5601 Notes: The Sandwich Estimator

Charles J. Geyer

July 13, 2013

Contents

1 Maximum Likelihood Estimation

Before we can learn about the "sandwich estimator" we must know the basic theory of maximum likelihood estimation.

1.1 Likelihood for One Observation

Suppose we observe data x, which may have any structure, scalar, vector, categorical, whatever, and is assumed to be distributed according to the probability density function f_θ. The probability of the data $f_\theta(x)$ thought of as a function of the parameter for fixed data rather than the other way around is called the *likelihood function*

$$L_x(\theta) = f_\theta(x). \tag{1}$$

For a variety of reasons, we almost always use the *log likelihood function*

$$l_x(\theta) = \log L_x(\theta) = \log f_\theta(x) \tag{2}$$

instead of the likelihood function (1). The way likelihood functions are used, it makes no difference if an arbitrary function of the data that does not depend on the parameter is added to a log likelihood, that is,

$$l_x(\theta) = \log f_\theta(x) + h(x) \tag{3}$$

is just as good a definition as (2), regardless of what the function h is.

1.2 Likelihood for Many IID Observations

Suppose X_1, …, X_n are IID random variables having common probability density function f_θ. Then the joint density function for the data vector $\mathbf{x} = (x_1, \ldots, x_n)$ is

$$f_{n,\theta}(\mathbf{x}) = \prod_{i=1}^{n} f_\theta(x_i)$$

and plugging this in for f_θ in (2) gives

$$l_n(\theta) = \sum_{i=1}^{n} \log f_\theta(x_i). \tag{4}$$

(We have a sum instead of a product because the log of a product is the sum of the logs.) We have changed the subscript on the log likelihood from x to n to follow convention. The log likelihood function l_n still depends on the whole data vector $\mathbf{x}$ as the right hand side of (4) makes clear even though the left hand side of (4) no longer mentions the data explicitly.

As with the relation between (2) and (3) we can add an arbitrary function of the data (that does not depend on the parameter) to the right hand side of (4) and nothing of importance would change.

1.3 Maximum Likelihood Estimation

The value of the parameter θ that maximizes the likelihood or log likelihood [any of equations (1), (2), or (3)] is called the *maximum likelihood estimate* (MLE) $\hat{\theta}$. Generally we write $\hat{\theta}_n$ when the data are IID and (4) is the log likelihood.

We are a bit unclear about what we mean by "maximize" here. Both local and global maximizers are used. Different theorems apply to each. Under some conditions, the global maximizer is the optimal estimator, "optimal" here meaning consistent and asymptotically normal with the smallest possible asymptotic variance. Under other conditions, the global maximizer may fail to be even consistent (which is the worst property an estimator can have, being unable to get close to the truth no matter how much data is available) but there exists a local maximizer that is optimal. Thus both global and local optimizers are of theoretical interest and neither is always better than the other. Thus both are used, although the difficulty of global optimization means that it is rarely used except for uniparameter models (when the one-dimensional optimization can be done by grid search).

Regardless of whether we use a local or global maximizer. It makes no difference which of (1), (2), or (3) we choose to maximize. Since the log function is increasing, (1) and (2) have the same maximizers. Since adding a constant does not change the location the maximum, (2) and (3) have the same maximizers.

1.4 Log Likelihood Derivatives

We are interested in the first two derivatives l'_x and l''_x of the log likelihood. (We would write l'_n and l''_n in the IID sample size n case.) Since the derivative of a term not containing the variable (with respect to which we are differentiating) is zero, there is no difference between the derivatives of (2) and (3).

It can be proved by differentiating the identity

$$\int f_\theta(x) \, dx = 1$$

under the integral sign that

$$E_\theta\{l'_X(\theta)\} = 0 \tag{5a}$$

$$\operatorname{var}_\theta\{l'_X(\theta)\} = -E_\theta\{l''_X(\theta)\} \tag{5b}$$

1.5 Fisher Information

Either side of the identity (5b) is called *Fisher information* (named after R. A. Fisher, the inventor of the method maximum likelihood and the creator of most of its theory, at least the original version of the theory). It is denoted $I(\theta)$, so we have two ways to calculate Fisher information

$$I(\theta) = \operatorname{var}_\theta\{l'_X(\theta)\} \tag{6a}$$

$$I(\theta) = -E_\theta\{l''_X(\theta)\} \tag{6b}$$

When we have IID data and are writing l_n instead of l_x, we write $I_n(\theta)$ instead of $I(\theta)$ because it is different for each sample size n.

Then (6b) becomes

$$\begin{aligned} I_n(\theta) &= -E_\theta\{l_n''(\theta)\} \\ &= -E_\theta\left\{\frac{d^2}{d\theta^2}\sum_{i=1}^n \log f_\theta(X_i)\right\} \\ &= -\sum_{i=1}^n E_\theta\left\{\frac{d^2}{d\theta^2}\log f_\theta(X_i)\right\} \\ &= -\sum_{i=1}^n E_\theta\left\{l_1''(\theta)\right\} \\ &= nI_1(\theta) \end{aligned}$$

because the expectation of a sum is the sum of the expectations and the derivative of a sum is the sum of the derivatives and because all the terms have the same expectation when the data are IID.

This means that if the data are IID, then we can use sample size one in the calculation of Fisher information (not for anything else) and then get the Fisher information for sample size n using the identity

$$I_n(\theta) = nI_1(\theta) \tag{7}$$

just proved.

1.6 Asymptotics of Log Likelihood Derivatives

1.6.1 The Law of Large Numbers

When we have IID data, the law of large numbers (LLN) applies to any average, in particular to the average

$$\frac{1}{n}l_n'(\theta) = \frac{1}{n}\sum_{i=1}^n \frac{d}{d\theta}\log f_\theta(X_i), \tag{8}$$

and says this converges to its expectation, which by (5a) is zero. Thus we have

$$\frac{1}{n}l_n'(\theta) \xrightarrow{P} 0. \tag{9a}$$

Similarly, the LLN applied to the average

$$-\frac{1}{n}l_n''(\theta) = -\frac{1}{n}\sum_{i=1}^n \frac{d^2}{d\theta^2}\log f_\theta(X_i) \tag{9b}$$

says this converges to its expectation, which by (5b) is $I_1(\theta)$. Thus we have

$$-\frac{1}{n} l''_n(\theta) \xrightarrow{P} I_1(\theta). \tag{9c}$$

1.6.2 The Central Limit Theorem

The central limit theorem (CLT) involves both mean and variance, and (5a) and (5b) only give us the mean and variance of l'_n. Thus we only get a CLT for that. The CLT says that for any average, and in particular for the average (8), when we subtract off its expectation and multiply by $\sqrt{n}$ the result converges in distribution to a normal distribution with mean zero and variance the variance of one term of the average. The expectation is zero by (5a). So there is nothing to subtract here. The variance is $I_1(\theta)$ by (5b) and the definition of Fisher information. Thus we have

$$\frac{1}{\sqrt{n}} l'_n(\theta) \xrightarrow{\mathcal{D}} \text{Normal}\bigl(0, I_1(\theta)\bigr). \tag{9d}$$

[we get $1/\sqrt{n}$ here because $\sqrt{n} \cdot (1/n) = 1/\sqrt{n}$.]

1.7 Asymptotics of Maximum Likelihood Estimators

So what is the point of all this? Assuming the MLE is in the interior of the parameter space, the maximum of the log likelihood occurs at a point where the derivative is zero. Thus we have

$$l'_n(\hat{\theta}_n) = 0. \tag{10}$$

That wouldn't seem to help us much, because the LLN and the CLT [equations (9a), (9c), and (9d)] only apply when θ is the unknown true population parameter value. But for large n, when $\hat{\theta}_n$ is close to θ (this assumes $\hat{\theta}_n$ is a consistent estimator, meaning it eventually does get close to θ) l'_n can be approximated by a Taylor series around θ

$$l'_n(\hat{\theta}_n) \approx l'_n(\theta) + l''_n(\theta)(\hat{\theta}_n - \theta) \tag{11}$$

We write $\approx$ here because we are omitting the remainder term in Taylor's theorem. This means we won't actually be able to prove the result we are heading for. Setting (11) equal to zero [because of (10)] and rearranging gives

$$\sqrt{n}(\hat{\theta}_n - \theta) \approx -\frac{\frac{1}{\sqrt{n}} l'_n(\theta)}{\frac{1}{n} l''_n(\theta)} \tag{12}$$

Now by (9a) and (9d) and Slutsky's theorem the right hand side converges to a normal random variable

$$-\frac{\frac{1}{\sqrt{n}} l_n'(\theta)}{\frac{1}{n} l_n''(\theta)} \xrightarrow{\mathcal{D}} \frac{Z}{I_1(\theta)} \tag{13}$$

where

$$Z \sim \text{Normal}\bigl(0, I_1(\theta)\bigr) \tag{14}$$

Using the fact that for any random variable z and any constant c we have $E(Z/c) = E(Z)/c$ and $\text{var}(Z/c) = \text{var}(Z)/c^2$ we get

$$\frac{Z}{I_1(\theta)} \sim \text{Normal}\bigl(0, I_1(\theta)^{-1}\bigr)$$

Thus finally, we get the big result about maximum likelihood estimates

$$\sqrt{n}(\hat{\theta}_n - \theta) \xrightarrow{\mathcal{D}} \text{Normal}\bigl(0, I_1(\theta)^{-1}\bigr). \tag{15}$$

Equation (15) is arguably the most important equation in theoretical statistics. It is really remarkable.

We may have no formula that gives the MLE as a function of the data. We may have no procedure for obtaining the MLE except to hand any particular data vector to a computer program that somehow maximizes the log likelihood. Nevertheless, theory gives us a large sample approximation (and a rather simple large sample approximation) to the sampling distribution of this estimator, an estimator that we can't explicitly describe!

Despite its amazing properties, we must admit two "iffy" issues about (15). First, we haven't actually proved it, and even if we wanted to make this course a lot more mathematical than it should be we couldn't prove it in complete generality. We could prove it under some conditions, but those conditions don't always hold. Sometimes (15) holds and sometimes it doesn't. There must be a thousand different proofs of (15) under various conditions in the literature. It has received more theoretical attention than any other result. But none of those proofs apply to all applications. An even if we could prove (15) to hold under all conditions (that's impossible, because there are counterexamples, applications where it doesn't hold, but assume we could), it still wouldn't tell us what we really want to know. It only asserts that for some sufficiently large n, perhaps much larger than the actual n of our actual data, the asymptotic approximation on the right hand side of (15) would be good. But perhaps it is no good at the actual n where we want to use it. Thus (15) has only heuristic value. The theorem gives no way to tell whether the approximation is good or bad at any particular n. Of course, now that we know all about the parametric bootstrap that shouldn't bother us. If we are worried about whether (15) is a good approximation, we simulate. Theory is no help.

The second "iffy" issue is that this whole discussion assumes the model is *exactly* correct, that the true distribution of the data has density f_θ for some θ. What if this assumption is wrong? That's the subject of Section 2 below.

Before we get to that we take care of a few loose ends.

1.8 Observed Fisher Information

Often Fisher information (6a) or (6b) is hard to calculate. Expectation involves integrals and not all integrals are doable. But the LLN (9c) gives us a consistent estimator of Fisher information, which is usually called *observed Fisher information*

$$\widehat{J}_n(\theta) = -l_n''(\theta) \tag{16}$$

To distinguish this from the other concept, we sometimes call $I_n(\theta)$ *expected Fisher information* although, strictly speaking, the "expected" is redundant.

Equation (15) can be rewritten (with another use of Slutsky's theorem)

$$\sqrt{I_n(\theta)} \cdot (\hat{\theta}_n - \theta) \xrightarrow{\mathcal{D}} \text{Normal}(0, 1) \tag{17a}$$

and yet another use of Slutsky's theorem gives

$$\sqrt{\widehat{J}_n(\theta)} \cdot (\hat{\theta}_n - \theta) \xrightarrow{\mathcal{D}} \text{Normal}(0, 1). \tag{17b}$$

1.9 Plug-In

Equations (15), (17a), and (17b) are not useful as they stand because we don't know the true parameter value θ. But still another use of Slutsky's theorem allows us to plug in the MLE

$$\sqrt{I_n(\hat{\theta}_n)} \cdot (\hat{\theta}_n - \theta) \xrightarrow{\mathcal{D}} \text{Normal}(0, 1) \tag{18a}$$

and

$$\sqrt{\widehat{J}_n(\hat{\theta}_n)} \cdot (\hat{\theta}_n - \theta) \xrightarrow{\mathcal{D}} \text{Normal}(0, 1). \tag{18b}$$

1.10 Sloppy Asymptotics

People typically rewrite (18a) and (18b) as

$$\hat{\theta}_n \approx \text{Normal}\bigl(\theta, I_n(\hat{\theta}_n)^{-1}\bigr) \tag{19a}$$

and

$$\hat{\theta}_n \approx \text{Normal}\bigl(\theta, \widehat{J}_n(\hat{\theta}_n)^{-1}\bigr). \tag{19b}$$

We call these "sloppy" because they aren't real limit theorems. In real math, you can't have an n in the limit of a sequence indexed by n. But that's what we have if we try to treat the right hand sides here as real limits.

But generally no harm is done. They both say that for "large n" the distribution of the MLE is approximately normal with mean θ (the true unknown parameter value) and approximate variance inverse Fisher information (either observed or expected and with the MLE plugged in).

1.11 Asymptotic Efficiency

It is a theorem (or again perhaps we should say many theorems with various conditions) that the MLE is the best possible estimator, asymptotically. Any other consistent and asymptotically normal estimator must have larger asymptotic variance. Asymptotic variance $I_1(\theta)^{-1}$ is the best possible.

That's a bit off topic. I just thought it should be mentioned.

1.12 Cauchy Location Example

We use the following data, which is assumed to come from some distribution in the Cauchy location family, that is, we assume the data satisfy

$$X_i = \theta + Z_i$$

where the Z_i are IID standard Cauchy.

```
> foo <- function(file) {
+      paste("http://www.stat.umn.edu/geyer/5601/mydata/",
+      file, sep = "")
+ }
> X <- read.table(foo("xc.txt"), header = TRUE)
> x <- X$x
> ### x <- 5 + rcauchy(50) ### if want different data
> stem(x, scale = 4)

  The decimal point is 1 digit(s) to the right of the |

  -0 | 5
  -0 | 21
   0 | 23444
   0 | 5555555669
   1 | 2
   1 |
   2 |
   2 |
   3 |
   3 |
   4 |
   4 | 9
```

Minus the log likelihood function is calculated by the R function,

```
> mlogl <- function(theta, x) sum(- dcauchy(x, theta, log = TRUE))
```

and the MLE is calculated by

```
> out <- nlm(mlogl, median(x), hessian = TRUE, x = x)
> print(out)

$minimum
[1] 56.63041

$estimate
[1] 4.873817

$gradient
[1] 1.647401e-07

$hessian
         [,1]
[1,] 13.17709

$code
[1] 1
```

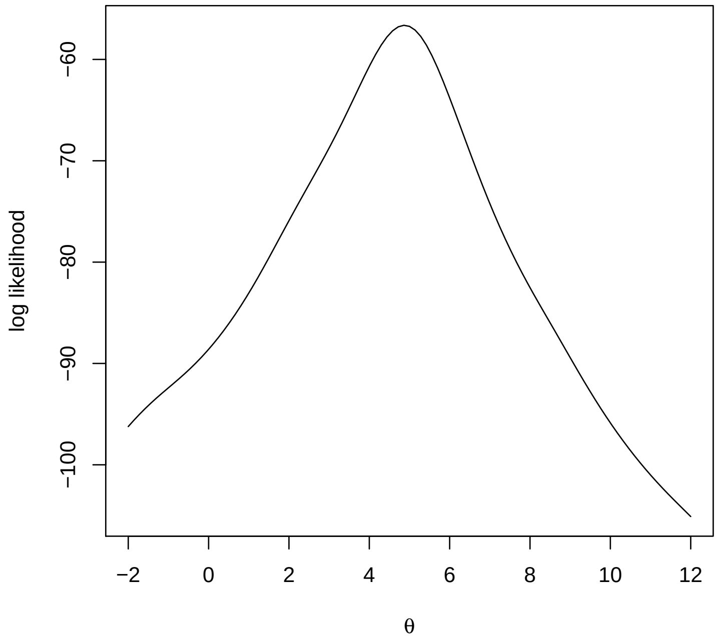

Figure 1: Log likelihood for Cauchy location model.

```
$iterations
[1] 3
```

To check our work, let us plot the log likelihood function. Figure 1 is produced by the following code

```
> fred <- function(theta)
+     return(- apply(as.matrix(theta), 1, mlogl, x = x))
> curve(fred, from = -2, to = 12, xlab = expression(theta),
+     ylab = "log likelihood")
```

It appears on p. 11 and seems to have a unique maximum in the region plotted at the solution (4.874) produced by the R function `nlm`.

Note that we have no guarantee that the MLE produced by `nlm` is the global maximizer of the log likelihood. It is clearly the nearest local maximizer to the starting point, which is the sample median. The fact that the sample median is already a good estimator of location guarantees that this MLE is the optimal estimator (assuming the population distribution really does satisfy our assumptions).

Now we use the approximation to Fisher information (the Hessian, which `nlm` has calculated by approximating derivatives with finite differences, to calculate a confidence interval for the true unknown θ

```
> theta.hat <- out$estimate
> info <- out$hessian
> conf.level <- 0.95
> crit <- qnorm((1 + conf.level) / 2)
> theta.hat + crit * c(-1, 1) / sqrt(info)

[1] 4.333886 5.413748
```

2 Misspecified Maximum Likelihood Estimation

2.1 Model Misspecification

A lot of what is said above survives model misspecification in modified form. By model misspecification, we mean the situation in which everything is the same as in Section 1 except that the true (unknown) probability density function g of the data is *not* of the form f_θ for any θ. The true distribution is *not* in the model we are using.

So our model assumption is wrong. What does that do?

2.2 Modifying the Theory under Model Misspecification

First it makes no sense to write E_θ or $\operatorname{var}_\theta$, as we did in (5a), (5b), (6a), and (6b) to emphasize that the distribution of the data is that having parameter θ. Now the true distribution has no θ because it isn't in the model. So we write E_g and var_g.

But more than that goes wrong. The reason we wrote E_θ and $\operatorname{var}_\theta$ in those equations was to emphasize that all the θ's must be the *same* θ in order for the differentiation under the integral sign to work. So under model misspecification we lose (5a), (5b), (6a), and (6b). Since these equations were the key to the whole theory, we need to replace them.

To replace (5a), we consider the expectation of the log likelihood

$$\lambda_g(\theta) = E_g\{l_X(\theta)\} \tag{20}$$

(the expectation being with respect to the true distribution of the data, which is indicated by the subscript g on the expectation operator). Suppose the function λ_g achieves its maximum at some point θ^* in the interior of the parameter space. Then at that point the derivative $\lambda'_g(\theta^*)$ is zero. Hence, assuming differentiation under the integral sign is possible, we have

$$E_g\{l'_X(\theta^*)\} = 0. \tag{21}$$

This looks enough like (5a) to play the same role in the theory.

The main difference is philosophical: θ^* is not the true unknown parameter value in the sense that f_{θ^*} is the true unknown probability density of the data. The true density is g, which is not (to repeat our misspecification assumption) any f_θ. So how do we interpret θ^*? It is (as we shall see below) what maximum likelihood estimates. In some sense maximum likelihood is doing the best job it can given what it is allowed to do. It estimates the θ^* that makes f_{θ^*} as close to g as an f_θ can get [where "close" means maximizing (20)].

To replace (5b), we have to face the problem that it just no longer holds. The two sides of (5b) are just not equal when the model is misspecified (and we replace E_θ and var_θ by E_g and var_g. But our use of (5b) was to define Fisher information as either side of the equation. When the two sides aren't equal our definition of Fisher information no longer makes sense.

We have to replace Fisher information by two different definitions

$$V_n(\theta) = \text{var}_g\{l'_n(\theta)\} \tag{22a}$$

$$J_n(\theta) = -E_g\{l''_n(\theta)\} \tag{22b}$$

These are our replacements for (6a) and (6b). When the model is not misspecified and $g = f_\theta$, then both of these are $I_n(\theta)$. When the model is misspecified, then they are different.

The identity (7) now gets replaced by two identities

$$V_n(\theta) = nV_1(\theta) \tag{23a}$$

$$J_n(\theta) = nJ_1(\theta) \tag{23b}$$

the first now holding because the variance of a sum is the sum of the variances when the summands are independent (and in the sum in question (8) the terms are independent because the X_i are IID) and the second now holding because the expectation of a sum is the sum of the expectations [the sum being (9b)].

2.3 Asymptotics under Model Misspecification

Now the law of large numbers (9c) gets replaced by

$$- \frac{1}{n} l_n''(\theta^*) \xrightarrow{P} J_1(\theta^*). \tag{24}$$

which we can also write given our definition of $\widehat{J}_n$, which we still use as a definition even though we shouldn't now call it "observed Fisher information,"

$$\frac{1}{n} \widehat{J}_n(\theta^*) \xrightarrow{P} J_1(\theta^*). \tag{25}$$

And the central limit theorem (9d) gets replaced by

$$\frac{1}{\sqrt{n}} l_n'(\theta^*) \xrightarrow{\mathcal{D}} \text{Normal}\bigl(0, V_1(\theta^*)\bigr). \tag{26}$$

Note that (24) and (26) are the same as (9c) and (9d) except that we had to replace $I_1(\theta)$ by $V_1(\theta^*)$ or $J_1(\theta^*)$, whichever was appropriate.

Then the whole asymptotic theory goes through as before with (13) and (14) being replaced by

$$- \frac{\frac{1}{\sqrt{n}} l_n'(\theta^*)}{\frac{1}{n} l_n''(\theta^*)} \xrightarrow{\mathcal{D}} \frac{Z}{J_1(\theta^*)} \tag{27}$$

and

$$Z \sim \text{Normal}\bigl(0, V_1(\theta^*)\bigr). \tag{28}$$

Again, we only had to replace $I_1(\theta)$ by $V_1(\theta^*)$ or $J_1(\theta^*)$, whichever was appropriate.

As before, the distribution of the right hand side of (27) is normal with mean zero. But the variance is now $V_1(\theta^*)/J_1(\theta^*)^2$ and does not simplify further. For reasons that will become clear in Section 3 below, we write this as $J_1(\theta^*)^{-1} V_1(\theta^*) J_1(\theta^*)^{-1}$.

So we arrive at the replacement of "arguably the most important equation in theoretical statistics" (15) under model misspecification

$$\sqrt{n}(\hat{\theta}_n - \theta^*) \xrightarrow{\mathcal{D}} \text{Normal}\bigl(0, J_1(\theta^*)^{-1} V_1(\theta^*) J_1(\theta^*)^{-1}\bigr). \tag{29}$$

Of course, (29) is useless for inference as it stands because we don't know θ^*. So we need "plug-in" here two. And we finally arrive at a replacement for (19a) and (19b). Curiously, they are replaced by the single equation

$$\hat{\theta}_n \approx \text{Normal}\bigl(\theta^*, \widehat{J}_n(\hat{\theta}_n)^{-1} \widehat{V}_n(\hat{\theta}_n) \widehat{J}_n(\hat{\theta}_n)^{-1}\bigr) \tag{30}$$

in which we have replaced $V_n(\theta)$ by an empirical estimate

$$\widehat{V}_n(\theta) = \sum_{i=1}^{n} l_n'(\theta)^2. \tag{31}$$

This makes sense because $l_n'(\theta^*)$ has expectation zero by (21) and hence its square estimates its variance.

2.4 The Sandwich Estimator

The asymptotic variance here

$$\widehat{J}_n(\hat{\theta}_n)^{-1}\widehat{V}_n(\hat{\theta}_n)\widehat{J}_n(\hat{\theta}_n)^{-1} \tag{32}$$

is called the *sandwich estimator*, the metaphor being that $\widehat{V}_n(\hat{\theta}_n)$ is a piece of ham between two pieces of bread $\widehat{J}_n(\hat{\theta}_n)^{-1}$.

It is remarkable that the whole theory of maximum likelihood goes through under model misspecification almost without change. The only changes are that we had to reinterpret a bit and substitute a bit, more precisely,

- the parameter value θ^* is no longer the "truth" but only the best approximation to the truth possible within the assumed model and
- the asymptotic variance becomes the more complicated "sandwich" (32) instead of the simpler "inverse Fisher information" appearing in (19a) or (19b).

2.5 Cauchy Location Example, Revisited

Although having log likelihood derivatives calculated automatically is convenient, it is better to do exact calculations when possible. To do that we need to know the formula for the density of the Cauchy distribution

$$f_\theta(x) = \frac{1}{\pi} \cdot \frac{1}{1 + (x - \theta)^2}$$

which makes one term of the log likelihood

$$\log f_\theta(x) = -\log(\pi) - \log\bigl[1 + (x - \theta)^2\bigr]$$

and we may drop the term $-\log(\pi)$, which does not contain the parameter θ if we please.

This makes the following R code

```
> logl <- expression(- log(1 + (x - theta)^2))
> scor <- D(logl, "theta")
> hess <- D(scor, "theta")
> print(scor)

2 * (x - theta)/(1 + (x - theta)^2)

> print(hess)

-(2/(1 + (x - theta)^2) - 2 * (x - theta) * (2 * (x - theta))/(1 +
    (x - theta)^2)^2)
```

calculate derivatives of one term of the log likelihood. Each of these objects, results of the R functions `expression` and `D`, have type `"expression"`. They are raw bits of R language stuffed into R objects.

From these we can define functions to calculate the log likelihood itself (summing over all the data) and its derivatives.

```
> mloglfun <- function(theta, x) sum(- eval(logl))
> gradfun <- function(theta, x) sum(- eval(scor))
> vfun <- function(theta, x) sum(eval(scor)^2)
> jfun <- function(theta, x) sum(- eval(hess))
```

For the convenience of `nlm`, our function `mloglfun` calculates minus the log likelihood. Our function `gradfun` calculates the first derivative of minus the log likelihood. Our function `vfun` calculates the sum of squares of first derivatives of terms of the log likelihood, thus calculating $\widehat{V}_n(\theta)$ as given by equation (31) above. Our function `jfun` calculates, the second derivative of minus the log likelihood $\widehat{J}_n(\theta)$ as given by equation (16) above.

In order for each of these functions to work, the argument `theta` must be a scalar and the argument `x` must be a vector. Then when the expression (the thingy of type `"expression"`, for example `logl`) is evaluated using the R function `eval`, the usual R vectorwise evaluation occurs and the result has one term for each element of `x`. The result vector is then summed using the `sum` function. The only magic is in `eval`. The function `sum` and the operators `-` and `^` work in the usual way, operating on the result of `eval`. In the `eval` the `theta` and `x` in the expression, are taken from the environment, which in this case are the `theta` and `x` that are the function arguments.

Having redefined `mlogl`, we try it out to see that it works. This time we omit the `hessian` argument to `nlm` because we now have `jfun` to calculate the Hessian.

```
> out2 <- nlm(mloglfun, median(x), x = x)
> print(out$estimate)

[1] 4.873817

> print(out2$estimate)

[1] 4.873817

> gradfun(out2$estimate, x)

[1] -3.194071e-05

> print(out2$gradient)

[1] 1.66198e-07
```

Looks like it works. The gradients don't agree because `nlm` is using finite difference approximation and `gradfun` is using exact derivatives, but clearly the derivative is nearly zero.

If we want `nlm` to use our `gradfun` we can do that too.

```
> fred <- function(theta, x) {
+     result <- mloglfun(theta, x)
+     attr(result, "gradient") <- gradfun(theta, x)
+     return(result)
+ }
> out3 <- nlm(fred, median(x), x = x)
> print(out3$estimate)

[1] 4.873819

> print(out3$gradient)

[1] 1.657468e-07
```

And we see that it makes almost no difference

```
> print(out2$estimate - out3$estimate)

[1] -2.43689e-06
```

So far nothing about the sandwich estimator. Now we do that

```
> theta.hat <- out3$estimate
> Vhat <- vfun(theta.hat, x)
> Jhat <- jfun(theta.hat, x)
> print(Vhat)

[1] 7.055237

> print(Jhat)

[1] 13.17518

> print(info)

         [,1]
[1,] 13.17709

> conf.level <- 0.95
> crit <- qnorm((1 + conf.level) / 2)
> theta.hat + crit * c(-1, 1) * sqrt(Vhat / Jhat^2)

[1] 4.478683 5.268956
```

This interval is shorter than the interval we calculated on p. 12 because `Vhat` is smaller than `Jhat`. With other data, the sandwich interval would be larger (when `Vhat` is larger than `Jhat`). In either case, this interval is semiparametric: the point estimate uses the likelihood equations for a very specific model (in this case Cauchy location) but the margin of error only uses Taylor series approximation of the likelihood function and the empirical distribution of the components of the score vector. In particular, the margin of error calculation does not assume the model is correct.

3 Multiparameter Models

3.1 Multivariable Calculus

When the parameter is a vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$ almost nothing changes. We must replace ordinary derivatives with partial derivatives, of course, since there are several variables θ_1, …, θ_p to differentiate with respect to. We indicate this by writing $\nabla l_n(\boldsymbol{\theta})$ in place of $l'_n(\theta)$ and $\nabla^2 l_n(\boldsymbol{\theta})$ in place of $l''_n(\theta)$.

3.1.1 Gradient Vectors

The symbol ∇, pronounced "del" indicates the vector whose components are partial derivatives. In particular, $\nabla l_n(\boldsymbol{\theta})$ is the vector having i-th component

$$\frac{\partial l_n(\theta)}{\partial \theta_i}.$$

Another name for ∇f is the *gradient vector* of the function f.

3.1.2 Hessian Matrices

The symbol ∇^2 pronounced "del squared" indicates the matrix whose components are second partial derivatives. In particular, $\nabla^2 l_n(\boldsymbol{\theta})$ is the matrix having i, j-th component

$$\frac{\partial^2 l_n(\theta)}{\partial \theta_i \partial \theta j}. \tag{33}$$

It is a theorem of multivariable calculus that the order of partial differentiation doesn't matter, in particular, that swapping i and j in (33) doesn't change the value. Hence $\nabla^2 l_n(\boldsymbol{\theta})$ is a symmetric $p \times p$ matrix.

Another name for $\nabla^2 f$ is the *Hessian matrix* of the function f.

3.1.3 Taylor Series

If f is a scalar valued function of a vector variable, then the Taylor series approximation keeping the first two terms is

$$f(\mathbf{x} + \mathbf{h}) \approx f(\mathbf{x}) + \mathbf{h}^T \nabla f(\mathbf{x})$$

Thus (11) becomes

$$\nabla l_n(\hat{\boldsymbol{\theta}}_n) \approx \nabla l_n(\boldsymbol{\theta}^*) + (\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^*)^T \nabla^2 l_n(\boldsymbol{\theta}^*) \tag{34}$$

which, using the fact that the left hand side is zero, can be rearranged to

$$\sqrt{n}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^*) \approx -\left(\tfrac{1}{n} \nabla^2 l_n(\boldsymbol{\theta}^*)\right)^{-1} \cdot \tfrac{1}{\sqrt{n}} \nabla l_n(\boldsymbol{\theta}^*) \tag{35}$$

which is the multiparameter analog of the left hand side of (13). We've also replaced $\boldsymbol{\theta}$ by $\boldsymbol{\theta}^*$ to do the misspecified case right away (the correctly specified case is a special case of the misspecified case).

Note that (35) is just like (12) except for some boldface and except that we had to replace the division operation in (12) by multiplication by an inverse matrix in (35). Without the matrix notation, we would be stuck. There is no way to describe a matrix inverse without matrices.

Before we can proceed, we need to review a bit of probability theory.

3.2 Multivariate Probability Theory

3.2.1 Random Vectors

A *random vector* is a vector whose components are random variables.

3.2.2 Mean Vectors

If $\mathbf{x} = (x_1, \ldots, x_p)$ is a random vector, then we say its *expectation* is the vector $\boldsymbol{\mu} = \mu_1, \ldots, \mu_p)$ where

$$\mu_i = E(x_i), \qquad i = 1, \ldots, p$$

and we write $\boldsymbol{\mu} = E(\mathbf{x})$, thus combining p scalar equations into one vector equation.

3.2.3 Variance Matrices

The *variance* of the random vector $\mathbf{x}$ is the $p \times p$ matrix $\mathbf{M}$ whose i, j-th component is

$$\operatorname{cov}(x_i, x_j) = E\{(x_i - \mu_i)(x_j - \mu_j)\}, \qquad i = 1, \ldots, p, \ j = 1, \ldots, p \tag{36}$$

(this is sometimes called the "covariance matrix" or the "variance-covariance matrix" because the diagonal elements are $\operatorname{cov}(x_i, x_i) = \operatorname{var}(x_i)$ or the "dispersion matrix" but we prefer the term *variance matrix* because it is the multivariate analog of the variance of a scalar random variable). We write $\operatorname{var}(\mathbf{x}) = \mathbf{M}$, thus combining p^2 scalar equations into one matrix equation.

The variance matrix $\mathbf{M}$ is also a symmetric $p \times p$ matrix because the value (36) is not changed by swapping i and j.

It is useful to have a definition of the variance matrix that uses vector notation

$$\operatorname{var}(\mathbf{x}) = E\{(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T\} \tag{37}$$

where, as before, $\boldsymbol{\mu} = E(\mathbf{x})$. The superscript T in (37) indicates the transpose of a matrix. When this notation is used the vector $\mathbf{x} - \boldsymbol{\mu}$ is treated like a $p \times 1$ matrix (a "column vector") so its transpose is $1 \times p$ (a "row vector") and the product is $p \times p$.

3.2.4 Linear Transformations

If x is a scalar random variable and a and b are scalar constants, then

$$E(a + bx) = a + bE(x) \tag{38a}$$

$$\operatorname{var}(a + bx) = b^2 \operatorname{var}(x) \tag{38b}$$

If $\mathbf{x}$ is a random vector and $\mathbf{a}$ is a constant vector and $\mathbf{B}$ a constant matrix, then

$$E(\mathbf{a} + \mathbf{B}\mathbf{x}) = \mathbf{a} + \mathbf{B}E(\mathbf{x}) \tag{39a}$$

$$\operatorname{var}(\mathbf{a} + \mathbf{B}\mathbf{x}) = \mathbf{B} \operatorname{var}(\mathbf{x})\mathbf{B}^T \tag{39b}$$

The right hand side of (39b) is the original "sandwich." It is where the "sandwich estimator" arises.

3.2.5 The Law of Large Numbers

If $\mathbf{X}_1$, $\mathbf{X}_2$, $\ldots$ is an IID sequence of random vectors having mean vector

$$\boldsymbol{\mu} = E(\mathbf{X}_i), \tag{40a}$$

then the multivariate law of large numbers (LLN) says

$$\overline{\mathbf{X}}_n \xrightarrow{P} \boldsymbol{\mu} \tag{40b}$$

where

$$\overline{\mathbf{X}}_n = \frac{1}{n} \sum_{i=1}^{n} \mathbf{X}_i. \tag{40c}$$

3.2.6 The Central Limit Theorem

If $\mathbf{X}_1$, $\mathbf{X}_2$, $\ldots$ is an IID sequence of random vectors having mean vector (40a) and variance matrix

$$\mathbf{M} = E\{(\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})^T\}, \tag{41a}$$

then the multivariate central limit theorem (CLT) says

$$\sqrt{n}(\overline{\mathbf{X}}_n - \boldsymbol{\mu}) \xrightarrow{\mathcal{D}} \operatorname{Normal}(\mathbf{0}, \mathbf{M}) \tag{41b}$$

where $\overline{\mathbf{X}}_n$ is given by (40c) and the distribution on the right hand side is the multivariate normal distribution with mean vector $\mathbf{0}$ and variance matrix $\mathbf{M}$.

3.3 Asymptotics of Log Likelihood Derivatives

3.3.1 Misspecified Models

With the probability review out of the way, we can continue with maximum likelihood.

Now the law of large numbers (24) gets replaced by

$$-\frac{1}{n}\nabla^2 l_n(\boldsymbol{\theta}^*) \xrightarrow{P} \mathbf{J}_1(\theta^*) \tag{42}$$

and the central limit theorem (26) gets replaced by

$$\frac{1}{\sqrt{n}}\nabla l_n'(\boldsymbol{\theta}^*) \xrightarrow{\mathcal{D}} \text{Normal}\big(\mathbf{0}, \mathbf{V}_1(\boldsymbol{\theta}^*)\big). \tag{43}$$

where

$$\mathbf{V}_n(\boldsymbol{\theta}) = \text{var}_g\{\nabla l_n(\boldsymbol{\theta})\} \tag{44a}$$

$$\mathbf{J}_n(\boldsymbol{\theta}) = -E_g\{\nabla^2 l_n(\boldsymbol{\theta})\} \tag{44b}$$

Note these are the same as (22a) and (22b) except for some boldface type and replacement of ordinary derivatives by ∇ and ∇^2. The boldface is significant though, both $\mathbf{V}_n(\boldsymbol{\theta})$ and $\mathbf{J}_n(\boldsymbol{\theta})$ are $p \times p$ symmetric matrices.

Now the right hand side of (35) has the limiting distribution $\mathbf{J}_1(\boldsymbol{\theta}^*)^{-1}\mathbf{Z}$, where $\mathbf{Z}$ is a random vector having the distribution on the right hand side of (43). Using (39b) we get the right hand side of the following for the asymptotic distribution of the MLE

$$\sqrt{n}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^*) \xrightarrow{\mathcal{D}} \text{Normal}\big(\mathbf{0}, \mathbf{J}_1(\boldsymbol{\theta}^*)^{-1}\mathbf{V}_1(\boldsymbol{\theta}^*)\mathbf{J}_1(\boldsymbol{\theta}^*)^{-1}\big). \tag{45}$$

Again, note, just like (29) except for some boldface. Of course, the reason it is just like (29) except for boldface is because we chose to write (29) so it had the "sandwich" form even though we didn't need it in the uniparameter case.

Using plug-in and "sloppy" asymptotics we get

$$\hat{\boldsymbol{\theta}}_n \approx \text{Normal}\big(\boldsymbol{\theta}^*, \widehat{\mathbf{J}}_n(\hat{\boldsymbol{\theta}}_n)^{-1}\widehat{\mathbf{V}}_n(\hat{\boldsymbol{\theta}}_n)\widehat{\mathbf{J}}_n(\hat{\boldsymbol{\theta}}_n)^{-1}\big) \tag{46}$$

to replace (30), where

$$\widehat{\mathbf{V}}_n(\boldsymbol{\theta}) = \sum_{i=1}^{n} \nabla l_n(\boldsymbol{\theta})\big(\nabla l_n(\boldsymbol{\theta})\big)^T \tag{47a}$$

$$\widehat{\mathbf{J}}_n(\boldsymbol{\theta}) = \sum_{i=1}^{n} \nabla^2 l_n(\boldsymbol{\theta}) \tag{47b}$$

3.3.2 Correctly Specified Models

Correctly specified models are a special case of what we have just done. When the model is correctly specified, we replace $\boldsymbol{\theta}^*$ by the true parameter value $\boldsymbol{\theta}$. Also we have the identity

$$\mathbf{V}_n(\boldsymbol{\theta}) = \mathbf{J}_n(\boldsymbol{\theta}) = \mathbf{I}_n(\boldsymbol{\theta}) \tag{48}$$

which is the multiparameter analog of (5b). This identity makes (45) simplify to

$$\sqrt{n}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}) \xrightarrow{\mathcal{D}} \text{Normal}\bigl(\mathbf{0}, \mathbf{I}_1(\boldsymbol{\theta})^{-1}\bigr). \tag{49}$$

and (46) becomes

$$\hat{\boldsymbol{\theta}}_n \approx \text{Normal}\bigl(\boldsymbol{\theta}^*, \mathbf{I}_n(\hat{\boldsymbol{\theta}}_n)^{-1}\bigr) \tag{50a}$$

$$\hat{\boldsymbol{\theta}}_n \approx \text{Normal}\bigl(\boldsymbol{\theta}^*, \widehat{\mathbf{J}}_n(\hat{\boldsymbol{\theta}}_n)^{-1}\bigr) \tag{50b}$$

$$\hat{\boldsymbol{\theta}}_n \approx \text{Normal}\bigl(\boldsymbol{\theta}^*, \widehat{\mathbf{V}}_n(\hat{\boldsymbol{\theta}}_n)^{-1}\bigr) \tag{50c}$$

One uses whichever plug-in estimate of the asymptotic variance is convenient.

It is interesting that before writing this handout, I only "knew" about the approximations (50a) and (50b) in the sense that those were the only ones I had names for, inverse *expected Fisher information* for the asymptotic variance in (50a) and inverse *observed Fisher information* for the asymptotic variance in (50b). Of course, I also knew about the approximation (50c) in the sense that the bits and pieces were somewhere in my mind, but I only though about $\widehat{\mathbf{V}}_n(\theta)$ in the context of model misspecification, never in the context of correctly specified models. The plug-in asymptotic variance estimator $\widehat{\mathbf{V}}_n(\hat{\boldsymbol{\theta}}_n)^{-1}$ used in (50c) probably needs a name, since it is just as good an approximation as those in (50a) and (50b). But it doesn't have a name, as far as I know. Ah, the power of terminology!

3.4 Cauchy Location-Scale Example

For a multiparameter model we go to the Cauchy location-scale family in which we assume the data satisfy

$$X_i = \mu + \sigma Z_i$$

where the Z_i are IID standard Cauchy. We call μ the location parameter and σ the scale parameter. When we "standardize" the data

$$Z_i = \frac{X_i - \mu}{\sigma}$$

we get the standard Cauchy distribution. This is just like every other standardization in statistics, except more general. Cauchy distributions do not have moments, so μ is not the mean (it is the center of symmetry and the median) and σ is not the standard deviation (2σ is the interquartile range)

```
> qcauchy(0.75) - qcauchy(0.25)
```

```
[1] 2
```

Now the formula for the density of the Cauchy distribution is

$$f_{\boldsymbol{\theta}}(x) = \frac{1}{\pi} \cdot \frac{1}{\sigma} \cdot \frac{1}{1 + \left(\frac{x-\mu}{\sigma}\right)^2}$$

where $\boldsymbol{\theta} = (\mu, \lambda)$ is the (two-dimensional) parameter vector, which makes one term of the log likelihood

$$l_x(\boldsymbol{\theta}) = -\log\left[1 + \left(\frac{x-\mu}{\sigma}\right)^2\right] - \log(\sigma)$$

and this time we have dropped the term $-\log(\pi)$, which does not contain any parameters.

This makes the following R code

```
> logl <- expression(- log(1 + ((x - mu) / sigma)^2) - log(sigma))
> scor.mu <- D(logl, "mu")
> scor.sigma <- D(logl, "sigma")
> hess.mu.mu <- D(scor.mu, "mu")
> hess.mu.sigma <- D(scor.mu, "sigma")
> hess.sigma.sigma <- D(scor.sigma, "sigma")
```

calculate the relevant partial derivatives.

From these we can define functions to calculate the log likelihood, its gradient vector, and its Hessian matrix.

```
> loglfun <- function(theta, x) {
+     mu <- theta[1]
+     sigma <- theta[2]
+     sum(eval(logl))
+ }
> gradfun <- function(theta, x) {
+     mu <- theta[1]
+     sigma <- theta[2]
+     grad.mu <- sum(eval(scor.mu))
+     grad.sigma <- sum(eval(scor.sigma))
+     c(grad.mu, grad.sigma)
+ }
> vfun <- function(theta, x) {
+     mu <- theta[1]
+     sigma <- theta[2]
+     s.mu <- eval(scor.mu)
+     s.sigma <- eval(scor.sigma)
+     v.mu.mu <- sum(s.mu^2)
+     v.mu.sigma <- sum(s.mu * s.sigma)
+     v.sigma.sigma <- sum(s.sigma^2)
+     matrix(c(v.mu.mu, v.mu.sigma, v.mu.sigma, v.sigma.sigma), 2, 2)
+ }
> jfun <- function(theta, x) {
+     mu <- theta[1]
+     sigma <- theta[2]
+     j.mu.mu <- sum(- eval(hess.mu.mu))
+     j.mu.sigma <- sum(- eval(hess.mu.sigma))
+     j.sigma.sigma <- sum(- eval(hess.sigma.sigma))
+     matrix(c(j.mu.mu, j.mu.sigma, j.mu.sigma, j.sigma.sigma), 2, 2)
+ }
```

These functions are messy. The computer algebra features in R are very limited. Mathematica it ain't. Nevertheless, that it has any computer algebra features at all is cool. We just do for each component of the gradient vector and each component of the Hessian matrix the same sort of thing we did in the one-parameter case. Then we assemble the vector or matrix in the last statement of the function. One difference from the one-parameter case is that now we are calculating the log likelihood (not minus the log likelihood) and its gradient, although we still have minus the Hessian because that is what $J_n(\theta)$ is.

Let's do it.

```
> fred <- function(theta, x) {
+     result <- (- loglfun(theta, x))
+     attr(result, "gradient") <- (- gradfun(theta, x))
+     return(result)
+ }
> theta.start <- c(median(x), IQR(x) / 2)
> out4 <- nlm(fred, theta.start, x = x)
> print(out4$estimate)

[1] 4.8774517 0.9672641

> print(out4$gradient)

[1] 3.416980e-08 1.932423e-08

> print(out3$estimate)

[1] 4.873819
```

The location parameter estimate only changes a little bit when we go to the two-parameter model.

Now we do what? Do we want confidence intervals or confidence regions? Our estimator is now a vector.

```
> theta.hat <- out4$estimate
> Vhat <- vfun(theta.hat, x)
> Jhat <- jfun(theta.hat, x)
> Mhat <- solve(Jhat) %*% Vhat %*% solve(Jhat)
> print(Vhat)

          [,1]      [,2]
[1,]  7.601141 -1.530151
[2,] -1.530151 13.775519

> print(Jhat)

          [,1]     [,2]
[1,] 13.775519 1.530151
[2,]  1.530151 7.601141

> print(Mhat)

            [,1]        [,2]
[1,]  0.04838327 -0.05177662
[2,] -0.05177662  0.25730936
```

I have no idea why the curious pattern of entries in $\widehat{V}_n(\theta)$ and $\widehat{J}_n(\theta)$, presumably something about the Cauchy likelihood. It is not the obvious bug that the code is not calculating what it claims to. The matrix `Mhat` is the sandwich estimator of the asymptotic variance of $\hat{\boldsymbol{\theta}}_n$.

One thing our theory fails to tell us is that expected Fisher information for the Cauchy location-scale model (or for any location-scale model with symmetric base distribution) is diagonal. Thus for large n we should have `Mhat` diagonal, assuming the true unknown distribution of the data (which may not be Cauchy) is symmetric about the true unknown location parameter μ.

For that reason, we shall ignore the (estimated) correlation of our estimates and produce (non-simultaneous) confidence intervals for each.

```
> theta.hat[1] + crit * c(-1, 1) * sqrt(Mhat[1, 1])

[1] 4.446334 5.308569

> theta.hat[2] + crit * c(-1, 1) * sqrt(Mhat[2, 2])

[1] -0.02694075  1.96146898
```

Our confidence interval for $\theta_1 = \mu$ is not that different from those obtained using the one-parameter model on p. 12 and on p. 18. This is because the true unknown σ does not seem to be that much different from the $\sigma = 1$ that gives the standard model. Our confidence interval for $\theta_2 = \sigma$ is so wide that it goes negative, which is ridiculous. By definition $\sigma > 0$. This is because the sample size n is really too small for asymptotic theory to work.

4 The Moral of the Story

Every application of maximum likelihood is a potential application of the theory of misspecified maximum likelihood. Just stop believing in the exact correctness of the model. As soon as you become a bit skeptical, you need the misspecified theory that gives (45) or (46).

This theory is on the borderline between parametrics and nonparametrics, in what is sometimes called *semiparametrics*. We have a parametric model: $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^*$ are parameters. And we use the model for maximum likelihood estimation: $\hat{\boldsymbol{\theta}}_n$ is a parameter estimate. But we don't base our inference, at least not completely, on believing the model. The "sandwich estimator" of asymptotic variance is nonparametric. Confidence intervals for

$\boldsymbol{\theta}^*$ based on these asymptotics [(45) or (46)] are valid large sample approximations regardless of the true unknown density function (g).

Of course, this is a bit misleading, because the very definition of $\boldsymbol{\theta}^*$ depends on g because $\boldsymbol{\theta}^*$ is the maximizer of the function λ_g, which depends on g as (20) shows. Perhaps we should have written $\boldsymbol{\theta}^*_g$ everywhere instead of $\boldsymbol{\theta}^*$.

The same sorts of procedures arise for all but the simplest of nonparametric procedures. Completely nonparametric procedures are available only in the simplest situations. Generally useful procedures for complicated situations involve some mix of parametric and nonparametric thinking. The deeper you get into nonparametrics, the more you run into this kind of "semiparametric" thinking.

5 Literature

I do not know of a good textbook treatment of this subject at the level of this handout (that's why I wrote it). The original paper on the subject was White (1982). It is quite technical and hard to read, but says essentially what this handout says but with proofs. This idea has since appeared in hundreds of theoretical papers.

The notation I use here comes from a recent PhD thesis in the School of Statistics (Sung, 2003) which extended this model misspecification idea to the situation where there are missing data and only Monte Carlo approximation of the likelihood is possible. (Then the asymptotic variance becomes much more complicated because it involves both sampling variability and Monte Carlo variability.)

References

Sung, Yun Ju (2003). *Model Misspecification in Missing Data.* Unpublished PhD thesis. University of Minnesota.

White Halbert (1982). Maximum likelihood estimation of misspecified model. *Econometrica* 50:1–25.