1 Maximum Likelihood Estimation

The idea behind maximum likelihood is to estimate the unknown parameter by the quantity that makes the probability of obtaining the observed data as large as possible. This probability is represented[1] by the likelihood function

$$L(\theta) = \prod_{i=1}^{n} f(d_i; \theta),$$

where $f(d_i; \theta)$ is the density or probability mass function evaluated at d_i.

Let $\widehat{\theta}$ denote the usual Maximum Likelihood Estimate (MLE). That is, it is the parameter value for which the likelihood function is greatest, over all $\theta \in \Theta$. Because the log is an increasing function, maximizing the likelihood is equivalent to maximizing the log likelihood, which will be denoted

$$\ell(\theta) = \ln L(\theta).$$

In elementary situations where the support of the distribution does not depend on the parameter, you get the MLE by closing your eyes, differentiating the log likelihood, setting the derivative to zero, and solving for θ. Then if you are being careful, you carry out the second derivative test; if $\ell''(\widehat{\theta}) < 0$, the log likelihood is concave down at your answer, and you have found the maximum. Here is an example, useful mostly to clarify ideas and serve as a contrast to more realistic cases. Let $D_1, \ldots, D_n$ be a random sample (independent and identically distributed random variables) from a distribution with density $f(y) = \frac{\theta}{(d+1)^{\theta+1}}$ for $d > 0$, where the unknown parameter θ is strictly greater than zero. The log likelihood is

$$\begin{aligned}
\ell(\theta) &= \ln \prod_{i=1}^{n} \frac{\theta}{(d_i+1)^{\theta+1}} \\
&= \sum_{i=1}^{n} \left(\ln \theta - (\theta+1) \ln(d_i+1) \right) \\
&= n \ln \theta - (\theta+1) \sum_{i=1}^{n} \ln(d_i+1)
\end{aligned}$$

Differentiating with respect to θ,

$$\begin{aligned}
\ell'(\theta) &= \frac{n}{\theta} - \sum_{i=1}^{n} \ln(d_i+1) \stackrel{\text{set}}{=} 0 \\
\Rightarrow \quad \theta &= \frac{1}{n} \sum_{i=1}^{n} \ln(d_i+1).
\end{aligned}$$

Carrying out the second derivative test,

$$\ell''(\theta) = -n\theta^{-2} = -\frac{n}{\theta^2} < 0,$$

so the log likelihood function is concave down and we have located a maximum. This justifies writing $\widehat{\theta} = \frac{1}{n} \sum_{i=1}^{n} \ln(d_i+1)$. In R, if the data were in a numeric vector called `d`, the MLE would be `thetahat = mean(log(d+1))`.

[1] If the data are discrete, the likelihood function is exactly the probability of observing the data that actually were observed. In the continuous case the likelihood function is approximately proportional to the probability of observing a data vector that falls into a small region surrounding the vector (point) that was observed.

Numerical maximum likelihood

Differentiating and setting the derivative to zero is all very well, and you will be asked to do it in some of the exercises. But in this course, as in much of applied statistics, you will find that you can write the log likelihood and differentiate it easily enough, but when you set the derivatives to zero, you obtain a set of equations that are impossible to solve explicitly. This means that the problem needs to be solved numerically. That is, you use a computer to calculate the value of the log likelihood for a set of parameter values, and you search until you have found the biggest one.

But how do you search? It's easy in one or two dimensions, but structural equation models can easily involve dozens, scores or even hundreds of parameters. It's a bit like being dropped by helicopter onto a mountain range, and asked to find the highest peak blindfolded. All you can do is walk uphill. The gradient is the direction of steepest increase, so walk that way. How big a step should you take? That's a good question. When you come to a place where the surface is level, or approximately level, stop. How level is level enough? That's another good question. Once you find a "level" place, you can check to see if the surface is concave down there. If so, you're at a maximum. Is it the global maximum (the real MLE), or just a local maximum? It's usually impossible to tell for sure. You can get the helicopter to drop you in several different places fairly far apart, and if you always arrive at the same maximum you will feel more confident of your answer. But it could still be just a local maximum that is easy to reach. The main thing to observe is that where you start is *very* important. Another point is that for realistically big problems, you need high-grade, professionally written software.

The following example is one that you can do by hand, though maybe not with your eyes closed. But it will serve to illustrate the basic ideas of numerical maximum likelihood.

Example 1.1

Let $D_1, \ldots, D_n$ be a random sample from a normal distribution with mean θ and variance θ^2. A sample of size 50 yields:

```
  5.85 -15.02 -13.24  -1.63  -0.07  -2.40  -3.02  -3.19  -5.16   0.79  -1.03 -10.69
-12.96  -4.55   0.57  -7.94  -6.80   2.95  -9.01  -9.33 -11.93  -7.13  10.34  -1.01
 -4.18  -1.30  -7.56  -1.25  -4.64  -4.88  -4.06  -1.91  -1.81  -6.92 -13.27  -5.52
  4.40 -12.17  -4.55  -5.82  -0.81 -19.28  -4.97  -7.78  -5.07  -5.45  -4.27  -4.98
 -9.56  -9.33
```

Find the maximum likelihood estimate of θ. You only need an approximate value; one decimal place of accuracy will do.

Again, this is a problem that can be solved explicitly by differentiation, and the reader is invited to give it a try before proceeding. Have the answer? Is it still the same day you started? Now for the numerical solution. First, write the log likelihood as

$$\begin{aligned}\ell(\theta) &= \ln \prod_{i=1}^{n} \frac{1}{|\theta|\sqrt{2\pi}} e^{-\frac{(d_i-\theta)^2}{2\theta^2}} \\ &= -n \ln|\theta| - \frac{n}{2}\ln(2\pi) - \frac{\sum_{i=1}^{n} d_i^2}{2\theta^2} + \frac{\sum_{i=1}^{n} d_i}{\theta} - \frac{n}{2}.\end{aligned}$$

We will do this in `R`. The data are in a file called `norm1.data`. Read it. Remember that > is the `R` prompt.

```
> D <- scan("norm1.data")
Read 50 items
```

Now define a function to compute the log likelihood.

```
loglike1 <- function(theta) # Assume data are in a vector called D
    {
    sumdsq <- sum(D^2); sumd <- sum(D); n <- length(D)
    loglike1 <- -n * log(abs(theta)) - (n/2)*log(2*pi) - sumdsq/(2*theta^2) +
                sumd/theta - n/2
    loglike1 # Return value of function
    } # End definition of function loglike1
```

Just to show how the function works, compute it at a couple of values, say $\theta = 2$ and $\theta = -2$.

```
> loglike1(2)
[1] -574.2965
> loglike1(-2)
[1] -321.7465
```

Negative values of the parameter look more promising, but it is time to get systematic. The following is called a *grid search*. It is brutal, inefficient, and usually effective. It is too slow to be practical for large problems, but this is a one-dimensional parameter and we are only asked for one decimal place of accuracy. Where should we start? Since the parameter is the mean of the distribution, it should be safe to search within the range of the data. Start with widely spaced values and then refine the search. All we are doing is to calculate the log likelihood for a set of (equally spaced) parameter values and see where it is greatest. After all, that is the *idea* behind the MLE.

```
> min(D); max(D)
[1] -19.28
[1] 10.34
> Theta <- -20:10
> cbind(Theta,loglike1(Theta))
      Theta
 [1,]   -20  -211.5302
 [2,]   -19  -208.6709
 [3,]   -18  -205.6623
 [4,]   -17  -202.4911
 [5,]   -16  -199.1423
 [6,]   -15  -195.6002
 [7,]   -14  -191.8486
 [8,]   -13  -187.8720
 [9,]   -12  -183.6580
[10,]   -11  -179.2022
[11,]   -10  -174.5179
[12,]    -9  -169.6565
[13,]    -8  -164.7513
[14,]    -7  -160.1163
[15,]    -6  -156.4896
[16,]    -5  -155.6956
[17,]    -4  -162.7285
[18,]    -3  -193.8796
[19,]    -2  -321.7465
[20,]    -1 -1188.0659
```

```
[21,]     0        NaN
[22,]     1 -1693.1659
[23,]     2  -574.2965
[24,]     3  -362.2463
[25,]     4  -289.0035
[26,]     5  -256.7156
[27,]     6  -240.6729
[28,]     7  -232.2734
[29,]     8  -227.8888
[30,]     9  -225.7788
[31,]    10  -225.0279
```

First, we notice that at $\theta = 0$, the log likelihood is indeed Not a Number. For this problem, the parameter space is all the real numbers except zero – unless one wants to think of a normal random variable with zero variance as being degenerate at μ; that is, $P(D = \mu) = 1$. (In his case, what would the data look like?)

But the log likelihood is greatest around $\theta = -5$. We are asked for one decimal place of accuracy, so,

```
> Theta <- seq(from=-5.5,to=-4.5,by=0.1)
> Loglike <- loglike1(Theta)
> cbind(Theta,Loglike)
      Theta   Loglike
 [1,]  -5.5 -155.5445
 [2,]  -5.4 -155.4692
 [3,]  -5.3 -155.4413
 [4,]  -5.2 -155.4660
 [5,]  -5.1 -155.5487
 [6,]  -5.0 -155.6956
 [7,]  -4.9 -155.9136
 [8,]  -4.8 -156.2106
 [9,]  -4.7 -156.5950
[10,]  -4.6 -157.0767
[11,]  -4.5 -157.6665
> thetahat <- Theta[Loglike==max(Loglike)]
>             # Theta such that Loglike is the maximum of Loglike
> thetahat
[1] -5.3
```

To one decimal place of accuracy, the maximum is at $\theta = -5.3$. It would be easy to refine the grid and get more accuracy, but that will do. This is the last time we will see our friend the grid search, but you may find the approach useful in homework.

Now let's do the search in a more sophisticated way, using `R`'s `nlm` (non-linear minimization) function. [2] The `nlm` function has quite a few arguments; try `help(nlm)`. The ones you always need are the first two: the name of the function, and a starting value (or vector of starting values, for multiparameter problems).

[2]The `nlm` function is good but generic. See Numerical Recipes for a really good discussion of routines for numerically minimizing a function. They also provide source code. The *Numerical Recipes* books have versions for the Pascal, Fortran and Basic languages as well as C. This is a case where a book definitely delivers more than the title promises. It may be a cookbook, but it is a very good cookbook written by expert chemists.

Where should we start? Since the parameter equals the expected value of the distribution, how about the sample mean? It is often a good strategy to use Method of Moment estimators as starting values for numerical maximum likelihood. Method of Moments estimation is reviewed in Section **??**.

One characteristic that `nlm` shares with most optimization routines is that it likes to *minimize* rather than maximizing. So we will minimize the negative of the log likelihood function. For this, it is necessary to define a new function, `loglike2`.

```
> mean(D)
[1] -5.051
> loglike2 <- function(theta) { loglike2 <- -loglike1(theta); loglike2 }
> nlm(loglike2,mean(D))
$minimum
[1] 155.4413

$estimate
[1] -5.295305

$gradient
[1] -1.386921e-05

$code
[1] 1

$iterations
[1] 4
```

By default, `nlm` returns a list with four elements; `minimum` is the value of the function at the point where it reaches its minimum, `estimate` is the value at which the minimum was located; that's the MLE. `Gradient` is the slope in the direction of greatest increase; it should be near zero. `Code` is a diagnosis of how well the optimization went; the value of 1 means everything seemed okay. See `help(nlm)` for more detail.

We could have gotten just the MLE with

```
> nlm(loglike2,mean(D))$estimate
[1] -5.295305
```

That's the answer, but the numerical approach misses some interesting features of the problem, which can be done with paper and pencil in this simple case. Differentiating the log likelihood separately for $\theta < 0$ and $\theta > 0$ to get rid of the absolute value sign, and then re-uniting the two cases since the answer is the same, we get

$$\ell'(\theta) = -\frac{n}{\theta} + \frac{\sum_{i=1}^n d_i^2}{\theta^3} - \frac{\sum_{i=1}^n d_i}{\theta^2}.$$

Setting $\ell'(\theta) = 0$ and re-arranging terms, we get

$$n\theta^2 + (\sum_{i=1}^n d_i)\theta - (\sum_{i=1}^n d_i^2) = 0.$$

Of course this expression is not valid at $\theta = 0$, because the function we are differentiating is not even defined there. The quadratic formula yields two solutions:

$$\frac{-\sum_{i=1}^n d_i \pm \sqrt{(\sum_{i=1}^n d_i)^2 + 4n\sum_{i=1}^n d_i^2}}{2n} = \frac{1}{2}\left(-\overline{d} \pm \sqrt{\overline{d}^2 + 4\frac{\sum_{i=1}^n d_i^2}{n}}\right), \tag{1}$$

where $\overline{d}$ is the sample mean.

Let's calculate these for the given data.

```
> meand <- mean(D) ; meandsq <- sum(D^2)/length(D)
> (-meand + sqrt(meand^2 + 4*meandsq) )/2
[1] 10.3463
> (-meand - sqrt(meand^2 + 4*meandsq) )/2
[1] -5.2953
```

The second solution is the one we found with the numerical search. What about the other one? Is it a minimum? Maximum? Saddle point? The second derivative test will tell us. The second derivative is

$$\ell''(\theta) = \frac{n}{\theta^2} - \frac{3\sum_{i=1}^n d_i^2}{\theta^4} + \frac{2\sum_{i=1}^n d_i}{\theta^3}.$$

Substituting 1 into this does not promise to be much fun, so we will be content with a numerical answer for this particular data set. Call the first root `t1` and the second one (our MLE) `t2`.

```
> t1 <- (-meand + sqrt(meand^2 + 4*meandsq) )/2 ; t1
[1] 10.3463
> t2 <- (-meand - sqrt(meand^2 + 4*meandsq) )/2 ; t2
[1] -5.2953
> n <- length(D)
> # Now calculaate second derivative at t1 and t2
> n/t1^2 - 3*sum(D^2)/t1^4 + 2*sum(D)/t1^3
[1] -0.7061484
> n/t2^2 - 3*sum(D^2)/t2^4 + 2*sum(D)/t2^3
[1] -5.267197
```

The second derivative is negative in both cases; they are both local maxima! Which peak is higher?

```
> loglike1(t1)
[1] -224.9832
> loglike1(t2)
[1] -155.4413
```

So the maximum we found is higher, which makes sense because it's within the range of the data. But we only found it because we started searching near the correct answer.

Let's plot the log likelihood function, and see what this thing looks like. We know that because the natural log function goes to minus infinity as its (positive) argument approaches zero, the log likelihood plunges to $-\infty$ at $\theta = 0$. A plot would look like a giant icicle and we would not be able to see any detail where it matters. So we will zoom in by limiting the range of the y axis. Here is the `R` code.

```
Theta <- seq(from=-15,to=20,by=0.25); Theta <- Theta[Theta!=0]
Loglike <- loglike1(Theta)
# Check where to break off the icicle
max(Loglike); Loglike[Theta==-3];  Loglike[Theta==3]

plot(Theta,Loglike,type='l',xlim=c(-15,20),ylim=c(-375,-155),
    xlab=expression(theta),ylab="Log Likelihood")
    # This is how you get Greek letters.
```

Figure 1: Log Likelihood for Example 1.1

includegraphics[width=3in]loglike1

Here is the picture. You can see the local maxima around $\theta = -5$ and $\theta = 10$, and also that the one for negative θ is a higher.

Presumably we would have reached the bad answer if we had started the search in a bad place. Let's try starting the search at $\theta = +3$.

```
> nlm(loglike2,3)
$minimum
[1] 283.7589

$estimate
[1] 64.83292

$gradient
[1] 0.701077

$code
[1] 4

$iterations
[1] 100
```

What happened?! The answer is way off, nowhere near the positive root of 10.3463. And the `minimum` (of *minus* the log likelihood) is over 283, when it would have been 224.9832 at $\theta = 10.3463$.

What happened was that the slope of the function was very steep at our starting value of $\theta = 3$, so `nlm` took a huge step in a positive direction. It was too big, and landed in a nearly flat place. Then `nlm` wandered around until it ran out of its default number of iterations (notice `iterations`=100). The exit `code` of 4 means maximum number of iterations exceeded.

It should be better if we start close to the answer, say at $\theta = 8$.

```
> nlm(loglike2,8)
$minimum
[1] 224.9832

$estimate
[1] 10.34629

$gradient
[1] -4.120564e-08

$code
[1] 1

$iterations
[1] 6
```

That's better. The moral of this story is clear. Good starting are *very* important.

Now let us look at an example of a multi-parameter problem where an explicit formula for the MLE is impossible, and numerical methods are required.

Example 1.2

Let $D_1, \ldots, D_n$ be a random sample from a Gamma distribution with parameters $\alpha > 0$ and $\beta > 0$. The probability density function is

$$f(x; \alpha, \beta) = \frac{1}{\beta^\alpha \Gamma(\alpha)} e^{-x/\beta} x^{\alpha-1}$$

for $x > 0$, and zero otherwise. Here is a random sample of size $n = 50$. For this example, the data are simulated using `R`, with known parameter values $\alpha = 2$ and $\beta = 3$. The seed for the random, number generator is set so the pseudo-random numbers can be recovered if necessary.

```
> set.seed(3201); alpha=2; beta=3
> D <- round(rgamma(50,shape=alpha, scale=beta),2); D
 [1] 20.87 13.74  5.13  2.76  4.73  2.66 11.74  0.75 22.07 10.49  7.26  5.82 13.08
[14]  1.79  4.57  1.40  1.13  6.84  3.21  0.38 11.24  1.72  4.69  1.96  7.87  8.49
[27]  5.31  3.40  5.24  1.64  7.17  9.60  6.97 10.87  5.23  5.53 15.80  6.40 11.25
[40]  4.91 12.05  5.44 12.62  1.81  2.70  3.03  4.09 12.29  3.23 10.94
> mean(D); alpha*beta
[1] 6.8782
[1] 6
> var(D); alpha*beta^2
[1] 24.90303
[1] 18
```

The parameter vector $\boldsymbol{\theta} = (\alpha, \beta)$, and the parameter space Θ is the first quadrant of $\mathbb{R}^2$.

$$\Theta = \{(\alpha, \beta) : \alpha > 0, \beta > 0\}$$

The log likelihood is

$$\begin{aligned}
\ell(\alpha, \beta) &= \ln \prod_{i=1}^{n} \frac{1}{\beta^\alpha \Gamma(\alpha)} e^{-d_i/\beta} d_i^{\alpha-1} \\
&= \ln \left(\beta^{-n\alpha} \, \Gamma(\alpha)^{-n} \exp(-\frac{1}{\beta} \sum_{i=1}^{n} d_i) \left(\prod_{i=1}^{n} d_i \right)^{\alpha-1} \right) \\
&= -n\alpha \ln \beta - n \ln \Gamma(\alpha) - \frac{1}{\beta} \sum_{i=1}^{n} d_i + (\alpha - 1) \sum_{i=1}^{n} \ln d_i.
\end{aligned}$$

The next step would be to partially differentiate the log likelihood with respect to α and β, set both partial derivatives to zero, and solve two equations in two unknowns. But even if you are confident that the gamma function is differentiable (it is), you will be unable to solve the equations. It has to be done numerically.

Define an `R` function for the minus log likelihood. Notice the `lgamma` function, a direct numerical approximation of $\ln \Gamma(\alpha)$. The plan is to numerically minimize the minus log likelihood function over all (α, β) pairs, for this particular set of data values.

```
> # Gamma minus log likelihood: alpha=a, beta=b
> gmll <- function(theta,datta)
+      {
+       a <- theta[1]; b <- theta[2]
+       n <- length(datta); sumd <- sum(datta); sumlogd <- sum(log(datta))
+       gmll <- n*a*log(b) + n*lgamma(a) + sumd/b - (a-1)*sumlogd
+      gmll
+      } # End function gmll
```

Where should the numerical search start? One approach is to start at reasonable estimates of α and β — estimates that can be calculated directly rather than by a numerical approximation. As in Example 1.1, Method of Moments estimators are a convenient, high-quality choice.

For a gamma distribution, $E(D) = \alpha\beta$ and $Var(D) = \alpha\beta^2$. So,

$$\alpha = \frac{E(D)^2}{Var(D)} \quad \text{and} \quad \beta = \frac{Var(D)}{E(D)}.$$

Replacing population moments by sample moments and writing $\widetilde{\alpha}$ and $\widetilde{\beta}$ for the resulting Method of Moments estimators, we obtain

$$\widetilde{\alpha} = \frac{\overline{D}^2}{S_D^2} \quad \text{and} \quad \widetilde{\beta} = \frac{S_D^2}{\overline{D}},$$

where $\overline{D}$ is the sample mean and S_D^2 is the sample variance. For these data, the Method of Moments estimates are reasonably close to the correct values of $\alpha = 2$ and $\beta = 3$, but they are not perfect. Parameter estimates are not the same as parameters!

```
> momalpha <- mean(D)^2/var(D); momalpha
[1] 1.899754
> mombeta <- var(D)/mean(D); mombeta
[1] 3.620574
```

Now for the numerical search. This time, we will request that the `nlm` function return the *Hessian* at the place where the search stops. The Hessian is defined as follows. Suppose we are minimizing a function $g(\theta_1, \ldots, \theta_k)$ – say, a minus log likelihood. The Hessian is a $k \times k$ matrix of mixed partial derivatives. It may be written in terms of its (i, j) element s

$$\mathbf{H} = \left[\frac{\partial^2 g}{\partial\theta_i \partial\theta_j} \right]. \tag{2}$$

In the following, notice how the `nlm` function assumes that the first argument of the function being minimized is a vector of arguments over which we should minimize, and any other arguments (in this case, the name of the data vector) can be specified by name in the `nlm` function call.

```
> gammasearch = nlm(gmll,c(momalpha,mombeta),hessian=T,datta=D); gammasearch
$minimum
[1] 142.0316

$estimate
[1] 1.805930 3.808674

$gradient
```

```
[1] 2.847002e-05 9.133932e-06

$hessian
         [,1]      [,2]
[1,] 36.68932 13.127271
[2,] 13.12727  6.222282

$code
[1] 1

$iterations
[1] 6

> eigen(gammasearch$hessian)$values
[1] 41.565137  1.346466
```

The `nlm` object `gammasearch` is a linked list. The item `minimum` is the value of the minus log likelihood function where the search stops. The item `estimate` is the point at which the search stops, so $\widehat{\alpha} = 1.805930$ and $\widehat{\beta} = 3.808674$. The `gradient` is

$$\left(-\frac{\partial \ell}{\partial \alpha}, -\frac{\partial \ell}{\partial \beta} \right)^\top .$$

Besides being the direction of steepest decrease, it's something that should be zero at the MLE. And indeed it is, give or take a bit of numerical inaccuracy.

The Hessian at the stopping place is in `gammasearch$hessian`. The Hessian is the matrix of mixed partial derivatives defined by

$$\mathbf{H} = \left[\frac{\partial^2 (-\ell)}{\partial \theta_i \partial \theta_j} \right] .$$

The rules about Hessian matrices are

- If the second derivatives are continuous, $\mathbf{H}$ is symmetric.
- If the gradient is zero at a point and $|\mathbf{H}| \neq 0$
  - If $\mathbf{H}$ is positive definite, there is a local minimum at the point.
  - If $\mathbf{H}$ is negative definite, there is a local maximum at the point.
  - If $\mathbf{H}$ has both positive and negative eigenvalues, the point is a saddle point.

The `eigen` command returns a linked list; one item is an array of the eigenvalues, and the other is the eigenvectors in the form of a matrix. Since for real symmetric matrices, positive definite is equivalent to all positive eigenvalues, it is convenient to check the eigenvalues to determine whether the numerical search has located a minimum. In this case it has. Finally, `code=1` means normal termination of the search, and `iterations=6` means the function took 6 steps downhill to reach its target.

It is very helpful to have the true parameter values $\alpha = 2$ and $\beta = 3$ for this example. $\widehat{\alpha} = 1.8$ seems pretty close, while and $\widehat{\beta} = 3.8$ seems farther off. This is a reminder of how informative confidence intervals and tests can be.

The Invariance Principle

The Invariance Principle of maximum likelihood estimation says that *the MLE of a function is that function of the MLE*. An example comes first, followed by formal details.

Example 1.3

Let $D_1, \ldots, D_n$ be a random sample from a Bernoulli distribution (1=Yes, 0=No) with parameter $\theta, 0 < \theta < 1$. The parameter space is $\Theta = (0, 1)$, and the likelihood function is

$$L(\theta) = \prod_{i=1}^{n} \theta^{d_i}(1-\theta)^{1-d_i} = \theta^{\sum_{i=1}^n d_i}(1-\theta)^{n-\sum_{i=1}^n d_i}.$$

Differentiating the log likelihood with respect to θ, setting the derivative to zero and solving yields the usual estimate $\widehat{\theta} = \overline{d}$, the sample proportion.

Now suppose that instead of the probability, we write this model in terms of the *odds* of $D_i = 1$, a re-parameterization that is often useful in categorical data analysis. Denote the odds by θ'. The definition of odds is

$$\theta' = \frac{\theta}{1-\theta} = g(\theta). \tag{3}$$

As θ ranges from zero to one, θ' ranges from zero to infinity. So there is a new parameter space: $\theta' \in \Theta' = (0, \infty)$.

To write the likelihood function in terms of θ', first solve for θ, obtaining

$$\theta = \frac{\theta'}{1+\theta'} = g^{-1}(\theta').$$

The likelihood in terms of θ' is then

$$\begin{aligned} L(g^{-1}(\theta')) &= \theta^{\sum_{i=1}^n d_i}(1-\theta)^{n-\sum_{i=1}^n d_i} \\ &= \left(\frac{\theta'}{1+\theta'}\right)^{\sum_{i=1}^n d_i} \left(1 - \frac{\theta'}{1+\theta'}\right)^{n-\sum_{i=1}^n d_i} \\ &= \left(\frac{\theta'}{1+\theta'}\right)^{\sum_{i=1}^n d_i} \left(\frac{1+\theta'-\theta'}{1+\theta'}\right)^{n-\sum_{i=1}^n d_i} \\ &= \frac{\theta'^{\sum_{i=1}^n d_i}}{(1+\theta')^n}. \end{aligned}$$

Note how re-parameterization changes the functional form of the likelihood function. The general formula is $L'(\theta') = L(g^{-1}(\theta'))$. For this example,

$$L'(\theta') = \frac{\theta'^{\sum_{i=1}^n d_i}}{(1+\theta')^n}. \tag{4}$$

At this point one could differentiate the log of (4) with respect to θ', set the derivative to zero, and solve for θ'. The point of the invariance principle is that this is unnecessary. The maximum likelihood estimator of $g(\theta)$ is $g(\widehat{\theta})$, so one need only look at (3) and write

$$\widehat{\theta'} = \frac{\widehat{\theta}}{1-\widehat{\theta}} = \frac{\overline{d}}{1-\overline{d}}\ .$$

It is often convenient to parameterize a statistical model in more than one way. The invariance principle can save a lot of work in practice, because it says that you only have to maximize the likelihood function once. It is useful theoretically too.

In Example 1.3, the likelihood function has only one maximum and the function g linking θ' to θ' is one-to-one, which is why we can write g^{-1}. This is the situation where the invariance principle is clearest and most useful. Here is a proof.

Let the parameter $\theta \in \Theta$, and re-parameterize by $\theta' = g(\theta)$. The new parameter space is $\Theta' = \{\theta' : \theta' = g(\theta), \theta \in \Theta\}$. The function $g : \Theta \to \Theta'$ is one-to-one, meaning that there exists a function g^{-1} such that $g^{-1}(g(\theta)) = \theta$ for all $\theta \in \Theta$. Suppose the likelihood function $L(\theta)$ has a unique maximum at $\widehat{\theta} \in \Theta$, so that for all $\theta \in \Theta$ with $\theta \neq \widehat{\theta}$, $L(\widehat{\theta}) > L(\theta)$. For every $\theta \in \Theta$,

$$L(\theta) = L(g^{-1}(g(\theta))) = L(g^{-1}(\theta')) = L'(\theta')$$

Maximizing $L'(\theta')$ over $\theta' \in \Theta'$ yields $\widehat{\theta'}$ satisfying $L'(\widehat{\theta'}) \geq L'(\theta')$ for all $\theta' \in \Theta'$. The invariance principle says $\widehat{\theta'} = g(\widehat{\theta})$.

Let $\theta_0 = g^{-1}(\widehat{\theta'})$ so that $g(\theta_0) = \widehat{\theta'}$. The objective is to show that this value $\theta_0 \in \Theta$ equals $\widehat{\theta}$. Suppose on the contrary that $\theta_0 \neq \widehat{\theta}$. Then because the maximum of $L(\theta)$ over Θ is unique, $L(\widehat{\theta}) > L(\theta_0)$. Therefore,

$$\begin{aligned} & L(g^{-1}(g(\widehat{\theta}))) > L(g^{-1}(g(\theta_0))) \\ \Rightarrow \quad & L'(g(\widehat{\theta})) > L'(g(\theta_0)) \\ \Rightarrow \quad & L'(g(\widehat{\theta})) > L'(\widehat{\theta'}). \end{aligned}$$

Since $g(\widehat{\theta}) \in \Theta'$, this contradicts $L'(\widehat{\theta'}) \geq L'(\theta')$ for all $\theta' \in \Theta'$, showing $\widehat{\theta} = \theta_0$. Not leaving anything to the imagination, we then have $g(\widehat{\theta}) = g(\theta_0) = \widehat{\theta'}$.

This concludes the proof, but it may be useful to establish the "obvious" fact that uniqueness of the maximum over Θ implies uniqueness of the maximum over Θ'. If $\widehat{\theta'_1}$ and $\widehat{\theta'_2}$ are two points in Θ' with $L'(\widehat{\theta'_1}) \geq L'(\theta')$ and $L'(\widehat{\theta'_2}) \geq L'(\theta')$ for all $\theta' \in \Theta'$, the preceding argument shows that $g(\widehat{\theta}) = \widehat{\theta'_1}$ and $g(\widehat{\theta}) = \widehat{\theta'_2}$. Because function values are unique, this can only happen if $\widehat{\theta'_1} = \widehat{\theta'_2}$

Interval Estimation and Testing

All the tests and confidence intervals here are based on large-sample approximations, primarily the Central Limit Theorem. See Section **??** for basic definitions and results. They are valid as the sample size $n \to \infty$, but frequently perform well for samples that are only fairly large. How big is big enough? This is a legitimate question, and the honest answer is that it depends upon the distribution of the data. In practice, people often just apply these tools almost regardless of the sample size, because nothing better is available. Some do it with their eyes closed, some squint, and some have their eyes wide open.

The basic result comes from the research of Abraham Wald (give a source) in the 1950s. *As the sample size n increases, the distribution of the maximum likelihood estimator* $\widehat{\boldsymbol{\theta}}_n$ *approaches a multivariate normal* with expected value $\boldsymbol{\theta}$ and variance-covariance matrix $\mathbf{V}_n(\boldsymbol{\theta})$. It is quite remarkable that anyone could figure this out, given that it includes cases like the Gamma, where no closed-form expressions for the maximum likelihood estimators are possible. The theorem in question is not true for every distribution, but it is true if the distribution of the data is not too strange. The precise meaning of "not too strange" is captured in a set of technical conditions called *regularity conditions*. Volume 2 of *Kendall's advanced theory of statistics* [**?**] is a good textbook source for the details.

If $\boldsymbol{\theta}$ is a $k \times 1$ matrix, then $\mathbf{V}_n(\boldsymbol{\theta})$ is a $k \times k$ matrix, called the *asymptotic covariance matrix* of the estimators. It's not too surprising that it depends on the parameter $\boldsymbol{\theta}$, and it also depends on the sample size n. Using the asymptotic covariance matrix, it is possible to construct a variety of useful tests and confidence intervals.

Fisher Information

The fact that $\mathbf{V}_n(\boldsymbol{\theta})$ depends on the unknown parameter will present no problem; substituting $\widehat{\boldsymbol{\theta}}_n$ for $\boldsymbol{\theta}$ yields an *estimated* asymptotic covariance matrix. So consider the form of the matrix $\mathbf{V}$.

Think of a one-parameter maximum likelihood problem, where we differentiate the log likelihood, set the derivative to zero and solve for θ; the solution is $\widehat{\theta}$. The log likelihood will be concave down at $\widehat{\theta}$, but the exact way it looks will depend on the distribution as well as the sample size. In particular, it could be almost flat at $\widehat{\theta}$, or it could be nearly a sharp peak, with extreme downward curvature. In the latter case, clearly the log likelihood is more informative about θ. It contains more information. One of the many good ideas of R. A. F. Fisher was that the second derivative reflects curvature, and and can be viewed as a measure of the information provided by the sample data. It is called the *Fisher Information* in his honour.

Now with increasing sample size, nearly all log likelihood functions acquire more and more downward curvature at the MLE. This makes sense – more data provide more information. But how about the information from just one observation? If you look at the second derivative of the log likelihood function,

$$\frac{\partial^2 \ell}{\partial \theta^2} = \frac{\partial^2}{\partial \theta^2} \ln \prod_{i=1}^{n} f(d_i;\theta) = \sum_{i=1}^{n} \frac{\partial^2}{\partial \theta^2} \ln f(d_i;\theta),$$

you see that it is the sum of n quantities. Each observation is contributing a piece to the downward curvature. But how much? Well, it depends on the particular data value x_i. But the data are a random sample, so in fact the contribution is a random quantity: $\frac{\partial^2}{\partial \theta^2} \ln f(X_i;\theta)$. How about the information one would *expect* an observation to contribute? Okay, take the expected value. Finally, note that because the curvature is down at the MLE, the quantity we are discussing is negative. But we want to call this "information," and it would be nicer if it were a positive number, so higher values meant more information. Okay, multiply by -1. This leads to the definition of the Fisher Information in a single observation:

$$I(\theta) = E\left[-\frac{\partial^2}{\partial \theta^2} \ln f(D_i;\theta)\right]. \tag{5}$$

The information is the same for $i = 1, \ldots, n$, and the Fisher Information in the entire sample is just $nI(\theta)$.

It was clear that Fisher was onto something good, because for many problems where the variance of $\widehat{\theta}$ can be calculated exactly, it is one divided by the Fisher Information. Subsequently Cramér and Rao discovered the *Cramér-Rao Inequality*, which says that for *any* statistic T that is an unbiased estimator of θ,

$$Var(T) \geq \frac{1}{nI(\theta)}.$$

That's impressive, because to have a small variance is a great property in an estimator; it means precise estimation. The Cramér-Rao inequality tells us that in terms of variance, one cannot do better than an unbiased estimator whose variance equals hte reciprocal (inverse) of the Fisher Information, and many MLEs do that. Subsequently, Wald[3] showed that under some regularity conditions, the variances of maximum likelihood estimators in general attain the Cramér-Rao lower bound as $n \to \infty$. Thus, to learn the asymptotic variance of $\widehat{\theta}$, you do not need an explicit formula for $\widehat{\theta}$. All you need is the Fisher Information. Also, in terms of variance nothing can beat maximum likelihood estimation, at least for large samples. So if the distribution of

[3]Need a reference

the data is known so you can write down the likelihood, it is difficult to justify any method of estimation other than maximum likelihood.

Calculating the expected value in (5) is often not too hard because taking the log and differentiating twice results in some simplification; it's a source of many fun homework problems. But still it can be a chore, especially for multiparameter problems, which will be taken up shortly. For larger sample sizes, the Law of Large Numbers (Section **??**) guarantees that the expected value can be approximated quite well by a sample mean, so that

$$I(\theta) = E\left(-\frac{\partial^2}{\partial\theta^2}\ln f(D_1;\theta)\right] \approx \frac{1}{n}\sum_{i=1}^{n} -\frac{\partial^2}{\partial\theta^2}\ln f(D_i;\theta).$$

This is sometimes called the *observed* Fisher Information.

Multiplying the observed Fisher Information by n to get the approximate information in the entire sample yields

$$\sum_{i=1}^{n} -\frac{\partial^2}{\partial\theta^2}\ln f(D_i;\theta) = \frac{\partial^2}{\partial\theta^2}\sum_{i=1}^{n} -\ln f(D_i;\theta) = \frac{\partial^2}{\partial\theta^2}\left(-\ln\prod_{i=1}^{n} f(D_i;\theta)\right).$$

That's just the second derivative of the minus log likelihood.

The parameter θ is unknown, so to get the *estimated* Fisher Information in the whole sample, substitute $\widehat{\theta}$. The result is

$$\frac{\partial^2}{\partial\theta^2}\left(-\ln\prod_{i=1}^{n} f(D_i;\widehat{\theta})\right).$$

That's the second derivative of minus the log likelihood, evaluated at the maximum likelihood estimate. And, it's a function of the sample data that is not a function of any unknown parameters; in other words it is a statistic. If you have already carried out the second derivative test to check that you really had a maximum, all you need to do to estimate the variance of $\widehat{\theta}$ is take the reciprocal of the second derivative and multiply by -1. It is truly remarkable how neatly this all works out.

Generalization to the multivariate case is very natural. Now the parameter is $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)^\top$ and the Fisher Information *Matrix* is a $k \times k$ matrix of (expected) mixed partial derivatives, defined by

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\theta}) = \left[-E\left(\frac{\partial^2}{\partial\theta_i\partial\theta_j} f(\mathbf{D}_1;\boldsymbol{\theta})\right)\right],$$

where the boldface $\mathbf{D}_i$ is an acknowledgement that the data might also be multivariate.

In the estimated observed Fisher Information evaluated at the MLE (which will simply be called the "Fisher Information Matrix" unless other wise noted), expected value is replaced by a sample mean and $\boldsymbol{\theta}$ is replaced by $\widehat{\boldsymbol{\theta}}$. The formula is

$$\boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{\theta}}) = \left[\frac{\partial^2}{\partial\theta_i\partial\theta_j}\left(-\ln\prod_{q=1}^{n} f(\mathbf{D}_q;\widehat{\boldsymbol{\theta}})\right)\right] = \left[\frac{\partial^2}{\partial\theta_i\partial\theta_j}\left(-\ell(\widehat{\boldsymbol{\theta}})\right)\right]. \tag{6}$$

In the one-dimensional case, one over the estimated Fisher Information is the (estimated) asymptotic variance of the maximum likelihood estimator. *In the multi-parameter case, the Fisher Information is a matrix, and the estimated asymptotic variance-covariance matrix is its inverse.* Denoting the estimated asymptotic covariance matrix by $\widehat{\mathbf{V}}_n$, we have

$$\widehat{\mathbf{V}}_n = \boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{\theta}}_n)^{-1}. \tag{7}$$

Now comes the really good part. Comparing Formula (6) for the Fisher Information to Formula (2) for the Hessian, we see that they are exactly the same. And *the Hessian evaluated at* $\widehat{\boldsymbol{\theta}}$ *is a by-product of the numerical search for the MLE*[4].

So to get the asymptotic covariance matrix, minimize minus the log likelihood, tell the software to give you the Hessian, and calculate the inverse by computer. The theoretical story may be a bit long here, but what you have to do in practice is quite simple.

Continuing with the Gamma distribution Example 1.2, the Hessian is

```
> gammasearch$hessian
         [,1]      [,2]
[1,] 36.68932 13.127271
[2,] 13.12727  6.222282
```

and the asymptotic covariance is just

```
> Vhat = solve(gammasearch$hessian); V
           [,1]       [,2]
[1,]  0.1111796 -0.2345577
[2,] -0.2345577  0.6555638 .
```

The diagonal elements of $\widehat{\mathbf{V}}$ are the estimated variances of the sampling distributions of $\widehat{\alpha}$ and $\widehat{\beta}$ respectively, and their square roots are the standard errors.

```
> SEalphahat = sqrt(Vhat[1,1]); SEbetahat = sqrt(Vhat[2,2])
```

In general, let θ denote an element of the parameter vector, let $\widehat{\theta}$ be its maximum likelihood estimator, and let the standard error of $\widehat{\theta}$ be written $S_{\widehat{\theta}}$. Then Wald's Central Limit Theorem for maximum likelihood estimators tells us that

$$Z = \frac{\widehat{\theta} - \theta}{S_{\widehat{\theta}}} \tag{8}$$

has an approximate standard normal distribution. In particular, for the Gamma example

$$Z_1 = \frac{\widehat{\alpha} - \alpha}{S_{\widehat{\alpha}}} \quad \text{and} \quad Z_2 = \frac{\widehat{\beta} - \beta}{S_{\widehat{\beta}}}$$

may be treated as standard normal.

Confidence Intervals

These quantities may be used to produce both tests and confidence intervals. For example, a 95% confidence interval for the parameter θ is obtained as follows.

$$\begin{aligned} 0.95 &\approx Pr\{-1.96 \leq Z \leq 1.96\} \\ &= Pr\left\{-1.96 \leq \frac{\widehat{\theta} - \theta}{S_{\widehat{\theta}}} \leq 1.96\right\} \\ &= Pr\left\{\widehat{\theta} - 1.96\, S_{\widehat{\theta}} \leq \theta \leq \widehat{\theta} + 1.96\, S_{\widehat{\theta}}\right\} \end{aligned}$$

[4]At least for generic numerical minimization routines like R's `nlm`. Some specialized methods like iterative proportional fitting of log-linear models and Fisher scoring (iteratively re-weighted least squares) for generalized linear models maximize the likelihood indirectly and do not require calculation of the Hessian.

This could also be written $\widehat{\theta} \pm 1.96\, S_{\widehat{\theta}}$.

If you are used to seeing confidence intervals with a $\sqrt{n}$ and wondering where it went, recall that $S_{\overline{X}} = \frac{S}{\sqrt{n}}$. The $\sqrt{n}$ is also present in the confidence interval for θ, but it is embedded in $S_{\widehat{\theta}}$.

Here are the 95% confidence intervals for the Gamma distribution example:

```
> alphahat = gammasearch$estimate[1]; betahat = gammasearch$estimate[2]
> Lalpha = alphahat - 1.96*SEalphahat; Ualpha = alphahat + 1.96*SEalphahat
> Lbeta = betahat - 1.96*SEbetahat; Ubeta = betahat + 1.96*SEbetahat
> cat("\nEstimated alpha = ",round(alphahat,2),"  95 percent CI from ",
+     round(Lalpha,2)," to ",round(Ualpha,2), "\n\n")

Estimated alpha =  1.81   95 percent CI from  1.15  to  2.46

> cat("\nEstimated beta = ",round(betahat,2),"  95 percent CI from ",
+     round(Lbeta,2)," to ",round(Ubeta,2), "\n\n")

Estimated beta =  3.81   95 percent CI from  2.22  to  5.4
```

Notice that while the parameter estimates may not seem very accurate, the 95% confidence intervals do include the true parameter values $\alpha = 2$ and $\beta = 3$.

Z-tests

The standard normal variable in (8) can be used to form a Z-test of $H_0 : \theta = \theta_0$ using

$$Z = \frac{\widehat{\theta} - \theta_0}{S_{\widehat{\theta}}}.$$

So for example, suppose the data represent time intervals between events occurring in time, and we wonder whether the events arise from a Poisson process. In this case the distribution of times would be exponential, which means $\alpha = 1$. To test this null hypothesis at the 0.05 level,

```
> Z = (alphahat-1)/SEalphahat; Z
[1] 2.417046
> pval = 2*(1-pnorm(abs(Z))); pval # Two-sided test
[1] 0.01564705
```

So, the null hypothesis is rejected, and because the value is positive, the conclusion is that the true value of α is greater than one[5].

[5]The following basic question arises from time to time. Suppose a null hypothesis is rejected in favour of a two-sided alternative. Are we then "allowed" to look at the sign of the test statistic and conclude that $\theta < \theta_0$ or $\theta > \theta_0$, or must we just be content with saying $\theta \neq \theta_0$? The answer is that directional conclusions are theoretically justified as well as practically desirable. Think of splitting up the two-sided level α test (call it the *overall test*) into two one-sided tests with significance level $\alpha/2$. The null hypotheses of these tests are $H_{0,a} : \theta \leq \theta_0$ and $H_{0,b} : \theta \geq \theta_0$. Exactly one of these null hypotheses will be rejected if and only if the null hypothesis of the overall test is rejected, so the set of two one-sided tests is fully equivalent to the overall two-sided test. And directional conclusions from the one-sided tests are clearly justified.

On a deeper level, notice that the null hypothesis of the overall test is the intersection of the null hypotheses of the one-sided tests, and its critical region (rejection region) is the union of the critical regions of the one-sided tests. This makes the two one-sided tests a set of *union-intersection multiple comparisons*, which are always simultaneously protected against Type I error at the significance level of the overall test. Performing the two-sided test and then following up with a one-sided test is very much like following up a statistically significant ANOVA with Scheffeé tests. Indeed, Scheffé tests are another example of union-intersection multiple comparisons. See [**?**] for details.

When statistical software packages display this kind of large-sample Z-test, they usually just divide $\widehat{\theta}$ by its standard error, testing the null hypothesis $H_0 : \theta = 0$. For parameters like regression coefficients, this is usually a good generic choice.

2 Wald Tests

The approximate multivariate normality of the MLE can be used to construct a larger class of hypothesis tests for *linear* null hypotheses. A linear null hypothesis sets a collection of linear combinations of the parameters to zero. Suppose $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)^\top$ is a $k \times 1$ vector. A linear null hypothesis can be written

$$H_0 : \mathbf{C}\boldsymbol{\theta} = \mathbf{h},$$

where $\mathbf{C}$ is an $r \times k$ matrix of constants, with rank r, $r \leq k$. As an example let $\boldsymbol{\theta} = (\theta_1, \ldots \theta_7)^\top$, and the null hypothesis is

$$\theta_1 = \theta_2, \quad \theta_6 = \theta_7, \quad \frac{1}{3}(\theta_1 + \theta_2 + \theta_3) = \frac{1}{3}(\theta_4 + \theta_5 + \theta_6).$$

This may be expressed in the form $\mathbf{C}\boldsymbol{\theta} = \mathbf{h}$ as follows:

$$\begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1 & 1 & 1 & -1 & -1 & -1 & 0 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Recall from Section **??** of this appendix that if $\mathbf{X} \sim N_k(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, and $\mathbf{C}$ is an $r \times k$ constant matrix of rank r, then

$$\mathbf{C}\mathbf{X} \sim N_r(\mathbf{C}\boldsymbol{\mu}, \mathbf{C}\boldsymbol{\Sigma}\mathbf{C}^\top)$$

and

$$(\mathbf{C}\mathbf{X} - \mathbf{C}\boldsymbol{\mu})^\top (\mathbf{C}\boldsymbol{\Sigma}\mathbf{C}^\top)^{-1} (\mathbf{C}\mathbf{X} - \mathbf{C}\boldsymbol{\mu}) \sim \chi^2(r).$$

Similar facts hold asymptotically — that is approximately, as the sample size n approaches infinity. Because (approximately) $\widehat{\boldsymbol{\theta}}_n \sim N_k(\boldsymbol{\theta}, \widehat{\mathbf{V}}_n)$,

$$\mathbf{C}\widehat{\boldsymbol{\theta}}_n \sim N_r(\mathbf{C}\boldsymbol{\theta}, \mathbf{C}\widehat{\mathbf{V}}_n\mathbf{C}^\top)$$

and

$$(\mathbf{C}\widehat{\boldsymbol{\theta}}_n - \mathbf{C}\boldsymbol{\theta})^\top (\mathbf{C}\widehat{\mathbf{V}}_n\mathbf{C}^\top)^{-1} (\mathbf{C}\widehat{\boldsymbol{\theta}}_n - \mathbf{C}\boldsymbol{\theta}) \sim \chi^2(r).$$

So, if $H_0 : \mathbf{C}\boldsymbol{\theta} = \mathbf{h}$ is true, we have the Wald test statistic

$$W_n = (\mathbf{C}\widehat{\boldsymbol{\theta}}_n - \mathbf{h})^\top (\mathbf{C}\widehat{\mathbf{V}}_n\mathbf{C}^\top)^{-1} (\mathbf{C}\widehat{\boldsymbol{\theta}}_n - \mathbf{h}) \sim \chi^2(r), \tag{9}$$

where again,

$$\widehat{\mathbf{V}}_n = \boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{\theta}})^{-1} = \left[\frac{\partial^2}{\partial\theta_i \partial\theta_j} \left(-\ell(\widehat{\boldsymbol{\theta}}) \right) \right]^{-1}.$$

Here is a test of $H_0 : \alpha = \beta$ for the Gamma distribution example. A little care must be taken to ensure that the matrices in (9) are the right size.

```
> #  H0: C theta = 0 is that alpha = beta <=> alpha-beta=0
> # Name C is used by R
> CC = rbind(c(1,-1)); is.matrix(CC); dim(CC)
[1] TRUE
[1] 1 2
> thetahat = as.matrix(c(alphahat,betahat)); dim(thetahat)
[1] 2 1
> W = t(CC%*%thetahat) %*% solve(CC%*%Vhat%*%t(CC)) %*% CC%*%thetahat
> W = as.numeric(W) # it was a 1x1 matrix
> pval2 = 1-pchisq(W,1)
> cat("Wald Test:  W = ", W, ", p = ", pval2, "\n")
Wald Test:  W =  3.245501 , p =  0.07161978
```

We might as well define a function to do Wald tests in general. In the function `WaldTest`, the null hypothesis is $\mathbf{L}\boldsymbol{\theta} = \mathbf{h}$, but that's just because the name `C` is used by `R` for contrasts. The function returns a pair of quantities, the Wald test statistic and the p-value.

```
> WaldTest = function(L,thetahat,h=0) # H0: L theta = h
+     {
+     WaldTest = numeric(2)
+     names(WaldTest) = c("W","p-value")
+     dfree = dim(L)[1]
+     W = t(L%*%thetahat-h) %*% solve(L%*%Vhat%*%t(L)) %*% (L%*%thetahat-h)
+     W = as.numeric(W)
+     pval = 1-pchisq(W,dfree)
+     WaldTest[1] = W; WaldTest[2] = pval
+     WaldTest
+     } # End function WaldTest
```

Here is the same test of $H_0 : \alpha = \beta$ done immediately above, just to test out the function. Notice that the default value of $\mathbf{h}$ in $H_0 : \mathbf{L}\boldsymbol{\theta} = \mathbf{h}$ is zero, so it does not have to be specified. The matrix `CC` has already been created, and the computed values are the same as before, naturally.

```
> WaldTest(CC,as.matrix(c(alphahat,betahat)))
         W    p-value
3.24550127 0.07161978
```

Here is a test of $H_0 : \alpha = 2, \beta = 3$, which happen to be the true parameter values. The null hypothesis is not rejected.

```
> C2 = rbind(c(1,0),
+           c(0,1) )
> WaldTest(C2,as.matrix(c(alphahat,betahat)),c(2,3))
        W   p-value
1.3305497 0.5141322
```

Finally, here is a test of $H_0 : \alpha = 1$, which was done earlier with a Z-test.

```
> WaldTest(t(c(1,0)),as.matrix(c(alphahat,betahat)),1)
         W    p-value
5.84210645 0.01564708
> Z; pval
```

```
[1] 2.417045
[1] 0.01564708
> Z^2
[1] 5.842106
```

The results of the Wald and Z tests are identical, with $W_n = Z^2$. In general, suppose the matrix $\mathbf{C}$ in $H_0 : \mathbf{C}\boldsymbol{\theta} = \mathbf{h}$ has just a single row, and that row contains one 1 in position j and all the rest zeros. Take a look at Formula (9) for the Wald test statistic. Pre-multiplying by $\mathbf{C}$ in $\mathbf{C}\widehat{\mathbf{V}}_n$ picks out row j of $\widehat{\mathbf{V}}_n$, and post-multiplying by $\mathbf{C}^\top$ picks out column j of the result, so that $\mathbf{C}\widehat{\mathbf{V}}_n\mathbf{C}^\top = \widehat{v}_{j,j}$, and inverting it puts it in the denominator. In the numerator, $(\mathbf{C}\widehat{\boldsymbol{\theta}}_n - \mathbf{h})^\top(\mathbf{C}\widehat{\boldsymbol{\theta}}_n - \mathbf{h}) = (\widehat{\theta}_j - \theta_{j,0})^2$, so that $W_n = Z^2$. Thus, squaring a large-sample Z-test gives a Wald chisquare test with one degree of freedom.

3 Likelihood Ratio Tests

Likelihood ratio tests fall into two categories, exact and large-sample. The main examples of exact likelihood ratio tests include are the standard F-tests and t-tests associated with regression and the analysis of variance for normal data. Here, we concentrate on the large-sample likelihood ratio tests.

Consider the following hypothesis-testing framework. The data are $D_1, \ldots, D_n$. The distribution of these independent and identically distributed random variables depends on the parameter θ, and we are testing a null hypothesis H_0.

$$\begin{array}{l} D_1, \ldots, D_n \stackrel{i.i.d.}{\sim} P_\theta,\ \theta \in \Theta, \\ H_0 : \theta \in \Theta_0 \text{ v.s. } H_A : \theta \in \Theta \cap \Theta_0^c, \end{array}$$

For example, let $D_1, \ldots, D_n \stackrel{i.i.d.}{\sim} N(\mu, \sigma^2)$. The null hypothesis is $H_0 : \mu = \mu_0$ v.s. versus $H_A : \mu \neq \mu_0$. The full parameter space is $\Theta = \{(\mu, \sigma^2) : -\infty < \mu < \infty, \sigma^2 > 0\}$ and the restricted parameter space is $\Theta_0 = \{(\mu, \sigma^2) : \mu = \mu_0, \sigma^2 > 0\}$. The full and restricted parameter spaces are shown in Figure 2.

Figure 2: Full versus reduced parameter spaces for $H_0 : \mu = \mu_0$ versus $H_A : \mu \neq \mu_0$

includegraphics[width=5in]Pictures/ParameterSpace

In general, the data have likelihood function

$$L(\theta) = \prod_{i=1}^{n} f(d_i; \theta),$$

where $f(d_i; \theta)$ is the density or probability mass function evaluated at d_i. Let $\widehat{\theta}$ denote the usual Maximum Likelihood Estimate (MLE). That is, it is the parameter value for which the likelihood function is greatest, over all $\theta \in \Theta$. Let $\widehat{\theta}_0$ denote the *restricted* MLE. The restricted MLE is the parameter value for which the likelihood function is greatest, over all $\theta \in \Theta_0$. This MLE is *restricted* by the null hypothesis $H_0 : \theta \in \Theta_0$. It should be clear that $L(\widehat{\theta}_0) \leq L(\widehat{\theta})$, so that the *likelihood ratio*.

$$\lambda = \frac{L(\widehat{\theta}_0)}{L(\widehat{\theta})} \leq 1.$$

The likelihood ratio will equal one if and only if the overall MLE $\widehat{\theta}$ is located in Θ_0. In this case, there is no reason to reject the null hypothesis.

Suppose that the likelihood ratio is strictly less than one. If it's a *lot* less than one, then the data are a lot less likely to have been observed under the null hypothesis than under the alternative hypothesis, and the null hypothesis is questionable. This is the basis of the likelihood ratio tests.

If λ is small (close to zero), then $\ln(\lambda)$ is a large negative number, and $-2\ln\lambda$ is a large positive number.

Tests will be based on

$$\begin{aligned} G &= -2\ln\left(\frac{\max_{\theta\in\Theta_0} L(\theta)}{\max_{\theta\in\Theta} L(\theta)}\right) \\ &= -2\ln\left(\frac{L(\widehat{\theta}_0)}{L(\widehat{\theta})}\right) \\ &= -2\ln L(\widehat{\theta}_0) - [-2\ln L(\widehat{\theta})] \\ &= 2\left(-\ell(\widehat{\theta}_0) - [-\ell(\widehat{\theta})]\right). \end{aligned} \tag{10}$$

Thus, the test statistic G is the *difference* between two -2 log likelihood functions. This means that to carry out a test, you can minimize $-\ell(\theta)$ twice, first over all $\theta \in \Theta$, and then over all $\theta \in \Theta_0$. The test statistic is the difference between the two minimum values, multiplied by two.

If the null hypothesis is true, then the test statistic G has, if the sample size is large, an approximate chisquare distribution, with degrees of freedom equal to the difference of the *dimension* of Θ and Θ_0. For example, if the null hypothesis is that 4 elements of θ equal zero, then the degrees of freedom are equal to 4. If the null hypothesis imposes r linearly independent linear restrictions on θ (as in $H_0 : \mathbf{C}\boldsymbol{\theta} = \mathbf{h}$), then the degrees of freedom equal r, the number or rows in $\mathbf{C}$. Another way to obtain the degrees of freedom is by counting the equal signs in the null hypothesis.

The p-value associated with the test statistic G is $Pr\{X > G\}$, where X is a chisquare random variable with r degrees of freedom. If $p < \alpha$, we reject H_0 and call the results "statistically significant." The standard choice is $\alpha = 0.05$.

Many null hypotheses are linear statements of the form $H_0 : \mathbf{C}\boldsymbol{\theta} = \mathbf{h}$, but some are not. To take a simple example, suppose you wanted to test $H_0 : \sigma^2 = \mu^2$ based on a normal random sample. It seems like the degrees of freedom should equal one, but can this be justified formally?

The original proof published in 1938 by Wilks [**?**] applies to linear null hypotheses, and if you look at high-level textbooks like the *Advanced Theory of Statistics* [**?**], you will find only Wilks' proof, without modification. A way around this that often works is to use the Invariance Principle on Page 11. Suppose the null hypothesis is that one or more non-linear functions of θ equal zero. If you can make those functions part of a function that is one-to-one, then re-parameterize. Your null hypothesis is now a linear null hypothesis in the new paraameter space. Wilks' theorem applies, and you are done. Furthermore, you don't have to literally re-parameterize. A glance at the proof of the Invariance Principle confirms that the likelihood ratio test statistic is the same under the original and re-parameterized models. Thus, the degrees of freedon equals he number of equals signs in the null hypothesis, period.

For the example of $H_0 : \sigma^2 = \mu^2$, let $\theta_1' = \sigma^2 - \mu^2$ and $\theta_2' = \mu$. The function is one-to-one, because $\mu = \theta_2'$ and $\sigma^2 = \theta_1' + \theta_2'^2$. The null hypothesis is $H_0 : \theta_1' = 0$. That's is a linear null hypothesis, so by Wilks' Theorem, the test statistic has a chi-squared distribution with $df = 1$.

Sometimes this lovely trick does not work. In a regression, it is easy to test the null hypothesis that β_1 and β_2 are both zero; this is a linear null hypothesis. But suppose that you want to test

the null hypothesis that β_1 *or* β_2 (or maybe both) are equal to zero. This is reasonable and attractive, because the alternative is that they are both non-zero, and it would be nice to have a single test for this. The null hypothesis is $H_0 : \beta_1\beta_2 = 0$, which is non-linear. Furthermore, any function that yields $\theta_1' = \beta_1\beta_2 = 0$ can't be one-to-one, because recovering β_1 or β_2 would potentially involve dividing by zero. Thus, while it would be perfectly possible to obtain the restricted MLE $\widehat{\theta}_0$ numerically and calculate the likelihood ratio statistic, its distribution under the null hypothesis is mysterious (to me). So, transforming a non-linear null hypothesis into a linear one by a one-to-one re-parameterization is a method that often works, but not always.

To illustrate the likelihood ratio tests, consider (one last time) the Gamma distribution Example 1.2. For comparison, the likelihood ratio method will be used test the same three null hypotheses that were tested earlier using Wald tests. They are

- $H_0 : \alpha = 1$
- $H_0 : \alpha = \beta$
- $H_0 : \alpha = 2, \beta = 3$

For $H_0 : \alpha = 1$, the restricted parameter space is $\Theta_0 = \{(\alpha, \beta) : \alpha = 1, \beta > 0\}$. Because the Gamma distribution with $\alpha = 1$ is exponential, the restricted MLE is $\widehat{\theta}_0 = (1, \overline{d})$. It is more informative, though, to use numerical methods.

To maximize the likelihood function (or minimize minus the log likelihood) over Θ_0, it might be tempting to impose the restriction on θ, simplify the log likelihood, and write the code for a new function to minimize. But this strategy is *not* recommended. It's time consuming, and mistakes are possible. In the `R` work shown below, notice how the function `gmll1` is just a "wrapper" for the unrestricted minus log likelihood function `gmll`. It is a function of β (and the data, of course), but all it does is call `gmll` with α set to one and β free to vary.

```
> gmll1 <- function(b,datta) # Restricted gamma minus LL with alpha=1
+     { gmll1 <- gmll(c(1,b),datta)
+       gmll1
+     } # End of function gmll1
> mean(D) # Resticted MLE of beta, just to check
[1] 6.8782
```

The next step is to invoke the nonlinear minimization function `nlm`. The second argument is a (vector of) starting value(s). Starting the search at $\beta = 1$ turns out to be unfortunate.

```
> gsearch1 <- nlm(gmll1,1,datta=D); gsearch1
$minimum
[1] 282.6288

$estimate
[1] 278.0605

$gradient
[1] 0.1753689

$code
[1] 4

$iterations
[1] 100
```

The answer `g1search$estimate=278.0605` is way off the correct answer of $\overline{d} = 6.8782$, it took 100 steps, and the exit code of 4 means the function ran out of the default number of iterations. Starting at the unrestricted $\widehat{\beta}$ works better.

```
> gsearch1 <- nlm(gmll1,betahat,datta=D); gsearch1
$minimum
[1] 146.4178

$estimate
[1] 6.878195

$gradient
[1] -1.768559e-06

$code
[1] 1

$iterations
[1] 7
```

That's better. Good starting values are important! Now the test statistic is easy to calculate.

```
> G = 2 * (gsearch1$minimum-gammasearch$minimum); pval = 1-pchisq(G,df=1)
> G; pval
[1] 8.772448
[1] 0.003058146
```

Let us carry out the other two tests, and then compare the Wald and likelihood ratio test results together in a table.

For $H_0 : \alpha = \beta$, the restricted parameter space is $\Theta_0 = \{(\alpha, \beta) : \alpha = \beta > 0\}$.

```
> gmll2 <- function(ab,datta) # Restricted gamma minus LL with alpha=1
+     { gmll2 <- gmll(c(ab,ab),datta)
+       gmll2
+     } # End of function gmll2
> abstart = (alphahat+betahat)/2
> gsearch2 <- nlm(gmll2,abstart,datta=D); gsearch2
Warning messages:
1: NaNs produced in: log(x)
2: NA/Inf replaced by maximum positive value
$minimum
[1] 144.1704

$estimate
[1] 2.562369

$gradient
[1] -4.991384e-07

$code
[1] 1
```

```
$iterations
[1] 4

> G = 2 * (gsearch2$minimum-gammasearch$minimum); pval = 1-pchisq(G,df=1)
> G; pval
[1] 4.277603
[1] 0.03861777
```

This seems okay; it only took 4 iterations and the exit code of 1 is a clean bill of health. But the warning messages are a little troubling. Probably they just indicate that the search tried a negative parameter value, outside the parameter space. The `R` function `nlminb` does minimization with bounds. Let's try it.

```
> gsearch2b <- nlminb(start=abstart,objective=gmll2,lower=0,datta=D); gsearch2b
$par
[1] 2.562371

$objective
[1] 144.1704

$convergence
[1] 0

$message
[1] "relative convergence (4)"

$iterations
[1] 5

$evaluations
function gradient
       7        8
```

Since `nlminb` gives almost the same restricted $\widehat{\alpha} = \widehat{\beta} = 2.5624$ (and no warnings), the warning messages from `nlm` were probably nothing to worry about.

Finally, for $H_0 : \alpha = 2, \beta = 3$ the restricted parameter space Θ_0 is a single point and no optimization is necessary. All we need to do is calculate the minus log likelihood there.

```
> G = 2 * (gmll(c(2,3),D)-gammasearch$minimum); pval = 1-pchisq(G,df=1)
> G; pval
[1] 2.269162
[1] 0.1319713
```

The top panel of Table 1 shows the Wald and likelihood ratio tests that have been done on the Gamma distribution data. But this is $n = 50$, which is not a very large sample. In the lower panel, the same tests were done for a sample of $n = 200$, formed by adding another 150 cases to the original data set. The results are typical; the χ^2 values are much closer except where they are far out on the tails, and both test lead to the same conclusions (though not always to the truth).

Like the Wald tests, likelihood ratio tests are very flexible and are distributed approximately as chi-square under the null hypothesis for large samples. In fact, they are *asymptotically equivalent*

Table 1: Tests on data from a gamma distribution with $\alpha = 2$ and $\beta = 3$

$n = 50$				
	Wald		Likelihood Ratio	
H_0	χ^2	p-value	χ^2	p-value
$\alpha = 1$	5.8421	0.0156	8.7724	0.0031
$\alpha = \beta$	3.2455	0.0762	4.2776	0.0386
$\alpha = 2, \beta = 3$	1.3305	0.5141	2.2692	0.1320
$n = 200$				
$\alpha = 1$	34.1847	5.01e-09	58.2194	2.34e-14
$\alpha = \beta$	0.9197	0.3376	0.9664	0.3256
$\alpha = 2, \beta = 3$	1.5286	0.4657	1.2724	0.2593

Table 2: Wald versus likelihood ratio: Type I error in 10,000 simulated datasets

	n				
Test	50	100	250	500	1000
Wald	1180	1589	1362	0749	0556
Likelihood Ratio	0330	0391	0541	0550	0522

under H_0, meaning that if the null hypothesis is true, the difference between the likelihood ratio statistic and the Wald statistic goes to zero in probability as the sample size approaches infinity.

Since the Wald and likelihood ratio tests are equivalent, does it matter which one you use? The answer is that usually, Wald tests and likelihood ratio tests lead to the same conclusions and their numerical values are close. But the tests are only equivalent as $n \to \infty$. When there is a meaningful difference, the likelihood ratio tests usually perform better, especially in terms of controlling Type I error rate for relatively small sample sample sizes.

Table 2 below contains the most extreme example I know. For a particular structural equation model with normal data (details don't matter for now), ten thousand data sets were randomly generated so that the null hypothesis was true. This was done for several sample sizes: $n = 50, 100, 250, 500$ and $1,000$. Using each of the 50,000 resulting data sets, the null hypothesis was tested with a Wald test and a likelihood ratio test at the $\alpha = 0.05$ significance level. If the asymptotic results held, we would expect both tests to reject H_0 500 times at each sample size.

So for this deliberately nasty example, the Wald test requires $n = 1,000$ before it settles down to something like the theoretical 0.05 significance level. The likelihood ratio test needs $n = 250$, and for smaller sample sizes it is conservative, with a Type I error rate somewhat *lower* than 0.05. In general, when the Wald and likelihood ratio tests have a contest of this sort, it is usually a draw. When there is a winner, it is always the likelihood ratio test, but the margin of victory is seldom as large as this.