Solving a System of Equations

Sometimes you may need to solve a system of simultaneous linear equations, as you probably learned in high school. It is easy to do if there are only a few variables, but if you are using Excel, the process can be slow with many variables, and you cannot do more than about 250 variables.

I was needing to solve equations with over 300 variables while fitting age, period and cohort functions to a matrix of mortality improvement rates. To do so I made a function in C which runs about 70x faster than Excel matrix methods and is limited only by available memory.

VBA declare for function:

Declare PtrSafe Function SolvEq Lib “WHGrad64.dll”[[1]](#footnote-1) (ByVal n as Long, mat As Double, vec As Double, sol As Double) As Double

where

n is the number of equations and variables; they must be the same.

mat is the coefficient matrix for the equations as a vector in row-major[[2]](#footnote-2) order; there must be n2 entries.

vec is the values for each equation; there must be n entries.

sol is the solution vector; the vector must be set up with n entries.

The return value is the maximum absolute difference between the expected and actual values, and it should be very close to zero. If the return value is negative, there is a memory error or the coefficient matrix is singular.

The function would be called using code such as:

d = SolvEq(n, mat(1), vec(1), sol(1))

To see what else you can do with WHGrad64.dll go to <http://www.howardfamily.ca/graduation>.

1. Use just the name of the dll if it is placed in a directory in your Path. Otherwise, the name must be fully qualified. For example, “C:\DownloadedTools\WHGrad64.dll”. [↑](#footnote-ref-1)
2. Note that VBA represents matrices in column-major order. Therefore, you will need to write the entries by loops, as suggested by the following code, where coeff is the coefficient matrix and mat is the input to the function.

Dim coeff(n, n) as double, mat(n \* n) as double, i as long, j as long, n as long

For i = 1 to n

 For j = 1 to n

 mat(n \* (i – 1) + j) = coeff(i, j)

 Next j

Next i [↑](#footnote-ref-2)