
Subject: filling in missing values
Posted by [Laura](#) on Tue, 25 Aug 2009 12:34:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi, I'm new to IDL and working on data files that assign a particular value for where the data is not available. But I want to replace such values with the nearest valid value in the array. Is there an efficient way to do that instead of loop?

Thanks a lot!

Subject: Re: filling in missing values
Posted by [Craig Markwardt](#) on Tue, 25 Aug 2009 15:50:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Aug 25, 8:34 am, Laura <haixia...@gmail.com> wrote:
> Hi, I'm new to IDL and working on data files that assign a particular
> value for where the data is not available. But I want to replace such
> values with the nearest valid value in the array. Is there an
> efficient way to do that instead of loop?

Yes. If you can tolerate always replacing using the good value on the left, then this would do the trick.

```
wh = where(data EQ BAD_VALUE, ct)
while ct GT 0 do begin
  data[wh] = data[wh-1]
  wh = where(data EQ BAD_VALUE, ct)
endwhile
```

It keeps on replacing until no values are left.

But if you are trying to replace bad values with local good values, why not use the median function?

```
medval = median(data, 5) ;; Sliding 5-point median
wh = where(data EQ BAD_VALUE, ct)
if ct GT 0 then data[wh] = medval[wh]
```

There are ways to be smarter about that. For example the median includes the bad values, and one may want to filter them out first.

Craig

Subject: Re: filling in missing values

Posted by [Laura](#) on Wed, 26 Aug 2009 15:17:31 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Aug 25, 11:50 am, Craig Markwardt <craig.markwa...@gmail.com> wrote:

> On Aug 25, 8:34 am, Laura <haixia...@gmail.com> wrote:

>

>> Hi, I'm new to IDL and working on data files that assign a particular
>> value for where the data is not available. But I want to replace such
>> values with the nearest valid value in the array. Is there an
>> efficient way to do that instead of loop?

>

> Yes. If you can tolerate always replacing using the good value on the
> left, then this would do the trick.

>

```
> wh = where(data EQ BAD_VALUE, ct)
> while ct GT 0 do begin
>   data[wh] = data[wh-1]
>   wh = where(data EQ BAD_VALUE, ct)
> endwhile
```

>

> It keeps on replacing until no values are left.

>

> But if you are trying to replace bad values with local good values,
> why not use the median function?

>

```
> medval = median(data, 5) ;; Sliding 5-point median
> wh = where(data EQ BAD_VALUE, ct)
> if ct GT 0 then data[wh] = medval[wh]
```

>

> There are ways to be smarter about that. For example the median
> includes the bad values, and one may want to filter them out first.

>

> Craig

Thanks a lot! I like the median one better!

--Laura
