test is somewhat informal. We determine whether or not the ML point estimates satisfy the inequalities — that is, whether $\hat{\beta}_1 < \hat{\beta}_2 < \hat{\beta}_3$ holds — for our main focus of attention in this section is the computation of the ML parameter estimates using the NR algorithm.

18. The cdf of a $N(0, 1)$ random variable is derived as follows:

```
f = e^(-x^2/2) / Sqrt[2 π];   domain[f] = {x, -∞, ∞};   Prob[x, f]
```

$$\frac{1}{2}\left(1 + \text{Erf}\left[\frac{x}{\sqrt{2}}\right]\right)$$

19. We refrain from using `Subscript` notation for parameters because `FindMinimum` / `FindMaximum`, which we apply later on, does not handle `Subscript` notation well.

20. The Hessian can be compiled as follows:

```
hessfC = Compile[{a2, a3, b1, b2, b3}, Evaluate[H]];
```

Mathematica requires large amounts of memory to successfully execute this command. In fact, around 43 MB of free RAM in the Kernel is needed for this one calculation; use `MemoryInUse[]` to check your own memory performance (Wolfram (1999, Section 2.13.4)). We can now compare the performance of the compiled function `hessfC` with the uncompiled function `hessf`. To illustrate, evaluate at the point $\lambda = (0, 0, 0, 0, 0)$:

```
λval = {0., 0., 0., 0., 0.};
```

Here is the compiled function:

```
hessfC @@ λval // Timing
```

```
{0.55 Second,
 {{-20.5475,  10.608,    -0.999693,       -5.00375,        -3.83075},
  { 10.608,   -174.13,    4.08597,         31.0208,         45.4937},
  {-0.999693, 4.08597,   -65.9319,        -4.54747*10^-13, -2.72848*10^-12},
  {-5.00375,  31.0208,    0.,              -77.5857,        2.27374*10^-13},
  {-3.83075,  45.4937,   -7.7307*10^-12,   1.3074*10^-12,   -78.8815}}}
```

… while here is the uncompiled function:

```
hessf[λval] // Timing
```

```
{2.03 Second,
 {{-20.5475,  10.608,    -0.999693,       -5.00375,        -3.83075},
  { 10.608,   -174.13,    4.08597,         31.0208,         45.4937},
  {-0.999693, 4.08597,   -65.9319,        -9.09495*10^-13,  4.54747*10^-13},
  {-5.00375,  31.0208,    4.54747*10^-13,  -77.5857,        2.27374*10^-13},
  {-3.83075,  45.4937,    1.36424*10^-12,   7.67386*10^-13, -78.8815}}}
```

The compiled function is about four times faster.

21. The strength of support for this would appear to be overwhelming judging from an inspection of the estimated asymptotic standard errors of $\hat{\beta}_1$, $\hat{\beta}_2$ and $\hat{\beta}_3$. A rule of thumb that compares the extent of overlap of intervals constructed as

$$\text{estimate} \pm 2 \times \text{(estimated standard deviation)}$$

finds only a slight overlap between the intervals about the second and third estimates. Formal statistical evidence may be gathered by performing a hypothesis test of multiple inequality restrictions. For example, one testing scenario could be to specify the maintained hypothesis as $\beta_1 = \beta_2 = \beta_3$, and the alternative hypothesis as $\beta_1 < \beta_2 < \beta_3$.