

Re: linalg[leastsqrs] in Maple V R4

Source: <http://sci.tech--archive.net/Archive/sci.math.symbolic/2006-08/msg00064.html>

- *From:* "lcw1964" <leslie.wright@xxxxxxxxxxxxxxx>
 - *Date:* 30 Jul 2006 19:08:33 -0700
-

I have decided to write my own leastsqrs routine using linalg(Svd) which seems to be a bit more consistent. This is how I have tested it out and have convinced myself it is adequate for my own purposes.

I execute `dvec:=evalf(linalg[Svd](u,Ubig,V))`. This returns a vector of the singular values, all of which are nonzero and not too small. To execute the least squares estimate as described well in section 2.6 of Numerical Recipes, I create the orthonormal "left" matrix `U:=evalm(delcols(Ubig, 10..72))`, and the diagonal matrix of inverses of the singular values by `isingmat:=diag(seq(1/dvec[i],i=1..9))`. The right matrix `V` is just what I need and can use it without modification. given this, the least squares solution is

```
sol:=evalm(V&*isingmat&*transpose(U));
```

I get the exact same result every time when I run everything after a restart, and the consistency is reassuring. The results are pretty close, at least in the first few digits, to the linalg[leastsqrs] output, though the consistency of the latter is not reassuring.

As a test, I attempt to reconstruct the original matrix `u` from my results, and compute its difference from the original:

```
evalm(U&*diag(seq(dvec[i],i=1..9))&*transpose(V);  
evalm("-u);
```

The output here reveals differences indicating that at most 2 or 3 SDs were lost. With `Digits:=60`, for example, this 72x9 matrix of differences has elements ranging between about $1E-57$ down to zero. For the stuff I am doing, this is tolerable and understandable.

I have no easy way of proving that the least squares result generated by this approach is "truer" or "better" than that given by the built-in leastsqrs routine—according to the theory it is supposed to be the "right" answer—but I do seem to think that it is at least more

Re: linalg[leastqsrs] in Maple V R4

consistent, and I will infer more accuracy and precision since the matrices and vectors generated by the Svd process that are prerequisites for the least squares calculation seem to maintain reasonable precision, as one can infer from the fairly close agreement when one reconstructs the original matrix from its decomposed parts.

This is an intermediate step in a larger algorithm