

# Inverse Theory week 7 part 2

## Nonlinearity

Hugh C. Pumphrey

November 1, 2006

### 1 Introduction

Throughout the last six lectures, we have assumed that  $\mathbf{x}$ , the quantity that we wish to estimate is linearly related to  $\mathbf{y}$ , the quantity we can measure, *i.e.* that we can write  $\mathbf{y} = \mathbf{K}\mathbf{x}$ , where  $\mathbf{K}$  is a matrix. In general, this is never exactly true –  $\mathbf{y}$  is related to  $\mathbf{x}$  by some more general function  $F$ . In the first lecture on profiles we noted that this was the case and that we could always linearise the problem by writing

$$\mathbf{y} = F(\mathbf{x}) \approx F(\mathbf{x}_L) + \mathbf{K}(\mathbf{x} - \mathbf{x}_L)$$

Here, we have expanded  $F(\mathbf{x})$  as a Taylor series in terms of the difference between  $\mathbf{x}$  and a linearisation profile  $\mathbf{x}_L$ . We can then choose new variables  $\mathbf{x} \leftarrow \mathbf{x} - \mathbf{x}_L$  and  $\mathbf{y} \leftarrow \mathbf{y} - F(\mathbf{x}_L)$  so that we can re-write the forward model as  $\mathbf{y} = \mathbf{K}\mathbf{x}$ . All of the work we have done to date has assumed that this approximation is either exact, or close enough to exact that it makes no practical difference.

In real life, this is often not the case. In this lecture, we will examine briefly how we can apply the techniques we have learned about to a problem in which the linear approximation is not accurate enough.

### 2 Degrees of nonlinearity

It is helpful to classify remote sounding problems into four groups, depending on how nonlinear their forward functions  $F(\mathbf{x})$  are.

- Linear.  $\mathbf{y} = \mathbf{K}\mathbf{x}$  is a good approximation
- Nearly linear.  $\mathbf{y} = F(\mathbf{x}) \approx F(\mathbf{x}_L) + \mathbf{K}(\mathbf{x} - \mathbf{x}_L)$  is a good approximation as long as  $\mathbf{x}_L$  is chosen sensibly.
- Moderately nonlinear. We have to use an iterative solution to find  $\hat{\mathbf{x}}$  but once we have found it we can apply the techniques we have learned to understand the errors described by the covariance matrix  $\hat{\mathbf{S}}$ .
- Very nonlinear. Every problem in this class is different and would need a different solution devised for it. We try extremely hard to avoid designing instruments whose forward functions are this nonlinear!

We will consider the middle two cases in a little more detail, but first it is worth noting that one can sometimes reduce the nonlinearity by applying some sort of transformation to  $\mathbf{x}$ . We did this in the first lecture by retrieving the Planck function  $B$  instead of the temperature.  $B$  is more linearly

related to the radiances than  $T$  and it is straightforward to convert from  $B$  to  $T$  and from  $T$  to  $B$ . A second important point to note is that there are two terms in our cost function.

$$C(\mathbf{x}) = (\mathbf{x} - \mathbf{x}_a)^T \mathbf{S}_a^{-1} (\mathbf{x} - \mathbf{x}_a) + (\mathbf{y} - F(\mathbf{x}))^T \mathbf{S}_y^{-1} (\mathbf{y} - F(\mathbf{x})) \quad (1)$$

The second term makes our problem nonlinear if  $F(\mathbf{x}) \approx F(\mathbf{x}_L) + \mathbf{K}(\mathbf{x} - \mathbf{x}_L)$  is a poor approximation. The first term introduces no nonlinearity because we chose it to be like that when we decided to assume *a priori* that  $\mathbf{x}$  had Gaussian statistics. Likewise, the constraint term in the Twomey-Tikhonov formula is a linear term because we choose it to be so. It is possible to use constraints which make the problem nonlinear, but nonlinearity is a nuisance, so this is rarely done in practice.

### 3 The nearly linear case

When we developed the MAP formula we assumed that we had made the change of variables described above, so that  $\mathbf{x}_L$  did not appear in the formula. It will now be helpful to re-write the formulae so that  $\mathbf{x}_L$  appears explicitly. We take Equation 1, the cost function which we wish to minimize and, instead of replacing  $F(\mathbf{x})$  with  $\mathbf{K}\mathbf{x}$ , we replace it with  $F(\mathbf{x}_L) + \mathbf{K}(\mathbf{x} - \mathbf{x}_L)$  to give:

$$C(\mathbf{x}) = (\mathbf{x} - \mathbf{x}_a)^T \mathbf{S}_a^{-1} (\mathbf{x} - \mathbf{x}_a) + (\mathbf{y} - \mathbf{y}_L - \mathbf{K}(\mathbf{x} - \mathbf{x}_L))^T \mathbf{S}_y^{-1} (\mathbf{y} - \mathbf{y}_L - \mathbf{K}(\mathbf{x} - \mathbf{x}_L))$$

where we have written  $\mathbf{y}_L$  for  $F(\mathbf{x}_L)$ . We can find the value of  $\mathbf{x}$  for which  $C$  is a minimum either as we did in the lecture on the MAP formula, or by differentiating  $C$  with respect to  $\mathbf{x}$  and setting the result to 0. The result (in both  $m$  and  $n$  forms) is:

$$\hat{\mathbf{x}} = \mathbf{x}_a + (\mathbf{S}_a^{-1} + \mathbf{K}^T \mathbf{S}_y^{-1} \mathbf{K})^{-1} \mathbf{K}^T \mathbf{S}_y^{-1} [\mathbf{y} - \mathbf{y}_L - \mathbf{K}(\mathbf{x}_a - \mathbf{x}_L)] \quad (2)$$

and

$$\hat{\mathbf{x}} = \mathbf{x}_a + \mathbf{S}_a \mathbf{K}^T (\mathbf{K} \mathbf{S}_a \mathbf{K}^T + \mathbf{S}_y)^{-1} [\mathbf{y} - \mathbf{y}_L - \mathbf{K}(\mathbf{x}_a - \mathbf{x}_L)] \quad (3)$$

This only leaves us with the question of what to choose as the linearisation point. One choice is to use the *a priori* as a linearisation point. Note, however, that for each different linearisation point used, we have to use the forward model (our best attempt to model the forward function  $F$ ) to calculate  $\mathbf{y}_L$  and  $\mathbf{K}$ . This can be very expensive, so it is common to do this in advance and to have the retrieval program read  $\mathbf{x}_L$ ,  $\mathbf{y}_L$  and  $\mathbf{K}$  from a file. It may only be practical to store these

things for a small number of different values of  $\mathbf{x}_L$  so it is common to have a more sophisticated way of obtaining  $\mathbf{x}_a$  than  $\mathbf{x}_L$ .

## 4 The moderately nonlinear case

### 4.1 Newton (Inverse Hessian) iteration

If the problem is moderately nonlinear, then applying Equation 2 or 3 will not give an answer that minimizes  $C$ . However it will in many cases give an answer that makes  $C$  smaller than  $C(\mathbf{x}_L)$ . We can therefore take this answer and use it as a new  $\mathbf{x}_L$ , recalculate  $\mathbf{y}_L$  and  $\mathbf{K}$  and re-apply the formula. If we repeat this process several times, the answer will hopefully settle down to a value that does minimize  $C$ . We therefore have an iterative solution that looks like this:

$$\mathbf{x}_{i+1} = \mathbf{x}_a + (\mathbf{S}_a^{-1} + \mathbf{K}_i^T \mathbf{S}_y^{-1} \mathbf{K}_i)^{-1} \mathbf{K}_i^T \mathbf{S}_y^{-1} [\mathbf{y} - F(\mathbf{x}_i) - \mathbf{K}(\mathbf{x}_a - \mathbf{x}_i)]$$

This formula (and its  $m$ -form variant) is called the Gauss-Newton or Inverse Hessian method. An alternative form of the equation is:

$$\mathbf{x}_{i+1} = \mathbf{x}_i + (\mathbf{S}_a^{-1} + \mathbf{K}_i^T \mathbf{S}_y^{-1} \mathbf{K}_i)^{-1} [\mathbf{K}_i^T \mathbf{S}_y^{-1} (\mathbf{y} - F(\mathbf{x}_i)) - \mathbf{S}_a^{-1} (\mathbf{x}_i - \mathbf{x}_a)] \quad (4)$$

The initial guess  $\mathbf{x}_0$  can be anything — it should not have any effect on the solution. It is common but not obligatory to use the a priori profile  $\mathbf{x}_a$  as the initial guess. A good initial guess is very important, though, as it reduces the number of iterations required to find the solution. Note that  $\mathbf{x}_a$  does *not* change from iteration to iteration.

This formula is only an example of the Inverse Hessian method, which can be used to find the minimum of any cost function  $C(\mathbf{x})$  — it doesn't have to be in the form of Equation 1. The general form of the Inverse Hessian formula is

$$\mathbf{x}_{i+1} = \mathbf{x}_i - [\nabla \mathbf{g}(\mathbf{x}_i)]^{-1} \mathbf{g}(\mathbf{x}_i) \quad (5)$$

where  $\mathbf{g}(\mathbf{x}) = \nabla C(\mathbf{x})$ , the first derivative of the cost function. The second derivative of the cost function,  $\nabla \mathbf{g}(\mathbf{x})$ , is called the Hessian, so it is clear why the method is called the Inverse Hessian method. If we expand out Equation 5 in the case where the cost function is given by Equation 1, then we recover Equation 4 — the derivation is in Rogers (p 85) if you want to look it up.

### 4.2 Marquadt-Levenberg iteration

For some classes of function  $F(\mathbf{x})$  the Inverse Hessian method will fail if the initial guess is not good enough. Instead of  $C$  getting smaller with each iteration it gets larger and  $\mathbf{x}$  becomes more obviously wrong. This is because the Inverse Hessian formula looks for a place where the gradient of  $C$  is zero. It may find a minimum, but it may also head off towards some other stationary point. In cases like this, it is necessary to resort to a more careful approach to finding the minimum. A simple idea is to move  $\mathbf{x}$  in the direction in which seems to make  $C$  become smaller faster than any other direction. This is called the method of steepest descent:

$$\mathbf{x}_{i+1} = \mathbf{x}_i - \gamma^{-1} \mathbf{g}(\mathbf{x}_i).$$

The parameter  $\gamma$  controls how large a step is taken. The steepest descents formula usually takes many steps to find the minimum so a popular approach is to combine it with the Inverse Hessian:

$$\mathbf{x}_{i+1} = \mathbf{x}_i - [\gamma + \nabla \mathbf{g}(\mathbf{x}_i)]^{-1} \mathbf{g}(\mathbf{x}_i)$$

This is known as the Marquadt-Levenberg formula and in the case of our usual cost function it looks like this:

$$\mathbf{x}_{i+1} = \mathbf{x}_i + (\mathbf{S}_a^{-1} + \mathbf{K}_i^T \mathbf{S}_y^{-1} \mathbf{K}_i + \gamma \mathbf{D})^{-1} [\mathbf{K}_i^T \mathbf{S}_y^{-1} (\mathbf{y} - F(\mathbf{x}_i)) - \mathbf{S}_a^{-1} (\mathbf{x}_i - \mathbf{x}_a)]$$

This is the same as Equation 4 apart from the  $\gamma \mathbf{D}$  term. The matrix  $\mathbf{D}$  is a scaling matrix to make  $\gamma$  dimensionless and ensure that the term has the same units as  $\mathbf{S}_a^{-1}$ . (In fact,  $\mathbf{S}_a^{-1}$  is a popular choice for  $\mathbf{D}$  which simplifies the formula slightly.)

The scalar parameter  $\gamma$  is the key to using this formula. If  $\gamma$  is small, the formula tends to the inverse Hessian formula. If  $\gamma$  is large, the formula becomes the method of steepest descents. The steepest descents method is more reliable than the inverse Hessian method but it converges very slowly, taking many iterations to find the minimum of  $C$ .

A common strategy for using the Marquadt-Levenberg formula is:

- If  $C$  decreases as the result of taking a step, then we are going the right way. We take  $\mathbf{x}_{i+1}$  as the new  $\mathbf{x}_i$ , make  $\gamma$  smaller and take another step.
- If  $C$  increases as the result of taking a step, then we are going the wrong way. We keep  $\mathbf{x}_i$  as it was, make  $\gamma$  larger and try to do the step again.

There are cleverer ways of controlling  $\gamma$  but in all cases, the idea is to have it be very small at the end of the process, so that we are using the rapidly-converging inverse Hessian formula once we are close enough to the solution for it to be well-behaved.

## 5 One-dimensional examples

In order to give a feel for how some of these formulae work, we demonstrate them in the example where  $\mathbf{x}$  has only one element and is therefore a scalar  $x$ . We let  $y$  also have only one element. Our cost function is now  $C(x) = (x - x_a)^2 / \sigma_a^2 + (y - F(x))^2 / \sigma_y^2$ , assuming that our a priori constraint is linear. The Inverse Hessian formula for finding the minimum of this is

$$x_{i+1} = x_i - \frac{C'(x)}{C''(x)}$$

where the  $'$  denotes differentiation with respect to  $x$ . The steepest descents formula is:

$$x_{i+1} = x_i - \frac{C'(x)}{\gamma}$$

and the Marquadt-Levenberg formula is:

$$x_{i+1} = x_i - \frac{C'(x)}{\gamma + C''(x)}.$$

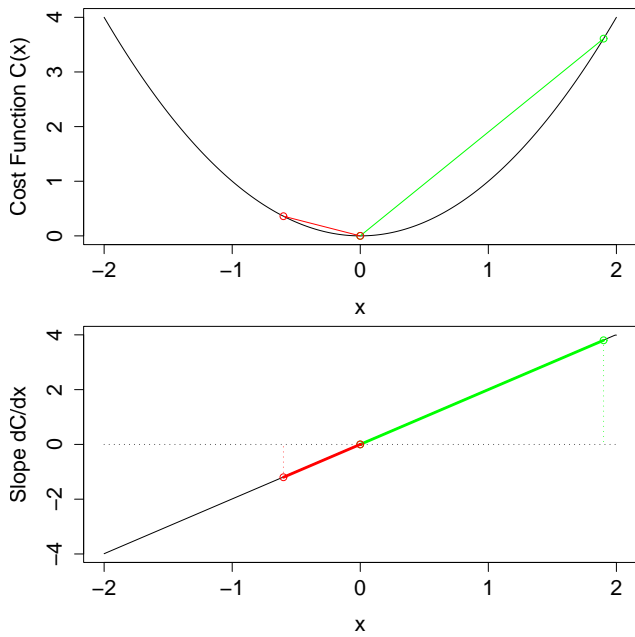


Figure 1: The top panel shows a 1-d cost function  $C(x)$ . The bottom panel shows its first derivative  $C'(x) = \frac{dC}{dx}$ . The Inverse Hessian method finds the tangent to  $C'(x)$  at its starting point and takes its next estimate as being where that tangent crosses the dotted line  $C'(x) = 0$ . In this case it will find the final answer in one step, wherever it starts from.

If  $F(x) = kx$  then the problem is linear and the cost function is a parabola. As Figure 1 shows, the Inverse Hessian method finds the minimum in one step.

If we allow  $F(x)$  to be slightly nonlinear as shown in Figure 2, then we find that the Inverse Hessian method still works, wherever you start it from, but if you start it too far away from the minimum of  $C$ , it will take several iterations before converging on the answer. The example shown in Figure 3 is altogether more difficult. The Inverse Hessian formula is unreliable in this sort of problem as it may not go towards the minimum. The Marquadt-Levenberg formula may take a large number of steps and have to adjust its  $\gamma$  parameter upwards at a few points, but it gets to the right answer in the end, wherever we start it. The worst sort of problems have more than one minimum. Figure 4 shows an example. The Marquadt-Levenberg formula does better than the Inverse Hessian in that it always finds a minimum and never a maximum. However it may be the wrong minimum. There are sophisticated search strategies available for finding the global minimum of a function, but in remote sensing applications it is usual to avoid these as they are too numerically expensive. Instead it is usual to design the instrument to avoid nonlinearities and to go to some effort to find as good an initial guess as possible so that the Marquadt-Levenberg or Inverse Hessian formula has as much chance as possible of finding the true minimum.

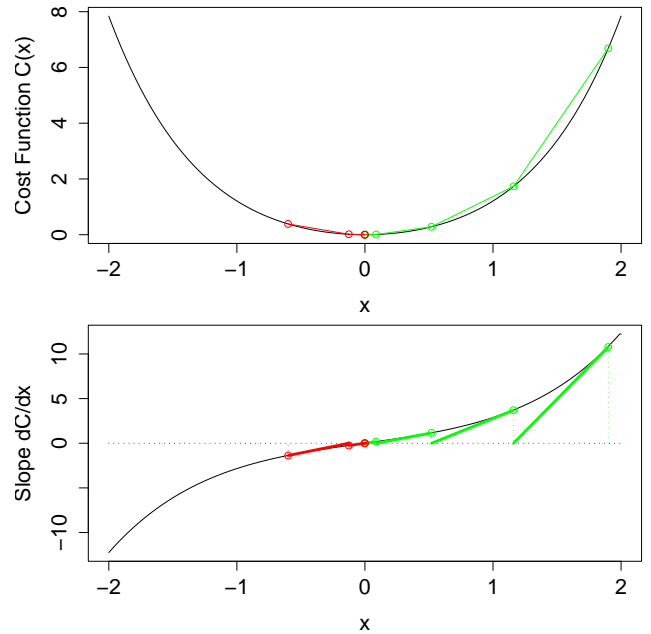


Figure 2: As Figure 1 but the cost function is no longer exactly parabolic, so  $C'(x)$  is no longer a straight line. The Inverse Hessian formula takes several iterations to find the minimum of  $C(x)$ . The further away it starts, the more iterations are required. The figure shows two attempts, starting at  $+1.9$  and  $-0.6$ .

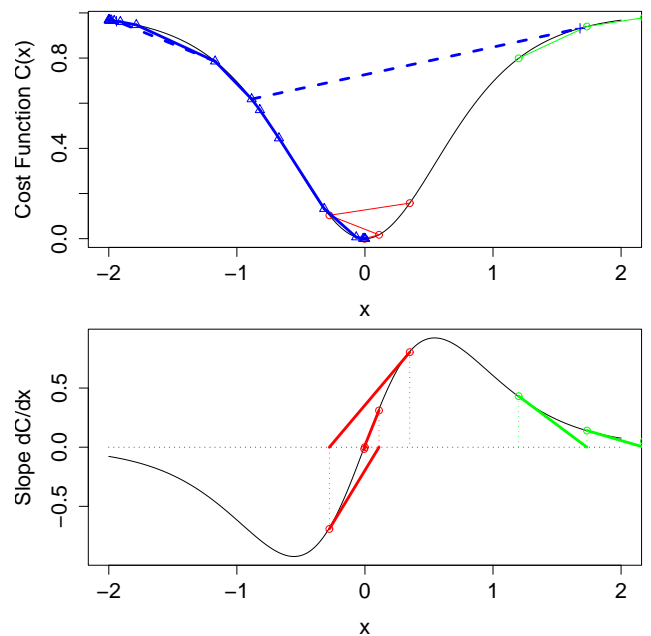


Figure 3: As Figure 1 but for a  $C$  with a more difficult sort of nonlinearity. Note how if we start the Inverse Hessian at  $-0.35$ , then it finds the answer, but if we start it at  $1.2$ , it goes off in the wrong direction. The Marquadt-Levenberg formula is shown starting at  $-2.0$ . Successful steps ( $C$  decreases) are shown with solid lines ending in a triangle while unsuccessful steps are shown with dashed lines ending in a cross.

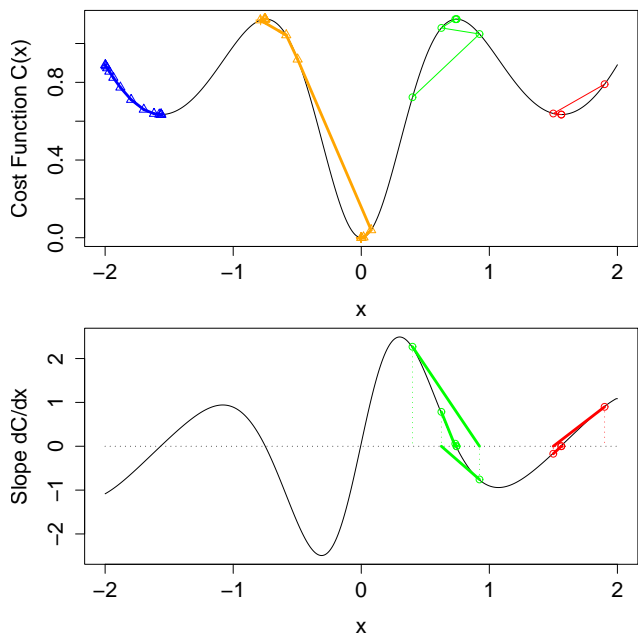


Figure 4: As previous figures, but for a  $C$  with several minima. The Inverse Hessian (circles) always finds a stationary point of  $C$  but it may be a maximum as well as a minimum. The Marquadt-Levenberg method always finds a local minimum, but it may not be the global minimum that we really wanted.