



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ «Информатика и системы управления»  
КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

---

## Отчёт

### по лабораторной работе №2

Название «Интервальные оценки»

---

Дисциплина «Математическая статистика»

---

Студент ИУ7-65Б

---

\_\_\_\_\_  
(подпись, дата)

Бугаенко А.П.  
(Фамилия И.О.)

Преподаватель

\_\_\_\_\_  
(подпись, дата)

Андреева Т.В.  
(Фамилия И.О.)

Москва, 2022

## 1 Цели и задачи работы

Цель работы — построение доверительных интервалов для математического ожидания и дисперсии нормальной случайной величины. Содержание работы:

1) Для выборки объёма  $n$  из генеральной совокупности  $X$  реализовать в виде программы на ЭВМ:

- а) вычисление точечных оценок  $\hat{\mu}(\vec{x}_n)$  и  $S^2(\vec{x}_n)$  математического ожидания  $MX$  и дисперсии  $DX$  соответственно;
- б) вычисление нижней и верхней границ  $\underline{\mu}(\vec{x}_n)$ ,  $\bar{\mu}(\vec{x}_n)$  для  $\gamma$ -доверительного интервала для математического ожидания  $MX$ ;
- в) вычисление нижней и верхней границ  $\underline{\sigma}^2(\vec{x}_n)$ ,  $\bar{\sigma}^2(\vec{x}_n)$  для  $\gamma$ -доверительного интервала для дисперсии  $DX$ ;

2) вычислить  $\hat{\mu}$ ,  $S^2$  из индивидуального варианта;

3) для заданного пользователем уровня доверия  $\gamma$  и  $N$  - объёма выборки из индивидуального варианта:

а) на координатной плоскости  $Oxy$  построить прямую  $y = \hat{\mu}(\vec{x}_N)$ , также графики функций  $y = \hat{\mu}(\vec{x}_n)$ ,  $y = \underline{\mu}(\vec{x}_n)$ ,  $y = \bar{\mu}(\vec{x}_n)$  как функций объёма  $n$  выборки, где  $n$  изменяется от 1 до  $N$ .

б) на другой координатной плоскости  $Ozn$  построить прямую  $z = S^2(\vec{x}_N)$ , также графики функций  $z = S^2(\vec{x}_n)$ ,  $z = \underline{\sigma}^2(\vec{x}_n)$  и  $z = \bar{\sigma}^2(\vec{x}_n)$  как функций объёма  $n$  выборки, где  $n$  изменяется от 1 до  $N$ .

## 2 Теоретическая часть

### 2.1 Формулы для вычисления точечных оценок математического ожидания и дисперсии

Пусть  $(x_1, \dots, x_n)$  - некая случайная выборка.

Оценка  $\hat{\mu}$  ( $\mu_{\text{X}}$ ) для математического ожидания  $\text{MX}$  вычисляется как  $\hat{\mu}(\vec{X}_n) = \overline{X_n} = \frac{1}{n} \sum_{i=1}^n X_i$ .

Оценка  $S^2$  ( $S_{\text{2-X}}$ ) для дисперсии (несмещённая) вычисляется как  $S^2(\vec{X}_n) = \frac{1}{n-1} \sum_{i=1}^n (X_i - \overline{X_n})^2$ .

### 2.2 Определение $\gamma$ -доверительного интервала для значения параметра распределения случайной величины

Пусть  $\vec{X}_n$  - случайная выборка  $n$  из генеральной совокупности  $X$  с функцией распределения  $F(x; \theta)$ , зависящей от параметра  $\theta$ , значение которого неизвестно.

Тогда интервальной оценкой с коэффициентов доверия  $\gamma$  ( $\gamma$ -доверительной интервальной оценкой) параметра  $\theta$  называют пару статистик  $\underline{\theta}(\vec{X}_n), \overline{\theta}(\vec{X}_n)$  таких, что:  
 $P\{\underline{\theta}(\vec{X}_n) < \theta < \overline{\theta}(\vec{X}_n)\} = \gamma$

Интервал  $(\underline{\theta}(\vec{X}_n), \overline{\theta}(\vec{X}_n))$  называют интервальной оценкой для параметра  $\theta$  с коэффициентов доверия  $\gamma$ , а  $\underline{\theta}(\vec{X}_n), \overline{\theta}(\vec{X}_n)$  называют соответственно нижней и верхней границами интервальной оценки. Смысл интервальной оценки состоит в том, что она представляет интервал со случайными границами, который с заданной вероятностью  $\gamma$  накрывает неизвестное истинное значение параметра  $\gamma$ .

Интервал  $(\underline{\theta}(\vec{X}_n), \overline{\theta}(\vec{X}_n))$  называют доверительным интервалом для параметра с коэффициентов доверия  $\gamma$ , где  $\vec{x}_n$  - любая реализация случайной выборки  $\vec{X}_n$ .

### 2.3 Формулы для вычисления границ $\gamma$ -доверительного интервала для математического ожидания и дисперсии

Для вычисления верхней и нижней границы  $\gamma$ -доверительного интервала для математического ожидания используются следующие формулы:

$$\underline{\mu}(\vec{X}_n) = \overline{X} - \frac{S(\vec{X})}{\sqrt{n}} t_{1-\alpha}(n-1) \text{ (u\_low)}$$

$$\overline{\mu}(\vec{X}_n) = \overline{X} + \frac{S(\vec{X})}{\sqrt{n}} t_{1-\alpha}(n-1) \text{ (u\_high)}$$

$\overline{X}$  - точечная оценка математического ожидания

$S^2(\vec{X})$  - точечная оценка дисперсии

$n$  - объём выборки

$\gamma$  - уровень доверия

$t_{1-\alpha}$  - квантиль уровня  $1 - \alpha$  для распределения Стьюдента с  $n - 1$  степенями свободы,  $\alpha = \frac{1-\gamma}{2}$

Для вычисления верхней и нижней границы  $\gamma$ -доверительного интервала для дисперсии:

$$\underline{\sigma}^2(\vec{x}_n) = \frac{(n-1)S^2(\vec{X})}{\chi_{1-\alpha}^2(n-1)} \text{ (sigma\_2\_low)}$$

$$\overline{\sigma}^2(\vec{x}_n) = \frac{(n-1)S^2(\vec{X})}{\chi_{\alpha}^2(n-1)} \text{ (sigma\_2\_high)}$$

$S^2(\vec{X})$  - точечная оценка дисперсии

$n$  - объём выборки

$\gamma$  - уровень доверия

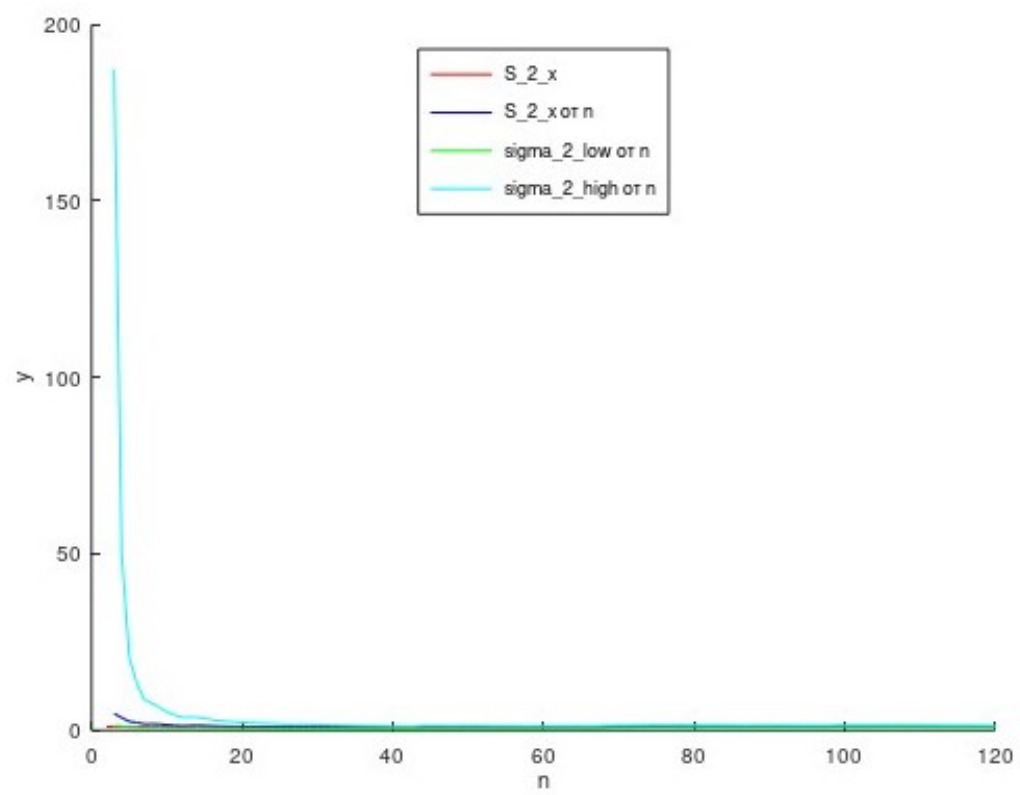
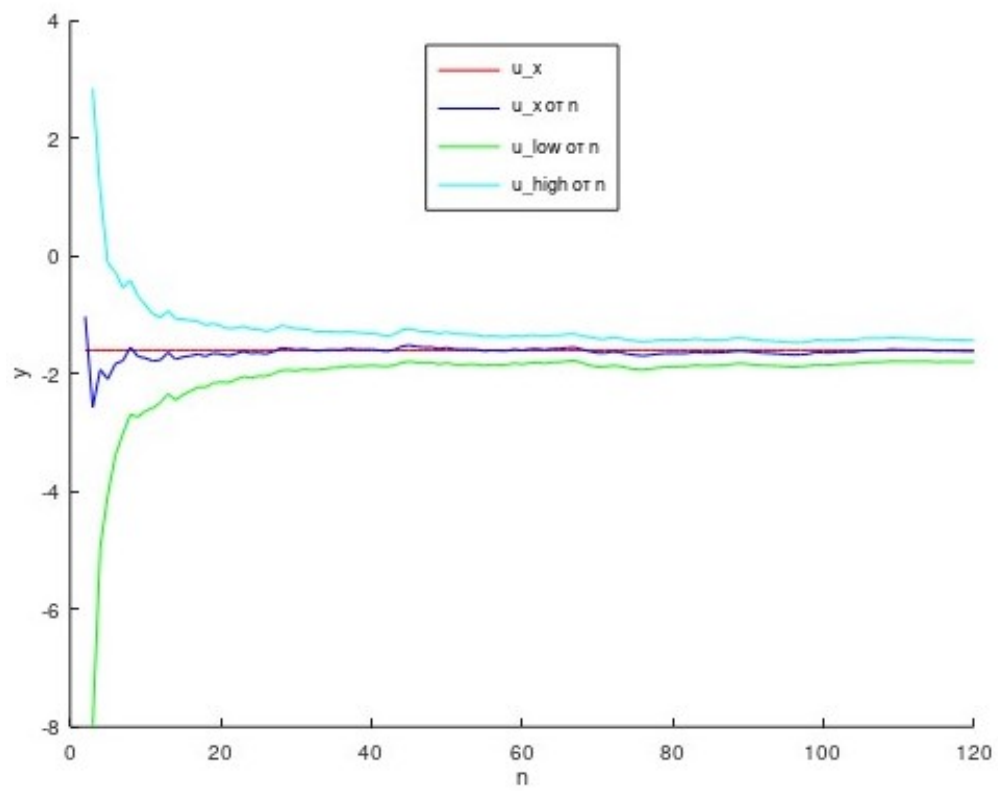
$\chi_{1-\alpha}(n-1)$  - квантиль уровня  $1-\alpha$  для распределения Стьюдента с  $n-1$  степенями свободы,

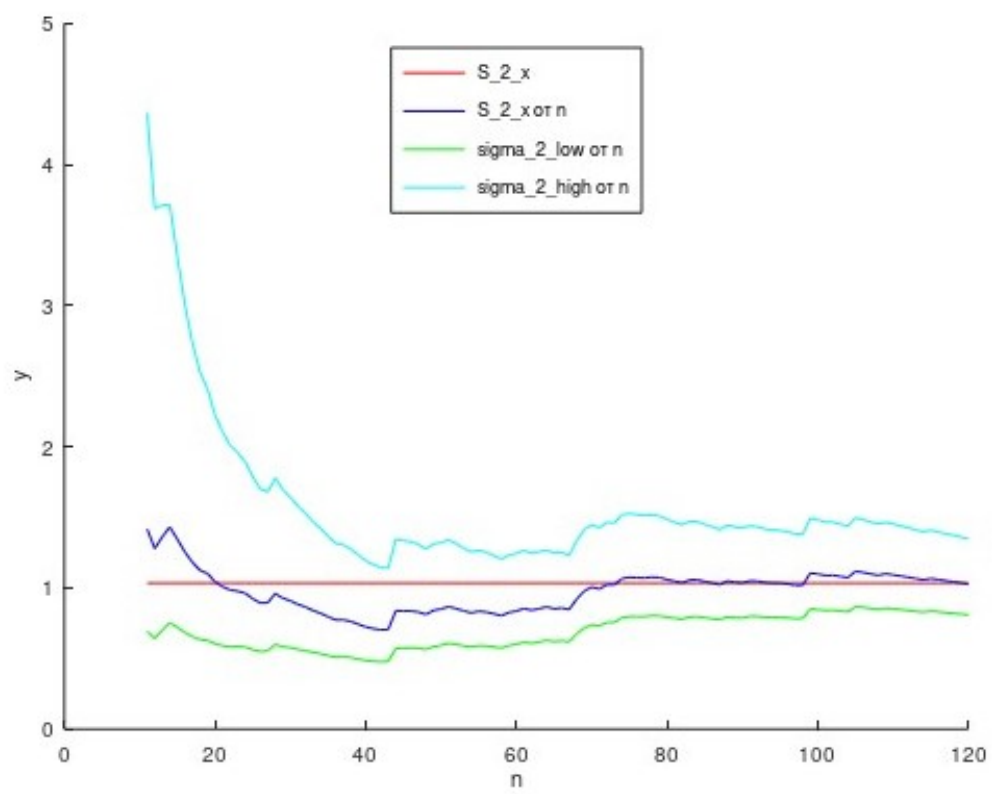
$$\alpha = \frac{1-\gamma}{2}$$

### 3 Практическая часть

#### 3.1 Результаты работы для выборки по варианту

```
1 Lab 2
2 n = 120
3 u_x = -1.6046
4 S_2_x = 1.0341
5 u_low = -1.7585
6 u_high = -1.4507
7 sigma_2_low = 0.8460
8 sigma_2_high = 1.2979
```





### 3.2 Листинг программы

```
1 disp("Lab 2")
2 pkg load statistics
3
4 # Выборка
5 x_unsorted =
    [-0.23,-1.03,-4.11,-0.65,-2.58,-0.79,-1.53,-0.18,-2.79,-1.97,-2.21,-1.59,-0.22,-3.18,-
6
7 # Сортировка массива данных
8 x = sort(x_unsorted);
9
10 # Нахождение длины массива
11 n = length(x)
12
13 # Оценка матожидания
14 function res = u_x_f(x)
15     res = (1/numel(x))*sum(x);
16 endfunction
17 u_x = u_x_f(x)
18
19 # Несмещённая оценка дисперсии
20 function res = S_2_x_f(x, u_x)
21     val = 0;
22     for i = 1:numel(x)
23         val +=(x(i) - u_x)*(x(i) - u_x);
24     endfor
25     res = val*(1/(numel(x) - 1));
26 endfunction
27 S_2_x = S_2_x_f(x, u_x)
28
29 gamma = 0.95;
30
31 # Нижняя граница доверительного интервала для матожидания
32 # tinv - функция распределения инверсии t студента
33 function res = u_low_f(u_x, S_2_x, n, gamma)
34     alpha = (1 - gamma)/2;
35     res = u_x - sqrt(S_2_x) / sqrt(n) * tinv(1 - alpha, n - 1);
36 endfunction
37
38 # Верхняя граница доверительного интервала для матожидания
39 function res = u_high_f(u_x, S_2_x, n, gamma)
40     alpha = (1 - gamma)/2;
41     res = u_x + sqrt(S_2_x) / sqrt(n) * tinv(1 - alpha, n - 1);
42 endfunction
43
44 u_low = u_low_f(u_x, S_2_x, n, gamma)
```



```

45 u_high = u_high_f(u_x, S_2_x, n, gamma)
46
47 # Нижняя граница доверительного интервала для дисперсии
48 function res = sigma_2_low_f(S_2_x, gamma, n)
49     alpha = (1 - gamma)/2;
50     res = ((n - 1) * S_2_x) / chi2inv(1 - alpha, n - 1);
51 endfunction
52
53 # Верхняя граница доверительного интервала для дисперсии
54 function res = sigma_2_high_f(S_2_x, gamma, n)
55     alpha = (1 - gamma)/2;
56     res = ((n - 1) * S_2_x)/chi2inv(alpha, n - 1);
57 endfunction
58
59 sigma_2_low = sigma_2_low_f(S_2_x, gamma, n)
60 sigma_2_high = sigma_2_high_f(S_2_x, gamma, n)
61
62 x = x_unsorted;
63 y = [];
64 # Построение графиков
65 figure();
66 hold on;
67 x_n = 2:1:n;
68 for i = 1:numel(x_n)
69     y(i) = u_x_f(x);
70 endfor
71
72 plot(x_n, y, 'r');
73 xlabel('n')
74 ylabel('y')
75
76 x_n = 2:1:n;
77 for i = 2:numel(x)
78     y(i - 1) = u_x_f(x(2:i));
79 endfor
80 plot(x_n, y, 'b');
81 xlabel('n')
82 ylabel('y')
83
84 for i = 2:numel(x)
85     u_x = u_x_f(x(2:i));
86     S_2_x = S_2_x_f(x(2:i), u_x);
87     y(i - 1) = u_low_f(u_x, S_2_x, i, gamma);
88 endfor
89
90 plot(x_n, y, 'g');
91 xlabel('n')

```

```

92 ylabel('y')
93
94 for i = 2:numel(x)
95     u_x = u_x_f(x(2:i));
96     S_2_x = S_2_x_f(x(2:i), u_x);
97     y(i - 1) = u_high_f(u_x, S_2_x, i, gamma);
98 endfor
99
100 plot(x_n, y, 'c');
101 xlabel('n')
102 ylabel('y')
103 legend ({'u\\_x', 'u\\_x от n', 'u\\_low от n', 'u\\_high от n'}, "location",
         "north");
104 hold off;
105
106 # Графики для дисперсии
107 figure();
108 hold on;
109 x_n = 2:1:n;
110 for i = 2:numel(x)
111     y(i - 1) = S_2_x_f(x, u_x);
112 endfor
113 plot(x_n, y, 'r');
114 xlabel('n')
115 ylabel('y')
116
117 x_n = 2:1:n;
118 for i = 2:numel(x)
119     u_x = u_x_f(x(2:i));
120     y(i - 1) = S_2_x_f(x(2:i), u_x);
121 endfor
122 plot(x_n, y, 'b');
123 xlabel('n')
124 ylabel('y')
125
126 for i = 2:numel(x)
127     u_x = u_x_f(x(2:i));
128     S_2_x = S_2_x_f(x(2:i), u_x);
129     y(i - 1) = sigma_2_low_f(S_2_x, gamma, i);
130 endfor
131
132 plot(x_n, y, 'g');
133 xlabel('n')
134 ylabel('y')
135
136 for i = 2:numel(x)
137     u_x = u_x_f(x(2:i));

```

```

138     S_2_x = S_2_x_f(x(2:i), u_x);
139     y(i - 1) = sigma_2_high_f(S_2_x, gamma, i);
140 endfor
141
142 plot(x_n, y, 'c');
143 xlabel('n')
144 ylabel('y')
145 legend ({ "S\\_2\\_x", "S\\_2\\_x от n", "sigma\\_2\\_low от n", "sigma\\_2\\_high от
        n"}, "location", "north");
146 hold off;
147
148 # Графики для дисперсии с n = 10
149 figure();
150 hold on;
151 x_n = 2:1:n;
152 for i = 2:numel(x)
153     y(i - 1) = S_2_x_f(x, u_x);
154 endfor
155 plot(x_n(10:numel(x_n)), y(10:numel(y)), 'r');
156 xlabel('n')
157 ylabel('y')
158
159 x_n = 2:1:n;
160 for i = 2:numel(x)
161     u_x = u_x_f(x(2:i));
162     y(i - 1) = S_2_x_f(x(2:i), u_x);
163 endfor
164 plot(x_n(10:numel(x_n)), y(10:numel(y)), 'b');
165 xlabel('n')
166 ylabel('y')
167
168 for i = 2:numel(x)
169     u_x = u_x_f(x(2:i));
170     S_2_x = S_2_x_f(x(2:i), u_x);
171     y(i - 1) = sigma_2_low_f(S_2_x, gamma, i);
172 endfor
173
174 plot(x_n(10:numel(x_n)), y(10:numel(y)), 'g');
175 xlabel('n')
176 ylabel('y')
177
178 for i = 2:numel(x)
179     u_x = u_x_f(x(2:i));
180     S_2_x = S_2_x_f(x(2:i), u_x);
181     y(i - 1) = sigma_2_high_f(S_2_x, gamma, i);
182 endfor
183

```

```

184 plot(x_n(10:numel(x_n)), y(10:numel(y)), 'c');
185 xlabel('n')
186 ylabel('y')
187 legend ({ "S\\_2\\_x", "S\\_2\\_x or n", "sigma\\_2\\_low or n", "sigma\\_2\\_high or
           n"}, "location", "north");
188 hold off;

```