Multivariable Optimization for the HP-67

by

Namir Shammas

This article presents an HP-67 program that performs multivariable optimization. The program calculates the minimum of a multivariable by performing sequential optimization on each variable, using Newton's method.

Usage

A Program prompts you to enter N (up to 5 variables), tolerance for the optimization for each variable, the tolerance for the norm (calculated for the variables), and the initial guesses for the optimum values of the variables.

Program calculates and displays the value of the minimized function and the values for the variables at the minimum.

Example

Consider the following function:

$f(X) = (X(1) - 2)^2 + (X(2) - 3)^2 + (X(3) + 1)^2$

The following table lists the input values:

Input Variable	Value
Number of variables	3
Tolerance for optimizing each variable	1e-5
Tolerance for the norms	1e-5
Initial value for X(1)	1
Initial value for X(2)	1
Initial value for X(3)	1

Using the above data, calculate the optimum function value and its coordinates. The Steps involved are:

Step	Task	Command/Input	Output
1	Switch to program mode and write the code for the targeted function under the label 08.	[GTO}[0][8] [PRGM] … [RTN] [PRGM]	
2	Switch to program mode and write the code to calculate the norm under the label 01.	[GTO}[0][1] [PRGM] … [RTN] [PRGM]	
3	Start the program.	[A]	(not relevant)
4	Enter the tolerance used to optimize each variable.	1 [EEX] [CHS] 5 [R/S]	(not relevant)
5	Enter the norm tolerance.	1 [EEX] [CHS] 5 [R/S]	(not relevant)
6	Enter the number of points.	3 [R/S]	1.00000
7	Enter the initial value for variable X(1).	1 [R/S]	2.0000
8	Enter the initial value for variable X(2).	1 [R/S]	3.0000
9	Enter the initial value for variable X(3).	1 [R/S]	(after showing intermediate values) 0.00000

Step	Task	Command/Input	Output
10	View the optimum value for variable X(1)	[R/S]	2.00000
11	View the optimum value for variable X(2)	[R/S]	3.00000
12	View the optimum value for variable X(3)	[R/S]	1.00000

Note that labels 01 and 08 in the program listings are geared to solve the above example.

Algorithm

```
Input N, X(), Toler, NormToler

Calculate Norm2

Do

  For I =1 to N

    Do
      Xt = X(I)
      h = 0.01 *(1 + |Xt|)
      F0 = F(X)
      X(I) = Xt + h
      Fp = F(X)
      X(I) = Xt - h
      Fm = F(X)
      X(I) = Xt
      Diff = (Fp - Fm) / (Fp - 2 * F0 + Fm) * h / 2
      X(I) = X(I) - Diff
    Loop Until |Diff| <= Toler

  Next I

  Norm1 = Norm2
  Calculate Norm2

Loop Until |(Norm2 - Norm1)| <= NormToler

Return X() and F(X)
```

Memory Map

R0 = N
R1 = Xt
R2 = h
R3 = F0
R4 = Fp
R5 = Fm
R6 = Toler
R7 = NormToler
R8 = Norm1
R9 = Norm2
RA = X(1)
RB = X(2)
RC = X(3)
RD = X(4)
RE = X(5)

Label Map

LBL A : Start program
LBL 0 : calculate index for accessing array X()
LBL 1 : calculate the norm
LBL 2 : I = I + 1 and recall N
LBL 3 : start input loop
LBL 4 : start main outer loop
LBL 5 : start of "For" loop
LBL 6 : start innermost loop
LBL 7 : start loop to display results
LBL 8 : function to be minimized
LBL 9 :

Source Code

The source code for the HP-67 program appears below. Please note the following:

- The blank lines are intentionally inserted to separate logical blocks of commands:

Program Step	*Comment*
*LBL A	
STO 0	store the number of variables
R↓	
STO 6	
R↓	

Program Step	*Comment*
STO 7	
GSB 0	
*LBL 3	start input loop
RC I	
19	
–	show the value for I to prompt for X(I)
R/S	
STO (i)	
GSB 2	I = I + 1 and recall N
X<=Y?	
GTO 3	end of input loop
GSB 1	
STO 9	Calculate Norm2
*LBL 4	---------- start main outer loop
GSB 0	initialize For loop variable
*LBL 5	start of "For" loop
*LBL 6	start innermost loop
RCL (i)	
STO 1	Xt = X(I)
PAUSE	display X(I)
ABS	
1	
+	
EEX 2	
/	
STO 2	h = 0.01 * (1 + |X(I)|)
GSB 8	
STO 3	F0 = F(X)
RCL 1	
RCL 2	
+	
STO (i)	X(I) = Xt + h

Program Step	*Comment*
GSB 8	
STO 4	Fp = F(X)
RCL 1	
RCL 2	
–	
STO (i)	X(I) = Xt - h
GSB 8	
STO 5	Fm = F(X)
RCL 1	
STO (i)	X(I) = Xt
RCL 4	
RCL 5	
–	
RCL 4	
RCL 5	
+	
RCL 3	
2	
*	
-	
/	
RCL 2	
*	
2	
/	Calculate Diff
STO– (i)	
ABS	
RCL 6	
X<=Y?	Toler <= |Diff|
GTO 6	end of inner most loop
RCL (I)	
–x–	display optimized X(I)
GSB 2	I = I + 1 and recall N
X<=Y?	

Program Step	*Comment*
GTO 5	end of For loop
RCL 8	
STO 9	Norm2 = Norm1
GSB 1	
STO 8	Calculate Norm1
RCL 9	
–	
ABS	
RCL 7	
X<=Y?	NormToler <= |Norm1 - Norm2|
GTO 4	---------- end of main outer loop
GSB 8	calculate f(X)
R/S	display optimum function value
GSB 0	
*LBL 7	start loop to display results
RCL (I)	
R/S	
GSB 2	I = I + 1 and recall N
X<=Y?	
GTO 7	
RTN	
*LBL 0	calculate index for accessing array X()
20	
ST I	
RTN	
*LBL 2	I = I + 1 and recall N
RC I	
1	
+	
ST I	
19	
–	

Program Step	*Comment*
RCL 0	
X<>Y	
RTN	
*LBL 1	calculate the norm
RCL A	
X^2	
RCL B	
X^2	
+	
RCL C	
X^2	
+	
SQRT	
RTN	
*LBL 8	function to be minimized
RCL A	
2	
–	
X^2	
RCL B	
3	
–	
X^2	
+	
RCL C	
1	
+	
X^2	
+	
RTN	