|  |  |  |  |
| --- | --- | --- | --- |
| **STEP** | **RESULT** | | |
| ***ValidateUserInput****()* |  | | |
| Read userInput | 1 | | |
| While userInput is valid |  | | |
| Call ***ProcessUserInput()*** | File = CorrelationAndRegressionInputData  EstimatedProxySize = 386 | | |
| X = 0 first column in the file  Xi | | Y = 2 third column in the file  Yi |
| 130  650  99  150  128  302  95  945  368  961 | | 186  699  132  272  291  331  199  1890  788  1601 |
| Else Return |  | | |
|  |  | | |
| ***ProcessUserInput****()* |  | | |
| Case IserInputof |  | | |
| UserInput = 1 : filepath = predefined test programs data file; estimatedProxySize = 386; X(ProgramsEstimatedProxySize)= 0; Y(ActualAddedAndModifiedSize) = 2; | Entered this case – see previous step | | |
| UserInput = 2: filepath = predefined test programs data file; estimatedProxySize = 386; X(ProgramsEstimatedProxySize)= 0; Y(ActualDevelopmentTime) = 3; |  | | |
| UserInput = 3: filepath = file with data from student workbook for Programs 2 to 6; estimatedProxySize = from student workbook; X(ProgramsEstimatedProxySize)= 0; Y(ActualAddedAndModifiedSize) = 1; |  | | |
| UserInput = 4: filepath = file with data from student workbook for Programs 2 to 6; estimatedProxySize = from student workbook; X(ProgramsEstimatedProxySize)= 0; Y(ActualDevelopmentTime) = 2; |  | | |
|  |  | | |
| ***ReadFile()*** |  | | |
| ***Read each line in file*** | eachLineInFile : 130,163,186,15  650,765,699,69.9  99,141,132,6.5  150,166,272,22.4  128,137,291,28.4  302,355,331,65.9  95,136,199,19.4  945,1206,1890,198.7  368,433,788,38.8  961,1130,1601,138.2 | | |
| Split by comma |  | | |
| Add values in each row to an array | I = 0 | numbersInRow[0] = 130  numbersInRow[1] = 163  numbersInRow[2] = 186  numbersInRow[3] = 15.0 | |
|  | I = 1 | numbersInRow[0] = 650  numbersInRow[1] = 765  numbersInRow[2] = 699  numbersInRow[3] = 69.9 | |
|  | I = 2 | numbersInRow[0] = 99  numbersInRow[1] = 141  numbersInRow[2] = 132  numbersInRow[3] = 6.5 | |
|  | I = 3 | numbersInRow[0] = 150  numbersInRow[1] = 166  numbersInRow[2] = 272  numbersInRow[3] = 22.4 | |
|  | I =4 | numbersInRow[0] = 128  numbersInRow[1] = 137  numbersInRow[2] = 291  numbersInRow[3] = 28.4 | |
|  | I = 5 | numbersInRow[0] = 302  numbersInRow[1] = 355  numbersInRow[2] = 331  numbersInRow[3] = 65.9 | |
|  | I = 6 | numberInRow[0]=95  numberInRow [1]=136  numberInRow [2]=199  numberInRow [3]=19.4 | |
|  | I = 7 | numbersInRow[0] = 945  numbersInRow[1] = 1206  numbersInRow[2] = 1890  numbersInRow[3] = 198.7 | |
|  | I = 8 | numbersInRow[0] = 368  numbersInRow[1] = 433  numbersInRow[2] = 788  numbersInRow[3] = 38.8 | |
|  | I = 9 | numbersInRow[0] = 961  numbersInRow[1] = 1130  numbersInRow[2] = 1601  numbersInRow[3] =138.2 | |
| Create linked list of corresponding indexes of the array so it contains values from a single column | |  | | --- | | ListOfRealNumbers.ElementAt<double[]>(0)[0]=130 | | ListOfRealNumbers.ElementAt<double[]>(1)[0]=650 | | ListOfRealNumbers.ElementAt<double[]>(2)[0]=99 | | ListOfRealNumbers.ElementAt<double[]>(3)[0]=150  ListOfRealNumbers.ElementAt<double[]>(4)[0]=128  ListOfRealNumbers.ElementAt<double[]>(5)[0]=302  ListOfRealNumbers.ElementAt<double[]>(6)[0]=95  ListOfRealNumbers.ElementAt<double[]>(7)[0]=945  ListOfRealNumbers.ElementAt<double[]>(8)[0]=368  ListOfRealNumbers.ElementAt<double[]>(9)[0]=961 | | | |
| |  | | --- | | ListOfRealNumbers.ElementAt<double[]>(0)[2]=186 | | ListOfRealNumbers.ElementAt<double[]>(1)[2]=699 | | ListOfRealNumbers.ElementAt<double[]>(2)[2]=132 | | ListOfRealNumbers.ElementAt<double[]>(3)[2]=272 | | ListOfRealNumbers.ElementAt<double[]>(4)[2]=291  ListOfRealNumbers.ElementAt<double[]>(5)[2]=331  ListOfRealNumbers.ElementAt<double[]>(6)[2]=199  ListOfRealNumbers.ElementAt<double[]>(7)[2]=1890  ListOfRealNumbers.ElementAt<double[]>(8)[2]=788  ListOfRealNumbers.ElementAt<double[]>(9)[2]=1601 | | | |
|  |  | | |
| ***CalculateRegressionParameters()*** |  | | |
| NumberOfSegments = LinkedList.Count //number of rows is a column | NumberOfSegments = 10 | | |
| Dof = NumberOfSegments – 2; //as required | Dof = 8 | | |
|  |  | | |
| regressionB1 = (sumOfXYProducts - productMeanXY) / (sumOfSquareX - productSquareMeanX) | (4303108.0 - 2445709.1999999997)/( 2540284.0- 1465358.4) = **1.7279** | | |
|  |  | | |
| regressionB0 = meanValueY - (**regressionB1** \* meanValueX); | 638.9 – (**1.7279\***382.8) **= -22.55253** | | |
|  |  | | |
| improvedPrediction = regressionB0 + regressionB1 \* EstimatedProxySize; | -22.55253 + 1.7279\*386 =**644.4294** | | |
|  |  | | |
| ***CalculateCorrelationParameters()*** |  | | |
|  |  | | |
| correlationR = (numberOfSegments \* sumOfXYProducts - sumOfX \* sumOfY) / Math.Sqrt((numberOfSegments \* sumOfSquareX - Math.Pow(sumOfX, 2)) \* (numberOfSegments \* sumOfSquareY - Math.Pow(sumOfY, 2))); | (10\*4303108.0-3828.0\*6389.0)/Math.Sqrt(10\*2540284.0-(3828.0)^2)\*(10\*7604693.0-(6389.0)^2) =**0.95449657410468258** | | |
| correlationR2 = Math.Pow(correlationR, 2); | 0.95449657410468258^2 **= 0.9110637099775758** | | |
|  |  | | |
| ***CalculateSignificance()*** |  | | |
|  |  | | |
| tInterval = (Math.Abs(CorrelationR) \* Math.Sqrt(NumberOfSegments - 2)) / (double)Math.Sqrt(1 - CorrelationR2); | |0.95449657410468258|\*Math.Sqrt(10-2)/Math.Sqrt(1-0.9110637099775758)=  9.052736157200183 | | |
| CalculateProbability(tInterval, NumberOfSegments, Dof); | NumberOfSegments = NumberOfSegments \* 8  Probability = 0.49999112413466984 | | |
| significance = 1 - 2 \* Probability; | **0.00001775173066032032 = 1.7752E-05** | | |
|  |  | | |
| ***CalculateNewTForRange()*** |  | | |
|  |  | | |
| Probability = 0.35; |  | | |
| Call FindValueOfT() with this Probability | tInterval **=** 1.1081457138061523 | | |
|  |  | | |
| ***CalculateStandardDeviation()*** |  | | |
| For each row in selected column from file |  | | |
| Sum += Math.Pow((Yi – regressionb0 - regressionb1\*Xi),2) | (186+22.5525-1.7279\*130)^2 +  (699+22.5525-1.7279\*650)^2 +  (132+22.5525-1.7279\*99)^2 +  (272+22.5525-1.7279\*150)^2 +  (291+22.5525-1.7279\*128)^2 +  (331+22.5525-1.7279\*302)^2 +  (199+22.5525-1.7279\*95)^2 +  (1890+22.5525-1.7279\*945)^2 +  (788+22.5525-1.7279\*368)^2 +  (1601+22.5525-1.7279\*961)^2 = 313301.28508205595 | | |
| sigma = Math.Sqrt((1 / NumberOfSegments – 2) \* sum); | Math.Sqrt(1/10-2)\* 313301.28508205595 = 197.89558013067648 | | |
|  |  | | |
| ***CalculateLastPartOfTheProduct()*** |  | | |
| For each row in selected column from file |  | | |
| SumBelow += Math.Pow((Xi – MeanX),2) | 130 - 382.79999999999995)^2 +  650- 382.79999999999995)^2 +  99 - 382.79999999999995)^2 +  150 - 382.79999999999995)^2 +  128 - 382.79999999999995)^2 +  302 - 382.79999999999995)^2 +  95 - 382.79999999999995)^2 +  945 - 382.79999999999995)^2 +  368 - 382.79999999999995)^2 +  961- 382.79999999999995)^2 = 1074925.6 | | |
|  |  | | |
| ThirdPart = Math.Sqrt(1+1/numberOfSegments + Math.Pow(estimatedProxySize – MeanX),2)/sumBelow) | Math.Sqrt(1+1/10 + (386-382.79999999999995)^2/1074925.6) = 1.0488133896173808 | | |
|  |  | | |
| ***CalculateFinalRange()*** |  | | |
|  |  | | |
| Range = tInterval\*Sigma\*thirdPart | 1.1081457138061523\*197.89558013067648\*1.0488133896173808 =  **230.0018** | | |
|  |  | | |
| ***CalculateRangeIntervals()*** |  | | |
|  |  | | |
| LPI = ImprovedPrediction - Range | 644.42938376386235 – 230.0018 = **414.4276** | | |
| UPI = ImprovedPrediction + Range | 644.42938376386235 + 230.0018 = **874.4311** | | |
|  |  | | |