Chapter 4

From sensitivity to optimization

4.1 Estimating a solution through regularization

In the previous chapters we have discussed in detail the discretization of the forward problem and the sensitivities with respect to parameters. In this chapter we discuss how, given data, we can pose a problem for an unknown parameter in the PDE.

Assume that the (discrete) forward problem, the discretized PDE, is written as

$$c(m,u) = 0 \quad \text{or} \quad u = u(m)$$

and that the data is given by a linear operation on the solution

$$d = Qu(m).$$

Let $d^{\rm obs}$ be some observed data. Our goal is to find a "reasonable" parameter m such that $u(m)$ fits the data. One may think that it is possible to obtain such a model by solving the optimization problem

$$\mathsf{misfit}(m) = \min_m \ \ \frac{1}{2}\|Qu(m) - d^{\rm obs}\|^2.$$

However, this problem often does not have a unique solution and even if it has a unique solution it is typically unstable.

To see why this is the case consider for a moment the DC resistivity problem of evaluating m given u in 1D and assume data everywhere, $d = u + \epsilon$. In this case we have that

$$(mu')' = q$$

and assuming that we have u everywhere we can attempt to obtain m by the following steps.

- Evaluate u' from the data, say, set $u' = d'$
- set $m(x) = d'(x)^{-1} \int_0^x q(t)dt$

Evaluating the integral should be straight forward however, the estimation of m requires the estimation of the derivative of u. Consider the data

$$d = u + \epsilon \sin(n\pi x)$$

that is d is polluted with a small, high frequency component. In this case we have that

$$d' = u' + \epsilon n \sin(n\pi x)$$

and for sufficiently large n the derivative of d is swamped by the noise, even though the data itself may have very little noise. For realistic applications the situation is worst since the data is finite and may not have sufficient support. It is therefore unadvised to attempt and solve the problem directly and some stable process is preferred.

We thus acknowledge the fact that there are infinitely many solutions, many solutions that give a reasonable fit to the data. The question that we ask is, which solution, out of the infinitely many we want to recover. Our strategy is to use optimization. The solution we want to recover minimizes a functional $R(m - m_{\rm ref})$ where $R(\cdot)$ is a function from $\mathbb{R}^n \rightarrow \mathbb{R}$ which we call the *regularizer*. For all regularizers we have that $R(0)$ minimizes $R(t)$. Since we have optimization in mind, a convex function is more useful than a non-convex one although non-convex functions have been used with some success in the past. The choice of $R(\cdot)$ is crucial! Different choices lead to very different solutions. Thousands of papers and many books have been written on justifying a particular choice. Obviously, for a meaningful solution of the problem we need to have $R(m - m_{\rm ref})$ small for the "true" solution. If $R(m_{\rm true} - m_{\rm ref})$ is not small then the resulting computed solution is likely to be far from the "true" solution. In the following sections we discuss different regularization techniques and their validity and computability.

To obtain and optimization problem for the model we need to consider the noise model. Our model is rather simple as we assume that the observed data is given by

$$Qu(m) + \epsilon = d^{\rm obs}$$

with ϵ is a vector in $\mathbb{R}^k$ of an iid normally distributed noise with 0 mean and σ^2 standard deviation. Consider now quantity $\epsilon^\top \epsilon$. Since each ϵ_i is random the scalar variable φ_d is also random. In fact, this variable has a χ^2 distribution. One can verify that

$$\mathbf{E}\ (\epsilon^\top \epsilon) = \sigma^2 k \qquad (4.1a)$$

$$\mathsf{Var}\ (\epsilon^\top \epsilon) = \sigma^2 \sqrt{k} \qquad (4.1b)$$

It is therefore reasonable to pose the following optimization problem in order to obtain a "reasonable" model that fits the data

$$\min \quad R(m - m_{\rm ref}) \qquad (4.2a)$$

$$\text{s.t} \quad \frac{1}{k}\|Qu(m) - d^{\rm obs}\|^2 \leq \sigma^2 \qquad (4.2b)$$

Obviously, the number σ^2 is not "set in stone". Especially when the number of data is not very large. Note that the standard deviation of the χ^2 function implies that this number has a variance of $\sigma^2 k^{-\frac{1}{2}}$. On the other hand, for very large scale problems where the number data is in the thousands this estimate of the noise is very accurate.

We now recall a general methodology for the solution of constrained optimization problems. The Lagrangian of this problem is

$$R(m - m_{\rm ref}) + \beta \left(\frac{1}{k} \|Qu(m) - d^{\rm obs}\|^2 - \sigma^2 \right) \tag{4.3}$$

where β is a Lagrange multiplier. The conditions for a minimum are

$$R'(m - m_{\rm ref}) + \frac{2\beta}{k} J(m)^\top (Qu(m) - d^{\rm obs}) \tag{4.4a}$$

$$\beta \left(\frac{1}{k} \|Qu(m) - d^{\rm obs}\|^2 - \sigma^2 \right) = 0 \tag{4.4b}$$

$$\beta \geq 0 \quad \frac{1}{k} \|Qu(m) - d^{\rm obs}\|^2 \leq \sigma^2 \tag{4.4c}$$

where $J(m) = -Q \nabla_u c^{-1} \nabla_m c$ is the sensitivity matrix.

If our reference model does not fit the data, then the solution is obtained when $\frac{1}{k}\|Qu(m) - d^{\rm obs}\|^2 = \sigma^2$. Define

$$\alpha = \frac{k}{2\beta}$$

and we see that the solution is equivalent to the unconstrained optimization problem

$$\min \quad \alpha R(m - m_{\rm ref}) + \frac{1}{2} \|Qu(m) - d^{\rm obs}\|^2 \tag{4.5}$$

for the appropriate choice of α. The problem Eq. (4.5) is often referred to as Tikhonov regularization. There are thousands of papers and many books that discuss this form of regularization.

4.2 Quadratic regularization

Maybe the most simple regularization is quadratic. Setting

$$R(m) = \frac{1}{2} \|Lm\|^2$$

Where L is some operator.

An important point need to be made here. Although it is simple to choose any discrete operator care must be taken such that the problem is scaled in the right way. Consider for example the case that L is the identity. The continuous analog of R is

$$R(m) = \frac{1}{2} \int_\Omega m(x)^2 \, dx$$

Again, using the midpoint method and assuming a regular grid of spacing h we obtain

$$R(m) = h^{\mathrm{dim}} m^\top m$$

where dim is the dimension of the problem. If we do not use the appropriate scaling then the solution of the problem on different grids is different, because this implies that different problems are solved on different grids.

Regularization operators that have been successfully used for many problems include $L = \nabla_h$, $L = \Delta_h$ (where h implies discretization of the differential operators) and variations and combination of thereof. These operators imply that the solution is expected to be smooth, with no discontinuities. For smooth problems such regularization is hard to beat.

Consider the special case where

$$c(m, u) = Au - Gm = 0$$

then

$$d^{\mathrm{obs}} = QA^{-1}Gm = Jm$$

In this case it is easy to obtain a closed form solution to the problem. Substituting the regularization into the optimization problem Eq. (4.5) we obtain

$$\min \quad \frac{\alpha}{2}\|Lm\|^2 + \frac{1}{2}\|Jm - d^{\mathrm{obs}}\|^2$$

and its solution is

$$\widehat{m} = (J^\top J + \alpha L^\top L)^{-1} J^\top d^{\mathrm{obs}}.$$

Also, it is easy to see that the problem is equivalent to the least-squares problem

$$\begin{pmatrix} J \\ \sqrt{\alpha} L \end{pmatrix} m = \begin{pmatrix} d^{\mathrm{obs}} \\ 0 \end{pmatrix}. \tag{4.6}$$

The advantage of this observation is that it is possible to use least-squares solvers for the solution of the problem without ever forming $J^\top J$ or even having J explicitly. Methods such as Conjugate Gradient (CG), Conjugate Gradient Least Squares (CGLS) and Least-Squares QR (LSQR) are very effective methods for the solution of such problems. These methods are iterative and require only matrix vector products of the form Jv and $J^\top w$.

```
function x = cgls(A,b,k)

x = zeros(n,1);
d = A'*b; r = b;
normr2 = d'*d;

for j=1:k

  Ad = A*d; alpha = normr2/(Ad'*Ad);
  x  = x + alpha*d;
  r  = r - alpha*Ad;
  s  = A'*r;
  normr2_new = s'*s;
  beta = normr2_new/normr2;
  normr2 = normr2_new;
  d = s + beta*d;
end
```

It is known that the convergence of conjugate gradient depends on the condition number of the system. Consider first the case that L is the identity. The matrix to be inverted is $J^\top J + \alpha I$. Since J is a discretization of an integral operator it is typically compact and its singular values are bounded from above, independent on the mesh size. Therefore, the eigenvalues of $J^\top J + \alpha I$ are bounded from above and below *independent* on the mesh size and the number of CG iteration is fixed. Next, consider the case that L is a differential operator and that $L^\top L$ is invertible. In this case the eigenvalues of $L^\top L$ cluster at infinity and the condition number of the matrix is mesh dependent. This can be easily avoided by preconditioning. Consider the preconditioned system $(L^\top L)^{-1}(J^\top J + \alpha L^\top L)$. It is easy to verify that the condition number of this system is also bounded independent of the mesh and therefore, the number of CG iteration is mesh independent.

Although the number of CG steps can be made mesh independent it is strongly depends on α. In fact, we should not confuse the words mesh-independent with small. To have a small number of iterations one must have an appropriate preconditioner. Preconditioning for ill-posed problems is an open field of research.

4.2.1 Programmer note Jv and $J^\top w$

For the problems discussed above, computing J is not recommended and should be avoided in practice. This implies that one needs a code to compute products of the form Jv and $J^\top w$. **Never** assume that two codes, one that compute Jv and one that computes $J^\top w$ are indeed adjoints of each other. A simple test is as follows. Choose random vectors v, w and compute (by using your code)

$$w^\top (Jv) \quad \text{and} \quad v^\top (J^\top w).$$

These expressions should be equal (up to roundoff errors). If they are not you likely have a bug in your code.

4.3 ℓ_1 Regularization

A different regularization from the one we have seen above uses the 1-norm rather than the 2-norm. It is rather well known that the one norm yields solution with many zeros and only very few nonzeros. This observation was used extensively by geophysicists in the 70's and 80's [7, 26, 31] to obtain so-called spiky solutions to inverse problem. Recently some proofs about the amount of sparsity under some strict conditions have been proved [6] and this has generated a "hot" trend within the inverse problem community, trying to solve almost all inverse problems with sparse-like solution. We now review some of the techniques for sparse recovery and discuss some of the applications its advantages and limits.

Consider first the case of imaging a star cluster. Obviously, stars are "spikes" and therefore this is a simple case of "sparse solution". Sparse solutions implies that most of the entries in m are zero and as explained above we minimize

$$\min_{m} \quad \frac{1}{2}\|Jm-d\|^2 + \alpha\|m\|_1 \tag{4.7}$$

If the model we require is not sparse we assume that we can express it using a basis function

$$m = Wz$$

where W is some basis and z are coefficients. The assumption is that the model can be expressed using only a few of the basis vectors in W and therefore we can minimize

$$\min_{z} \quad \frac{1}{2}\|JWz-d\|^2 + \alpha\|z\|_1 \tag{4.8}$$

The choice of W is crucial. It is easy to see that it is possible to choose W that yield sparse solutions without any advantage compared with the 2-norm solutions introduced in the previous section. For example, if we choose $W = V$ where V are the right hand singular vector matrix of J then, it is easy to verify that we simply obtain the truncated SVD solution. Choosing W judicially is problem dependent and for many inverse problems the appropriate W is hard to find.

The difficulty with this regularization is that it is not linear and even worst, it is not differentiable. The question is, how to effectively solve such problems. Here we discuss two main approaches.

First, it is possible to use Iterative Reweighted Least Squares (IRLS). IRLS has been used in the past for many problems with much success. IRLS is a simple strategy that linearly converges for the solution of the problem. Rather than solving the original non-differentiable problem we "regularize" the regularizer. Defining

$$\|m\|_{1,\epsilon} = \sum_i \sqrt{m_i^2+\epsilon}$$

we replace the one norm with a differentiable function, minimizing

$$\min_{m} \quad \frac{1}{2}\|Jm-d\|^2 + \alpha\|m\|_{1,\epsilon} \tag{4.9}$$

Then, replace the problem with a sequence of quadratic problems of the form

$$\min_{m_k} \quad \frac{1}{2}\|Jm_k-d\|^2 + \frac{\alpha}{2}m_k^\top \ \mathrm{diag}\left(\frac{1}{\sqrt{m_{k-1}^2+\epsilon}}\right) m_k \tag{4.10}$$

Hence the name, iterative reweighted least squares. The advantage of this approach is that one can use tools developed for the quadratic problem.

There are two disadvantage to IRLS. First each iteration can be rather expensive, solving a linear systems of equations to high accuracy when this may not be needed. Second, the choice of ϵ may not be easy and pose more difficulty.

A second approach for the solution of the ℓ_1 problem is to replaced the non-smooth problem by a smooth optimization problem with inequality constraints.

Setting $m = p - q$ with both $p, q \geq 0$, we show that the optimization problem Eq. (4.7) i s equivalent to the following optimization problem

$$\begin{aligned} \min_{p,q} \quad & \frac{1}{2}\|J(p-q) - b\|^2 + \alpha e^\top (p+q) \\ \text{s.t} \quad & p, q \geq 0, \end{aligned} \tag{4.11}$$

where $e = [1, \ldots, 1]^\top$. A very effective method, that does not require matrix inversion is the gradient projection method. We will discuss this method in the next section.

4.4 Total variation and Huber

One celebrated method of regularization is the total variation. The idea here is to obtain a piecewise constant solution. Let us consider this regularization in 1D first. In continuous setting, the regularization operator can be written as

$$TV(m) = \int_\Omega \left| \frac{dm}{dx} \right| \, dx$$

To see the effect of this regularizer, assume that $m(x)$ is a piecewise constant function on $[0,1]$ with value a for $x < \frac{1}{2}$ and b otherwise. Then, it is easy to verify that

$$TV(m) = b - a$$

thus, TV regularization penalizes the jump but allows it. To see why TV may prefer a jumpy or non-smooth solution, consider the following interpolation and extrapolation problem. Assume that $m(0.25) = 0.25$ and $m(0.75) = 0.75$. Assume we would like to recover m everywhere in $[0,1]$. Consider first a linear interpolation and extrapolation. Obviously, a linear function is a very smooth function. The interpolation leads to $m_1(x) = x$ which obviously fits the data. It is easy to see that $TV(m_1(x)) = 1$. Now consider the function

$$m_2 = \begin{cases} 0.25 & x < x_M \\ 0.75 & \text{otherwise} \end{cases}$$

where x_M is any point in the interval $[0.25, 0.75]$. It is easy to calculate that $TV(m_2(x)) = 0.5$. Thus, TV regularization generally prefer non-smooth solutions over the smooth ones.

4.4.1 Discretization in 1D

Discretization in 1D is straight forward. Assume that $m \in [0,1]$ then, we divide the interval into cells by the nodes $\{x_1, \ldots, x_n\}$. The mid of each cell is numbered as $\{x_{\frac{3}{2}}, \ldots, x_{n-\frac{1}{2}}\}$. Assume that m is discretized in nodes then, define the discrete approximation

$$TV_h(m) = h \sum \frac{|m_{j+1} - m_j|}{h}$$

which is a second order approximation to the continuous TV function.

As usual, it is beneficial to think about this regularization in matrix form. If we let D be an $(n-1)\times n$ difference matrix

$$D = \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{pmatrix}$$

then, we can write

$$TV_h(m) = e^\top |Dm|$$

Again, to work with the non-differentiability it is possible to smooth the origin. One way to achieve this is to use the Huber function

$$\rho(t,\theta) = \begin{cases} \frac{t^2}{2\theta} & |t| \le \theta \\ |t| & \text{otherwise} \end{cases}$$

Using this approximation we obtain

$$TV_h^\theta(m) = e^\top \rho(Dm).$$

It is easy to verify that

$$\nabla TV_h^\theta(m) = D^\top \mathrm{diag}\left(\frac{\rho'(Dm)}{Dm}\right) Dm.$$

Using the derivative we can now suggest a method for the solution of the optimization problem

$$\min_m \ \frac{1}{2}\|Jm-d\|^2 + \alpha TV(m).$$

The gradient is

$$J^\top(Jm-d) + \alpha D^\top \mathrm{diag}\left(\frac{\rho'(Dm)}{Dm}\right) Dm.$$

The lagged diffusivity method uses the following fixed point iteration for the solution of the problem

$$J^\top(Jm_k-d) + \alpha D^\top \mathrm{diag}\left(\frac{\rho'(Dm_{k-1})}{Dm_{k-1}}\right) Dm_k = 0$$

Although the iteration converge slowly, it tends to have satisfactory results (at least in the eyeball norm) within a few iterations.

4.4.2 Discretization in 2D

The discretization in 2D is slightly more complicated. Consider the 2D grid and consider the cell who's corners are $[i,j],[i+1,j],[i,j+1],[i+1,j+1]$. We can approximate the derivatives on the edges

$$\begin{aligned}
(m_x)_{i+\frac{1}{2},j} &= \frac{1}{h}(m_{i+1,j} - m_{i,j}) + \mathcal{O}(h^2) \\
(m_x)_{i+\frac{1}{2},j+1} &= \frac{1}{h}(m_{i+1,j+1} - m_{i,j+1}) + \mathcal{O}(h^2) \\
(m_y)_{i,j+\frac{1}{2}} &= \frac{1}{h}(m_{i,j+1} - m_{i,j}) + \mathcal{O}(h^2) \\
(m_y)_{i+1,j+\frac{1}{2}} &= \frac{1}{h}(m_{i+1,j+1} - m_{i+1,j}) + \mathcal{O}(h^2)
\end{aligned}$$

Now, to obtain a second order approximation for the TV function we *average the squares* (rather than square the average), and summing over all cells multiplied with their associated volumes, obtaining

$$\begin{aligned}
TV_h(m) = \sqrt{2}h \sum ((m_{i+1,j} - m_{i,j})^2 + (m_{i+1,j+1} - m_{i,j+1})^2 \\
+ (m_{i,j+1} - m_{i,j})^2 + (m_{i+1,j+1} - m_{i+1,j})^2)^{\frac{1}{2}}
\end{aligned}$$

Again, we would like to obtain a matrix form for this function. Let D be the 1D difference matrix defined in the previous chapters. Then, the x derivative can be written as

$$D_x = I \otimes D$$

and the y derivative can be written as

$$D_y = D \otimes I$$

where $\otimes$ is a kronecker product and I is an identity matrix. Then, we can write

$$TV_h(m) = he^\top (A_y(D_x m)^2 + A_x(D_y m)^2)^{\frac{1}{2}}$$

where $A_{x,y}$ are averages matrices that average from the edges of the cells to the cell centers. It is easy to see that $A_{x,y}$ can be also obtained by using kroneker products.

Once again, to avoid the problem of non-differentiability, it is possible to replace the non-differentiable TV function by a corresponding smoothed approximation.

4.5 A comparative study

In this section we examine different regularization techniques and see their effect on a simple model problem. Rather than using a forward problem taken from PDE's we take a forward problem that has similar structure given by the equation

$$d(x) = \int_\Omega K(x, \xi) m(\xi) \, d\xi$$

This is an integral equation of the first kind with a kernel $K(\vec{x}, \vec{\xi})$.

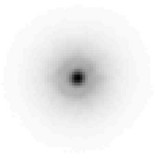

Figure 4.1. *The Kernel function for our experiment .*

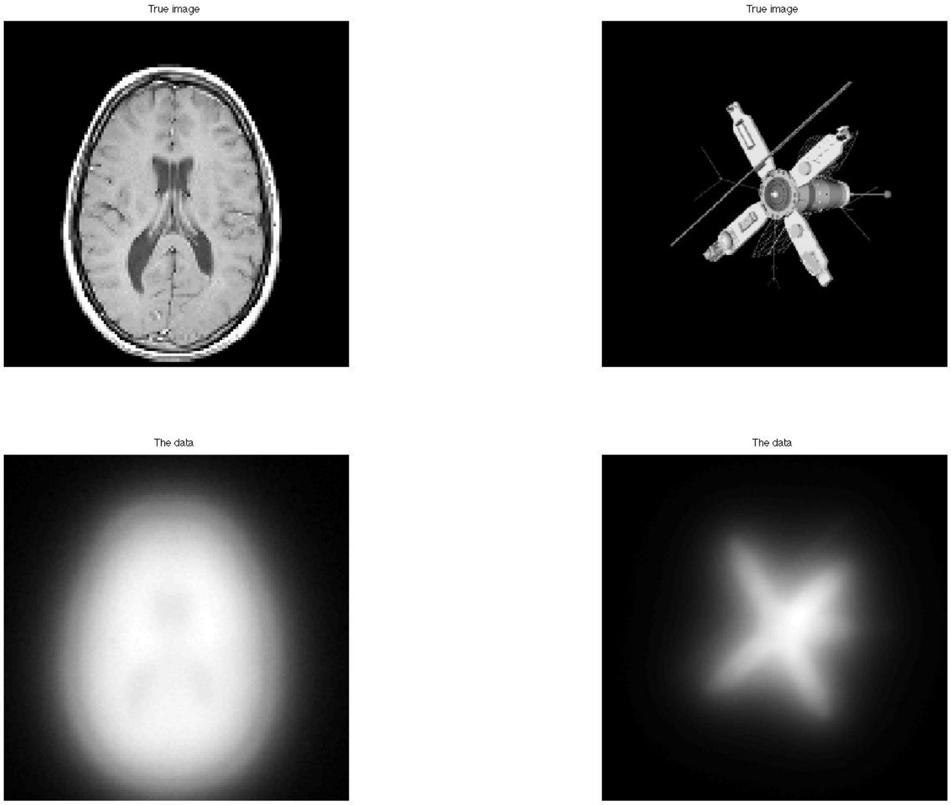

Figure 4.2. *The models and data associated with them .*

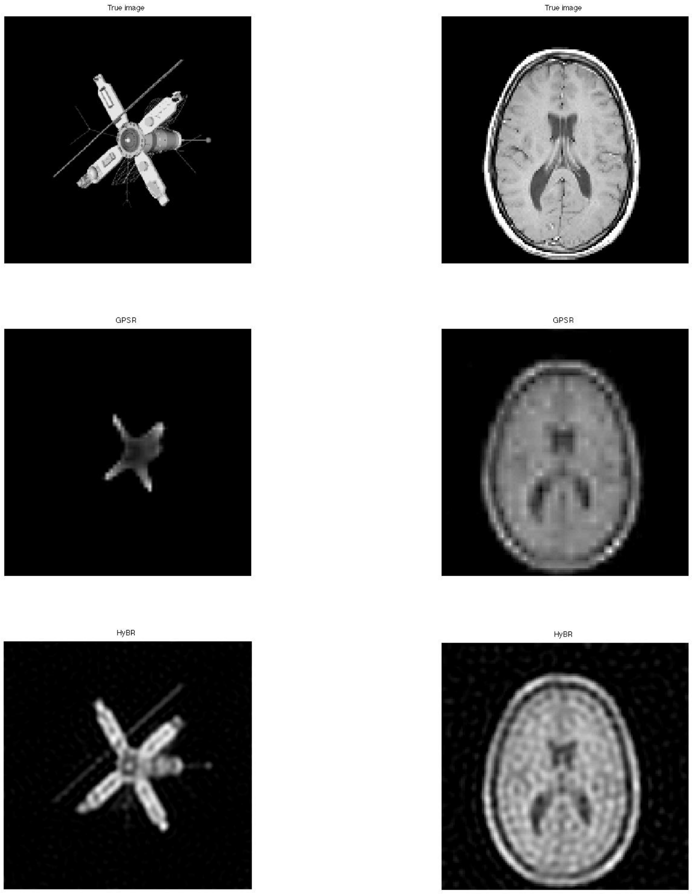

Figure 4.3. *Reconstructions given different regularizers* .

For our experiment we generate the kernel by the using the package restore tools written by James Nagy. The Kernel is plotted in Figure 4.1. Using the Kernel we blur two images. An MRI image and a satellite image. The images and the blurred data are presented in Figure 4.2 This blur is a relatively strong one and it is similar to action of many sensitivity calculations. We now use different algorithms for the recovery of these objects. Below, in Figure 4.3 we present the results of a hybrid regularization which is an approximation to the L_2 regularizer, the L_1 results as obtained by the package GPSR (by Steve Wright). For both problems we chose the regularization parameter such that the data fit by the discrepancy principle is obtained.

It is important to note that at least for the problems here, the results are very different however, it is not easy to say that one form of regularization seriously outperform another. In my experience this is the case for many ill-posed problems and although some publications suggest that using simple, quadratic regularization is unadvised, I would beg the differ.