Inverse Theory Methods in Experimental Physics

Edward Sternin

PHYS 5P10: 2018-02-26

Framework: indirectly observed data

Inverting the Fredholm Integral Equation of the 1st kind (FEI):

$$S(\omega) = \int g(x)\, K(\omega, x)\, dx \tag{1}$$

- $S(\omega)$ — what you measure experimentally (*e.g.* an NMR spectrum)
- $g(x)$ — what you want to know (*e.g.* chemical shifts)
- $K(\omega, x)$ — a *kernel* function

is, in general, an *ill-posed* problem. Only a few choice kernels allow a complete inverse calculation: $S(\omega) \longrightarrow g(x)$, *e.g.* a Fourier transform.

Most other kernels require numerical solutions.

A new research paradigm

An unexpected benefit: a "new research paradigm"[1], where data analysis is an essential part of the experiment itself. The "model" of the traditional paradigm, where it is kept scrupulously separate from the experiment, becomes *a part* of it, subject to computer-based tweaking and straining.
This yields estimates of the parameters, quantitative analysis of the misfit of the model to the experimental data, and, ultimately, better models.

[1] J.V.Beck, *Perspective on the Relation of Current Engineering Practice to Inverse Problems*, Michigan State University, 1998. http://www.me.ua.edu/research/inverse-problems/perspective.html.

A new research paradigm

The language of this new paradigm is the language of the inverse problems, and its range of applicability is vast:

- medical, industrial, and geophysical imaging, using X-ray or magnetic resonance or electrical impedance measurements;
- analysis of EEG traces in neuroscience and experimental psychology;
- inverse heat problems in the casting of steel;
- scattering problems in physics and geological exploration;
- extraction of material properties from indirect bulk measurements.

Ill-posedness and what to do with it

Existence: a solution $g(x)$ may not exist at all

Then: ask not for the "true" $g(x)$, but for "a reasonable approximation $\tilde{g}(x)$". Here, "reasonable" is in the least-squares sense, a minimum of misfit

$$\Psi = \left\| S(\omega) - \int \tilde{g}(x) K(\omega, x) dx \right\|^2$$

This misfit norm, the "distance" between the measured $S(\omega)$ and the approximation calculated as an integral over $\tilde{g}(x)$, is sometimes called the least-squares error norm. A minimum of misfit ensures compatibility of the fit with the measured data.

Ill-posedness and what to do with it

Uniqueness: many $g(x)$ may satisfy Eq.1 equally well

Then: bring in additional physical input. Of all compatible solutions, choose those that satisfy those additional constraints. *E.g.* a reasonable physical function is "smooth", so minimize $\|g''(x)\|^2$.
Each problem is unique and may need a different optimum constraint. Test using numerical modeling (*e.g.* Monte Carlo).

Ill-posedness and what to do with it

Stability: a small perturbation in $S(\omega)$ (*e.g.* experimental noise) may cause a large jump in $g(x)$

Then: regularize the problem, enforcing local stability of the inverse $S(\omega) \longrightarrow g(x)$ mapping

Formally, lack of stability means that two "adjacent" forward mappings S_1, S_2 do not have "adjacent" origins g_1, g_2, *i.e.* that one cannot find two numbers η and $\varepsilon(\eta)$ so that from $\|g_2 - g_1\| \leq \eta$ follows $\|S_2 - S_1\| \leq \varepsilon(\eta)$, independently of the choice of g_1 and g_2. Lack of stability is a distinguishing feature of all ill-posed inverse problems.

Regularization

Discretization: just choosing the right set of $\{x_i\}$ for which to ask about $g(x_i)$ may do the trick.
Often done implicitly, but there are mathematical implications.
cf. Nyquist limit, noise folding, and zero filling in FT NMR

Truncation of SVD: expand the transformation matrix in its "singular values", and then truncate the expansion, ignoring the lower values which are responsible for most instabilities.
cf. Taylor series expansion of functions

Regularization

Tikhonov regularization: bring in additional constraints, *i.e.* minimize

$$\Psi\{g(x)\} = \left\| S(\omega) - \int g(x) K(\omega, x) dx \right\|^2 + \lambda T\{g(x)\}$$

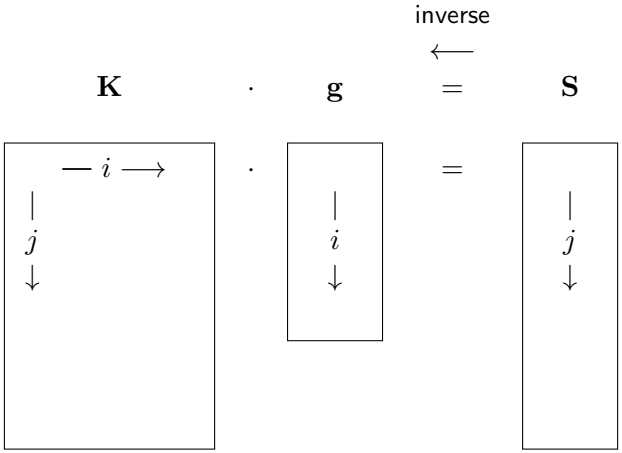
$$T\{g(x)\} = \begin{cases} \|g\|^2 & \text{(Tikhonov)} \\ \|g''\|^2 & \text{(Phillips)} \\ -\int g \log g \, dx & \text{(Shannon, maximum entropy)} \end{cases}$$

where λ controls the balance between compatibility with the data, and the regularizing effects of $T\{g\}$

Regularization by discretization

In the ω-direction, a natural discretization of a FIE occurs when a discrete set of data sampling points ω_i is considered. Discretization in the x-direction is realized when, for example, g is approximated with a sum of δ-functions, $\sum_i g_i \, \delta(x - x_i)$, on a pre-selected grid $\{x_i\}$ with the unknown coefficients g_i, or when $g(x)$ is assumed to be a linear function within each of the pre-selected intervals $[g_i, g_{i+1}]$. This is often done implicitly in the "standard" analysis of the experimental data, but strictly speaking, such discretizations impose strong restrictions on a possible solution g. This alone may sufficiently stabilize the inverse problem to overcome its ill-posedness.

A discrete formulation of the FIE problem



Ranges of i, j define the problem as even-, over-, or under-determined.

Underdetermined ?!

For an experimentalist, it is somewhat startling to think of a fit to *more* "parameters" than there are "measured data points", but this simply underscores the fact that regularized solutions of inverse problems are fundamentally different from least-squares fits.
The discrete inverse problem is equivalent to calculating the inverse matrix, $\mathbf{K}^{-1}$ such that $\mathbf{g} = \mathbf{K}^{-1}\mathbf{f}$. If such an inverse does not exist (a majority of cases), the problem becomes the calculation of the *pseudo-inverse*.
The right kind of numerical strategy is important, and much depends on the experience and judgment of the person performing the calculation: "*successful inverse problem solving is strongly dependent on the analyst*"[2]

[2] J. C. Santamarina and D. Fratta. *Discrete signals and inverse problems. An introduction for engineers and scientists*. John Wiley & Sons, 2005.

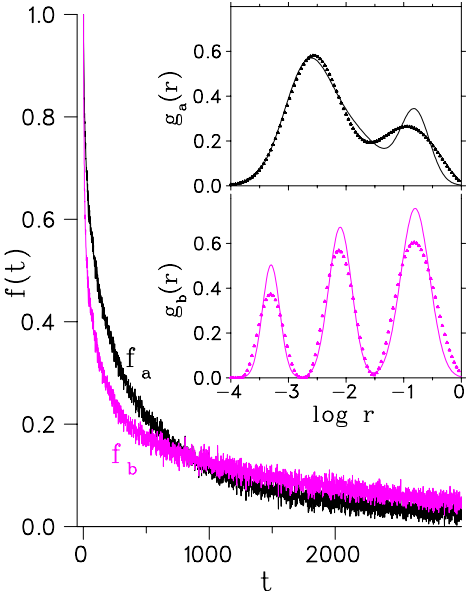
Example: multi-exponential decay

$$f(t) = \int_{r_{min}}^{r_{max}} g(r) e^{-rt}\, dr$$

where t represents time and r has the physical meaning of a relaxation rate. The data $f(t)$ is a decay curve in the linearly-sampled time domain, and the desired inverse solution $g(r)$ is a distribution of relaxation rates.

Our example: a broad asymmetric distribution of relaxation rates simulated in the logarithmically-sampled range of $r_{min} = 10^{-3}$ to $r_{max} = 10^{0}$, with random normally-distributed noise added at 1% of the maximum $f(t)$ intensity.

Example: multi-exponential decay



Two simulated relaxation-rate distribution functions, $g_a(r)$ and $g_b(r)$ (smooth lines in the insert), are used to generate two time-domain signals, each with 1% random noise is added. The resulting $f_a(t)$ and $f_b(t)$ are very similar.

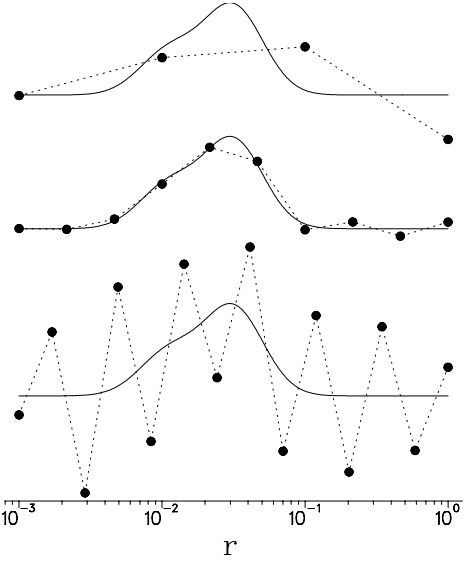
For now, ignore the points in the insert.

Discretization strategy

- Select enough points in the grid of parameter values to allow for the entire physically relevant range to be covered with sufficient resolution to reproduce all of the essential features in the data; and no more.
- Acquire enough data points, sufficiently spread out in the observation domain to resolve contributions from different parameter values.
- Over-determined problems, by at least a factor of 2, are easier to solve.
- Higher parameter grid densities require better signal-to-noise ratio in the data.

Use numerical simulation and testing (trial-and-error) to establish a good discretization scheme for a given problem. Near the optimum, the results of the calculation should be largely independent of the exact choice of the discretization grids.

Discretization example

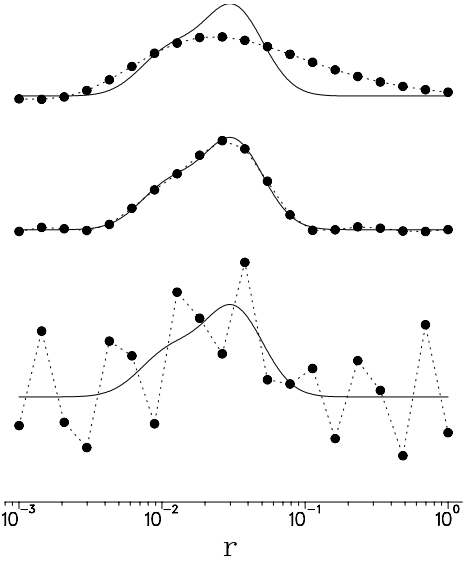


Regularizing effects of discretization for a distribution of relaxation rates. Level of discretization is varied by changing the total number of points across the parameter space. At just the right level of regularization (**middle plot**), the re-calculated $\tilde{g}$ is a faithful representation of the true g.

SVD truncation strategy

- Many discrete ill-posed problems exhibit a gradual decrease in the size of their singular values.
- As the kernel matrix is expanded in its singular values, the lower-valued ones tend to magnify the effects of noise in the measured data.
- Regularize by truncating the SVD expansion!
- This has to be done in the region free of rapid changes in the size of the singular values.

SVD truncation example



Overly smooth pseudo-inverse solution (**top**) fails to reproduce some of the features of the true g, and insufficient regularization produces an unstable pseudo-inverse (**bottom**). The optimum SVD truncation (**middle**) faithfully reproduces the true g.

Tikhonov regularization

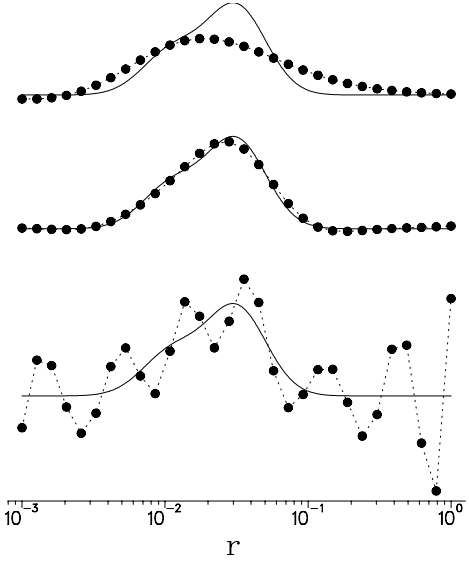
Applying Tikhonov regularization to a noisy experimental dataset

$$\Psi\{g\} = \left\| f(t) - \int g(r)K(r,t)dr \right\|^2 + \lambda \left\| g(r) \right\|^2$$

crucially depends on the right choice for λ: if λ value is too small, a stable solution will not be found; if λ is too large, essential physical features in the solution will be obscured.

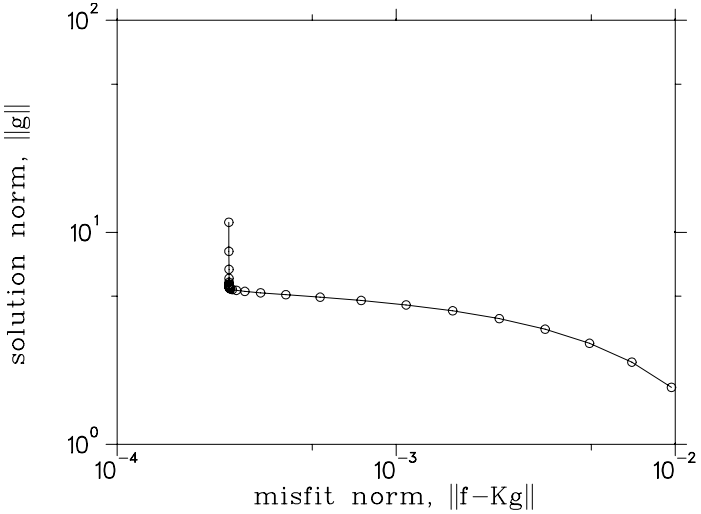
This form of the regularization term gives preference to "small" $g(x)$: in the discrete case, those with as few as possible non-zero values in $\{g_i\}$.

Tikhonov regularization



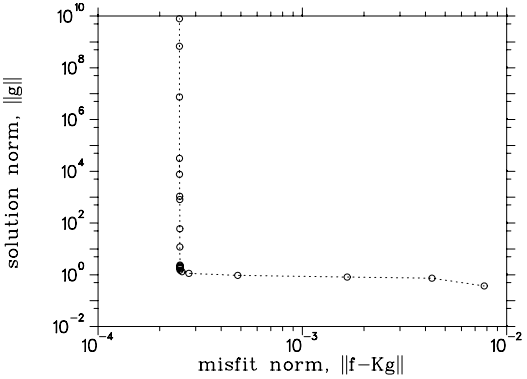
Overly "small" pseudo-inverse solution (**top**) fails to reproduce some of the features of the true g, and insufficient regularization produces an unstable pseudo-inverse (**bottom**). The optimum choice of λ (**middle**) faithfully reproduces the true g.

Optimum regularization: the L-curve

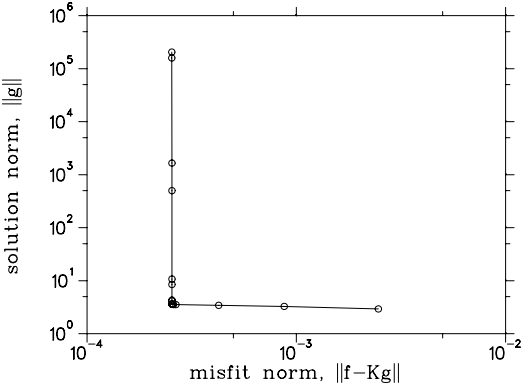


L-curve is a plot of the norm of the regularized solution $\|\tilde{g}\|$ versus the corresponding misfit norm $\|\mathbf{f} - \mathbf{K}\tilde{\mathbf{g}}\|$, as the regularization is varied. Upper left: loss of stability in $\tilde{g}$; lower right: loss of compatibility with the data; the corner corresponds to the optimum regularization.

Optimum regularization: the L-curve

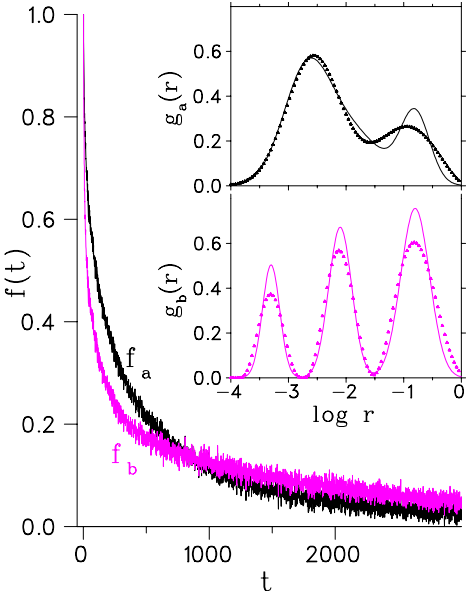


L-curve used in selecting optimum grid density in r



L-curve used in selecting optimum SVD truncation

Example: multi-exponential decay



Two simulated relaxation-rate distribution functions, $g_a(r)$ and $g_b(r)$ are successfully recovered from the two, essentially indistinguishable, simulated noisy datasets $f_a(t)$ and $f_b(t)$.

This time, do focus on the points in the insert!

The results of the inversion using Tikhonov regularization, of $f_a(t)$ and $f_b(t)$, reproduce the true exponential rate distributions faithfully.

Skeleton code

```
function [g] = regularize(t,f,r,K,svd_n,lambda)
  m=length(f); n=length(r);
  if m < n then
    error("For overdetermined problems only, need more data");
  end;
  [U,S,V]=svd(K);        //  SVD of the kernel matrix
  nt=n;
  if svd_cnt > 0 then    //  if requested, truncate singular values
    nt=max(min(n,svd_cnt),2);   //  less than n, but not too few!
  end;
  sl=S(nt,nt);           //  Tikhonov regularization
  for k=1:nt
    sl(k,k)=S(k,k)/(S(k,k)^2+lambda);
  end;
  g=V(1:n,1:nt)*sl*U(1:m,1:nt)'*f;    // return g(r)
endfunction;
```

Skeleton code, cont'd

```
////////////////////////////////////////////////////////////
//  multi-exponential inverse analysis of time decay curves
//    f(t) = \int g(r) exp(-r*t) dr
////////////////////////////////////////////////////////////
//    r = vector or r values
//    n = number of points in r
//    g = vector of unknowns, g(r)
//    t = vector of t values
//    m = number of points in t
//    f = vector of measured data, f(t)
//    K = (m x n) kernel matrix
//    svd_cnt = keep only this many singular values
//    lambda = Tikhonov regularization parameter
//    datafile = (noisy) data, in two columns: (t,f)
////////////////////////////////////////////////////////////
```

Skeleton code, cont'd

```
datafile='test.dat';
r_steps=30;      // r-grid has this many points,
r_min=1e-3;      //      from this minimum,
r_max=1e0;       //      to this maximum
svd_cnt=11;      // set to 0 for no SVD truncation‘
lambda=2e-4;     // set to 0 for no Tikhonov regularization

//  read in the time-domain data
fd=mopen(datafile); [n,t,f]=mfscanf(-1,fd,"%f %f"); mclose(fd);

//  create a vector of r values, logarithmically spaced
r_inc = (log(r_max)-log(r_min)) / (r_steps-1);
r = exp([log(r_min):r_inc:log(r_max)]);

//  set up our kernel matrix, normalize by the step in r
K = exp(-t*r) * r_inc;
```

Skeleton code, cont'd

```
//  call the inversion routine
[g] = regularize(t,f,r,K,svd_cnt,lambda);

//  plot the original data f(t) and our misfit
scf(1); clf; plot(t,f,'-');
misfit = f - K*g ;
Psi = sum(misfit.^2)/(length(f)-1);
scale=0.2*max(f)/max(misfit);
plot(t,scale*misfit,'-r');
legend('input data, f(t)','misfit, x'+string(scale));
xtitle('LS error norm = '+string(Psi),'t','f(t)');

//  separately, plot the result of the inversion
scf(2); clf; plot(log10(r),g,':o');
xtitle('inverse solution','log r','g(r)');
```

Homework

- Review the concept of SVD decomposition.
- Analyze the skeleton code of `regularize()`; make sure you understand every line.
- Modify the main code by adding appropriate loops, *etc.* to reproduce the results presented for the exponential example, including the L-curves, on the sample data provided in `/work/5P10/test.dat`
- Automate the optimum selection of parameter λ. One possible approach is to seek the value that corresponds to the shortest distance to the origin on the L-curve. Be efficient: vary the step size in λ depending on how strong the dependence on λ is.
- Optionally, use your program to analyze a real experimental data set (to be provided).