Subject: newbie sees iterative solution, wants to know the IDL Way
Posted by Tom Roche on Mon, 21 Jul 2008 04:28:49 GMT
View Forum Message <> Reply to Message

<best viewed in monospace font/>

From what I read, iteration is not the way to go with IDL. So I'm wondering, how to translate the following iterative solution:

I'm reading netCDF data with dimensions={time, lat, lon}, from which I want to get the "seasonality" such that, for each spacetime, I get the proportion of the total over time at that point in space. E.g. if I had input data for 4 space points (j,k) @ 2 time points (i):

```
(k=0) j=0 j=1
  i=0: 0   1
  i=1: 2   3

(k=1)
  i=0: 4   5
  i=1: 6   7
```

it would map to

```
      0    1/4
      1    3/4

      2/5  5/12
      3/5  7/12
```

My coding background is such that I quickly see an iterative solution like

```
  nTime=size of time dimension;
  nLat=size of latitude dimension;
  nLon=size of longitude dimension;

  create array summing[nLat][nLon];
  for (int i = 0; i < nTime; i++)
    for (int j = 0; j < nLat; j++)
      for (int k = 0; k < nLon; k++)
        // gotta handle input=NaN
        if (isNumber(input[i][j][k]))
          summing[j][k] += input[i][j][k];

  create array seasonality[nTime][nLat][nLon];
  for (int i = 0; i < nTime; i++)
    for (int j = 0; j < nLat; j++)
```

```
    for (int k = 0; k < nLon; k++)
      // gotta handle input=NaN, summing={0, NaN}
      if (isNumber(input[i][j][k]) &&
          isNumber(summing[j][k]) &&
          (input[i][j][k] != 0))
        seasonality[i][j][k] = input[i][j][k] / summing[j][k];
```

Is there a better, IDLer way?

TIA, Tom Roche <Tom_Roche@pobox.com>
