Subject: Re: Compute area between curves
Posted by jameskuyper on Fri, 17 Oct 2008 19:43:56 GMT
View Forum Message <> Reply to Message

mystea wrote:
> Hi James,
>
> I would like to thank you for your reply. It is truly helpful.
>
> Here is an example:
>
> x=dindgen(200)+1
> y=x^(-2)

For this function, the second derivative f2 = 6*x^(-4). The sixth derivative f6 = 5040*x^-8. Within the range of integration, the maximum values of f2 and d6 both occur at x =1, so f2_max = 6 and f6_max=5040. Therefore, the maximum step size for which INT_TABULATED gives better results than TSUM is

    (315*f2_max/(32*f6_max))^0.25 = 0.329

Your data is tabulated with a spacing of 1.0, which is more than 3 times too large to get better results with INT_TABULATED than with TSUM. Try the following:

    x = 0.25*dindgen(800)+1.0

You get much nicer results that way.

> q=dblarr(200)
> for i=1,199 do q[i]=int_tabulated(x[0:i],y[0:i])
>
> plot,q
> plot,y
> plot,deriv(y)
>
> plot,sort(q)
>
> x2=x
> x2[60:199]=x2[60]+x2[60]*dindgen(140)
> y2=x2^(-2)

For the second portion of your data, the spacing is 61 units. The maximum values of f2 and f6 occur in that region occur at x = 61:

    f2_max = 6*x[60]^(-4)
    f6_max = 5040*x[60]^(-8)

For this portion of the data, the minimum step size needed is

```
(315*f2_max/(32*f6_max))^0.25 = 20.07
```

Once again, the spacing of your tabulated data is more than 3 times too large for INT_TABULATED to give you better results than TSUM. The moral of this story is that, if you can, you should sample your data more closely. If you can't do that, you'll have to integrate it using a more robust integration method, such as TSUM.

```
> q2=dblarr(200)
> for i=1,199 do q2[i]=int_tabulated(x2[0:i],y2[0:i])
> plot,q2
> plot,deriv(y2)
```

It would be more meaningful to use

```
plot, x2, q2
plot, x2, deriv(x2,y2)
```

However, give the range of values involved, you won't actually be able to see anything on that second plot. For a more readable plot, use

```
plot, x2, -deriv(x2,y2), /xlog,/ylog
```

For comparison, the analytical result is:

```
oplot, x2, 2*x2^(-3)
```

Which seems to be a pretty accurate match.

...
> As you can see, the results of integration is wiggling here.
> It's not necessary that the polynomial goes negative, its
> probably just because the routine is using different polynomials when
> new terms are introduced.

No, the issue is that the best-fit polynomial for under-sampled data isn't a particularly good fit. That polynomial has wiggles that don't match the shape of the function actually being tabulated.

> I also figured out that the deriv routine went crazy under some
> circumstance, too.
> In these cases, deriv(y) should go to zero, yet it was oscillating
> crazy.

I didn't see any oscillations in deriv(y), or deriv(x2,y2). Could you explain that more precisely?
