Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

**Лабораторна робота № 3**

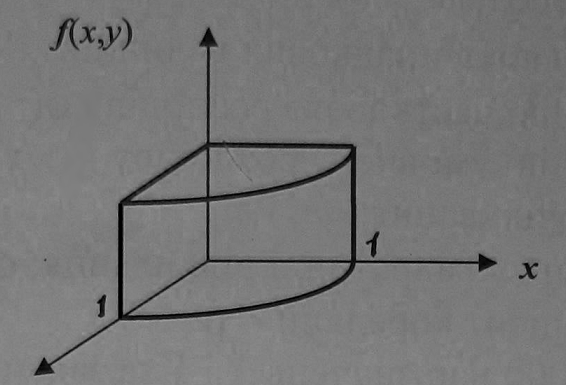
*Системи випадкових величин*

Виконав студент групи ІО-01

Редько Олександр

# Завдання

Задана функція розподілу двох випадкових величин.



Легко визначити, що .

# Аналітичні розрахунки

## Функція часткового розподілу

## Функція умовного розподілу

## Математичне очікування

## Середньоквадратичне відхилення

Тоді,

Аналогічно визначається :

Отже,

## Коефіцієнт кореляції

# Програмна генерація вибірки 1000 пар значень *X*і *Y*за заданою функцією

## Експериментальне визначення мат. очікування, середньоквадратичного відхилення, коефіцієнта кореляції

mean X = 0.4212182111179136

mean Y = 0.42249320873915647

standartdeviation X = 0.26581552590063506

standartdeviation Y = 0.2614796868170328

correlation = -0.3201958265604174

## Лістинг коду

**import**java.util.Random;

**publicclass**SystemsOfRandomNumbers {

/\*\*

\*величинавибірки

\*/

**privatestaticfinalint***NUMBERS* = 1000;

**publicstaticvoid**main(String[] args) {

**double**[] arrayX = **newdouble**[*NUMBERS*];

**double**[] arrayY = **newdouble**[*NUMBERS*];

*generateArrayX*(arrayX);

*generateArrayY*(arrayY, arrayX);

System.*out*.println("\n mean X = " + *mean*(arrayX));

System.*out*.println("\n mean Y = " + *mean*(arrayY));

System.*out*.println("\n standartdeviation X = " + *standartDeviation*(arrayX));

System.*out*.println("\n standartdeviation Y = " + *standartDeviation*(arrayY));

System.*out*.println("\n correlation = " + *correlation*(arrayX, arrayY));

}

/\*\*

\*Генераціячислаx.

\*Реалізуєтьсяякзнаходженнякоренярівнянняметодомполовинногоділення

\***@param**rчисловід0до1

\***@return**згенерованийxвід0до1

\*/

**publicstaticdouble**solveEquation (**double** r){

**double** a = 0, b = 1, x = a, xt, eps = 0.00000000001;

**do**{

xt = x;

x = (b + a)/2;

**if**(*FuncX*(x, r) \* *FuncX*(a, r) > 0)

a = x;

**else**

b = x;

}**while** (Math.*abs*(x - xt) >= eps);

**return** x;

}

/\*\*

\*Генераціямасивузx,розподіленихзазаданимфункцієющільності

\*вірогідності(четвертачастинациліндра)

\***@param**array

\*/

**publicstaticvoid**generateArrayX(**double** [] arrayX){

**double**[] arrayRandoms = **newdouble**[*NUMBERS*];

*generateRandom*(arrayRandoms);

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

arrayX[i] = *solveEquation*(arrayRandoms[i]);

}

}

**publicstaticvoid**generateArrayY(**double**[] arrayY, **double**[] arrayX){

**double**[] arrayRandoms = **newdouble**[*NUMBERS*];

*generateRandom*(arrayRandoms);

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

arrayY[i] = arrayRandoms[i] \* Math.*sqrt*(1 - arrayX[i] \* arrayX[i]);

}

}

/\*\*

\*Інтегралвідфункціїчастковогорозподілу

\***@param**xx

\***@param**rвипадковечисловід0до1

\***@return**

\*/

**publicstaticdouble**FuncX(**double** x, **double** r){

**return** 4 \* (Math.*asin*(x)/2 + x \* Math.*sqrt*(1 - x \* x)/2)/Math.*PI* - r;

}

/\*\*

\*Математичнеочікування.

\*Реалізуєтьсяяксереднєзначеннявибірки

\***@param**arrayмасиввибірки

\***@return**averматематичнеочікуванняm

\*/

**publicstaticdouble**mean(**double**[] array){

**double**Sum = 0.0;

**double**aver = 0.0;

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

Sum += array[i];

}

aver = Sum/array.length;

**return**aver;

}

/\*\*

\*Середньоквадратичневідхиленнявипадковоївеличини

\***@param**array вибірка

\***@return**значеннясигма

\*/

**publicstaticdouble**standartDeviation(**double**[] array){

**double**disp = 0;

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

disp += Math.*pow*((array[i] - *mean*(array)), 2);

}

disp = disp/array.length;

**return**Math.*sqrt*(disp); //сигма - корінь із дисперсії

}

/\*\*

\*Коваріаціясистемдвохвеличин

\***@param**arrayXвибіркаX

\***@param**arrayYвибіркаY

\***@return**cov

\*/

**publicstaticdouble**covariance(**double**[] arrayX, **double**[] arrayY){

**double**cov = 0;

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

cov += (arrayX[i] - *mean*(arrayX)) \* (arrayY[i] - *mean*(arrayY));

}

cov = cov/arrayX.length;

**return**cov;

}

/\*\*

\*Коефіцієнткореляції

\***@param**arrayXвибіркаX

\***@param**arrayYвибіркаY

\***@return**cor=cov/(sigmaX,sigmaY)

\*/

**publicstaticdouble**correlation(**double**[] arrayX, **double**[] arrayY){

**return***covariance*(arrayX, arrayY)/(*standartDeviation*(arrayX) \*

*standartDeviation*(arrayY));

}

/\*\*

\*Генеруємасиврандомнихчиселізінтервалу[0,1]

\***@param**array

\*/

**publicstaticvoid**generateRandom(**double** [] array){

Random R = **new**Random();

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

array[i] = R.nextDouble();

}

}

/\*\*

\*Методдлядрукунаекранмасиву

\***@param**array

\*/

**publicstaticvoid**showArray(**double**[] array){

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

System.*out*.printf(" " + array[i]);

}

}

}