"ESSENTIAL ALGORITHMS TO STANDARD DEVIATION"

"AN ALGORITHM MUST BE SEEN TO BE BELIEVED" *By Donald Knuth*

Basant Gera (40082433)[*]
Dept. of Computer Science & Software Engineering
Concordia University
basantgera29@gmail.com

Submitted to : Pankaj Kamthan* & Team*
Dept. of Computer Science & Software Engineering
Concordia University
kamthan@cse.concordia.ca

Basant Gera (40082433) Problem Solution : 3 Function : F10

1 Conventional Algorithm for standard deviation

$functionStandardDevatation()$
$Sum \leftarrow 0.0$
$Mean \leftarrow 0.0$
$Var \leftarrow 0.0$
$SD \leftarrow 0.0$
$Number \leftarrow 0.0$
$iLength \leftarrow 0$ $myArray$ is an array of n doubles.
$iLength$=$myArray$.count()
$Sum \leftarrow Sum + Sum$
for iteration bounds sum+=$myArray$
end for $SD = squareRoot(SD/iLength)$
return SD

$functionsquareRoot()$
$t \leftarrow 0.0$
$squareRoot = number/2$
do
$t = squareRoot$
$squareRoot = (t + (number/t))/2$
$while$
$((t - squareRoot)! = 0)$
$return t$

2 Multi pass or n-pass Algorithm for Standard deviation

$functionStandardDevatation()$
$a \leftarrow 0.0$
$n \leftarrow 0$
$Sum \leftarrow 0.0$
$sq_s m \leftarrow 0.0$
$Mean \leftarrow 0.0$
$Var \leftarrow 0.0$
$SD \leftarrow 0.0$
$If n = 0 return 0.0$
$for iteration bounds$ sum+=$myArray$

sq_sum += $myArray * myArray$
$endfor$
$double mean = sum/n$
$variance = sq_sum/n - mean * mean$
$return sqrt(variance)$

$function squareRoot()$
$t \leftarrow 0.0$
$squareRoot = number/2$
do
$t = squareRoot$
$squareRoot = (t + (number/t))/2$
$while$
$((t - squareRoot)! = 0)$
$return t$

3 Multi pass or n-pass algorithm for standard deviation VS conventional algorithm for standard deviation

- Multi pass or n-pass algorithm for standard deviation: Can take more inputs of data in collection in form of file and can be faster and much more efficient to calculate standard deviation.Apart from that have issue is that memory stack will get full since it uses recursion over iteration.

- Conventional algorithm for standard deviation:Conventional approach is easy to understand but takes time to calculate standard deviation if frequency of number are high and input data is large.Apart from that it is easy to error chances are more. Therefore, Multi pass or n-pass algorithm is faster for standard deviation and much more efficient than Conventional algorithm for standard deviation

4 Advantages of Multi pass or n-pass algorithm for standard deviation

- Faster and much more efficient to use.
- Can take large set of data to solve standard devation problem.

5 Disadvantage of Multi pass or n-pass algorithm for standard deviation

- Most of the time memory stack get full and that's why the use of internal garbage collector start working which delete the unused memory which get fill in stack
- Difficult to develop in any language.

References

[1] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1255808/

[2] https://www.robertniles.com/stats/stdev.shtml

[3] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3487226/