# Calculate Z

/\* Variables \*/

Var beta0, beta1, beta2, beta3; //---> Historic data of user

Var wk, xk, yk, zk; //---> User new program input

Var zk = (beta0) + (wk\*beta1) + (xk\*beta1) + (yk\*beta2) + (zk\*beta3);

Return zk;

# Deviation2 -> result()

Var wk, xk, yk;

HisotircData data;

Gauss betas;

Var N = data.size;

Var sum = 0;

Var firstElement = (1/(N - 4));

Var w = data.getLetterList(‘w’);

Var x = data.getLetterList(‘x’);

Var y = data.getLetterList(‘y’);

Var z = data.getLetterList(‘z’);

For i=0 -> N-1

Sum += z[i] - betas.getBeta(0) - ( betas.getBeta(1)\*w[i] ) - ( betas.getBeta(2)\*x[i] ) - ( betas.getBeta(3)\*y[i] );

Return sqrt( firstElement \* sum );

# Range -> result()

SearchXValue x;

Deviation2 sigma;

HIstoricData data;

Gauss gauss;

Var result

Var wk

Var xk

Var yk

Var N = data.size;

For i to N

//The same for each letter

wSub = pow(data.getLetterList(‘w’) - data.getAvg(‘w’), 2)

Var part3 = 1 + (1/N) +( pow(wk-data.getAvg(‘w’),2) /wSub ) + x... + y...;

Return x.Search(0.35, N-4) \* sigma.result() \* sqrt( part3 ) ;

# Gauss -> betas

|  |
| --- |
| 1. I = 1  2. LET P=AK , I = max{|AJ,I| : I <= J <= N} (Find the pivot value which is the largest number in the column below the I,I position.)  3. IF P = 0 THEN EXIT (If the remainder of this column is zero, a unique solution does not exist.)  4. IF K > I THEN FOR J = I TO N + 1 SWAP A(I,J) AND A(K,J) (Get the pivot value into the I,I position.)  5. FOR J = I + 1 TO N + 1 LET AI,J = AI,J / AI,I (Get a one in the I,I position.) SET AI,I = 1.  6. FOR L = I + 1 to N multiply row I by -AL,I and add to row L (Zero out the column below the I,I position.)  7. I = I + 1. IF I < N THEN GO TO 2.  8. EXIT to back-substitution |

W[ 1142, 863, 1065, 554, 983, 256 ]

X[ 1060, 995, 3205, 120, 2896, 485 ]

Y[ 325, 98, 23, 0, 120, 88 ]

Z[ 201, 98, 162, 54, 138, 61 ]