
Subject: Re: getting derivatives from spline
Posted by lbuseett@yahoo.it on Sat, 11 Apr 2009 16:11:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 11 Apr, 08:56, Jeremy Bailin <astroco...@gmail.com> wrote:
> Does anyone know a way of getting SPLINE or any of its friends to
> return the derivatives of the interpolating function at the desired
> abscissae in addition to the interpolated values?
>
> Or, another way of getting at this question: what is everyone's
> favourite algorithm for numerically calculating the derivatives of
> (slightly) noisy data?
>
> -Jeremy.

Hi Jeremy,

I would consider Savitzky and Golay smoothing. The IDL "SAVGOL"
routine allows to compute the derivatives of the de-noised data.

Hope this helps,

Lorenzo

Subject: Re: getting derivatives from spline
Posted by jameskuyper on Mon, 13 Apr 2009 16:54:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

Jeremy Bailin wrote:
> Does anyone know a way of getting SPLINE or any of its friends to
> return the derivatives of the interpolating function at the desired
> abscissae in addition to the interpolated values?
>
> Or, another way of getting at this question: what is everyone's
> favourite algorithm for numerically calculating the derivatives of
> (slightly) noisy data?

Don't use splines to fit noisy data. The spline treats the noise as
being just as real as the signal. Particularly with higher-order
splines, it's easy to end up with interpolated data which looks even
noisier than the raw data.

Instead, fit the noisy data to a model with fewer free parameters than
you have data points. The difference between the number of data points
and the number of free parameters is a measure of the amount of
information you're discarding. If you've made a good choice of model,
the information you're discarding will be about the noise; with a bad

model, you'll be discarding information about the signal. Choosing the right model for your data is therefore a key step in this process.

The fundamental problem with using splines is that the number of data points is the same as the number of free parameters - you aren't discarding any noise.

Subject: Re: getting derivatives from spline
Posted by [Jeremy Bailin](#) on Tue, 14 Apr 2009 12:50:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Apr 13, 12:54 pm, jameskuyper <jameskuy...@verizon.net> wrote:

> Jeremy Bailin wrote:

>> Does anyone know a way of getting SPLINE or any of its friends to
>> return the derivatives of the interpolating function at the desired
>> abscissae in addition to the interpolated values?

>

>> Or, another way of getting at this question: what is everyone's
>> favourite algorithm for numerically calculating the derivatives of
>> (slightly) noisy data?

>

> Don't use splines to fit noisy data. The spline treats the noise as
> being just as real as the signal. Particularly with higher-order
> splines, it's easy to end up with interpolated data which looks even
> noisier than the raw data.

>

> Instead, fit the noisy data to a model with fewer free parameters than
> you have data points. The difference between the number of data points
> and the number of free parameters is a measure of the amount of
> information you're discarding. If you've made a good choice of model,
> the information you're discarding will be about the noise; with a bad
> model, you'll be discarding information about the signal. Choosing the
> right model for your data is therefore a key step in this process.

>

> The fundamental problem with using splines is that the number of data
> points is the same as the number of free parameters - you aren't
> discarding any noise.

In the long run, that's obviously the correct thing to do. But I was trying to see if I can come up with a quick way of getting an idea of what the differential distribution of a quantity looks like given its cumulative distribution, where the "noise" is really just coming from the fact that the distribution isn't sampled that well in the tail.

In other words, what I'm really looking for is a quick way of doing an adaptively-binned differential distribution. Which, now that I express it like that, is easier than screwing around with getting derivatives

out of a fit to the cumulative distribution, so I'll just go ahead and do that!

-Jeremy.
