Subject: Re: Compute area between curves
Posted by mystea on Wed, 15 Oct 2008 05:35:09 GMT
View Forum Message <> Reply to Message

On Oct 14, 4:26 am, James Kuyper <jameskuy...@verizon.net> wrote:
> mystea wrote:
>> Hi everyone,
>
>> I am also working on a topic where I need to numerically calculate an
>> integral
>> of a tabulated function. However, what I need is an indefinite
>> integral, namely,
>> the area under a curve as a function of x-coordinate.
>
> You can't calculate the true indefinite integral using numerical
> methods; that's something that can only be done by using a symbolic math
> program like Mathematica.
>
>> The procedure int_tabulated only calculates the definite integral,
>> given tabulated
>> f and its x-coordinates x. Let's say both f and x are double array of
>> length nl.
>
>> I tried the following fix:
>
>> integral=dblarr(nl)
>> for i=1, nl-1 do integral[i]=int_tabulated(x[0:i],f[0:i])
>
> What you're getting by this method is not the indefinite integral, but a
> tabulation of definite integrals. This can represent the indefinite
> integral, in much the same sense that your x and f arrays represent the
> function you want to integrate, but it is not the indefinite integral
> itself.
>
>> I thought it will work but not quite! Turns out that in general, the
>> result
>> integral will not be monotone even if f are always positive.
>
> That should not be the case for the true integral of a function that is
> always positive, assuming that the x values are sorted.
>
> However, numerical integration always produces no better than an
> approximation. INT_TABULATED uses a " a five-point Newton-Cotes
> integration formula", which is basically derived from fitting those five
> points to a polynomial. The best-fit polynomial could go to negative
> values within the range of integration, even if all of the data it is
> being fitted to is positive; in that case, the integral could decrease
> with increasing x, for some values of x. That seems unlikely, however,

> if your function is tabulated with sufficient detail.
>
> Could you give a simple example that demonstrates the problem you've seen?

I tried "tsum" and suddenly every problem was solved!

I found that the warning in the int_tabulated help file must be taken very seriously:

Warning:
Data that is highly oscillatory requires a sufficient number of samples for an accurate integral approximation.

My data was not oscillatory. However, I tried to find its first derivative using "deriv" and
found that its first derivatives are oscillatory.

So the motto is: thou shalt not use int_tabulated when the result from deriv is oscillatory. huh?
