СОДЕРЖАНИЕ

ВВЕДЕНИЕ3

1. ПОСТАНОВКА ЗАДАЧИ4
2. ПОСЛЕДОВАТЕЛЬНОСТИ СЛУЧАЙНЫХ СОБЫТИЙ 6
   1. Разработка метода получения последовательностей случайных событий программным путем на основе системы MATLAB 6
   2. Верификация разработанного алгоритма 7
3. РАСЧЁТ ВЕРОЯТНОСТЕЙ СРАБАТЫВАНИЯ КОМБИНАЦИОННОЙ СХЕМЫ10
   1. Разработка комбинационной схемы10
   2. Расчёт вероятностей срабатывания комбинационной схемы11

3.3. Разработка имитационного алгоритма срабатывания комбинационной схемы14

3.4. Верификация разработанного алгоритма 14

1. ЧИСЛОВЫЕ ХАРАКТЕРИСТИКИ СЛУЧАЙНЫХ ВЕЛИЧИН17
   1. Разработка алгоритма нахождения числовых характеристик случайных величин17
   2. Верификация разработанного алгоритма 18

ЗАКЛЮЧЕНИЕ25

БИБЛИОГРАФИЧЕСКИЙ СПИСОК26

ПРИЛОЖЕНИЕ А27

*Изм.*

№ Документа

*Подпись*

*Дата*

*Лист*

3

Курсовой проект

*Разработал*

Воронин И.Ю.

*Руковод.*

Заикина Е.Н.

*Т.контр.*

*Н.контр.*

*Утв.*

ПОЯСНИТЕЛЬНАЯ  
ЗАПИСКА

*Лит*

*Листов*

ИСб-22о

ВВЕДЕНИЕ

Целью проведения курсовой работы является закрепление, углубление и обобщение знаний и навыков моделирования случайных событий на ЭВМ, а также оценки их основных характеристик. В процессе выполнения курсовой работы совершенствуется техника программирования на языке пакета математических расчётов MatLab.

Курсовое проектирование включает следующие этапы:

1. изучение математического метода решения задачи;
2. получение последовательностей случайных событий программным путем;
3. составление комбинационной схемы;
4. расчет вероятностей срабатывания комбинационной схемы;
5. разработка имитационного алгоритма срабатывания комбинационной схемы;
6. разработка алгоритма нахождения числовых характеристик случайных величин;
7. разработка тестового примера;
8. написание текста программы;
9. отладка программы;
10. проведение вычислительного эксперимента;
11. оформление технической документации на программу.
12. ПОСТАНОВКА ЗАДАЧИ

Применить изученные методы получения последовательностей случайных событий программным путём на основе системы *MatLab* к конкретному эксперименту. Рассчитать текущую частоту случайных событий, реализованных в проводимом эксперименте. Убедиться, что случайные события, произошедшие в данном эксперименте, обладают свойством стохастической устойчивости. Оценить вероятность этих событий.

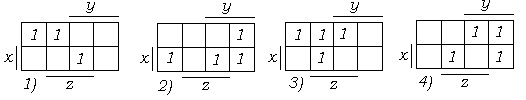
Таблица 1 – Вариант второго задания.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № |  |  |  |  |  |  |  |  |  |  |
| 4 | 0.3 | 0.8 | 0.3 | 0.8 | 0.3 | 0.8 | 0.20 | 0.25 | 0.06 | 0.96 |

Выполнить теоретический расчёт вероятностей срабатывания комбинационных схем и найти оценки этих вероятностей экспериментальным путём. Сравнить теоретические и экспериментальные результаты. Оценить применимость теорем сложения и умножения вероятностей и формулы полной вероятности для вычисления вероятностей сложных событий на примере работы комбинационных схем.

Таблица 2 – Вариант второго задания.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | *am* | *aM* | *bm* | *bM* | *cm* | *cM* |
| IV | 0.2 | 0.7 | 0 | 0.3 | 0.1 | 0.5 |



Вспомнить методы нахождения числовых характеристик случайных величин. Также необходимо произвести экспериментальные исследования зависимости точности оценок числовых характеристик от объема выборки случайной величины.

MU = 5;

SIGMA = 3;

Таблица 3 – Вариант третьего задания.

|  |  |  |  |
| --- | --- | --- | --- |
| № | Вид распределения | Команда генерациислучайной величины | Команда вычисления  M1 и |
| Непрерывные распределения | | | |
| 4 | Логнормальное | R=lognrnd(MU,SIGMA,m,n) | [M,V]=lognstat(MU,  SIGMA) |

2. ПОСЛЕДОВАТЕЛЬНОСТИ СЛУЧАЙНЫХ СОБЫТИЙ

2.1. Разработка метода получения последовательностей случайных событий программным путем на основе системы MATLAB.

Для выполнения поставленной задачи совершим следующие действия:

1. Создадим матрицу , элементами  которой являются случайные равномерно распределенные числа, лежащие в диапазоне от 0 до 1. Число строк матрицы *m* = 5, число столбцов *n =* 1000.

2. Проверить наличие элементов в матрице A, выведя на экран ее первые 10 столбцов.

Будем считать событием  попадание числа в промежуток .

3. Создать М-функцию , которая возвращает единицу, если выполняется условие , и возвращает 0, если это условие не выполнено.

4. С помощью функции *logzn* из матрицы  получить матрицу , элементы которой равны 1, если событие произошло, и равны 0, если не произошло. Для этого написать и сохранить соответствующую М-функцию.

5. Написать М-функцию , где *v* – вектор размера *m*, состоящий из нулей и единиц. Сохранить ее в М-файле.

6. Рассчитать зависимости  частот событий от числа испытаний для 0 и всех пяти *k* и изобразить их графически в линейном и полулогарифмическом (по оси *x*) масштабах. Найти *аналитически* вероятности событий , учтя тип распределения получаемого с помощью функции *rand*.

Функция logzn(am,aM,x), принимающая на вход нижний (am) и верхний (aM) пределы промежутка и затем проверяемое число, возвращает 1 в случае, если переданные значения удовлетворяют условию am ≤ x ≤ aM, иначе возвращает 0.Функция заполнения матрицы частот – logzn1 - в качестве параметров принимает вектор случайных чисел. Возвращает матрицу из «0» и «1» (0 – событие не произошло, то есть число из вектора случайных чисел не входит в необходимый интервал случайного события, а 1 – событие произошло), строки которой соответствуют номеру случайного события

Функция fregp(*v,m*) подсчитывает частоту встречаемости единиц в строке матрицы для определенного события. В качестве параметров принимает матрицу *v* из «0» и «1» - вектор, хранящий частоту исследуемого события для каждого числа испытаний, и номер событий *m*, соответствующий номеру строки в матрице, над которой будет производиться анализ. Возвращает вектор частоту встречаемости единиц в строке матрицы для каждого числа испытаний. Так как все элементы строки – единицы или нули, для расчета частоты элементы строки матрицы суммируются и делятся на количество элементов.

Для получения последовательности случайных событий программным путем на основе системы MATLAB была написана программа, приведенная в приложении A.

2.2 Верификация разработанного алгоритма

Для вычисления вероятности попадания величины в заданный интервал, необходимо найти площадь под кривой распределения, опирающийся на этот интервал. Так как функция rand генерирует равномерное непрерывное распределение, вычисления сведутся к простому нахождению площади прямоугольника (рис.1).

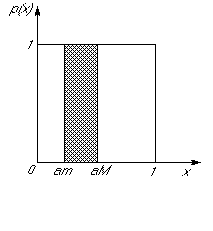


Рисунок 1 – Равномерное непрерывное распределение

5

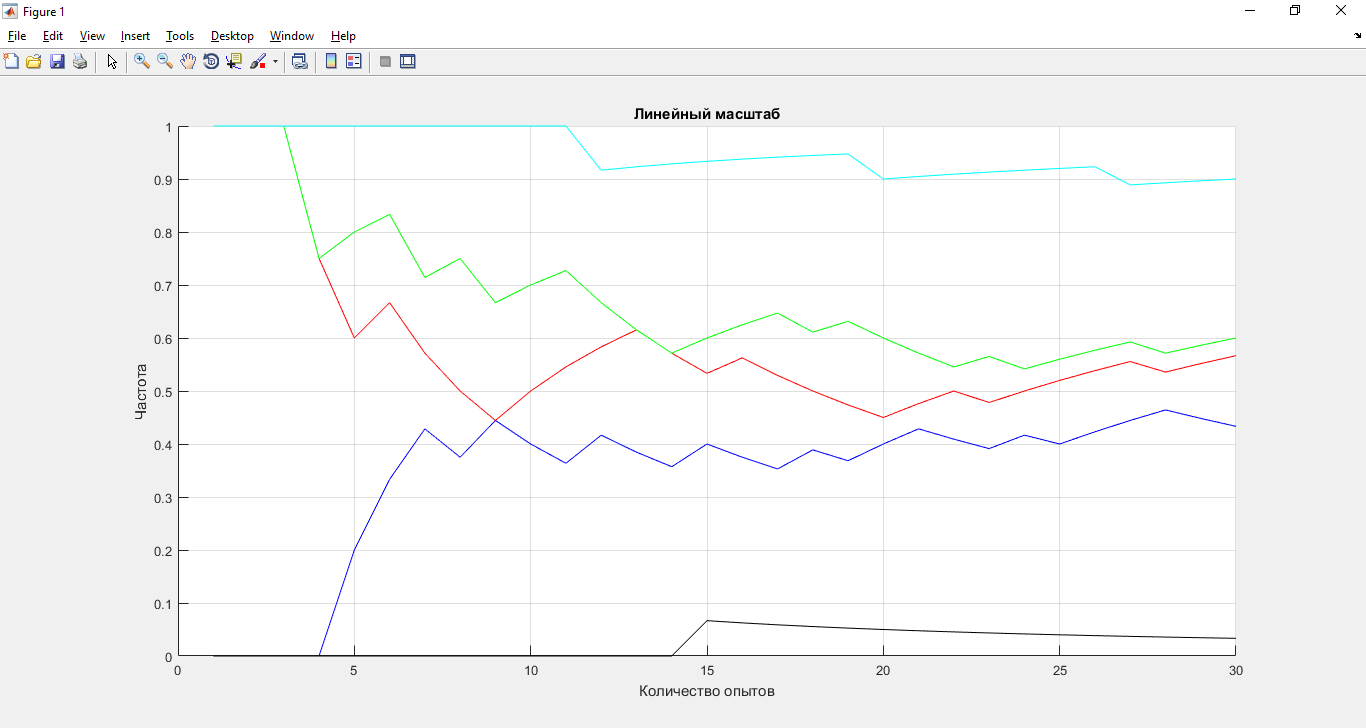


Рисунок 2 – Графики зависимости частоты событий от числа испытаний в линейном масштабе

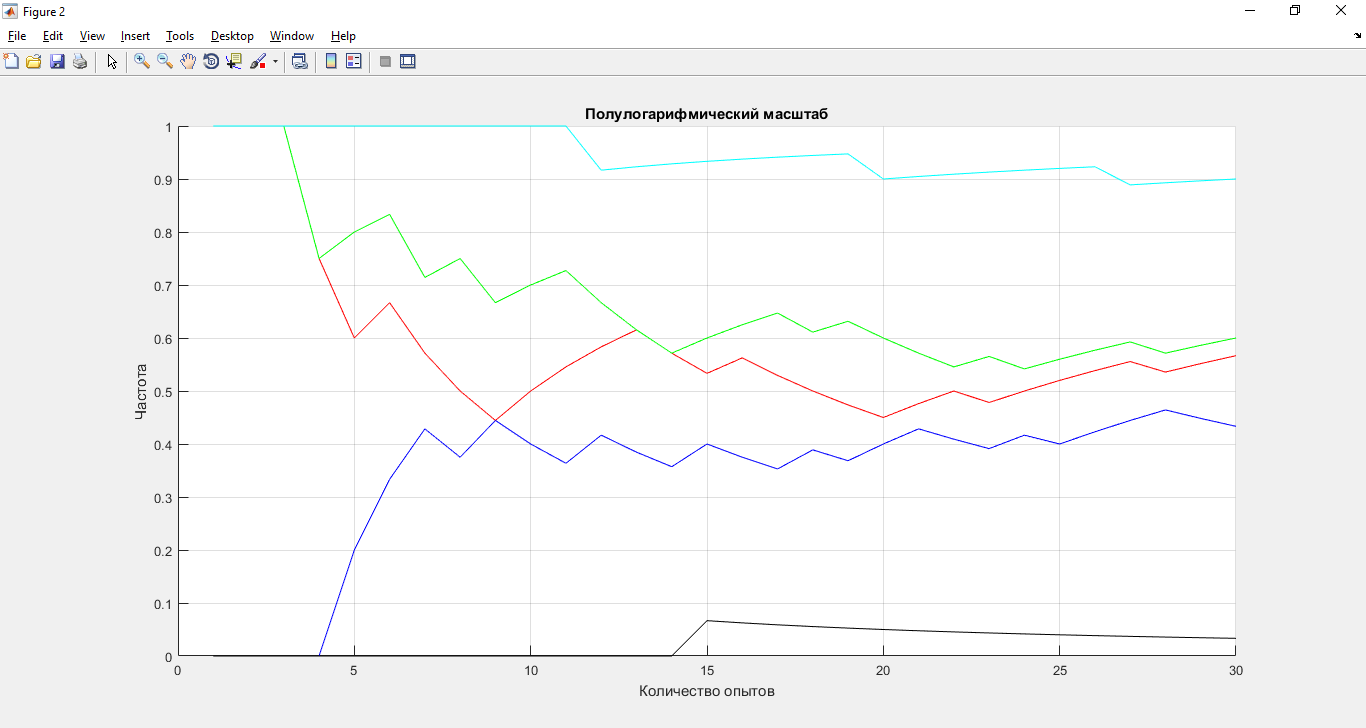


Рисунок 3 - Графики зависимости частоты событий от числа испытаний в полулогарифмических масштабе

3. РАСЧЁТ ВЕРОЯТНОСТЕЙ СРАБАТЫВАНИЯ КОМБИНАЦИОННОЙ СХЕМЫ

3.1 Разработка комбинационной схемы

Составим по данной карте Карно найдём минимальную ДНФ включения лампочки. Получаем следующую запись:

Изобразим данную формулу при помощи комбинационной схемы (рис.4)

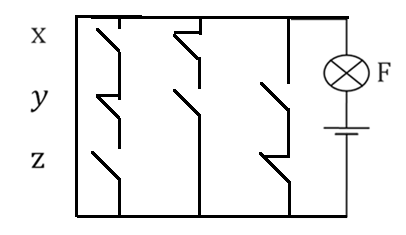


Рисунок 4 – Комбинационной схема.

Также изобразим графически промежутки, что заданы вариантом задания. (рис.5)

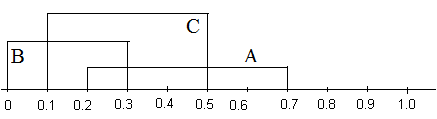


Рисунок 5 – Интервалы событий

3.2 Расчёт вероятностей срабатывания комбинационной схемы

Произведём аналитический расчёт по формулам сложения-умножения. Рассчитаем вероятности событий и их дополнений.

Теоретические значения вероятностей нажатия кнопок:

p(A) = p(x) = 0.5;

p(B) = p(y) = 0.3;

p(C) = p(z) = 0.4;

Для независимых событий:

В данном случае будут использоваться следующие формулы сложения и умножения:

Произведём расчёт:

+ (P()+P)\*P() =

=0,18 + 0,15 + 0,14 + (0,18 + 0,15) \* 0,14 = 0,5162

Итак, для независимых событий P(F) = 0,5162

Для зависимых событий:

В данном случае все произведения вероятностей рассчитывать с использованием теоремы умножения, а все условные вероятности найти графоаналитическим способом

В данном случае будут использоваться следующие формулы сложения и умножения:

*, где*

Изначально рассчитаем вспомогательные величины условных вероятностей:

Произведём расчёт на основе полученных величин:

Итак, для зависимых событий P(F) = 0,6.

Пусть дана гипотеза S1, что кнопка «Y» нажата.

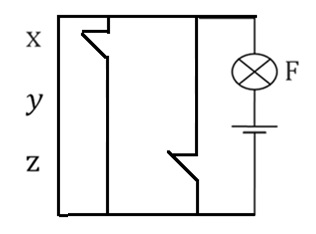


Рисунок 6 – Кнопка Y нажата.

Аналитический расчёт по формуле полной вероятности.

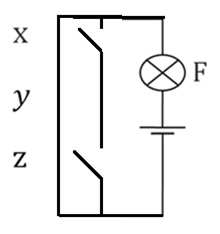


Рисунок 7 – Кнопка Y не нажата.

Тогда:

Для независимых событий:

P(F / S1) = P ( ) = P() + P() = 0,5 + 0,6 = 1

Для зависимых событий:

P(F / S1) = P ( )= P() + P() - P() = 0,5 + 0,6 - P() \* P() = 1,1 - 0,4 = 0,7

Гипотеза S2 будет представлена ситуацией, если кнопка «Y» не нажата и соответственно в таком случае вероятность будет равна:

Для независимых событий:

P(F / S2) = P = P(x) \* P(z) = 0,5 \* 0,4 = 0,2;

Для зависимых событий:

P(F / S2) = P = P(x) \* P(z \ x)= 0,5 \* 0,3 = 0,315;

Рассчитаем формулу полной вероятности:

Для независимых событий:

= 0,3 \* 1 + 0,7 \* 0,2 = 0,44;

Для зависимых событий:

0,3 \* 0,7 + 0,7 \* 0,15 = 0,315;

3.3. Разработка имитационного алгоритма срабатывания комбинационной схемы

Для имитации срабатывания комбинационной схемы необходимо выполнить следующую последовательность действий:

Создадим матрицу случайных чисел, состоящую из 1000 строк и 4 столбцов. Необходимо создать три вектора и заполнить их числами «0» либо «1» в зависимости от попадания соответствующего элемента матрицы случайных чисел в исследуемые интервалы. Для каждого вектора использовать отдельный столбец матрицы случайных чисел (для первого вектора – первый столбец, для второго – второй и т.д.). Далее следует реализовать функцию заполнения вектора нулями либо единицами в соответствии с выписанной минимальной ДНФ для независимых (зависимых) событий.

Рассчитать частоту реализации минимальной ДНФ для независимых (зависимых) событий, создав соответствующий вектор, содержащий частоты события для каждого числа испытаний.

Построить соответствующие графики.

3.4. Верификация разработанного алгоритма

При расчётах была получена вероятность включения лампочки для:

* независимых событий 0.44
* а для зависимых 0.315.

На рисунках 7 и 8 представлены графики оценки вероятности для независимых и зависимы событий в линейном масштабе

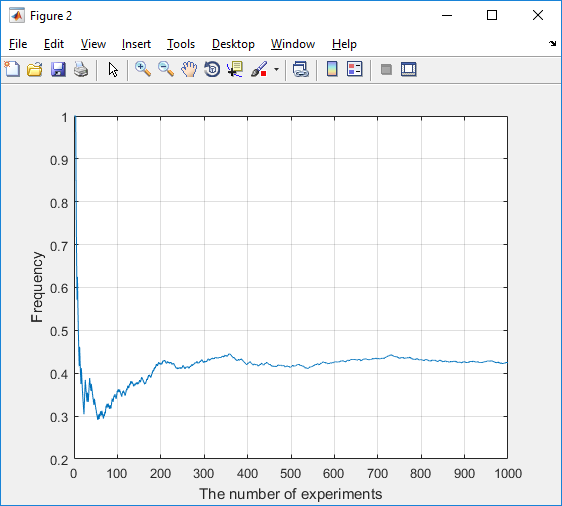


Рисунок 7 – Анализ незав­исимых событий

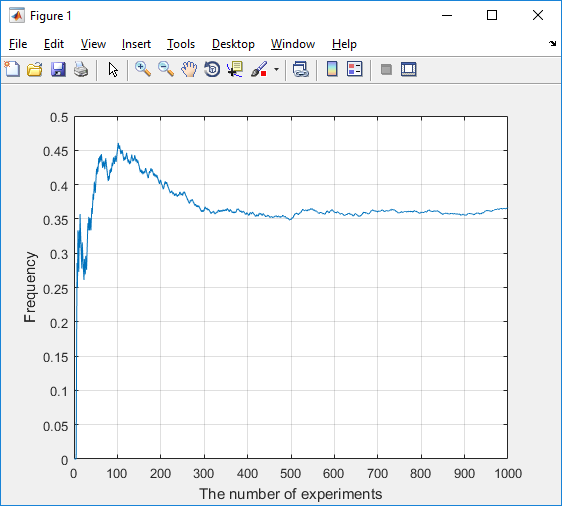


Рисунок 8 – Анализ зависимых событий.

Теоретические близки к экспериментальным, что подтверждает правильность и точность их вычислений.

0,42 0,44 (Абс. погрешность: 0,02)

0,315 0,36 (Абс. погрешность: 0,045)

4. ЧИСЛОВЫЕ ХАРАКТЕРИСТИКИ СЛУЧАЙНЫХ ВЕЛИЧИН

4.1 Разработка алгоритма нахождения числовых характеристик случайных величин.

Для произведения расчетов заполним вектора случайными числами с помощью функции lognrnd (MU,m,n), которая генерирует экспоненциальное распределение. После этого алгоритм выполняет вычисление оценок МО, центральных моментов k-го порядка (k = 1,2,3,4), дисперсии, коэффициентов эксцесса и асимметрии. Затем рассчитаем зависимости оценок вышеуказанных числовых характеристик от числа испытаний, после чего построим график зависимостей.

Данный алгоритм использует вспомогательную функцию для расчета среднего арифметического некоторого распределения до определенного испытания. В качестве параметров она принимает вектор чисел и номер элемента, до которого необходимо вычислить среднее арифметическое предшествующих чисел. Она используется при вычислении оценок числовых характеристик непрерывной случайной величины до n-го испытания (n = 1000).

Данная задача будет выполняться по завершению следующих этапов:

1. Написать программу в MATLAB вычисляющую оценки моментов , , , , , оценки коэффициента асимметрии и эксцесса.

2. При помощи данной программы вычислить зависимости указанных оценок от количества испытаний *N (*1 ≤ *N* ≤ 1000) и отобразить их на графике в линейном и полулогарифмическом масштабе.

3. Вычисление теоретические значения математического ожидания и дисперсию и сравнение их с экспериментальными значениями.

4. Установим вид теоретических кривых, которые описывают закон данного логнормального распределения, при помощи команды disttool

5. Установим, как меняются эмпирические распределения данной случайно величины, при помощи команды randtool, при последовательном выборе ее числа отсчетов *N* = 100, 200, 500, 1000.

4.2 Верификация разработанного алгоритма

Мат. ожидание и дисперсия в логнормальном распределении при MU = 0.5 и SIGMA = 0.25 равны:

M = 1.7011

V = 0.1866

Математическое ожидание = 1.718122

Центральный момент первого порядка случайной величины = 3.0509e-15

Центральный момент второго порядка случайной величины = 0.1944

Центральный момент третьего порядка случайной величины = 0.0716

Центральный момент четвертого порядка случайной величины = 0.1575

Оценка коэффициента асимметрии = 0.8355

Оценка коэффициента эксцесса = 107.1719

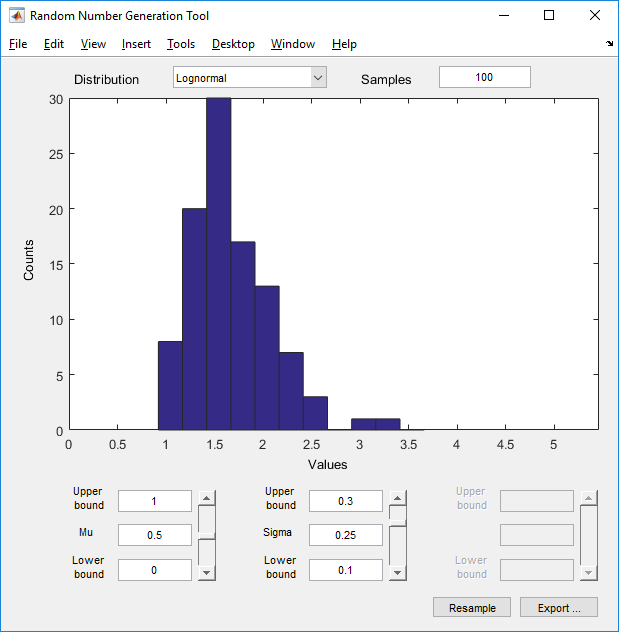


Рисунок 9 – Логнормальное распределение 100 элементов

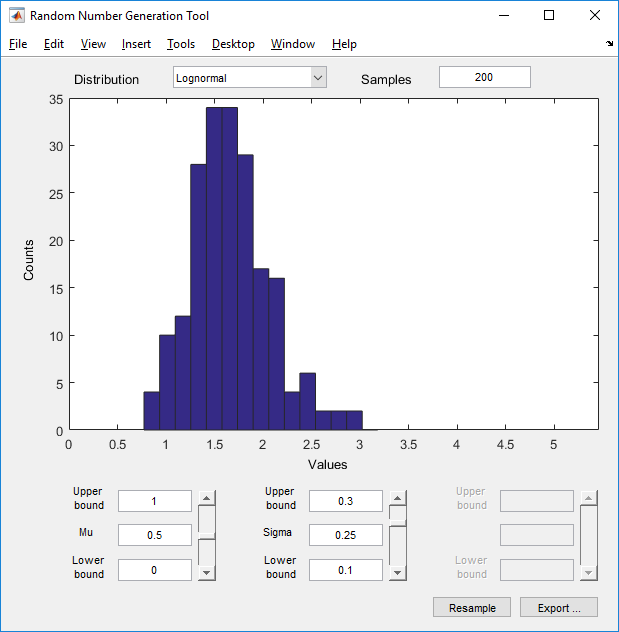


Рисунок 10 – Логнормальное распределение 200 элементов

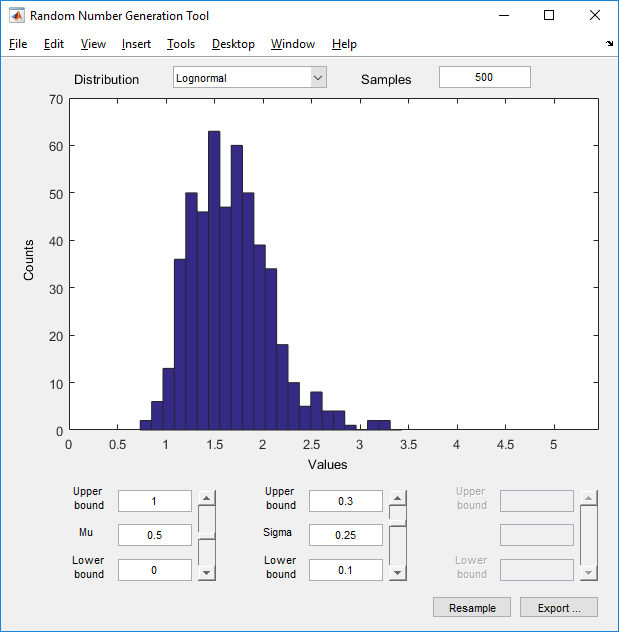


Рисунок 11 – Логнормальное распределение 500 элементов

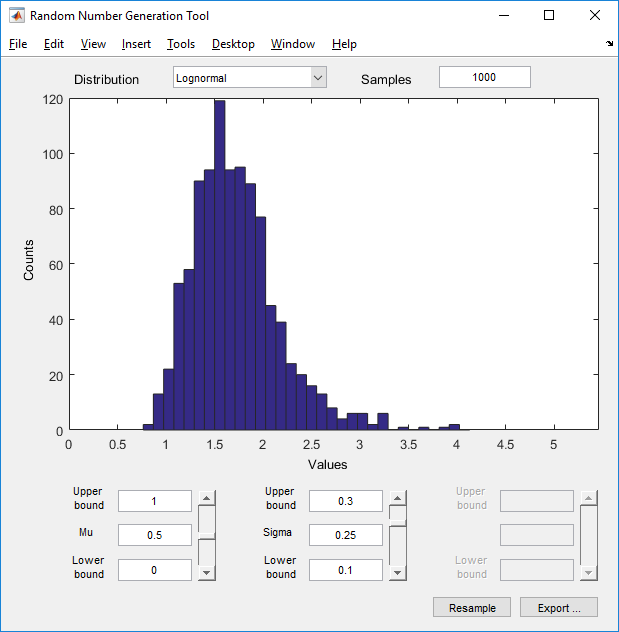


Рисунок 12 – Логнормальное распределение 1000 элементов

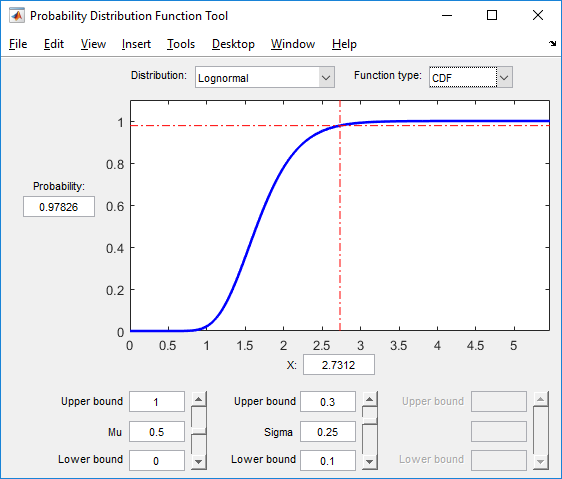


Рисунок 13 - Интегральная функция распределения данной с.в.

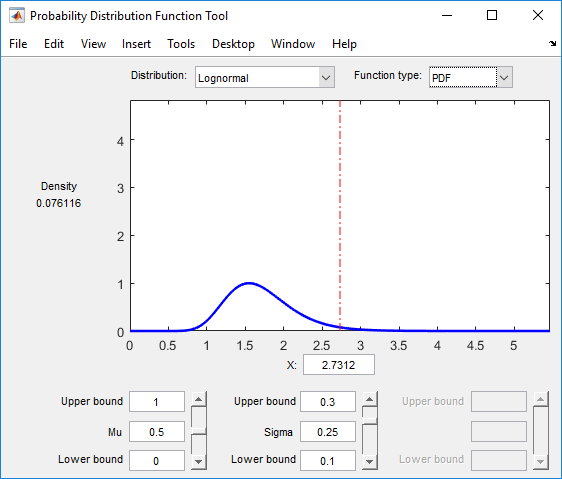


Рисунок 14 - Кривая плотности вероятности данной непрерывной с.в.

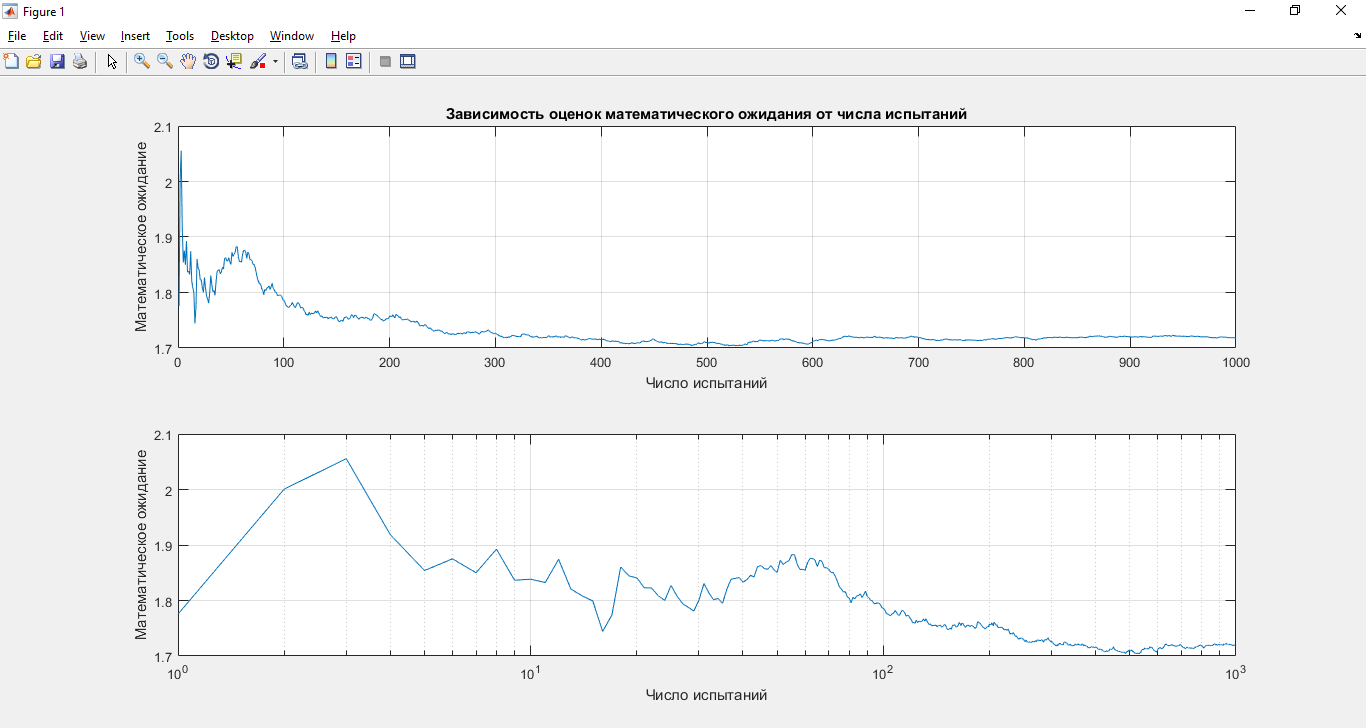


Рисунок 15 – Характеристика математического ожидания

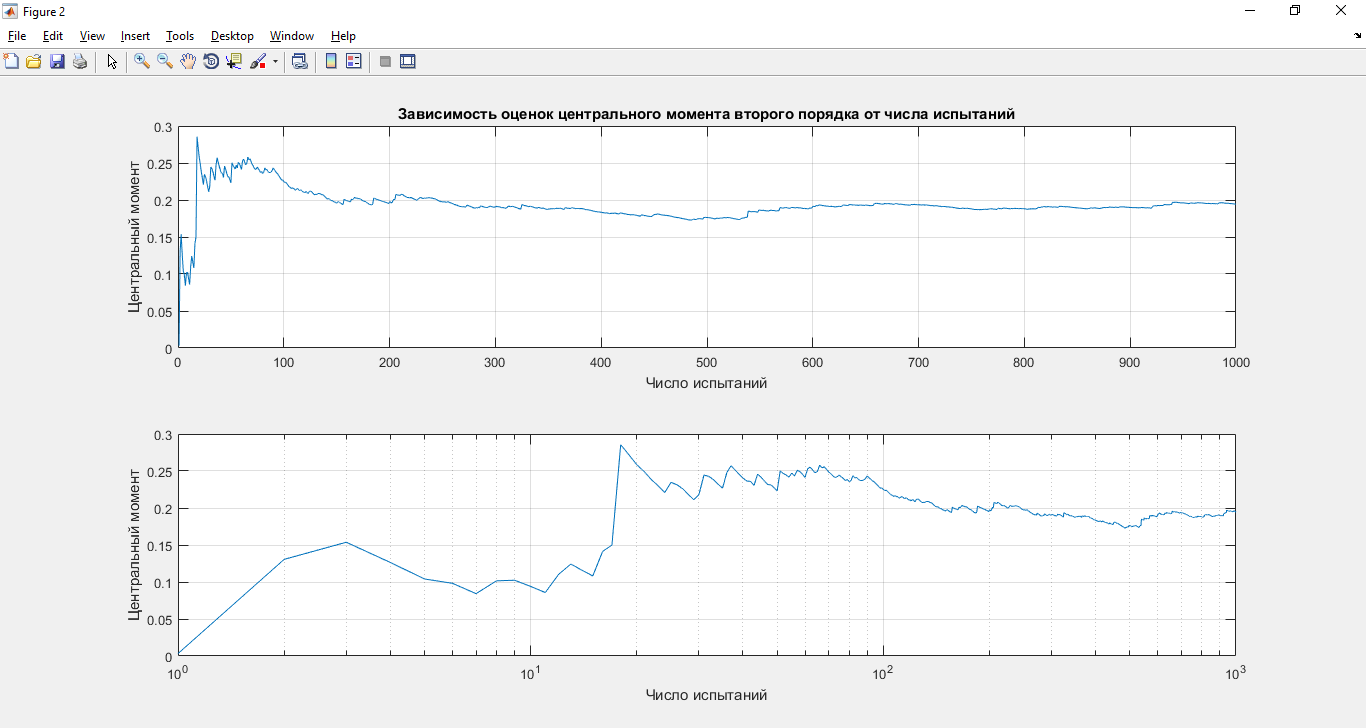


Рисунок 16 - Характеристика центрального момента 2-го порядка

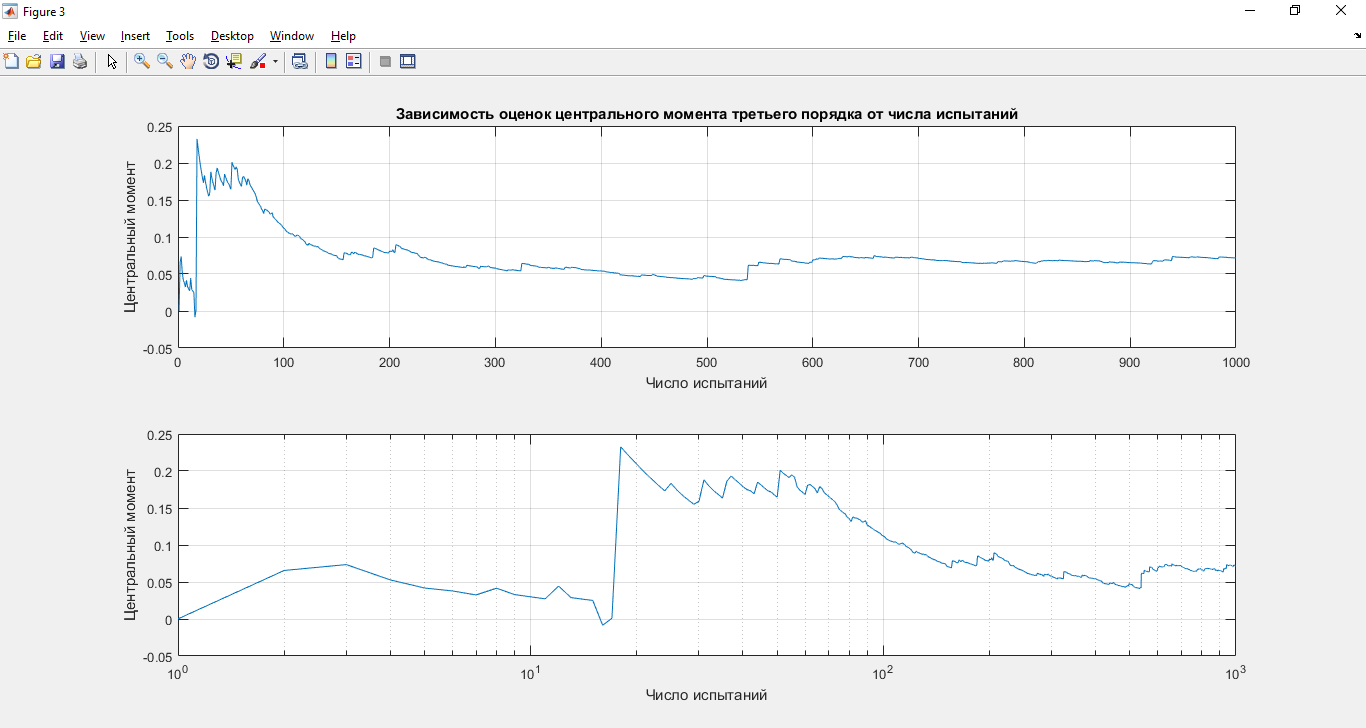


Рисунок 17 – Характеристика центрального момента 3-го порядка

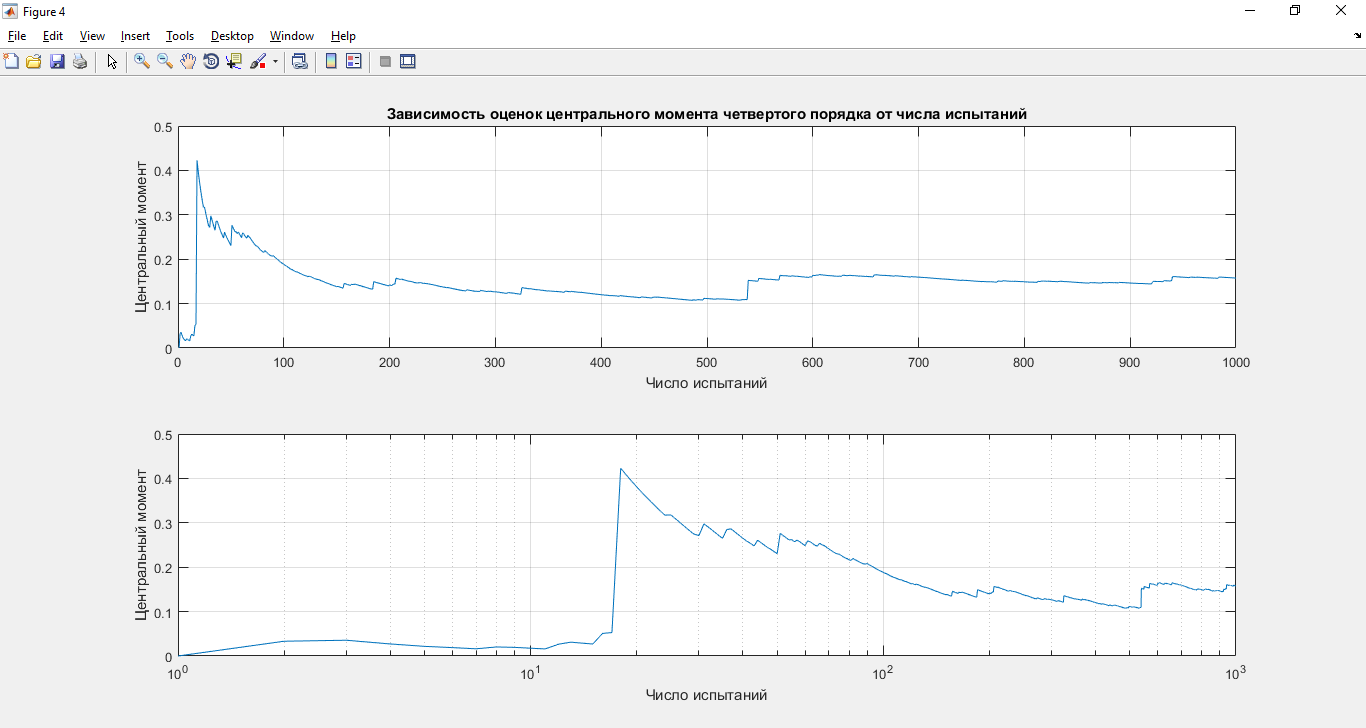


Рисунок 18 - Характеристика центрального момента 4-го порядка

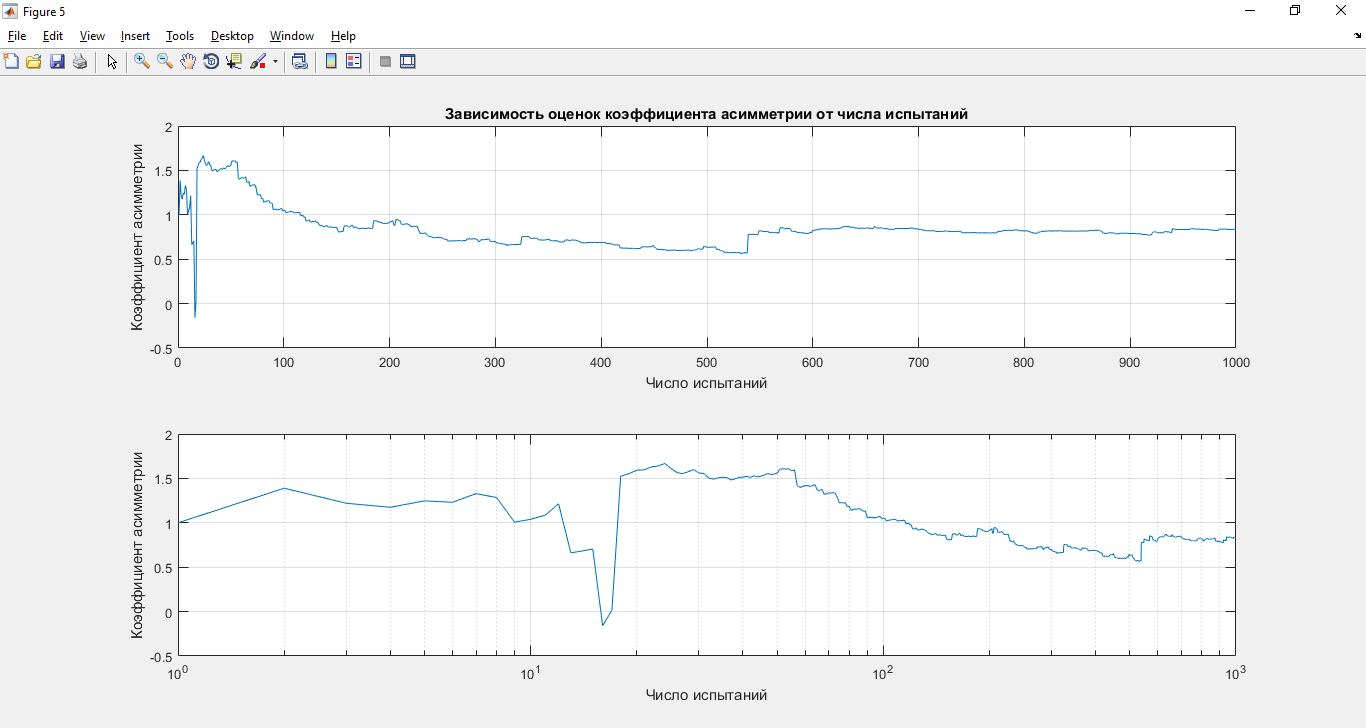


Рисунок 19 - Характеристика коэффициента асимметрии

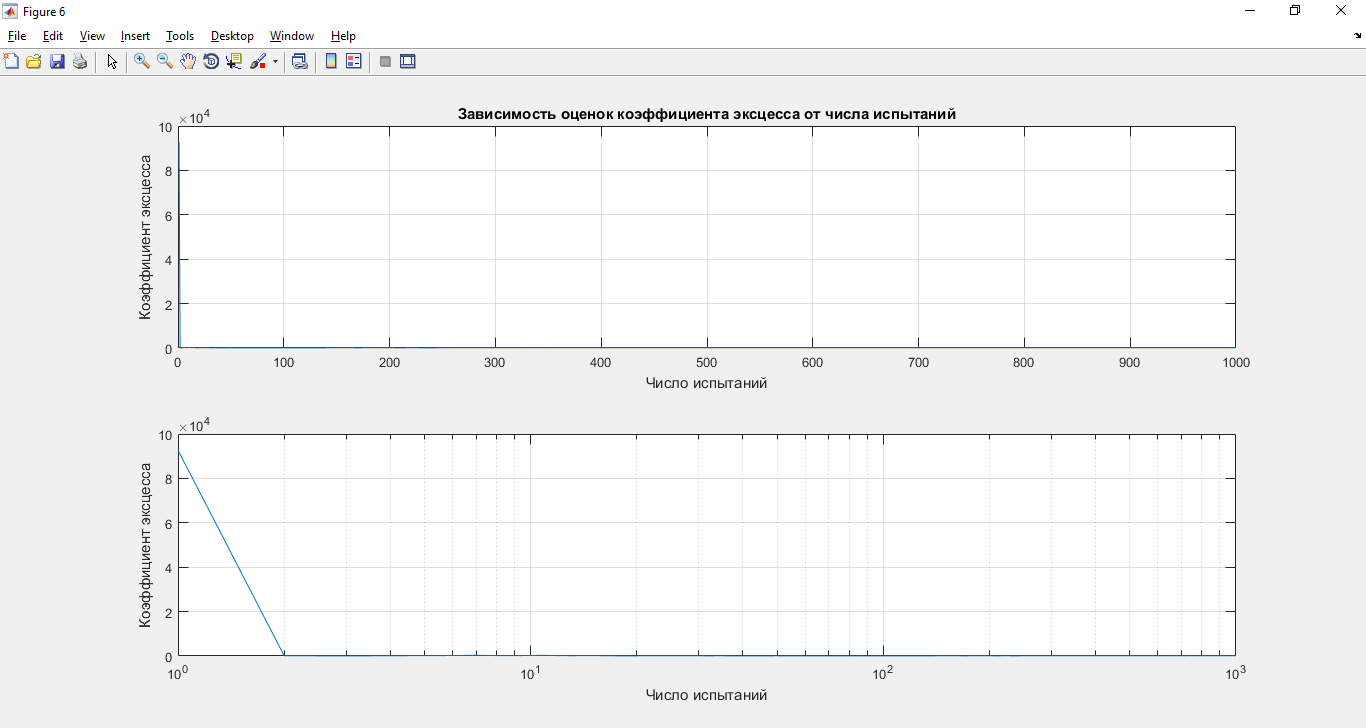


Рисунок 20 – Характеристика коэффициента эксцесса

Сравнивая графики теоретических кривых и графики зависимости числовых характеристик от числа экспериментов, полученные в результате выполнения написанной программы, можно сделать вывод, что программа работает правильно.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения курсовой работы были закреплены навыки моделирования случайных событий на ЭВМ в среде *MatLab*, а также оценки их числовых характеристик.

Была выполнена разработка метода получения последовательностей случайных событий программным путем, аналитически рассчитаны вероятности исходов для каждого из экспериментов. Также была написана программа на языке *MatLab*, по результатам которой можно сделать следующий вывод относительно данной части курсовой работы: числа, сгенерированные генератором псевдослучайных чисел, обладают свойством стохастической устойчивости, что иллюстрируют соответствующие графики.

Также были выполнены теоретические расчеты вероятностей срабатывания комбинационных схем. Результаты, полученные для независимых событий путем использования теорем о сложении и умножении вероятностей и формулы полной вероятности, совпали в полной мере. Сравнив результаты, полученные в ходе выполнения теоретических расчетов с результатами, полученными экспериментально, можно сделать вывод относительно того, что вероятности подсчитаны верно, а погрешности являются незначительными.

Также были изучены методы нахождения числовых характеристик случайных величин, а также проведены исследования зависимости точности приблизительных оценок от размера выборки случайной величины.

Было определено, что требуемая точность достигается с увеличением числа экспериментов в силу стохастической устойчивости.

Значения, полученные в результате эксперимента в значительной мере, совпадают с теоретическими.

Цели и этапы курсового проектирования были выполнены.

# БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Агапов Г.И. Задачник по теории вероятностей: Учебное пособие/Г.И.Агапов.-М.:Высшая школа, 1986.-80с.

2. Скворцов В. В. Теория вероятностей? Это интересно!/В.В.Скворцов.- М.: Мир, 1993.-118с.

3. Вентцель Е.С.Теория вероятностей и ее инженерные приложения/ Е.С. Вентцель, Л.А.Овчаров.-М.:Наука,1988.- 480с.

4. Чистяков В.П. Курс теории вероятностей: Учебник для вузов/В.П.Чистяков.- М.: Агар, 1996.- 256с.

ПРИЛОЖЕНИЕ А

В Приложении А содержится код программы с комментариями, написанных в программе MatLab, а также вспомогательные функции, которые использовались при расчётах.

Код программы для пункта 1:

clc;

m = 5;

n = 1000;

A = rand(m ,n);

min = [0.3, 0.3, 0.3, 0.2, 0.06;];

max = [0.8, 0.8, 0.8, 0.25, 0.96;];

B (m,n)=0;

for i=1:1:m

for j=1:1:n

B(i,j) = logzn(min(1,i),max(1,i),A(i,j));

end

end

arr1 = fregp(B(1,1:N),N);

arr2 = fregp(B(2,1:N),N);

arr3 = fregp(B(3,1:N),N);

arr4 = fregp(B(4,1:N),N);

arr5 = fregp(B(5,1:N),N);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

hold on

title('Линейный масштаб')

plot(arr1,'r','LineWidth',0.5)

plot(arr2,'g','LineWidth',0.5)

plot(arr3,'b','LineWidth',0.5)

plot(arr4,'k','LineWidth',0.5)

plot(arr5,'c','LineWidth',0.5)

xlabel('Количество опытов')

ylabel('Частота')

grid on

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

figure %новое окно

hold on

title('Полулогарифмический масштаб')

semilogx(arr1,'r','LineWidth',0.5)

semilogx(arr2,'g','LineWidth',0.5)

semilogx(arr3,'b','LineWidth',0.5)

semilogx(arr4,'k','LineWidth',0.5)

semilogx(arr5,'c','LineWidth',0.5)

grid on

xlabel('Количество опытов')

ylabel('Частота')

Вспомогательная функция logzn:

function res = logzn(am,aM,x)

res = 0;

if (x<aM && x>am)

res = 1;

end;

end

Вспомогательная функция freqp:

function y = fregp(v,m)

k=0;

for i = 1:m

k = k+v(i);

y(i) = k/i;

end

end

Код программы для пункта 2:

n=1000; % Число испытаний

L=rand(4, n); % Задаём случайные данные для эксперимента

%LL=L(: , 1:10) их просмотр

% Преобразование в массив "0" и "1", для независимых событий

for i=1:n

A(i)=logzn(0.2, 0.7, L(1, i));

B(i)=logzn(0, 0.3, L(2, i));

C(i)=logzn(0.1, 0.5, L(3, i));

end;

% Расчёт вероятности включения лампочки для независимых событий

F=(~A&B | A&(~B)&C | B&~C); % Вычисление булевой функции срабатывания лампочки

PFN=mean(F) % Вероятность включения лампочки для независимых событий

% Графическое представление оценки вероятности включения лампочки для независимых событий

for j=1:n

QFN(j)=freqp(F, j); % Вычисление вектора частоты для независимых событий

end

figure;

plot(QFN); % Построение графика для независимых событий

grid on;

xlabel('The number of experiments');

ylabel('Frequency');

% Преобразование в массив "0" и "1", для зависимых событий

for i=1:n

A1(i)=logzn(0.2, 0.7, L(4, i));

B1(i)=logzn(0, 0.3, L(4, i));

C1(i)=logzn(0.1, 0.5, L(4, i));

end;

% Расчёт вероятности включения лампочки для зависимых событий

F1= (~A1&B1 | A1&(~B1)&C1 | B1&~C1); % Вычисление булевой функции срабатывания лампочки

PFZ=mean(F1) % Вероятность включения лампочки для зависимых событий

% Графическое представление оценки вероятности включения лампочки для зависимых событий

for j=1:n

QFZ(j)=freqp (F1, j); % Вычисление вектора частоты для зависимых событий

end

figure;

plot(QFZ); % Построение графика для зависимых событий

grid on;

xlabel('The number of experiments')

ylabel('Frequency')

Код программы для пункта 3:

randtool;%параметры распределения

MU = 0.5;

SIGMA = 0.25;

%размернсть

m = 1;

n = 1000;

fprintf('Мат. ожидание и дисперсия в логонормальном распределении при MU = 0.5 и sigma = 0.25 равны: \n');

[M,V] = lognstat(MU,SIGMA)

r = lognrnd(MU,SIGMA,m,n);

M1=mean(r);

fprintf('Математическое ожидание = %f\n',M1);

fprintf ('Центральный момент первого порядка случайной величины = ');

mu1 = helpfunc((r-M1),n);

disp(mu1);

fprintf ('Центральный момент второго порядка случайной величины = дисперсии = ');

mu2 = helpfunc(power((r-M1),2),n);

disp(mu2);

fprintf ('Центральный момент третьго порядка случайной величины =');

mu3 = helpfunc(power((r-M1),3),n);

disp(mu3);

fprintf ('Центральный момент четвертого порядка случайной величины = ');

mu4 = helpfunc(power((r-M1),4),n);

disp(mu4);

fprintf ('Оценка коэффициента асимметрии = ');

y1 = mu3/(sqrt(mu2^3));

disp(y1);

fprintf ('Оценка коэффициента эксцесса = ');

y2 = (mu4/(mu2^4))-3;

disp(y2);

for i=1:n

M(i)=helpfunc(r,i);

end;

figure

subplot(2,1,1);

plot(M), grid on

title('Зависимость оценок математического ожидания от числа испытаний');

xlabel('Число испытаний');

ylabel('Математическое ожидание');

subplot(2,1,2);

semilogx(M), grid on

xlabel('Число испытаний');

ylabel('Математическое ожидание');

for i=1:n

mu2(i)=helpfunc(power((r-M1),2),i);

end

figure

subplot(2,1,1);

plot(mu2), grid on

title('Зависимость оценок центрального момента второго порядка от числа испытаний');

xlabel('Число испытаний');

ylabel('Центральный момент');

subplot(2,1,2);

semilogx(mu2), grid on

xlabel('Число испытаний');

ylabel('Центральный момент');

for i=1:n

mu3(i)=helpfunc(power((r-M1),3),i);

end

figure

subplot(2,1,1);

plot(mu3), grid on

title('Зависимость оценок центрального момента третьего порядка от числа испытаний');

xlabel('Число испытаний');

ylabel('Центральный момент');

subplot(2,1,2);

semilogx(mu3), grid on

xlabel('Число испытаний');

ylabel('Центральный момент');

for i=1:n

mu4(i)=helpfunc(power((r-M1),4),i);

end

figure

subplot(2,1,1);

plot(mu4), grid on

title('Зависимость оценок центрального момента четвертого порядка от числа испытаний');

xlabel('Число испытаний');

ylabel('Центральный момент');

subplot(2,1,2);

semilogx(mu4), grid on

xlabel('Число испытаний');

ylabel('Центральный момент');

for i=1:n

y1(i)=helpfunc(power((r-M1),3),i)/((mu2(i)^3)^0.5);

end

figure

subplot(2,1,1);

plot(y1), grid on

title('Зависимость оценок коэффициента асимметрии от числа испытаний');

xlabel('Число испытаний');

ylabel('Коэффициент асимметрии');

subplot(2,1,2);

semilogx(y1), grid on

xlabel('Число испытаний');

ylabel('Коэффициент асимметрии');

for i=1:n

y2(i)=helpfunc(power((r-M1),4),i)/(power(mu2(i),4))-3;

end

figure

subplot(2,1,1);

plot(y2), grid on

title('Зависимость оценок коэффициента эксцесса от числа испытаний');

xlabel('Число испытаний');

ylabel('Коэффициент эксцесса');

subplot(2,1,2);

semilogx(y2), grid on

xlabel('Число испытаний');

ylabel('Коэффициент эксцесса');