KALASALINGAM
Academy of Research and Education
DEEMED TO BE UNIVERSITY
Estd.U/S 3 of UGC Act 1956. Accredited by NAAC with "A" Grade
Anand Nagar, Krishnankoil - 626126. Srivilliputtur (Via), Virudhunagar (Dt), Tamil Nadu | info@kalasalingam.ac.in | www.kalasalingam.ac.in

DEPARTMENT OF MATHEMATICS

COURSE NAME: STATISTICS FOR ENGINEERS

COURSE CODE : 211MAT1302

TOPIC : FIDING REGREESION,RANK CORRELATION AND CORRLEATION COEFFICIENT USING PYTHON

STUDENT DETAILS

NAME : ANNAM HEMANTH KUMAR

REGISTER NUMBER : 99220041106

DEPARTMENT : COMPUTER SCIENCE AND ENGINEERING

YEAR : II ND YEAR

SLOT : III

SUBMITTED TO : DR. A. SUMITRA

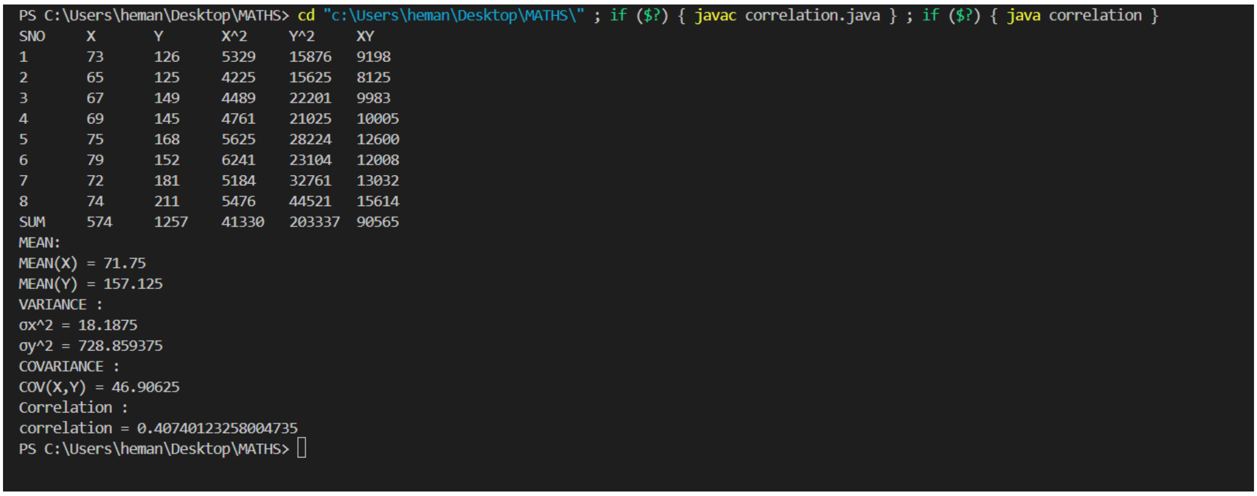
CORRELATION :

X	73	65	69	75	75	79	72	74
Y	126	125	149	145	168	152	181	211

#JAVA

```
public class correlation
{
    public static void main(String[] args)
    {
        int[] X = {73 , 65 , 67 , 69 , 75 , 79 , 72 , 74 };
        int[] Y = { 126 , 125 , 149 , 145 , 168 , 152 , 181, 211};

        int[] x_square = new int[X.length];
        int[] y_square = new int[Y.length];
        int[] xy = new int[X.length];
        int xsum =0, ysum =0, xysum =0, xsquaresum =0 , ysquaresum =0;

        for (int i = 0; i < X.length; i++)
        {

            x_square[i] = X[i]*X[i];
            y_square[i] = Y[i]*Y[i];
            xy[i] = X[i]*Y[i];

            xsum += X[i];
            ysum += Y[i];

            xsquaresum += x_square[i];
            ysquaresum += y_square[i];
            xysum += xy[i];
        }


        System.out.println("SNO\tX\tY\tX^2\tY^2\tXY");
        for (int i = 0; i < X.length; i++)
        {
            System.out.println((i+1) + "\t" + X[i] + "\t" + Y[i] + "\t" +
x_square[i] + "\t" + y_square[i] + "\t" + xy[i]);
        }

        System.out.println("SUM\t" + xsum + "\t" + ysum + "\t" + xsquaresum +
"\t" + ysquaresum + "\t" + xysum );


        System.out.println("MEAN:");
```

```java
        double meanx =(double) xsum/X.length ;
        double meany =(double) ysum/Y.length ;

        System.out.println( "MEAN(X) = " + meanx  );

        System.out.println("MEAN(Y) = " + meany );

        System.out.println("VARIANCE : ");

        double varx =  ( (double)xsquaresum/X.length) - meanx*meanx ;

        double vary = ((double)ysquaresum/Y.length) - meany*meany ;

        System.out.println("\u03c3x^2 = " + varx );
        System.out.println("\u03c3y^2 = " + vary );

        System.out.println("COVARIANCE : ");

        double cov =( (double) xysum/X.length) - meanx*meany ;

        System.out.println("COV(X,Y) = " + cov  );

        System.out.println("Correlation : ");

        double correlation = cov/  ((Math.pow(varx, 0.5)*Math.pow(vary, 0.5)
)) ;

        System.err.println("correlation = " + correlation );
    }
}
```



```
PS C:\Users\heman\Desktop\MATHS> cd "c:\Users\heman\Desktop\MATHS\" ; if ($?) { javac correlation.java } ; if ($?) { java correlation }
SNO     X       Y       X^2     Y^2     XY
1       73      126     5329    15876   9198
2       65      125     4225    15625   8125
3       67      149     4489    22201   9983
4       69      145     4761    21025   10005
5       75      168     5625    28224   12600
6       79      152     6241    23104   12008
7       72      181     5184    32761   13032
8       74      211     5476    44521   15614
SUM     574     1257    41330   203337  90565
MEAN:
MEAN(X) = 71.75
MEAN(Y) = 157.125
VARIANCE :
σx^2 = 18.1875
σy^2 = 728.859375
COVARIANCE :
COV(X,Y) = 46.90625
Correlation :
correlation = 0.40740123258004735
PS C:\Users\heman\Desktop\MATHS> 
```

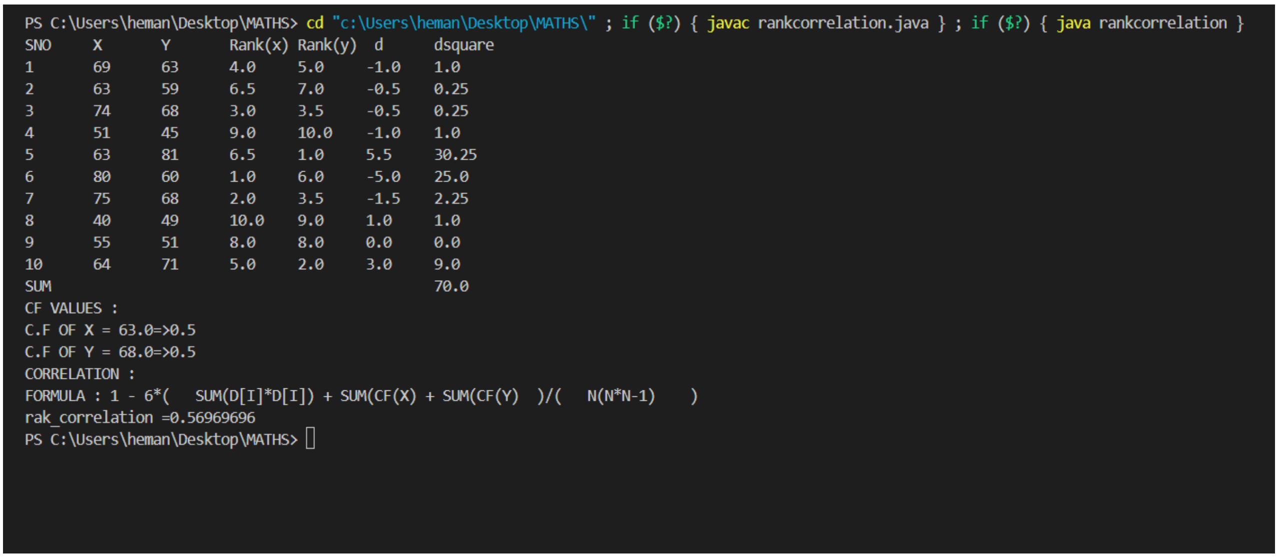
RANK CORRELATION :

X	69	63	74	51	63	80	75	40	55	64
Y	63	59	68	45	81	60	68	49	51	71

```
import java.util.ArrayList;
import java.util.Arrays;
import java.util.HashMap;
import java.util.HashSet;
import java.util.List;
import java.util.Map;
import java.util.Set;

class element
{
    int data;
    ArrayList<Integer> rankholder;
    int count;
    element(int data)
    {
        this.data = data;
        rankholder = new ArrayList<Integer>();
        count =0;
    }
    void addelement(int data, int rank)
    {
        rankholder.add(rank);
        count++;
    }
    float getrank()
    {
        int ranksum =0;
        for(int i=0; i<rankholder.size(); i++)
        {
            ranksum = ranksum + rankholder.get(i);
        }
        float rank = (float) ranksum/rankholder.size();
        return rank;
    }

    int getcount()
    {
        return count;
    }
}
```

```
public class rankcorrelation
{
    public static void main(String[] args)
    {
        int[] X = { 69 , 63 , 74 , 51 , 63 , 80 , 75 , 40  , 55 , 64};
        int[] Y = { 63 , 59 , 68 , 45 , 81 , 60 , 68 , 49 , 51 , 71  };

        ArrayList<Float> CF = new ArrayList<Float>();

        ArrayList<Float> CX = new ArrayList<Float>();
        ArrayList<Float> CY = new ArrayList<Float>();

        float[] xrank = rankcal(X , CF , CX);
        float[] yrank = rankcal(Y , CF , CY);


        float[] d = new float[X.length];

        float dsquare =0;

        for(int i=0; i<X.length; i++)
        {
            d[i] = xrank[i]-yrank[i];
            float t = d[i]*d[i];
            dsquare = dsquare + t;
        }

        int n = X.length;

        float rank_correlation = 1-( 6*(dsquare + CF.get(0) + CF.get(1)
)/(n*(n*n-1))  );


        System.out.println("SNO\tX\tY\tRank(x)\tRank(y)\t d\tdsquare");
        for(int i =0 ; i< X.length ; i++ )
        {
            System.out.println( (i+1)+"\t" +X[i] + "\t" + Y[i] +"\t" +
xrank[i] + "\t"  + yrank[i] +  "\t" + d[i] + "\t" + d[i]*d[i] );
        }
        System.out.println("SUM\t\t\t\t\t\t" + dsquare);

        System.out.println("CF VALUES :");

        for(int i =0 ; i< CX.size() ; i = i+2)
        {
            System.out.println( "C.F OF X = " +CX.get(i) + "=>" + CX.get(i+1)
);
        }
        for(int i =0 ; i< CY.size() ; i = i+2)
        {
            System.out.println( "C.F OF Y = " +CY.get(i) + "=>" + CY.get(i+1)
);
        }

        System.out.println("CORRELATION :");

        System.out.println("FORMULA : 1 - 6*(   SUM(D[I]*D[I]) + SUM(CF(X) +
SUM(CF(Y)  )/(   N(N*N-1)    ) ");

        System.out.println("rak_correlation =" + rank_correlation);

    }

    public static float[] rankcal(int[] b  , ArrayList<Float> CF ,
ArrayList<Float> CT )
    {
        int[] a = new int[b.length];

        for(int i=0; i<b.length; i++)
        {
            a[i] = b[i];
        }

        Arrays.sort(a ) ;
        for (int i = 0; i < a.length / 2; i++) {
            int temp = a[i];
            a[i] = a[a.length - 1 - i];
            a[a.length - 1 - i] = temp;
        }

        element[] arr= new element[a[0] + 1];

        int[] uniqueArray = removeDuplicates(a);


        for (int i = 0; i < uniqueArray.length ; i++)
        {
            arr[uniqueArray[i]] = new element(uniqueArray[i]);
        }

        for (int i = 0; i <a.length ; i++)
        {
            arr[a[i]].addelement(a[i] , i+1 );
        }
        float[] rank = new float[a.length];

        for (int i = 0; i < b.length; i++)
        {
            rank[i] = arr[b[i]].getrank();
        }

        float cf=0;

        int[] a_repeted = findRepeatedNumbers(b);

        for(int i =0 ; i< a_repeted.length ; i++)
        {
            CT.add((float) a_repeted[i]);

            int m = arr[a_repeted[i]].getcount();

            cf +=  (float)m*( (m*m) -1)/12;

            CT.add( (float)m*( (m*m) -1)/12);

        }

        CF.add(cf);

        return rank;
    }

    public static int[] removeDuplicates(int[] array) {
        // Create a HashSet to store unique elements
        Set<Integer> set = new HashSet<>();

        // Add elements from the array to the set (duplicates will
automatically be removed)
        for (int num : array) {
            set.add(num);
        }

        // Convert the set back to an array
        int[] uniqueArray = new int[set.size()];
        int index = 0;
        for (int num : set) {
            uniqueArray[index++] = num;
        }

        return uniqueArray;
    }
```

```java
    public static int[] findRepeatedNumbers(int[] array) {
        Map<Integer, Integer> countMap = new HashMap<>();
        List<Integer> repeatedNumbersList = new ArrayList<>();

        for (int num : array) {
            countMap.put(num, countMap.getOrDefault(num, 0) + 1);
        }

        for (Map.Entry<Integer, Integer> entry : countMap.entrySet()) {
            if (entry.getValue() > 1) {
                repeatedNumbersList.add(entry.getKey());
            }
        }

        int[] repeatedNumbersArray = new int[repeatedNumbersList.size()];
        for (int i = 0; i < repeatedNumbersList.size(); i++) {
            repeatedNumbersArray[i] = repeatedNumbersList.get(i);
        }

        return repeatedNumbersArray;
    }


}
```



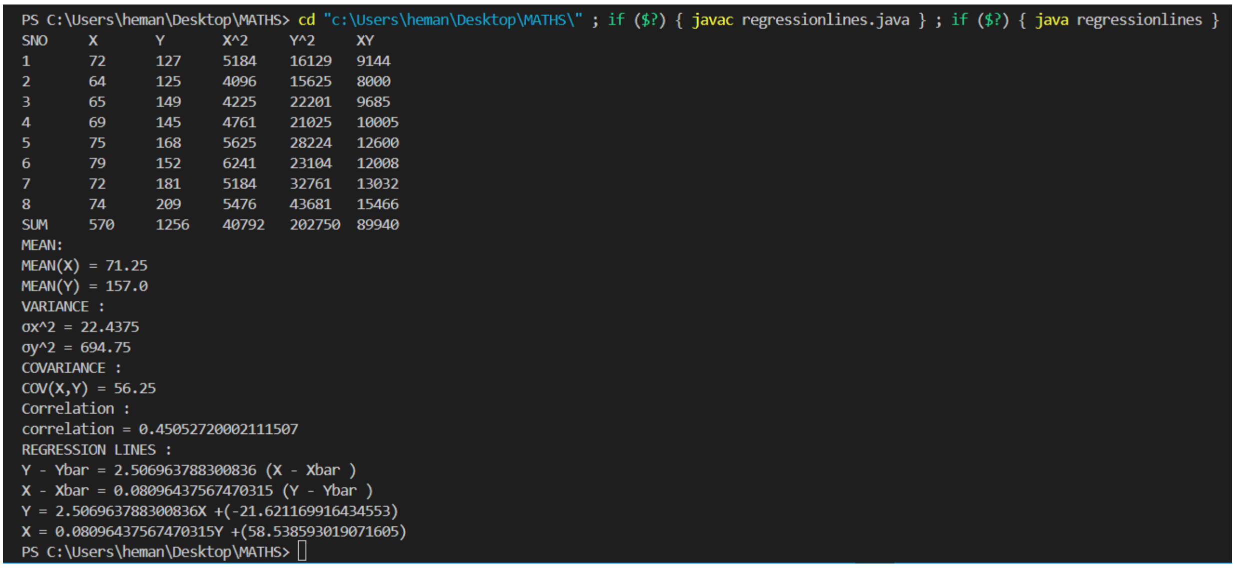
```
PS C:\Users\heman\Desktop\MATHS> cd "c:\Users\heman\Desktop\MATHS\" ; if ($?) { javac rankcorrelation.java } ; if ($?) { java rankcorrelation }
SNO     X       Y       Rank(x) Rank(y)  d      dsquare
1       69      63      4.0     5.0     -1.0    1.0
2       63      59      6.5     7.0     -0.5    0.25
3       74      68      3.0     3.5     -0.5    0.25
4       51      45      9.0     10.0    -1.0    1.0
5       63      81      6.5     1.0     5.5     30.25
6       80      60      1.0     6.0     -5.0    25.0
7       75      68      2.0     3.5     -1.5    2.25
8       40      49      10.0    9.0     1.0     1.0
9       55      51      8.0     8.0     0.0     0.0
10      64      71      5.0     2.0     3.0     9.0
SUM                                             70.0
CF VALUES :
C.F OF X = 63.0=>0.5
C.F OF Y = 68.0=>0.5
CORRELATION :
FORMULA : 1 - 6*(    SUM(D[I]*D[I]) + SUM(CF(X) + SUM(CF(Y)   )/(    N(N*N-1)     )
rak_correlation =0.56969696
PS C:\Users\heman\Desktop\MATHS> 
```

REGESSION LINES :

X	72	64	65	69	75	79	72	74
Y	127	125	149	145	168	152	181	209

```
public class regressionlines
{
    public static void main(String[] args)
    {
        int[] X = {72 , 64 , 65 , 69 , 75 , 79 , 72 , 74 };
        int[] Y = { 127 , 125 , 149 , 145 , 168 , 152 , 181, 209};

        int[] x_square = new int[X.length];
        int[] y_square = new int[Y.length];
        int[] xy = new int[X.length];
        int xsum =0, ysum =0, xysum =0, xsquaresum =0 , ysquaresum =0;

        for (int i = 0; i < X.length; i++)
        {

            x_square[i] = X[i]*X[i];
            y_square[i] = Y[i]*Y[i];
            xy[i] = X[i]*Y[i];

            xsum += X[i];
            ysum += Y[i];

            xsquaresum += x_square[i];
            ysquaresum += y_square[i];
            xysum += xy[i];
        }


        System.out.println("SNO\tX\tY\tX^2\tY^2\tXY");
        for (int i = 0; i < X.length; i++)
        {
            System.out.println((i+1) + "\t" + X[i] + "\t" + Y[i] + "\t" +
x_square[i] + "\t" + y_square[i] + "\t" + xy[i]);
        }

        System.out.println("SUM\t" + xsum + "\t" + ysum + "\t" + xsquaresum +
"\t" + ysquaresum + "\t" + xysum );


        System.out.println("MEAN:");
        double meanx =(double) xsum/X.length ;
        double meany =(double) ysum/Y.length ;

        System.out.println( "MEAN(X) = " + meanx  );

        System.out.println("MEAN(Y) = " + meany );

        System.out.println("VARIANCE : ");

        double varx =  ( (double)xsquaresum/X.length) - meanx*meanx ;

        double vary = ((double)ysquaresum/Y.length) - meany*meany ;

        System.out.println("\u03c3x^2 = " + varx );
        System.out.println("\u03c3y^2 = " + vary );

        System.out.println("COVARIANCE : ");

        double cov =( (double) xysum/X.length) - meanx*meany ;

        System.out.println("COV(X,Y) = " + cov  );

        System.out.println("Correlation : ");

        double correlation = cov/  ((Math.pow(varx, 0.5)*Math.pow(vary, 0.5)
)) ;

        System.err.println("correlation = " + correlation );

        System.out.println("REGRESSION LINES :" );

        double coefx = correlation*(Math.pow(vary, 0.5)/(Math.pow(varx,
0.5)));
        double coefy = correlation*(Math.pow(varx, 0.5)/(Math.pow(vary,
0.5)));

        System.out.println("Y - Ybar = " + coefx + " (X - Xbar )" );
        System.out.println("X - Xbar = " + coefy + " (Y - Ybar )" );

        System.out.println("Y = " +coefx+ "X +(" + (meany-coefx*meanx)  +")"
);
        System.out.println("X = " +coefy+ "Y +(" + (meanx-coefy*meany) + ")"
);



    }
```

```
}
```



```
PS C:\Users\heman\Desktop\MATHS> cd "c:\Users\heman\Desktop\MATHS\" ; if ($?) { javac regressionlines.java } ; if ($?) { java regressionlines }
SNO     X       Y       X^2     Y^2     XY
1       72      127     5184    16129   9144
2       64      125     4096    15625   8000
3       65      149     4225    22201   9685
4       69      145     4761    21025   10005
5       75      168     5625    28224   12600
6       79      152     6241    23104   12008
7       72      181     5184    32761   13032
8       74      209     5476    43681   15466
SUM     570     1256    40792   202750  89940
MEAN:
MEAN(X) = 71.25
MEAN(Y) = 157.0
VARIANCE :
σx^2 = 22.4375
σy^2 = 694.75
COVARIANCE :
COV(X,Y) = 56.25
Correlation :
correlation = 0.45052720002111507
REGRESSION LINES :
Y - Ybar = 2.506963788300836 (X - Xbar )
X - Xbar = 0.08096437567470315 (Y - Ybar )
Y = 2.506963788300836X +(-21.621169916434553)
X = 0.08096437567470315Y +(58.538593019071605)
PS C:\Users\heman\Desktop\MATHS> 
```