
Subject: Re: convolution of vsini
Posted by [wlandsman](#) on Wed, 23 Jan 2013 16:49:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

The IDL astronomy library has the function LSF_ROTATE

http://idlastro.gsfc.nasa.gov/ftp/pro/astro/lsf_rotate.pro

to return a convolution kernel for rotating a stellar spectrum under typical assumptions (e.g. constant limb darkening). You can use one of the many interpolation routines (e.g. `interp()`) to do the interpolation.

On Wednesday, January 23, 2013 7:10:12 AM UTC-5, idlhelp wrote:

> Does anyone know the IDL routine with the help of which I can first do the interpolation onto a wavelength grid and then get the kernel for vsini and then finally interpolate back onto my original wavelength grid.

>

>

>

> thanks

Subject: Re: convolution of vsini
Posted by [abc](#) on Wed, 23 Jan 2013 20:20:52 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Wednesday, January 23, 2013 5:49:33 PM UTC+1, wlandsman wrote:

> The IDL astronomy library has the function LSF_ROTATE

>

>

>

> http://idlastro.gsfc.nasa.gov/ftp/pro/astro/lsf_rotate.pro

>

>

>

> to return a convolution kernel for rotating a stellar spectrum under typical assumptions (e.g. constant limb darkening). You can use one of the many interpolation routines (e.g. `interp()`) to do the interpolation.

>

>

>

> On Wednesday, January 23, 2013 7:10:12 AM UTC-5, idlhelp wrote:

>

>> Does anyone know the IDL routine with the help of which I can first do the interpolation onto a wavelength grid and then get the kernel for vsini and then finally interpolate back onto my original wavelength grid.

>

>>

```
>  
>>  
>  
>>  
>  
>> thanks
```

thanks,

I am using that library. I have checked the program with different vsini values. The only things which I didn't understand is there is a huge offset in the flux at vsini=0.0 and vsini=10 km/sec. I didn't figure it out where I am making mistake. And this is how I am performing the calculation

```
dlambda = vsini/c  
npoints = ALOG(lammax/lammin)/dlambda+1  
interlam = lammin * EXP(dlambda * (DINDGEN(npoints)-1))  
interflux = INTERPOL( flux, lam, interlam )
```

and then get the kernel as

```
lsf = lsf_rotate(delta_V, vsini)  
fold = CONVOL( interflux, lsf, /CENTER, /EDGE_TRUNCATE)  
flux_new=INTERPOL(fold,interlam,wave)
```
