|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| /\* Strategy:  \* 1. Calculate X with the formula value  \* 2. Calculate P for that x with a dof = n-2  \* 3. Now we can return the calculation of the tailArea as 1\*2\*P  \*/  Double tailArea( r, arrayList )  //Step 1  Var numeratorX = abs(r) \* sqrt(n-2);  Var denominatorX = sqrt(1-r\*r);  Var x = numeratorX / denominatorX;  //Step 2  Var valueOfP = CalculationP(x, 0.00001,n-2).calculate();  //step3  Var tailArea = 1-2\*valueOfP;  Return tailArea; | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | Step | input | numeratorX | denominatorX | x | valueOfP | tailArea | Output | | 1 | R = 0,5  List =[1,2,3] | 0,5\*1=0,5 | - | - | - | - | - | | 1.1 | R =0,5 | - | 0,86602 | - | - | - | - | | 1.2 | Num =0,50  Den = 0,86 | - | - | 0,58 | - | - | - | | 2 | X = 0,58  E = 0,00001  Dof = 1 | - | - | - | 0.16730265756863133 | - | - | | 3 | P = 0.16730265756863133 | - | - | - | - | .6653946848627373 | .6653946848627373 | |