|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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»\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Лабораторная работа №2*

*По предмету: «Математическая статистика»*

***Тема: «*Интервальные оценки*»***

Преподаватели: Волков И. К.,

Власов П. А.,

Студент: Мирзоян С.А.,

Группа: ИУ7-65Б

Вариант: 12

Москва, 2020 г.

**Введение**

**Цель** **работы:** построение доверительных интервалов для математическо-го ожидания и дисперсии нормальной случайной величины.

**Содержание** **работы:**

1. Для выборки объема *n* из нормальной генеральной совокупности *X* реализовать в виде программы на ЭВМ

a. вычисление точечных оценок и

математического ожидания *MX* и дисперсии *DX* соответственно;

b. вычисление нижней и верхней границ , для *γ*-доверительного интервала для математического ожидания *MX*;

c. вычисление нижней и верхней границ для *γ*-доверительного интервала для математического ожидания *DX*;

2. вычислить *μ* и *S*2 для выборки из индивидуального варианта;

3. для заданного пользователем уровня доверия *γ* и *N* – объема выборки из индивидуального варианта:

a. на координатной плоскости построить прямую *y* = *μ*( *xN* ), также графики функций *y* = *μ*(*xn*), *y* = *μ*(*xn*) и *y* = *μ*(*xn*) как функций объема *n* выборки, где *n* изменяется от 1 до *N*;

b. на другой координатной плоскости *Ozn* построить прямую *z* = *S*2( *xN* ), также графики функций *z* = *S*2(*xn*), *z* = *σ*2(*xn*) и

*z* = *σ*2(*xn*) как функций объема *n* выборки, где *n* изменяется от 1 до *N*.

**Теоретическая** **часть**

В данной части рассмотрены формулы для вычисления величин, границ *γ*-доверительного интервала, а также определение *γ*-доверительного интервала.

**Определение** *γ***-доверительного** **интервала** **для** **значения** **параметра** **распределения** **случайной** **величины**

**Интервальной** **оценкой** **с** **коэффициентом** **доверия** *γ* (*γ*-доверительной

интервальной оценкой) параметра *θ* называют пару статистик *θ*() и *θ*() таких, что

(1)

**1.2** **Формулы** **для** **вычисления** **величин**

**Оценка** **математического** **ожидания:**

**Несмещённая** **оценка** **дисперсии:**

**Выборочная** **дисперсия:**

**1.3** **Формулы** **для** **вычисления** **границ** *γ***-доверительного** **интервала**

Пусть *Xn* — случайная выборка объема *n* из генеральной совокупности *X*, распределенной по нормальному закону с параметрами *μ* и *σ*2.

**Оценка** **для** **математического** **ожидания** **при** **известной** **дисперсии**

где *Xn* — оценка математического ожидания, n — объем выборки,

*S*(*Xn*) — точечная оценка дисперсии случайной выборки *Xn*,

*hα*(*n* − 1) — квантиль уровня *α* распределения Стьюдента с *n* − 1 степенями свободы.

*t*

**Оценка** **для** **дисперсии**

где n — объем выборки,*S*(*Xn*) — точечная оценка дисперсии случайной выборки *Xn*,

*χα*(*n* − 1) — квантиль уровня *α* для распределения *χ*2 с *n* − 1 степенями свободы.

2

**Листинг**

1. function lab2()
2. clear all;
4. **%**Мат. ожидание
5. function mu **=** expect(X)
6. mu **=** mean(X); **%**mean **-** арифмитичекое среднее элементов массива
7. end
8. **%**Дисперсия
9. function sSqr **=** variance(X)
10. sSqr **=** var(X); **%**var **-** возвращает дисперсию ээлементов массива
11. end
12. **%**Массив мат. ожиданий
13. function muArray **=** expectMas(X, N)
14. muArray **=** [];
15. **for** i **=** N
16. muArray **=** [muArray, mean(X(1:i))];
17. end
18. end
19. **%**Массив дисперсий
20. function varArray **=** varianceArray(X, N)
21. varArray **=** [];
22. **for** i **=** N
23. varArray **=** [varArray, var(X(1:i))];
24. end
25. end
27. X **=** [11.89,9.60,9.29,10.06,9.50,8.93,9.58,6.81,8.69,9.62,9.01,10.59,10.50,11.53,9.94,8.84,8.91,6.90,9.76,7.09,11.29,11.25,10.84,10.76,7.42,8.49,10.10,8.79,11.87,8.77,9.43,12.41,9.75,8.53,9.72,9.45,7.20,9.23,8.93,9.15,10.19,9.57,11.09,9.97,8.81,10.73,9.57,8.53,9.21,10.08,9.10,11.03,10.10,9.47,9.72,9.60,8.21,7.78,10.21,8.99,9.14,8.60,9.14,10.95,9.33,9.98,9.09,10.35,8.61,9.35,10.04,7.85,9.64,9.99,9.65,10.89,9.08,8.60,7.56,9.27,10.33,10.09,8.51,9.86,9.24,9.63,8.67,8.85,11.57,9.85,9.27,9.69,10.90,8.84,11.10,8.19,9.26,9.93,10.15,8.42,9.36,9.93,9.11,9.07,7.21,8.22,9.08,8.88,8.71,9.93,12.04,10.41,10.80,7.17,9.00,9.46,10.42,10.43,8.38,9.01];
29. N **=** 10:100;
30. **%**N **=** 1:40
32. gamma **=** 0.99;
33. alpha **=** (1 **-** gamma)**/**2;
35. **%**Подсчет мат.ожидания и дисперсии
36. mu **=** expect(X);
37. sSqr **=** variance(X);
39. muArray **=** expectMas(X, N);
40. varArray **=** varianceArray(X, N);
42. figure
43. plot([N(1), N(end)], [mu, mu], 'm');
44. hold on;
45. plot(N, muArray, 'g');
47. **%**tinv **-** возвращает обратную кумулятивную функцию распределения
48. **%**(icdf) распределения t Студента
49. Ml **=** muArray **-** sqrt(varArray.**/**N).**\***tinv(1 **-** alpha, N **-** 1);
50. Mh **=** muArray **+** sqrt(varArray.**/**N).**\***tinv(1 **-** alpha, N **-** 1);
52. plot(N, Ml, 'b');
53. plot(N, Mh, 'r');
54. grid on;
55. hold off;
57. fprintf('µ = %.2f\n', mu);
58. fprintf('S^2 = %.2f\n\n', sSqr);
59. fprintf('µ\_low = %.2f\n', Ml(end));
60. fprintf('µ\_high = %.2f\n', Mh(end));

63. figure
64. plot([N(1), N(end)], [sSqr, sSqr], 'm');
65. hold on;
67. **%**chi2inv **-** возвращает обратную кумулятивную функцию распределения
68. **%**(icdf) распределения хи**-**квадрат со степенями свободы nu
69. Sl **=** varArray.**\***(N **-** 1).**/**chi2inv(1 **-** alpha, N **-** 1);
70. Sh **=** varArray.**\***(N **-** 1).**/**chi2inv(alpha, N **-** 1);
71. plot(N, varArray, 'g');
72. plot(N, Sl, 'b');
73. plot(N, Sh, 'r');
74. grid on;
75. hold off;
76. fprintf('sigma^2\_low = %.2f\n', Sl(end));
77. fprintf('sigma^2\_high = %.2f\n', Sh(end));
78. end

**Результат работы программы**

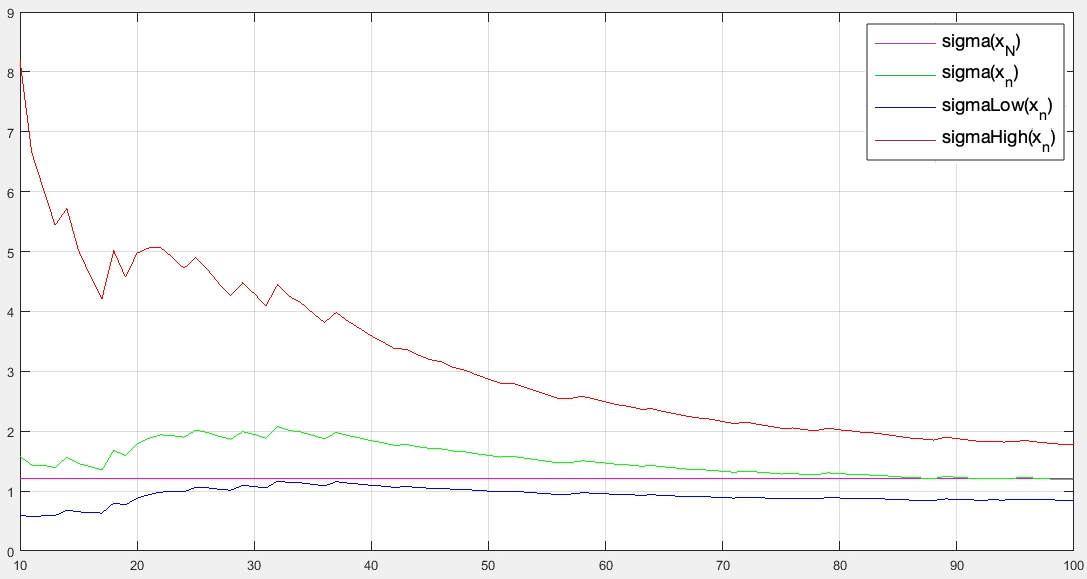
****

Рис.1 Оценка для дисперсии

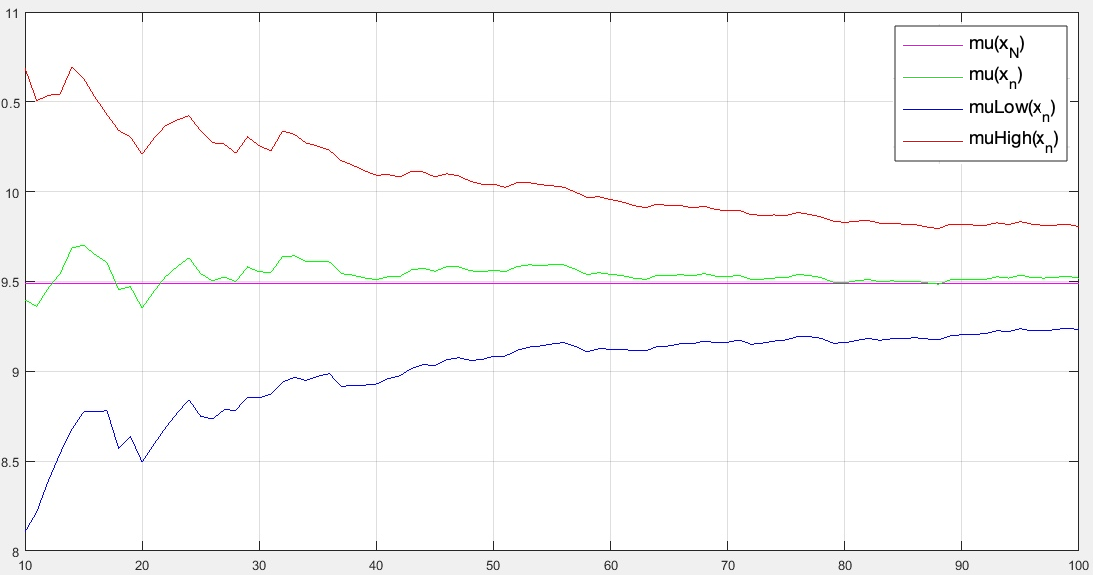


Рис. 2 Оценка для математического ожидания