Subject: Re: Can this be vectorized? Posted by davis on Tue, 26 Oct 1999 07:00:00 GMT

View Forum Message <> Reply to Message

On 26 Oct 1999 09:08:26 GMT, Struan Gray <struan.gray@sljus.lu.se> wrote:

- > In IDL an alternative to vectorisation is to use the fast built-in
- > function HISTOGRAM. In your case you should take the histogram of 'I'
- > and use the REVERSE_INDICES keyword to get a list of which elements
- > are in which bin. Summing the same elements of 'X' will give you the
- > answer you want. This will work even if 'I' and 'X' are not in
- > ascending order. There is some memory overhead, but it is of the same
- > order as creating working copies of your original data.

Let's suppose that the integer array 'I' is:

```
I = [0, 1, 1, 2, 3, 4, 4, 4, 5]

and `X' = [a, b, c, d, e, f, g, h, i].

Then, I want `Y' to be:

Y = [a, (b+c), d, e, (f+g+h), i]

The histogram `H' is:

H = [1, 2, 1, 1, 3, 1]
```

and the REVERSE_INDICES array `R' is:

$$R = [7, 8, 10, 11, 12, 15, 16, 0, 1, 2, 3, 4, 5, 6, 7, 8]$$

It seems to me that I would have to loop over all terms in the histogram via (pseudocode!):

```
\begin{split} H &= H[\text{where}(H != 0)] \qquad \text{\# get rid of non-zero elements} \\ j &= 0 \\ \text{for } i = 0 \text{ to } i = \text{length}(H) \\ &\quad \text{if } R[i] != R[i+1] \\ &\quad J = [ \ R[i] : R[i+1] - 1 \ ] \\ &\quad Y[j] = \text{sum } (X[J]) \\ &\quad \text{endif} \\ \text{next } i \end{split}
```

Of course this is useful as long as `l' contains many repeated elements so that the histogram will be small. Unfortunately, in my case this is unlikely.

At least I have a better understanding of the REVERSE_INDICES keyword. Thanks, --John