

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. ПЕТРА ВЕЛИКОГО

ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ И МЕХАНИКИ

КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ

ЛАБОРАТОРНАЯ РАБОТА №2
ХАРАКТЕРИСТИКИ ПОЛОЖЕНИЯ
ВЫБОРКИ

3 КУРС, ГРУППА 3630102/70301

Студент

Лебедев К.С.

Преподаватель

Баженов А. Н.

САНКТ-ПЕТЕРБУРГ
2020 г.

Содержание

1. Постановка задачи	3
2. Теория	3
3. Реализация	3
4. Результаты	4
5. Выводы	6
6. Список литературы	6

1 Постановка задачи

Сгенерировать выборки размеров 20, 60, 100 элементов для 5ти распределений. Для каждой выборки вычислить \bar{x} , $med\ x$, Z_R , Z_Q , Z_{tr} , при $r = \frac{n}{4}$.

Распределения:

- Стандартное нормальное распределение
- Стандартное распределение Коши
- Распределение Лапласа
- Равномерное распределение
- Распределение Пуассона

2 Теория

1. Выборочное среднее [3]:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (1)$$

2. Выборочная медиана [4]:

$$med\ x = \begin{cases} x_{k+1}, & n = 2k + 1 \\ \frac{1}{2}(x_k + x_{k+1}), & n = 2k \end{cases} \quad (2)$$

3. Полусумма экстремальных значений [5]:

$$Z_R = \frac{1}{2}(x_1 + x_n) \quad (3)$$

4. Полусумма квантилей [6]:

$$Z_Q = \frac{1}{2}\left(Z_{\frac{1}{4}} + Z_{\frac{3}{4}}\right) \quad (4)$$

5. Усечённое среднее [7]:

$$Z_{tr} = \frac{1}{n - 2r} \sum_{i=r+1}^{n-r} x_i \quad (5)$$

3 Реализация

Для генерации выборки был использован *Python 3.6*: модуль *random* библиотеки *numpy* [1] для генерации случайных чисел с различными распределениями. Также с помощью библиотеки *numpy* были вычислены характеристики положения.

После вычисления характеристик положения 1000 раз находится среднее значение и дисперсия:

$$E(z) = \frac{1}{n} \sum_{i=1}^n z_i \quad (6)$$

$$D(z) = E(z^2) - E^2(z) \quad (7)$$

4 Результаты

n = 20	average	med	Zr	Zq	Ztr r = n/4
E =	-0.006700	0.018546	0.012695	0.000594	0.015402
D =	0.049919	0.072927	0.147937	0.057251	0.057085
n = 60	average	med	Zr	Zq	Ztr r = n/4
E =	0.001934	0.001927	0.001174	0.003355	-0.001168
D =	0.017054	0.024812	0.103528	0.020604	0.018967
n = 100	average	med	Zr	Zq	Ztr r = n/4
E =	0.000708	0.000648	0.010756	0.001811	-0.003775
D =	0.009767	0.015795	0.097672	0.012952	0.012131

Нормальное распределение

n = 20	average	med	Zr	Zq	Ztr r = n/4
E =	0.006851	0.004852	0.008633	0.000993	0.002342
D =	0.054811	0.034301	0.409060	0.044605	0.034997
n = 60	average	med	Zr	Zq	Ztr r = n/4
E =	0.003625	-0.002591	-0.003029	-0.003954	-0.003168
D =	0.016641	0.010034	0.417957	0.016484	0.010047
n = 100	average	med	Zr	Zq	Ztr r = n/4
E =	-0.001436	-0.002020	0.035100	0.006703	0.001789
D =	0.010006	0.006088	0.424029	0.009739	0.005931

Распределение Лапласа

n = 20	average	med	Zr	Zq	Ztr r = n/4
E =	22.758779	-0.006740	78.929010	-0.002305	0.013057
D =	774990.43	0.141492	2423114.9	0.318424	0.178163
n = 60	average	med	Zr	Zq	Ztr r = n/4
E =	0.464150	-0.005508	16.718463	-0.005078	0.007213
D =	599.91366	0.043940	410697.47	0.090639	0.046947
n = 100	average	med	Zr	Zq	Ztr r = n/4
E =	1.667554	-0.003077	128.15939	0.001487	-0.013537
D =	5879.3866	0.024712	9178141.9	0.053342	0.027383

Распределение Коши

n = 20	average	med	Zr	Zq	Ztr r = n/4
E =	1.987600	1.870500	2.515500	1.912250	1.868300
D =	0.099786	0.178480	0.303010	0.121644	0.129565
n = 60	average	med	Zr	Zq	Ztr r = n/4
E =	1.994817	1.936000	2.968500	1.954750	1.828900
D =	0.032801	0.051904	0.241258	0.031765	0.045717
n = 100	average	med	Zr	Zq	Ztr r = n/4
E =	1.996890	1.968500	3.168000	1.971250	1.831700
D =	0.021158	0.027758	0.280776	0.012861	0.026473

Распределение Пуассона

n = 20	average	med	Zr	Zq	Ztr r = n/4
E =	0.012552	-0.006810	-0.000923	0.004960	0.009501
D =	0.049610	0.131448	0.013292	0.076374	0.086512
n = 60	average	med	Zr	Zq	Ztr r = n/4
E =	-0.003837	-0.007174	-0.000018	-0.010948	-0.008353
D =	0.017882	0.047638	0.001874	0.023592	0.031756
n = 100	average	med	Zr	Zq	Ztr r = n/4
E =	-0.000175	-0.000765	-0.001623	-0.002510	0.002546
D =	0.009693	0.029956	0.000587	0.014596	0.019309

Равномерное распределение

5 Выводы

В процессе работы вычислены значения характеристик положения для определённых распределений на выборках разной мощности и получено следующее ранжирование характеристик положения:

1. Стандартное нормальное распределение

$$\bar{x} < Z_{tr} < Z_Q < med\ x < Z_R$$

2. Стандартное распределение Коши

$$med\ x < Z_Q < Z_{tr} < \bar{x} < Z_R$$

3. Распределение Лапласа (коэффициент масштаба $\sqrt{2}$ коэффициент сдвига равен нулю)

$$med\ x < Z_{tr} < \bar{x} < Z_Q < Z_R$$

4. Равномерное распределение на отрезке $[-\sqrt{3}, \sqrt{3}]$

$$Z_R < \bar{x} < Z_{tr} < Z_Q < med\ x$$

5. Распределение Пуассона (значение мат ожидания равно 3)

$$\bar{x} < Z_{tr} < Z_Q < med\ x < Z_R$$

6 Список литературы

- [1] Модуль numpy - <https://physics.susu.ru/vorontsov/language/numpy.html>
- [2] Формулы распределений - https://vk.com/doc184549949_491827451

[3] Выборочное среднее - https://en.wikipedia.org/wiki/Sample_mean_and_covariance

[4] Выборочная медиана - http://femto.com.ua/articles/part_1/2194.html

[5] Полусумма экстремальных значений - <https://studopedia.info/8-56888.html>

[6] Квартили - <https://studfiles.net/preview/2438125/page:13/>

[7] Усечённое среднее - <https://ole-olesko.livejournal.com/15773.html>

```
1
2 import numpy
3 import sys
4
5
6 POISSON_PARAM = 2
7 UNIFORM_LEFT = -numpy.sqrt(3)
8 UNIFORM_RIGHT = numpy.sqrt(3)
9 LAPLAS_COEF = numpy.sqrt(2)
10 selection = [20, 60, 100]
11 selection = numpy.sort(selection)
12
13 def standart_normal(x):
14     return (1 / numpy.sqrt(2*numpy.pi)) * numpy.exp(- x * x / 2)
15
16
17 def standart_cauchy(x):
18     return 1 / (numpy.pi * (1 + x*x))
19
20
21 def laplace(x):
22     return 1 / LAPLAS_COEF * numpy.exp(-LAPLAS_COEF * numpy.abs(x))
23
24
25 def uniform(x):
26     flag2 = x <= UNIFORM_RIGHT
27     flag1 = x >= UNIFORM_LEFT
28     return 1 / (UNIFORM_RIGHT - UNIFORM_LEFT) * flag1 * flag2
29
30
31 def poisson(x):
32     k = POISSON_PARAM
33     return (numpy.power(x, k) / numpy.math.factorial(k)) * numpy.exp(-x)
34
35
36 func_dict = {
37     'normal': standart_normal,
38     'cauchy': standart_cauchy,
39     'laplace': laplace,
40     'uniform': uniform,
41     'poisson': poisson
42 }
43
44
45 def generate_laplace(x):
46     return numpy.random.laplace(0, 1/LAPLAS_COEF, x)
47
48
49 def generate_uniform(x):
50     return numpy.random.uniform(UNIFORM_LEFT, UNIFORM_RIGHT, x)
51
52
53 def generate_poisson(x):
54     return numpy.random.poisson(POISSON_PARAM, x)
55
```

```

56
57 generate_dict = {
58     'normal': numpy.random.standard_normal,
59     'cauchy': numpy.random.standard_cauchy,
60     'laplace': generate_laplace,
61     'uniform': generate_uniform,
62     'poisson': generate_poisson
63 }
64
65
66 def Zr(x):
67     return (numpy.amin(x) + numpy.amax(x))/2
68
69
70 def Zq(x):
71     return (numpy.quantile(x, 1/4) + numpy.quantile(x, 3/4) ) /2
72
73
74 def Ztr(x):
75     length = x.size
76     r = (int)(length / 4)
77     sum = 0
78     for i in range(r, length - r):
79         sum += x[i]
80     return sum/(length - 2 * r)
81
82
83 pos_characteristic_dict = {
84     'average': numpy.mean,
85     'med': numpy.median,
86     'Zr': Zr,
87     'Zq': Zq,
88     'Ztr r = n/4': Ztr
89 }
90
91 pos_char_name = [
92     'average',
93     'med',
94     'Zr',
95     'Zq',
96     'Ztr r = n/4'
97 ]
98
99
100 def E(z):
101     return numpy.mean(z)
102
103
104 def D(z):
105     return numpy.var(z)
106
107
108
109 f = open('out.csv', 'w')
110 sys.stdout = f
111
112
113 def research(dist_type):
114     print('_____')
115     print(dist_type)
116     for num in selection:
117         #print(num)
118         print_table = {
119             'E': [],
120             'D': []
121         }

```



```

122     for pos_name in pos_char_name:
123         z = []
124         for i in range(0, 1000):
125             arr = numpy.sort(generate_dict[dist_type](num))
126             z.append(pos_characteristic_dict[pos_name](arr))
127         print_table['E'].append(E(z))
128         print_table['D'].append(D(z))
129
130     print()
131     print("%-10s;" % ('n = %i' % num), end=" ")
132     for pos_name in pos_char_name:
133         print("%-12s;" % pos_name, end=" ")
134
135     print()
136     print("%-10s;" % ('E ='), end=" ")
137     for e in print_table['E']:
138         print("%-12f;" % e, end=" ")
139
140     print()
141     print("%-10s;" % ('D ='), end=" ")
142     for d in print_table['D']:
143         print("%-12f;" % d, end=" ")
144     print()
145
146
147
148
149 research('normal')
150 research('cauchy')
151 research('laplace')
152 research('uniform')
153 research('poisson')

```