Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

**ЛАБОРАТОРНА РОБОТА № 3**

з дисципліни «Теорія імовірності» на тему

**«Системи неперервних випадкових величин»**

ВИКОНАЛА:

студентка ІІ курсу ФІОТ

групи ІО-64

Бровченко Анастасія

Залікова - 6403

ПЕРЕВІРИВ:

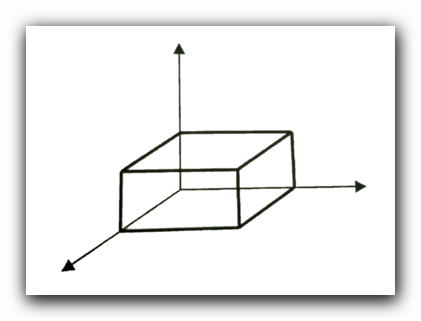
доц. Марковський О.П.

Київ – 2017

ЗАВДАННЯ

Аналітично визначити ф-ю часткового та умовного розподілу, математичне очікування, середньоквадратичне відхилення, коефіцієнт кореляції.

Експериментально визначити мат. очікування, середньоквадратичні відхилення, коефіцієнт кореляції.



АНАЛІТИЧНІ РОЗРАХУНКИ

Об’єм даної фігури V=1.

Припустимо, ця фігура – куб з ребром а=1.

Знайдемо висоту h=V/(a\*а)=1.

Часткова функція:

φx== =1;

*𝜑y==*  =1;

Умовна функція:

*=1;*

*=1.*

Математичне очікування:

;

.

Дисперсія:

= 0.083;

= 0.083.

Середнє квадратичне відхилення

;

.

Коваріація

0

Коефіцієнт кореляції

 .

КОД ПРОГРАМИ

Файл *f\_real.py*

**import** random, math  
  
*# сторона куба: a = 1  
# висота куба: h = 1/math.pow(a, 2)***def** gen\_xy(n):  
 *"""Генеруємо пари (х,у)"""* X = [random.random() **for** i **in** range(n)]  
 Y = [random.random() **for** j **in** range(n)]  
 **return** X, Y  
  
  
**def** mean(N):  
 *"""Математичне очікування"""* s = 0  
 **for** i **in** N:  
 s += i  
 m = s/len(N)  
 **return** m  
  
  
**def** cov(X, Y):  
 *"""Коваріація"""* n = 0  
 **for** i **in** range(len(X)):  
 n += ((X[i] - mean(X)) \* (Y[i] - mean(Y)))  
 n = n/len(X)  
 **return** n  
  
  
**def** deviation(N):  
 *"""Середньоквадратичне відхлення"""* v = 0  
 m = mean(N)  
 **for** i **in** range(len(N)):  
 v += math.pow((N[i] - m), 2)  
 v = v/len(N)  
 **return** math.sqrt(v)  
  
  
**def** corel(X,Y):  
 *"""Коефіцієнт кореляції"""* **return** cov(X, Y) / (deviation(X) \* deviation(Y))

Файл *GUI.py*

**from** tkinter **import** \*  
**from** PIL **import** ImageTk, Image  
**import** math  
**import** third\_sem.teor\_ver.lab3.f\_real **as** f\_real  
  
**def** counts():  
 X, Y = f\_real.gen\_xy(n)  
 l = **'Мат очікування X = {Mx}\n'**\  
 **'Мат очікування Y = {My}\n'**\  
 **'Дисперсія X = {Dx}\n'**\  
 **'Дисперсія Y = {Dy}\n'**\  
 **'Середньоквадр. відхилення X = {Sx}\n'**\  
 **'Середньоквадр. відхилення Y = {Sy}\n'**\  
 **'Коваріація = {cov}\n'**\  
 **'Коеф кореляції = {corel}\n'**.format(Mx=f\_real.mean(X), My=f\_real.mean(Y),  
 Dx=math.pow(f\_real.deviation(X), 2), Dy=math.pow(f\_real.deviation(Y), 2),  
 Sx=f\_real.deviation(X), Sy=f\_real.deviation(Y),  
 cov=f\_real.cov(X, Y), corel=f\_real.corel(X, Y))  
 **return** l  
  
**def** but\_bind():  
 **if** ent\_n.get().isalnum():  
 **global** n  
 n = int(ent\_n.get())  
 ent\_n[**'state'**] = DISABLED  
  
 l[**'text'**] = counts()  
 l[**'fg'**] = **'black'** l[**'font'**] = **'Arial 14'  
 else**:  
 l[**'text'**] = **'\*Перевірте введену довжину вибірки на зайві символи'** l[**'fg'**] = **'red'** l[**'font'**] = **'Arial 14 bold'  
  
  
if** \_\_name\_\_ == **'\_\_main\_\_'**:  
 root = Tk()  
  
 Label(root, text=**'Функція щільності\n(куб зі стороною 1)'**, font=**'Arial 16 bold'**).grid(row=0, column=0)  
  
 canvas = Canvas(root,width=400, height=300)  
 canvas.grid(row=1, column=0)  
 pilImage = Image.open(**"photo.jpg"**)  
 image = ImageTk.PhotoImage(pilImage)  
 imagesprite = canvas.create\_image(200, 150, image=image)  
  
 Label(root, text=**'Введіть довжину вибірки'**, font=**'Arial 14 bold'**).grid(row=2, column=0)  
  
 ent\_n = Entry(root, width=10, bd=3, bg=**'light cyan'**, font=**'Arial 13'**)  
 ent\_n.insert(END, **'5000'**)  
 ent\_n.grid(row=3, column=0, pady=10)  
  
 Button(root, text=**'OK'**,font=**'Arial 14 bold'**, bg=**'pale green'**, command=but\_bind).grid(row=4, column=0)  
  
 l = Label(root, text=**''**, justify=LEFT)  
 l.grid(row=5, column=0)  
  
 root.mainloop()

РЕЗУЛЬТАТ РОБОТИ ПРОГРАМИ

