**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ **«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ ИМ. В. Г. ШУХОВА»**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

**ЛАБОРАТОРНАЯ РАБОТА №0**

**Дисциплина: Теория надежности**

Выполнил: ст. группы ВТ-31

Подкопаев Антон Валерьевич

Проверил: доц. каф. ПО и ВТАС

Кабалянц Петр Степанович

**Белгород 2020**

**Задания для выполнения к работе**

**дискретно**

1. Получить выборку 1000 чисел в промежутке от 0 до 1, используя random.
2. Используя имитационные методы, получить выборку заданной случайной величины (последние три цифры зачетной книжки).
3. Вычислить теоретически мат. ожидание и дисперсию, а также их статистические оценки.
4. Используя χ-квадрат проверить гипотезу о том, что полученная выборка является выборкой из значений случайной дискретной величины, заданной в варианте.

**непрерывно**

1. Используя имитационные методы из пункта 1, получить выборку для показательного распределения с параметром λ = 13/5
2. Вычислить теоретически мат. ожидание и дисперсию, а также их статистические оценки.
3. Проверить гипотезу о том, что это значения из показательного распределения с параметром λ.

**Ход выполнения работы**

**Непрерывная случайная величина**

Сгенерируем 1000 случайных величин с равномерным распределением при λ = 13/5:

Изображение выглядит как снимок экрана, монитор, компьютер, черный

Автоматически созданное описание

Получим по формуле набор случайных величин показательного распределения:

Изображение выглядит как снимок экрана, монитор, компьютер, черный

Автоматически созданное описание

Вычислим среднее арифметическое и дисперсию:

= 0.380

D(x) = 0.141

Проверим гипотезу о том, что полученная выборка соответствует показательному закону распределения. Используем критерий Пирсона ()

= 0.78 ; 6, следовательно гипотеза верна

**Дискретная случайная величина**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 |
|  |  | 0.357 | 0.464 | 0.072 |

Вычислим среднее арифметическое и дисперсию:

= 1.498

D(x) = 0.608

Вычислим теоретические и фактические частоты:

Теоретические частоты:

[107, 357, 464, 71]

Полученные частоты:

[115, 351, 458, 76]

Проверим гипотезу о том, что полученная выборка соответствует показательному закону распределения. Используем критерий Пирсона ()

= 2.683 ; 6, следовательно гипотеза верна

*Приложение*

# This Python file uses the following encoding: utf-8

import random

import math

import numpy

from prettytable import PrettyTable

import matplotlib.pyplot as pyplot

def getRandomNumbers(sz):

result = []

for i in range (0, sz):

result.append(random.random())

return result

if \_\_name\_\_ == "\_\_main\_\_":

l = 13/5

sz = 1000

#Получаем случайные числа

randomNumbers = getRandomNumbers(sz)

expRandomNumbers = numpy.multiply(numpy.log(randomNumbers), - 1/l)

#print("Полученные числа: ")

#print(numpy.round(expRandomNumbers, 3))

#Вычисляем мат ожидание и дисперсию

mean = numpy.mean(expRandomNumbers)

mean2 = numpy.mean(numpy.square(expRandomNumbers))

disp = mean2 - mean\*\*2

print("Непрерывная СВ")

print("Среднее: ", numpy.mean(expRandomNumbers))

print("Дисперсия: ", disp)

#Находим интервалы разбиения

max = numpy.max(expRandomNumbers)

x = numpy.linspace(0, max, num = 5)

intervals = [0, 0, 0, 0, 0] # n

n = [0, 0, 0, 0]

for X in expRandomNumbers:

i = 0

while not (x[i] < X <= x[i+1]):

i+=1

n[i]+=1

p = [0, 0, 0, 0]

for i in range (0, 4):

p[i] = math.exp(-l \* x[i]) - math.exp(-l \* x[i+1])

nt = numpy.multiply(p, sz)

k = 0

for i in range (0, 4):

k += (n[i] - nt[i]) \*\* 2 / n[i]

if (k <= 6):

print ("K\_в = ", round(k, 3), "<= 6, ==> гипотеза верна")

else :

print ("K\_в = ", round(k, 3), ">= 6, ==> гипотеза не верна")

# ДИСКРЕТНАЯ СВ

i = 3

j = 10

k = 1

p[0] = round(i / (2 \* (i + j + k)), 3)

p[1] = round(j / (2 \* (i + j + k)), 3)

p[2] = round((i + j) / (2 \* (i + j + k)), 3)

p[3] = round((2 \* k) / (2 \* (i + j + k)), 3)

n = [0, 0, 0, 0]

for i in range(0, sz):

if (randomNumbers[i] <= p[0]): n[0] += 1

elif (p[0] < randomNumbers[i] <= p[0] + p[1]): n[1] += 1

elif (p[0] + p[1] < randomNumbers[i] <= p[0] + p[1] + p[2]): n[2] += 1

elif (p[0] + p[1] + p[2] <= randomNumbers[i]) : n[3] += 1

print("Дискретная СВ")

print("Закон распределения СВ согласно варианту:", p)

avg = 0

for i in range (0, 4):

avg += i \* p[i]

dispersion = 0

for i in range (0, 4):

dispersion += i\*\*2 \* p[i]

dispersion -= avg\*\*2

print("Среднее:", avg)

print("Дисперсия: ", dispersion)

n1 = [0, 0, 0, 0]

for i in range (0, 4):

n1[i] = sz \* p[i]

print("Теоретические частоты: ")

print(n1)

print("Полученные частоты: ")

print(n)

k = 0

for i in range(0, 4):

k += ((n[i] - n1[i])\*\*2)/n1[i]

if (k <= 6):

print ("K\_в = ", round(k, 3), "<= 6, ==> гипотеза верна")

else :

print ("K\_в = ", round(k, 3), ">= 6, ==> гипотеза не верна")

dExprandomNumbers = []

for i in range (len(n)):

for k in range(int(n[i])):

dExprandomNumbers.append(i)

title, p = pyplot.subplots(3)

p[0].hist(randomNumbers)

p[1].hist(expRandomNumbers, 10)

p[2].hist(dExprandomNumbers, 4)

pyplot.show()