

Министерство образования Российской Федерации  
Московский Государственный Технический  
Университет им. Н.Э. Баумана

Отчет по лабораторной работе №1  
По курсу «Математическая статистика»

**Тема: «Гистограмма и эмпирическая  
функция распределения»**

Вариант 11

Студент: **Медведев А.В.**  
Группа: **ИУ7-62**

Преподаватель: **Власов П.А.**

Москва, 2019

## Содержание

Формулы для вычисления	3
Определения	3
Текст программы	4
Результаты расчетов для выборки из индивидуального варианта	9

## Формулы для вычисления

Реализация случайной выборки:

$$(x_1, x_2, \dots, x_n)$$

Максимальное значение выборки:

$$M_{max} = \max(x_1, x_2, \dots, x_n)$$

Минимальное значение выборки:

$$M_{min} = \min(x_1, x_2, \dots, x_n)$$

Размах выборки:

$$R = M_{max} - M_{min}$$

Выборочное среднее:

$$\hat{\mu}(\vec{X}) = \bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

Выборочная дисперсия:

$$S^2(\vec{X}) = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$$

## Определения

**Эмпирическая плотность распределения**, соответствующая реализации  $\vec{x}_n$  случайной выборки  $\vec{X}_n$  - функция

$$f_n(x) = \begin{cases} \frac{n_i}{n\Delta} & , \text{ если } x \in J_i \\ 0 & , \text{ иначе} \end{cases}$$

где  $X_1$ - 1-й член вариационного ряда выборки,

$X_i$ -  $i$ -й член вариационного ряда выборки,

$\Delta = \frac{X_{(n)} - X_{(1)}}{m} = \frac{|J|}{m}$  - длина полуинтервала,

$m = [\log_2 n] + 2$  - количество полуинтервалов интервала  $J = [X_{(1)}; X_{(n)}]$ ,

$J_i = [X_{(min)} + (i-1)\Delta; X_{(min)} + i\Delta), i = \overline{1, m-1}$

$J_m = [X_{(min)} + (m-1)\Delta; X_{(min)} + m\Delta]$

$n_i$  = число элементов выборки принадлежащих  $J_i, i = \overline{1, n}$

**Гистограмма** - график функции  $f_n(x)$ , которая представляет собой кусочно-постоянную функцию

**Эмпирическая функция распределения**, построенная по выборке - отображение  $F_n : R \rightarrow R$ , определенное правилом

$$F_n(x) = \frac{n(x, \vec{x})}{n},$$

где  $n(x, \vec{x})$  - количество элементов выборки, которые меньше  $x$ , а  $n$  - объем этой выборки.

Если все элементы этой выбоки попарно различны, то

$$F_n(x) = \begin{cases} 0 & , x \leq x_{(1)} \\ \frac{i}{n} & , x \in (x_{(i)}; x_{(i+1)}] \\ 1 & , x > x_{(n)} \end{cases}$$

## Текст программы

Листинг 1: Функция Main

```
1
2 function main()
3     sample = importdata('data(var11).txt');
4     sample = sort(sample);
5
6     [Max] = getMaxValue(sample);
7     [Min] = getMinValue(sample);
8     [Range] = getRange(sample);
9     fprintf("min value: %.2f \n", Min);
10    fprintf("max value: %.2f \n", Max);
11    fprintf("range      : %.2f \n", Range);
12    fprintf("N          : %d \n", length(sample));
13
14    [MX] = getExpectedValue(sample);
15    [DX] = getDispersionValue(sample);
16
17    fprintf("MX          : %.3f \n", MX);
18    fprintf("DX          : %.3f \n", DX);
19
20    [GroupTable, m] = Group(sample);
21    for i=1:m-1
22        fprintf("[%5.2f;%5.2f] ", GroupTable(1,i),
23                GroupTable(1,i+1));
24
25    fprintf("[%5.2f;%5.2f]\n", GroupTable(1,m), Max);
26    for i=1:m
27        fprintf("%13d ", GroupTable(2,i));
28    end
29    fprintf("\n\n\n");
30
31    figure(1);
```

```

32     grid;
33     hold on;
34     HistogramAndDensity(sample)
35
36     figure(2);
37     grid;
38     hold on;
39     EmpiricalAndDensity(sample)
40 end

```

Листинг 2: Функции вычисления максимального и минимального значения

```

1 function [Max] = getMaxValue(sample)
2     Max = max(sample);
3 end
4
5 function [Min] = getMinValue(sample)
6     Min = min(sample);
7 end

```

Листинг 3: Функция вычисления размаха выборки

```

1 function [Range] = getRange(sample)
2     Range = getMaxValue(sample) - getMinValue(sample);
3 end

```

Листинг 4: Функции вычисления оценок

```

1 function [MX] = getExpectedValue(sample)
2     n = length(sample);
3     MX = sum(sample)/n;
4 end
5
6 function [DX] = getDispersionValue(sample)
7     n = length(sample);
8     MX = getExpectedValue(sample);
9     DX = sum((sample - MX).^2) / n;
10 end

```

Листинг 5: Группировка значений

```
1 function [GroupTable, m] = Group(sample)
2     n = length(sample);
3     m = floor(log2(n)) + 1;
4     fprintf("m          : %d \n", m);
5
6     GroupTable = zeros(2,m);
7     Delta = getRange(sample)/m;
8     fprintf("Delta      : %d \n", Delta);
9     for k = 0:m-1
10         GroupTable(1, k+1) = sample(1)+Delta*k;
11     end
12
13     count = 0;
14     for i = 1:n
15         for j = 1:m-1
16             if GroupTable(1,j) <= sample(i) && sample(i) <
17                 GroupTable(1, j+1)
18                 GroupTable(2, j) = GroupTable(2, j) + 1;
19                 count = count + 1;
20                 break;
21             end
22         end
23     end
24     GroupTable(2, m) = n - count;
25 end
```

Листинг 6: Гистограмма и график функции плотности

```

1 %
2 function [] = HistogramAndDensity(sample)
3     [min] = getMinValue(sample);
4     [max] = getMaxValue(sample);
5     count = length(sample);
6     Delta = (max-min)/(count-1);
7
8     Graph = zeros(2, count);
9     [MX] = getExpectedValue(sample);
10    [DX] = getDispersionValue(sample);
11
12    for i = 1:count
13        X = min + Delta*(i-1);
14        Graph(1,i) = X;
15        Graph(2,i) = NormalDensityDistribution(X, MX, DX);
16    end
17
18    [GroupTable, n] = Group(sample);
19    x = zeros(n+1);
20    y = zeros(n+1);
21    znam = length(sample)*getRange(sample)/n;
22    for i = 1:n
23        x(i) = GroupTable(1,i);
24        y(i) = GroupTable(2,i) ./ znam;
25    %     fprintf("%f %f \n", x(i), y(i));
26    end
27    x(n+1) = max;
28    y(n+1) = y(n);
29
30    %
31    stairs(x, y), grid;
32    %
33    plot(Graph(1,:), Graph(2,:), 'r'), grid;
34
35 end
36
37 function [y] = NormalDensityDistribution(x, mx, dx)
38     y = exp(-((x-mx).^2)/2/dx)/sqrt(2*pi*dx);
39 end

```

Листинг 7: Графики эмпирической функции и функции распределения случайной величины

```

1 %
2 function EmpiricalAndDensity(sample)
3     [min] = getMinValue(sample);
4     [max] = getMaxValue(sample);
5     count = length(sample);
6     Delta = (max-min)/(count-1);
7
8     Graph = zeros(2,count);
9     [MX] = getExpectedValue(sample);
10    [DX] = getDispersionValue(sample);
11
12    syms t;
13    for i = 1:count
14        X = min + Delta*(i-1);
15        Graph(1,i) = X;
16        Graph(2,i) = NormalDistribution(X, MX, DX,t);
17        %fprintf("%f %f \n", Graph(1,i), Graph(2,i));
18    end
19
20
21    F = zeros(count);
22    for i = 1:count
23        F(i) = EmpiricFunc(sample(i), sample, count);
24        %fprintf("EmpiricFunc %f %f \n", sample(i), F(i));
25    end
26    %
27    stairs(sample, F),grid;
28    %
29
30    plot(Graph(1,:), Graph(2,:), 'r'),grid;
31 end
32
33 function [y] = NormalDistribution(x,mx,dx,t)
34     %syms t;
35     y = 1/sqrt(2*pi*dx) * int( exp(-((t-mx).^2)/2/dx), t, -
36         Inf, x);
37 end
38
39 function [Fi] = EmpiricFunc(x, sample, n)

```



```

38     count = 0;
39     for i = 1:n
40         if (sample(i) <= x)
41             count = count + 1;
42         else
43             continue;
44         end
45     end
46     Fi = count/n;
47 end

```

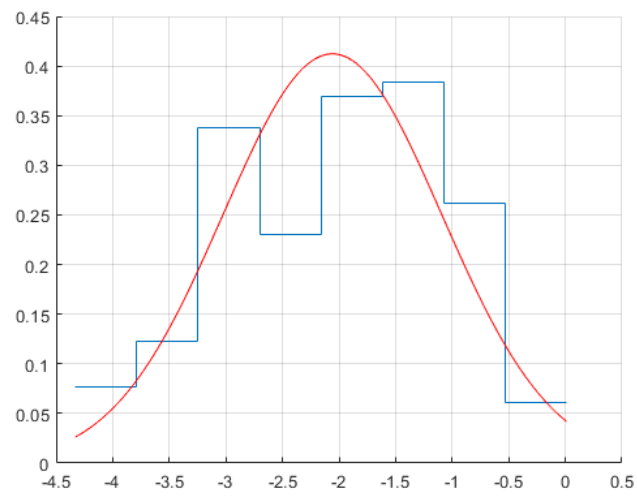
## Результаты расчетов для выборки из индивидуального варианта

$M_{min}$	-4.33
$M_{max}$	0.01
R	4.34
N	120
MX	-2.058
DX	0.936

Группировка значений выборки в m=8 интервалов

[-4.33;-3.79)	[-3.79;-3.25)	[-3.25;-2.70)	[-2.70;-2.16)
5	8	22	15
[-2.16;-1.62)	[-1.62;-1.08)	[-1.08; -0.53]	[-0.53; 0.01]
24	25	17	4

Построение на одной координатной плоскости гистограммы и графика функции плотности распределения вероятностей нормальной случайной величины с математическим ожиданием  $\hat{\mu}$  и дисперсией  $S^2$ :



Построение на другой координатной плоскости графика эмпирической функции распределения и функции распределения нормальной случайной величины с математическим ожиданием  $\hat{\mu}$  и дисперсией  $S^2$ :

