

Re: Smoothing Spline Algorithm

Source: <http://sci.tech-archive.net/Archive/sci.math/2007-04/msg02238.html>

- *From:* "Randy Poe" <poespam-trap@xxxxxxxxxx>
 - *Date:* 16 Apr 2007 11:43:26 -0700
-

On Apr 16, 1:39 pm, "woess...@xxxxxxxxxx" <woess...@xxxxxxxxxx> wrote:

I'm looking for a good reference on smoothing splines (using cubic natural splines). Specifically, I'm looking for an efficient algorithm to compute a smoothing spline (preferably with derivation). I have cooked up my own MATLAB code but I fear it's pretty inefficient. I've been able to reduce the problem to inverting a $3N-4$ banded matrix with bandwidth $2N$ (N is the number of knots).

However, according to this website:

http://www.physics.utah.edu/~detar/phyics6720/handouts/cubic_spline/cu...

the problem can be further reduced to solving a tridiagonal system. Unfortunately, they don't offer the algorithm or any kind of derivation. And I've been unable to reduce the problem further.

When I've had to do this, I used an approach which as I recall came from the SIAM Journal on Applied Mathematics, circa 1970.

While normal spline interpolation involves solving a linear system, my recollection is that the smoothing problem is not linear but I could be wrong.

As I recall, the idea is to minimize the total energy of a curve which is bound in two ways: A "stiffness" constant causes the curve to want to be straight. The energy of curvature is the integral of $S*|f''(x)|^2$, where $f(x)$ is the spline curve and S is the free parameter.

The second component of energy comes from tying the curve to the data points with rubber bands, so the energy is proportional to $k*|f(x) - y|^2$ where k is the spring constant of the rubber bands.

You can then ask what shape minimizes the total energy, the sum of these two terms, and that gives your

Re: Smoothing Spline Algorithm

smoothing spline.

http://www.physics.utah.edu/~detar/phyics6720/handouts/cubic_spline/cubic_spline/node2.html

Ah -- I see that what I described is exactly the problem being described here. But as you say they don't offer an algorithm.

I'm pretty sure about the journal reference where I first saw this. And I'm also pretty sure that there was enough information there for a young grad student to put together a working algorithm in FORTRAN.

Sorry I can't be more specific.

– Randy

.