



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

ИНСТИТУТ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

КАФЕДРА ВЫСШЕЙ МАТЕМАТИКИ

Лабораторная работа 2

по курсу «Теория вероятностей и математическая статистика, часть 2»

ВАРИАНТ 46

Тема: Первичная обработка выборки из

непрерывной генеральной совокупности

Выполнил:
Студент 3-го курса
Успенский А.А.

Группа: КМБО-03-19

МОСКВА – 2022

Задание

Задание 1. Получить выборку, сгенерировав $N = 200$ псевдослучайных чисел, распределенных по нормальному закону с параметрами $a = (-1)^V \cdot 0,1V$ и σ^2 , где $\sigma = 0,005V + 1$.

Задание 2. Получить выборку, сгенерировав $N = 200$ псевдослучайных чисел, распределенных по показательному закону с параметром $\lambda = 3 + (-1)^V \cdot 0,01 \cdot V$

Задание 3. Получить выборку, сгенерировав 200 псевдослучайных чисел, распределенных равномерно на отрезке $[a, b]$, где $a = (-1)^V \cdot 0,02 \cdot V$, $b = a + 6$.

V –номер варианта.

Построить:

- 1) статистический ряд;
- 2) полигон относительных частот;
- 3) график эмпирической функции распределения;

Найти:

- 1) выборочное среднее;
- 2) выборочную дисперсию с поправкой Шеппарда;
- 3) выборочное среднее квадратическое отклонение;
- 4) выборочную моду;
- 5) выборочную медиану;
- 6) выборочный коэффициент асимметрии;
- 7) выборочный коэффициент эксцесса.

Составить таблицы:

- 1) сравнения относительных частот и теоретических вероятностей попадания в интервалы;
- 2) сравнения рассчитанных характеристик с теоретическими значениями.

Вычисления проводить с точностью до **0,00001**.

Краткие теоретические сведения

Нормальное распределение

Характеристика	Значение
Математическое ожидание	a
Дисперсия	σ^2
Среднее квадратическое отклонение	σ
Мода	a
Медиана	a
Коэффициент асимметрии	0
Коэффициент эксцесса	0

Плотность

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{(-\frac{(x-a)^2}{2\sigma^2})}$$

Функция распределение

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^x e^{(-\frac{(t-a)^2}{2\sigma^2})} dt$$

Показательное распределение

Характеристика	Значение
Математическое ожидание	λ^{-1}
Дисперсия	λ^{-2}
Среднее квадратическое отклонение	λ^{-1}
Мода	0
Медиана	$\frac{\ln 2}{\lambda}$
Коэффициент асимметрии	2
Коэффициент эксцесса	6

Плотность

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0, \\ 0, & x < 0. \end{cases}$$

Функция распределение

$$F(x) = \begin{cases} 1 - e^{-\lambda x}, & x \geq 0, \\ 0, & x < 0. \end{cases}$$

Равномерное распределение на отрезке

Характеристика	Значение
Математическое ожидание	$\frac{a+b}{2}$
Дисперсия	$\frac{(b-a)^2}{12}$
Среднее квадратическое отклонение	$\frac{b-a}{2\sqrt{3}}$
Мода	$\frac{a+b}{2}$
Медиана	$\frac{a+b}{2}$
Коэффициент асимметрии	0
Коэффициент эксцесса	$-\frac{6}{5}$

Плотность

$$f(x) = \begin{cases} \frac{1}{b-a}, & x \in [a, b] \\ 0, & x \notin [a, b] \end{cases}$$

Функция распределения

$$F(x) = \begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & a \leq x < b \\ 1, & x \geq b \end{cases}$$

Ряд распределения - структурная группировка с целью выделения характерных свойств и закономерностей изучаемой совокупности.

Математическое ожидание — понятие среднего значения случайной величины в теории вероятностей.

Дисперсия — отклонение величины от ее математического ожидания.

Среднеквадратическое отклонение – показатель рассеивания значений случайной величины относительно ее математического ожидания.

Мода – значение во множестве наблюдений, которое встречается наиболее часто.

Медиана – возможное значение признака, которое делит вариационный ряд выборки на две равные части.

Коэффициент асимметрии используется для проверки распределения на симметричность, а также для грубой предварительной проверки на нормальность.

Если плотность распределения симметрична, то выборочный коэффициент асимметрии равен нулю, если левый хвост распределения тяжелее – больше нуля, легче – меньше.

Коэффициент эксцесса используется для проверки на нормальность.

Нормальное распределение имеет нулевой эксцесс. Если хвосты распределения «легче», а пик острее, чем у нормального распределения, то коэффициент эксцесса положительный; если хвосты распределения «тяжелее», пик «приплюснутый», чем у нормального распределения, то отрицательный.

Математическое ожидание:

$$M(X) = \bar{x} = \frac{1}{N} \sum_{k=1}^m x_k^* n_k = \sum_{k=1}^m x_k^* w_k$$

Дисперсия с поправкой Шеппарда:

$$D(X) = s_B^2 = \sum_{i=1}^m (x_i^* - \bar{x})^2 \cdot w_i - \frac{h^2}{12}, \text{ где } h = \frac{a_m - a_0}{m}$$

Среднее квадратическое отклонение:

$$\sigma = \bar{\sigma} = \sqrt{s_B^2}$$

Мода:

$$\bar{M}_0 = a_{k-1} + h \cdot \frac{w_k - w_{k-1}}{2w_k - w_{k-1} - w_{k+1}}$$

Медиана:

$$\bar{M}_e = a_{k-1} + \frac{h}{w_k} \cdot \left(\frac{1}{2} - \sum_{i=1}^{k-1} w_i \right), \text{ если } \sum_{i=1}^{k-1} w_i < \frac{1}{2} < \sum_{i=1}^k w_i;$$

Выборочный момент k-го порядка:

$$\bar{\mu}_k = \overline{x^k} = \sum_{i=1}^m (x_i^*)^k \cdot w_i, \bar{\mu}_1 = \bar{x}$$

Коэффициент асимметрии:

$$\bar{\gamma}_1 = \frac{\bar{\mu}_3^0}{\bar{\sigma}^3}$$

Коэффициент эксцесса:

$$\bar{\gamma}_2 = \frac{\bar{\mu}_4^0}{\bar{\sigma}^4} - 3$$

Средства языка программирования

Для расчёта статистических исследований я использую язык Python. В программе расчёта используются следующие библиотеки:

NumPy — это библиотека языка Python, добавляющая поддержку больших многомерных массивов и матриц, вместе с большой библиотекой высокоуровневых (и очень быстрых) математических функций для операций с этими массивами.

Pandas — это библиотека для обработки и анализа данных. Работа pandas с данными строится поверх библиотеки NumPy, являющейся инструментом более низкого уровня. Предоставляет специальные структуры данных и операции для манипулирования числовыми таблицами и временными рядами.

Matplotlib — это библиотека для визуализации данных. Построение графиков, диаграмм и гистограмм.

Scipy.stats – этот модуль содержит большое количество вероятностных распределений, а также растущую библиотеку статистических функций.

Стоит описать некоторые команды для генерации и визуализации данных:

sps.norm(a, sd).rvs(size) - генерация N псевдослучайных чисел, распределенных по нормальному закону с параметрами a и σ ;

sps.expon(lm).rvs(size) – генерация N псевдослучайных чисел, распределенных по показательному закону с параметром λ ;

sps.uniform(a, b).rvs(size) – генерация N псевдослучайных чисел, распределенных равномерно на отрезке $[a, b]$.

pd.DataFrame – визуализация данных в виде таблицы. Используется для визуализации статистического ряда.

Результаты расчётов

Задание 1

$$a = 4.6; \sigma = 1.23.$$

Выборка из 200 элементов:

2.14992	1.81745	2.30375	1.72707	1.25813	1.52412	1.65144	1.86560	2.08559	2.20372
2.58229	2.41247	2.61796	2.41218	2.30579	2.33617	2.35270	2.45254	2.47230	2.59822
3.05920	2.82790	3.10751	2.81761	2.64590	2.66093	2.78912	3.00543	3.03485	3.06624
3.29912	3.17644	3.37818	3.16833	3.11233	3.16175	3.16522	3.22526	3.23466	3.32269
3.55253	3.51956	3.65529	3.47194	3.39235	3.43106	3.45065	3.53417	3.54933	3.55578
3.82083	3.80932	3.85153	3.74361	3.66390	3.67746	3.71210	3.81051	3.81713	3.84439

3.94103	3.90617	4.00158	3.90013	3.86440	3.87863	3.89470	3.91108	3.92613	3.97546
4.18934	4.12875	4.20267	4.12250	4.03412	4.05978	4.12176	4.13500	4.15080	4.18960
4.34664	4.25702	4.36222	4.23455	4.20785	4.20796	4.23178	4.29767	4.33669	4.35899
4.49629	4.45160	4.51728	4.41159	4.38671	4.38820	4.39574	4.45890	4.47601	4.50133
4.63200	4.58159	4.65974	4.57625	4.53367	4.55292	4.56754	4.60287	4.62972	4.64361
4.81468	4.73160	4.81468	4.72365	4.68293	4.69054	4.70690	4.78855	4.81415	4.81766
4.85991	4.83948	4.87437	4.83242	4.81766	4.82367	4.82719	4.84292	4.84861	4.86577
5.02054	4.98038	5.03294	4.94144	4.92852	4.93626	4.94038	5.01356	5.01693	5.02434
5.17213	5.12273	5.19329	5.11995	5.04061	5.06709	5.10874	5.15446	5.16789	5.18496
5.33190	5.25969	5.38625	5.24495	5.22017	5.22688	5.22754	5.27318	5.29903	5.33606
5.66085	5.55425	5.67932	5.53144	5.43060	5.46697	5.50179	5.56433	5.60122	5.67544
5.92392	5.83657	6.02607	5.79476	5.73547	5.76101	5.78677	5.84417	5.89531	5.93037
6.45662	6.21019	6.66601	6.19188	6.07975	6.10157	6.14694	6.24075	6.34853	6.48076
7.12071	6.87742	7.16699	6.84370	6.69621	6.77436	6.82098	6.97632	7.01078	7.12447

Сортированная выборка

1.25813	1.52412	1.65144	1.72707	1.81745	1.86560	2.08559	2.14992	2.20372	2.30375
2.30579	2.33617	2.35270	2.41218	2.41247	2.45254	2.47230	2.58229	2.59822	2.61796
2.64590	2.66093	2.78912	2.81761	2.82790	3.00543	3.03485	3.05920	3.06624	3.10751
3.11233	3.16175	3.16522	3.16833	3.17644	3.22526	3.23466	3.29912	3.32269	3.37818
3.39235	3.43106	3.45065	3.47194	3.51956	3.53417	3.54933	3.55253	3.55578	3.65529

3.66390	3.67746	3.71210	3.74361	3.80932	3.81051	3.81713	3.82083	3.84439	3.85153
3.86440	3.87863	3.89470	3.90013	3.90617	3.91108	3.92613	3.94103	3.97546	4.00158
4.03412	4.05978	4.12176	4.12250	4.12875	4.13500	4.15080	4.18934	4.18960	4.20267
4.20785	4.20796	4.23178	4.23455	4.25702	4.29767	4.33669	4.34664	4.35899	4.36222
4.38671	4.38820	4.39574	4.41159	4.45160	4.45890	4.47601	4.49629	4.50133	4.51728
4.53367	4.55292	4.56754	4.57625	4.58159	4.60287	4.62972	4.63200	4.64361	4.65974
4.68293	4.69054	4.70690	4.72365	4.73160	4.78855	4.81415	4.81468	4.81766	4.81468
4.81766	4.82367	4.82719	4.83242	4.83948	4.84292	4.84861	4.85991	4.86577	4.87437
4.92852	4.93626	4.94038	4.94144	4.98038	5.01356	5.01693	5.02054	5.02434	5.03294
5.04061	5.06709	5.10874	5.11995	5.12273	5.15446	5.16789	5.17213	5.18496	5.19329
5.22017	5.22688	5.22754	5.24495	5.25969	5.27318	5.29903	5.33190	5.33606	5.38625
5.43060	5.46697	5.50179	5.53144	5.55425	5.56433	5.60122	5.66085	5.67544	5.67932
5.73547	5.76101	5.78677	5.79476	5.83657	5.84417	5.89531	5.92392	5.93037	6.02607
6.07975	6.10157	6.14694	6.19188	6.21019	6.24075	6.34853	6.45662	6.48076	6.66601
6.69621	6.77436	6.82098	6.84370	6.87742	6.97632	7.01078	7.12071	7.12447	7.16699

Группированная выборка

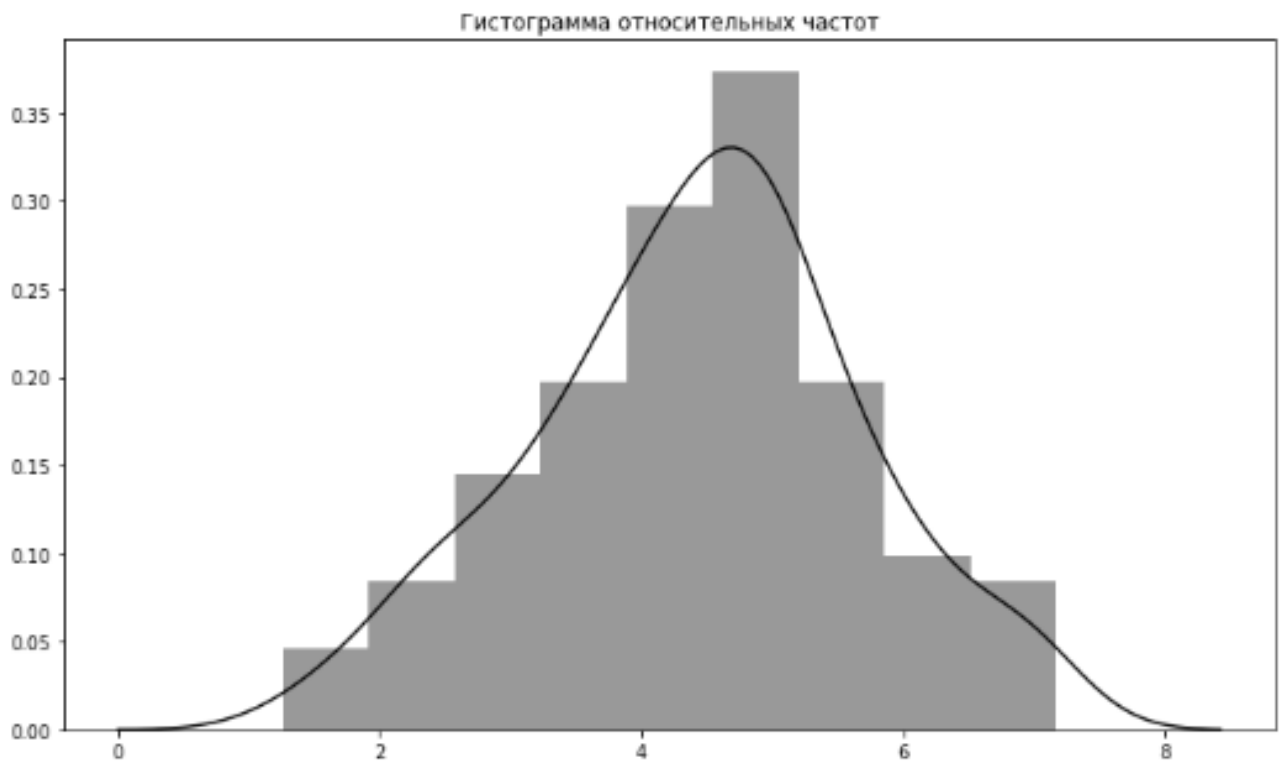
Интервалы	n_i	w_i
[1.25813, 1.99674]	6	0.030
(1.99674, 2.73534]	16	0.080
(2. 73534, 3.47395]	22	0.110
(3. 47395, 4.21256]	38	0.190
(4.21256, 4.95116]	52	0.260
(4.95116, 5.68977]	36	0.180
(5.68977, 6.42838]	17	0.085

(6. 42838, 7.16699]	13	0.065
---------------------	----	-------

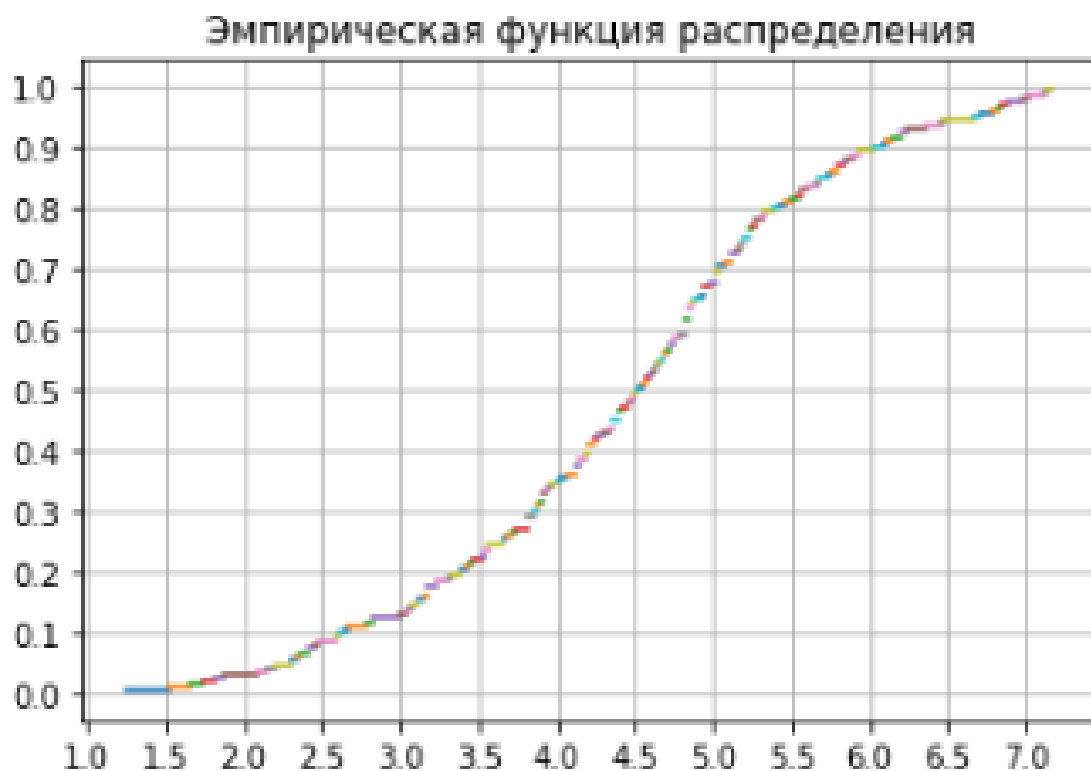
Ассоциированный статистический ряд

x_i^*	n_i	w_i
1.62744	6	0.030
2.36604	16	0.080
3.10465	22	0.110
3.84326	38	0.190
4.58186	52	0.260
5.32047	36	0.180
6.05908	17	0.085
6.797684	13	0.065

Гистограмма относительных частот



Эмпирическая функция распределения



Результаты расчетов требуемых характеристик

- Выборочное среднее: 4.43738;
- Выборочная дисперсия с поправкой Шеппарда: 1.55533;
- Выборочное среднее квадратическое отклонение: 1.24713;
- Выборочная мода: 4.58186;
- Выборочная медиана: 4.52548;
- Выборочный коэффициент асимметрии: -0.12039;
- Выборочный коэффициент эксцесса: -0.23359.

Задание 2

$$\lambda = 3.46$$

Выборка из 200 элементов:

0.01752	0.00704	0.01303	0.01059	0.00559	0.01063	0.01083	0.01257	0.01364	0.01420
0.03497	0.02122	0.03231	0.02201	0.01803	0.02919	0.03115	0.03194	0.03304	0.03448
0.04727	0.03639	0.04417	0.03809	0.03575	0.03858	0.04046	0.04081	0.04557	0.04633
0.08149	0.05260	0.06754	0.05356	0.05009	0.05495	0.05799	0.06326	0.07168	0.07602
0.09136	0.08271	0.08805	0.08370	0.08173	0.08398	0.08477	0.08539	0.08962	0.08974
0.10496	0.09418	0.10316	0.09510	0.09369	0.09562	0.09876	0.09921	0.10374	0.10473
0.11780	0.10704	0.11495	0.10726	0.10620	0.10819	0.10890	0.11158	0.11568	0.11710
0.14464	0.12504	0.13679	0.12573	0.12399	0.12621	0.13131	0.13447	0.13785	0.14291
0.18071	0.14737	0.16080	0.15440	0.14661	0.15556	0.15633	0.15635	0.16566	0.17130
0.20943	0.19325	0.20063	0.19459	0.18651	0.19671	0.19733	0.19929	0.20167	0.20532
0.23351	0.21656	0.22678	0.21691	0.21506	0.22101	0.22441	0.22469	0.22942	0.23280
0.25834	0.23868	0.25393	0.24006	0.23539	0.24017	0.24105	0.25234	0.25426	0.25748
0.27995	0.26369	0.27532	0.26425	0.25926	0.26507	0.26611	0.27471	0.27687	0.27758
0.32421	0.29701	0.31428	0.30047	0.29179	0.30356	0.30855	0.31055	0.31494	0.32175
0.37507	0.32995	0.35111	0.33650	0.32627	0.33770	0.34368	0.34502	0.35575	0.36037
0.46670	0.40296	0.45068	0.40928	0.37911	0.41981	0.42701	0.43861	0.45337	0.46248
0.56485	0.50214	0.53058	0.50242	0.47880	0.50406	0.51166	0.52171	0.53488	0.54301
0.68927	0.57582	0.65713	0.58550	0.56870	0.61315	0.63986	0.64498	0.66547	0.67621

0.83776	0.70648	0.77002	0.71616	0.70320	0.73461	0.76326	0.76655	0.77462	0.80939
1.47998	0.85892	1.16189	0.86071	0.84364	0.87241	0.90109	0.94003	1.36705	1.40168

Сортированная выборка:

0.00559	0.00704	0.01059	0.01063	0.01083	0.01257	0.01303	0.01364	0.01420	0.01752
0.01803	0.02122	0.02201	0.02919	0.03115	0.03194	0.03231	0.03304	0.03448	0.03497
0.03575	0.03639	0.03809	0.03858	0.04046	0.04081	0.04417	0.04557	0.04633	0.04727
0.05009	0.05260	0.05356	0.05495	0.05799	0.06326	0.06754	0.07168	0.07602	0.08149
0.08173	0.08271	0.08370	0.08398	0.08477	0.08539	0.08805	0.08962	0.08974	0.09136
0.09369	0.09418	0.09510	0.09562	0.09876	0.09921	0.10316	0.10374	0.10473	0.10496
0.10620	0.10704	0.10726	0.10819	0.10890	0.11158	0.11495	0.11568	0.11710	0.11780
0.12399	0.12504	0.12573	0.12621	0.13131	0.13447	0.13679	0.13785	0.14291	0.14464
0.14661	0.14737	0.15440	0.15556	0.15633	0.15635	0.16080	0.16566	0.17130	0.18071
0.18651	0.19325	0.19459	0.19671	0.19733	0.19929	0.20063	0.20167	0.20532	0.20943
0.21506	0.21656	0.21691	0.22101	0.22441	0.22469	0.22678	0.22942	0.23280	0.23351
0.23539	0.23868	0.24006	0.24017	0.24105	0.25234	0.25393	0.25426	0.25748	0.25834
0.25926	0.26369	0.26425	0.26507	0.26611	0.27471	0.27532	0.27687	0.27758	0.27995
0.29179	0.29701	0.30047	0.30356	0.30855	0.31055	0.31428	0.31494	0.32175	0.32421
0.32627	0.32995	0.33650	0.33770	0.34368	0.34502	0.35111	0.35575	0.36037	0.37507
0.37911	0.40296	0.40928	0.41981	0.42701	0.43861	0.45068	0.45337	0.46248	0.46670
0.47880	0.50214	0.50242	0.50406	0.51166	0.52171	0.53058	0.53488	0.54301	0.56485
0.56870	0.57582	0.58550	0.61315	0.63986	0.64498	0.65713	0.66547	0.67621	0.68927

0.70320	0.70648	0.71616	0.73461	0.76326	0.76655	0.77002	0.77462	0.80939	0.83776
0.84364	0.85892	0.86071	0.87241	0.90109	0.94003	1.16189	1.36705	1.40168	1.47998

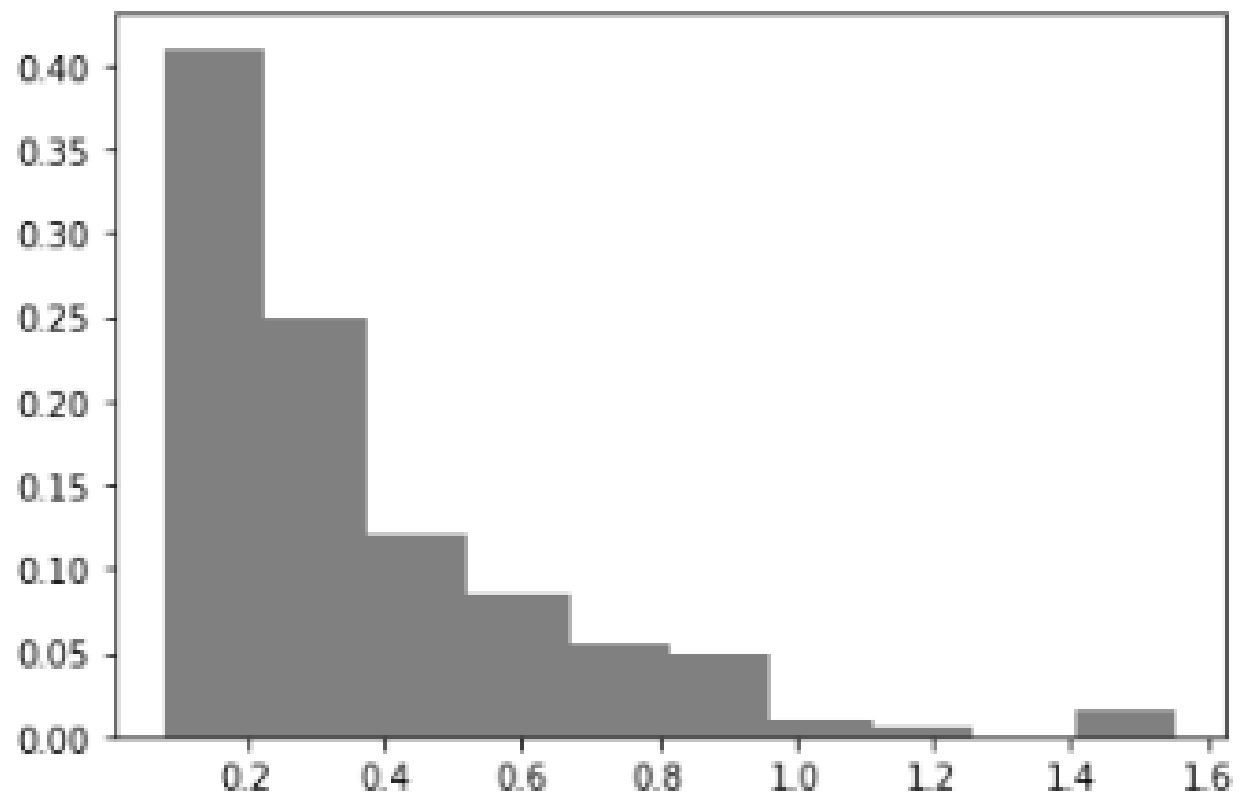
Группированная выборка

Интервалы	n_i	w_i
[0.00559, 0.18989]	91	0.455
(0.18989, 0.37419]	58	0.290
(0.37419, 0.55849]	20	0.100
(0.55849, 0.74279]	15	0.075
(0.74279, 0.92709]	11	0.055
(0.92709, 1.11139]	1	0.005
(1.11139, 1.29568]	1	0.005
(1.29568, 1.47998]	3	0.015

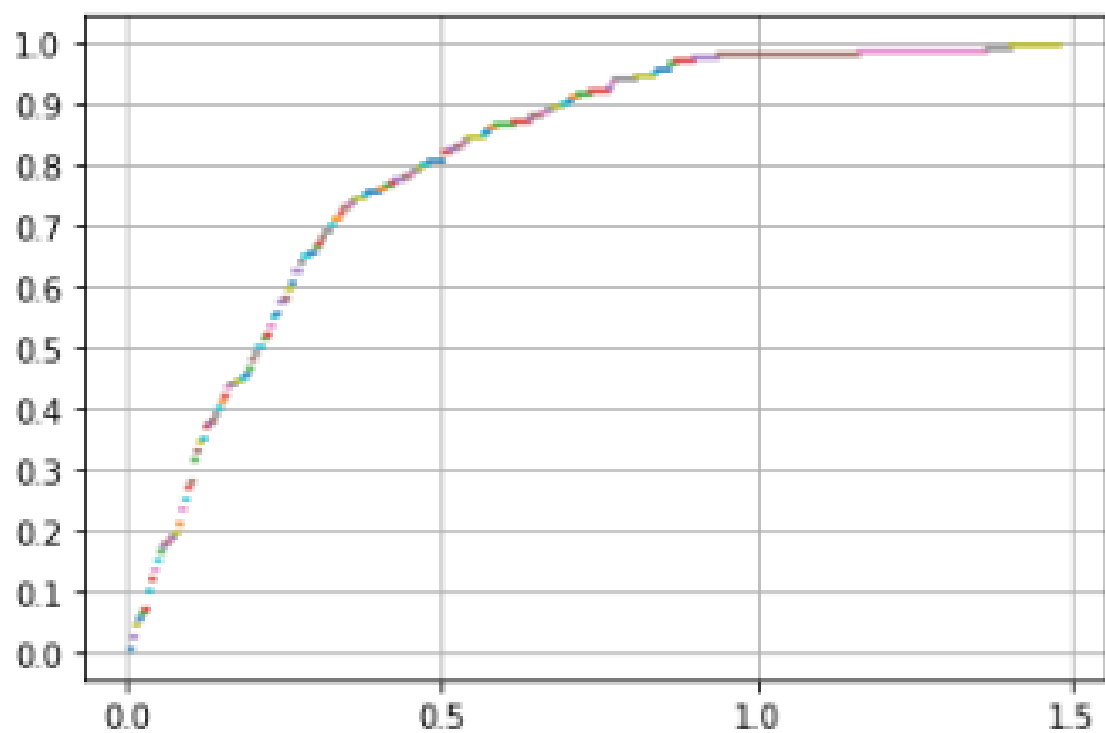
Ассоциированный статистический ряд

x_i^*	n_i	w_i
0.09774	91	0.455
0.28204	58	0.290
0.46634	20	0.100
0.65064	15	0.075
0.83494	11	0.055
1.01924	1	0.005
1.20354	1	0.005
1.38783	3	0.015

Гистограмма относительных частот



Эмпирическая функция распределения



Результаты расчетов требуемых характеристик

- Выборочное среднее: 0.28689;
- Выборочная дисперсия с поправкой Шеппарда: 0.07546;
- Выборочное среднее квадратическое отклонение: 0.27470;
- Выборочная мода: 0.09774
- Выборочная медиана: 0.21224
- Выборочный коэффициент асимметрии: 1.70761;
- Выборочный коэффициент эксцесса: 3.42792.

Задание 3

$$a = 0.92; b = 6.92$$

Выборка из 200 элементов

1.10255	0.96087	1.12508	0.92203	0.95797	0.96293	0.96747	0.97825	1.01538	1.03989
1.49995	1.17400	1.51284	1.13940	1.15717	1.20037	1.27560	1.35057	1.46125	1.49311
1.80873	1.62559	1.81192	1.55442	1.57155	1.63206	1.64895	1.74796	1.75393	1.76746
2.08999	1.86954	2.11960	1.83147	1.86778	1.87068	1.88207	1.90528	1.92699	1.95271
2.34468	2.22746	2.35336	2.12494	2.15699	2.24550	2.25345	2.28229	2.32040	2.32738
2.83398	2.56734	2.85527	2.49579	2.53571	2.57359	2.72241	2.73666	2.75356	2.81423
3.10758	2.96810	3.11770	2.87184	2.88844	2.97703	3.04607	3.06817	3.06819	3.07947
3.36382	3.16223	3.38608	3.14415	3.16140	3.26795	3.29837	3.30791	3.35635	3.35793
3.63821	3.47399	3.64972	3.39185	3.42436	3.47872	3.49631	3.53548	3.56696	3.61924
3.89014	3.69870	3.90404	3.66798	3.68065	3.74500	3.82730	3.84277	3.86819	3.87830
4.15752	4.00423	4.17684	3.92053	3.95246	4.01898	4.02591	4.04527	4.04823	4.05324
4.47414	4.23591	4.48527	4.19394	4.22332	4.25102	4.39307	4.39631	4.42584	4.43972
4.89896	4.59609	4.90190	4.54036	4.56470	4.62118	4.62704	4.67591	4.70494	4.86652
5.06795	4.96207	5.12419	4.90434	4.93063	4.98602	4.99435	4.99644	5.01942	5.06771
5.42752	5.14579	5.48281	5.13080	5.13653	5.18069	5.20928	5.28708	5.30818	5.35323

5.71922	5.57951	5.76422	5.53252	5.54690	5.58780	5.61134	5.62275	5.63167	5.65022
6.01513	5.85750	6.03627	5.78881	5.83999	5.87861	5.91046	5.91867	5.94082	5.96450
6.27752	6.03909	6.29129	6.03756	6.03891	6.05607	6.05668	6.14657	6.25194	6.27366
6.63033	6.31586	6.63687	6.30510	6.30906	6.46848	6.48496	6.52667	6.56120	6.57341
6.90512	6.69780	6.91446	6.65035	6.65137	6.72406	6.73537	6.87337	6.90309	6.90428

Сортированная выборка:

0.92203	0.95797	0.96087	0.96293	0.96747	0.97825	1.01538	1.03989	1.10255	1.12508
1.13940	1.15717	1.17400	1.20037	1.27560	1.35057	1.46125	1.49311	1.49995	1.51284
1.55442	1.57155	1.62559	1.63206	1.64895	1.74796	1.75393	1.76746	1.80873	1.81192
1.83147	1.86778	1.86954	1.87068	1.88207	1.90528	1.92699	1.95271	2.08999	2.11960
2.12494	2.15699	2.22746	2.24550	2.25345	2.28229	2.32040	2.32738	2.34468	2.35336
2.49579	2.53571	2.56734	2.57359	2.72241	2.73666	2.75356	2.81423	2.83398	2.85527
2.87184	2.88844	2.96810	2.97703	3.04607	3.06817	3.06819	3.07947	3.10758	3.11770
3.14415	3.16140	3.16223	3.26795	3.29837	3.30791	3.35635	3.35793	3.36382	3.38608
3.39185	3.42436	3.47399	3.47872	3.49631	3.53548	3.56696	3.61924	3.63821	3.64972
3.66798	3.68065	3.69870	3.74500	3.82730	3.84277	3.86819	3.87830	3.89014	3.90404
3.92053	3.95246	4.00423	4.01898	4.02591	4.04527	4.04823	4.05324	4.15752	4.17684
4.19394	4.22332	4.23591	4.25102	4.39307	4.39631	4.42584	4.43972	4.47414	4.48527
4.54036	4.56470	4.59609	4.62118	4.62704	4.67591	4.70494	4.86652	4.89896	4.90190
4.90434	4.93063	4.96207	4.98602	4.99435	4.99644	5.01942	5.06771	5.06795	5.12419
5.13080	5.13653	5.14579	5.18069	5.20928	5.28708	5.30818	5.35323	5.42752	5.48281
5.53252	5.54690	5.57951	5.58780	5.61134	5.62275	5.63167	5.65022	5.71922	5.76422

5.78881	5.83999	5.85750	5.87861	5.91046	5.91867	5.94082	5.96450	6.01513	6.03627
6.03756	6.03891	6.03909	6.05607	6.05668	6.14657	6.25194	6.27366	6.27752	6.29129
6.30510	6.30906	6.31586	6.46848	6.48496	6.52667	6.56120	6.57341	6.63033	6.63687
6.65035	6.65137	6.69780	6.72406	6.73537	6.87337	6.90309	6.90428	6.90512	6.91446

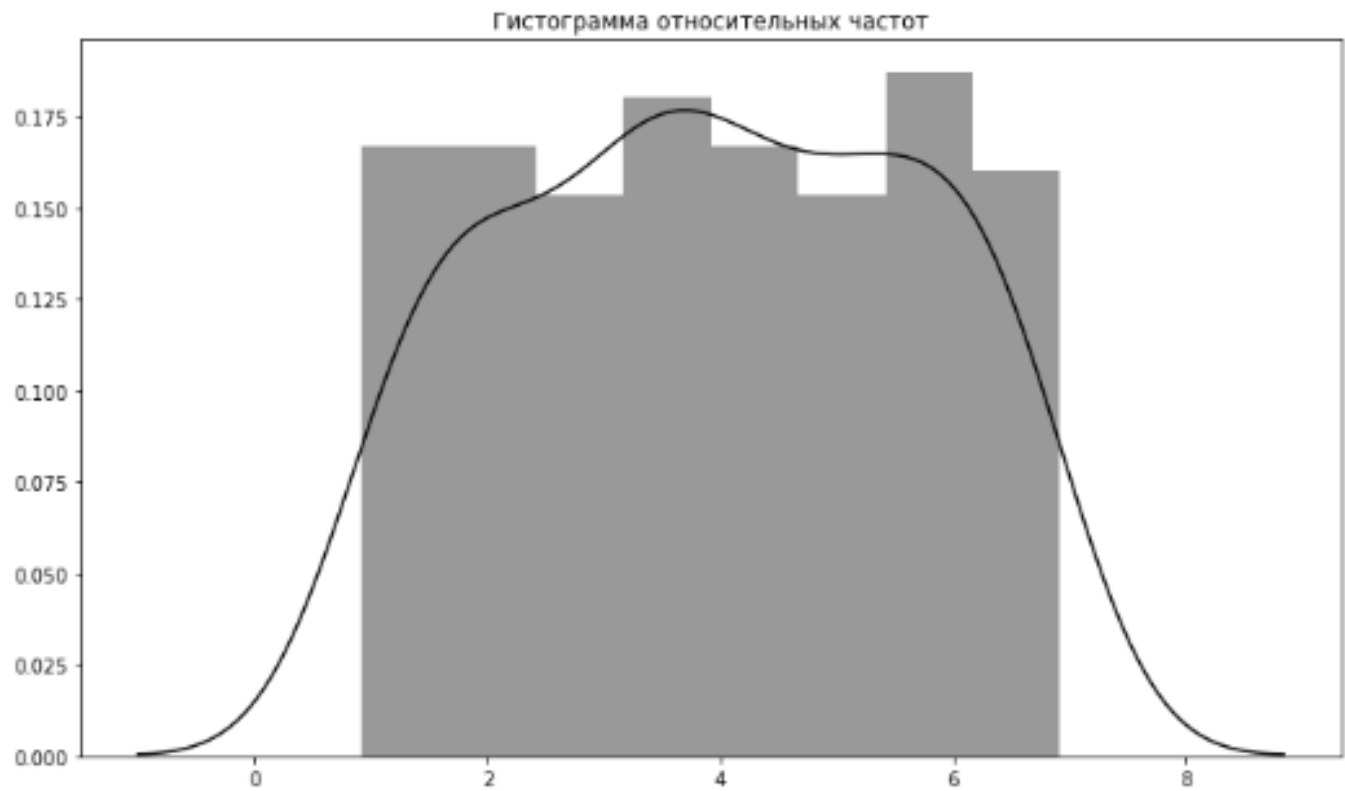
Группированная выборка

Интервалы	n_i	w_i
(0.92202, 1.67108]	25	0.125
(1.67108, 2.42013]	25	0.125
(2.42013, 3.16919]	23	0.115
(3.16919, 3.91824]	27	0.135
(3.98967, 4.66729]	25	0.125
(4.66729, 5.41635]	23	0.115
(5.41635, 6.16540]	28	0.140
(6.16540, 6.91446]	24	0.120

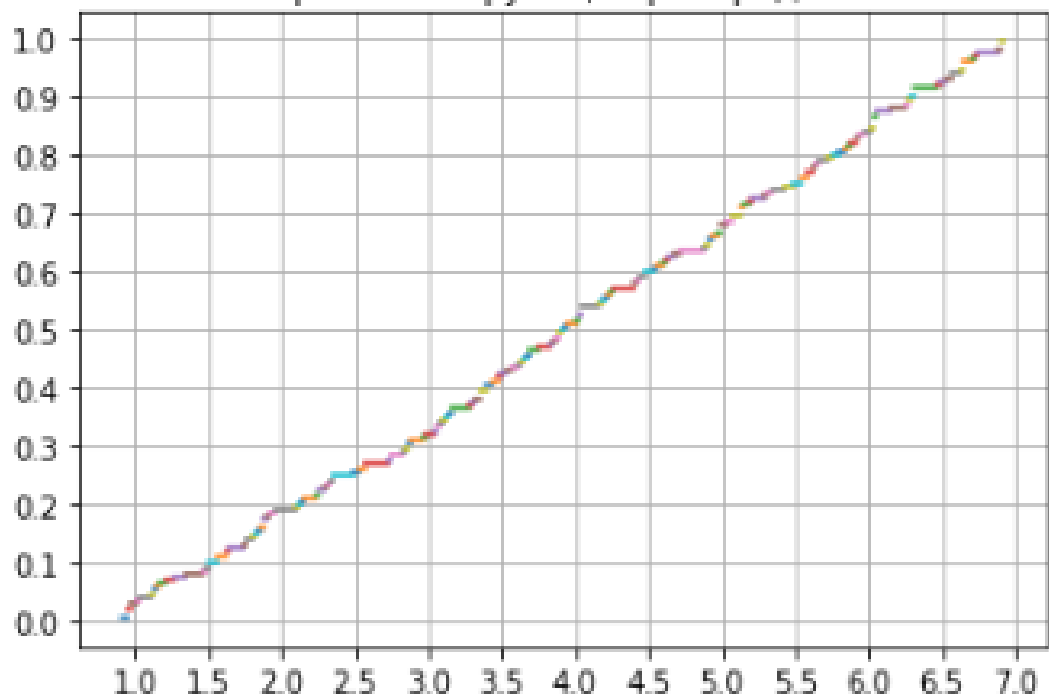
Ассоциированный статистический ряд

x_i^*	n_i	w_i
1.29655	25	0.125
2.04561	25	0.125
2.79466	23	0.115
3.54372	27	0.135
4.29277	25	0.125
5.04182	23	0.115
5.79088	28	0.140
6.53993	24	0.120

Гистограмма относительных частот



Эмпирическая функция распределения
Эмпирическая функция распределения



Результаты расчетов требуемых характеристик

- Выборочное среднее: 3.94304;
- Выборочная дисперсия с поправкой Шеппарда: 3.02291;
- Выборочное среднее квадратическое отклонение: 1.73865;
- Выборочная мода: 5.79088;
- Выборочная медиана: 3.91824;
- Выборочный коэффициент асимметрии: -0.03819;
- Выборочный коэффициент эксцесса: -1.14953.

Анализ результатов

Нормальное распределение

Название показателя	Экспериментальное значение	Теоретическое значение	Абсолютное отклонение	Относительное отклонение
Выборочное среднее	4.43738	4.6	0.16262	0.03535
Выборочная дисперсия с поправкой Шеппарда	1.55533	1.5129	0.01713	0.02805
Выборочное среднее квадратичное	1.24713	1.23	0.03739	0.01393
Выборочная мода	4.58186	4.6	0.02805	0.00609
Выборочная медиана	4.52548	4.6	0.07452	0.01620
Выборочный коэффициент асимметрии	-0.12039	0	-0.12039	-
Выборочный коэффициент эксцесса	-0.23359	0	-0.23359	-

Интервалы	w_i	p_i	$ w_i - p_i $
[1.25813, 1.99674]	0.030	0.00677	0.00823
(1.99674, 2.73534]	0.080	0.01396	0.03896
(2. 73534, 3.47395]	0.110	0.04681	0.04318
(3. 47395, 4.21256]	0.190	0.15007	0.01993
(4.21256, 4.95116]	0.260	0.15088	0.04912
(4.95116, 5.68977]	0.180	0.18551	0.05949
(5.68977, 6.42838]	0.085	0.14145	0.00355
(6. 42838, 7.16699]	0.065	0.04761	0.02239
$\sum w_i = 1.00$		$\sum p_i = 0.98883$	$\Delta_{max} = 0.05949$

Показательное распределение

Название показателя	Экспериментальное значение	Теоретическое значение	Абсолютное отклонение	Относительное отклонение
Выборочное среднее	0.28689	0.28902	0.00213	0.00737
Выборочная дисперсия с поправкой Шеппарда	0.07546	0.08354	0.00808	0.09672
Выборочное среднее квадратичное	0.27470	0.28902	0.01432	0.04955
Выборочная мода	0.09774	0	0.09774	-
Выборочная медиана	0.21224	0.20033	0.01191	0.05945
Выборочный коэффициент асимметрии	1.70761	2	0.29239	0.14619
Выборочный коэффициент эксцесса	3.42792	6	2.57208	0.42868

Интервалы	w_i	p_i	$ w_i - p_i $
[0.00559, 0.18989]	0.455	0.43126	0.02374
(0.18989, 0.37419]	0.290	0.25471	0.03029
(0.37419, 0.55849]	0.100	0.10176	0.00824
(0.55849, 0.74279]	0.075	0.01585	0.01585
(0.74279, 0.92709]	0.055	0.02118	0.02882
(0.92709, 1.11139]	0.005	0.01763	0.00237
(1.11139, 1.29568]	0.005	0.00108	0.00608
(1.29568, 1.47998]	0.015	0.00086	0.00086
$\sum w_i = 1.00$ $\sum p_i = 0.99475$ $\Delta_{max} = 0.03029$			

Равномерное распределение на отрезке

Название показателя	Экспериментальное значение	Теоретическое значение	Абсолютное отклонение	Относительное отклонение
Выборочное среднее	3.94304	3.92	0.02303	0.00588
Выборочная дисперсия с поправкой Шеппарда	3.02291	3	0.02290	0.00763
Выборочное среднее квадратичное	1.73865	1.73205	0.00659	0.00380
Выборочная мода	5.79088	3.92	1,87088	0.477265
Выборочная медиана	3.91824	3.92	0.00176	0.00044
Выборочный коэффициент асимметрии	-0.03819	0	0.03819	-
Выборочный коэффициент эксцесса	-1.14953	-1.2	0.05047	-0.04206

Интервалы	w_i	p_i	$ w_i - p_i $
(0.92202, 1.67108]	0.125	0.12762	0.03162
(1.67108, 2.42013]	0.125	0.12762	0.00338
(2.42013, 3.16919]	0.115	0.12762	0.02662
(3.16919, 3.91824]	0.135	0.12762	0.03338
(3.98967, 4.66729]	0.125	0.12762	0.01838
(4.66729, 5.41635]	0.115	0.12762	0.02338
(5.41635, 6.16540]	0.140	0.12762	0.01338
(6.16540, 6.91446]	0.120	0.12762	0.04662
$\sum w_i = 1.00$		$\sum p_i = 0.98704$	$\Delta_{max} = 0.04662$

Список литературы

- 1) Математическая статистика [Электронный ресурс]: метод. указания по выполнению лаб. работ / А.А. Лобузов – М.: МИРЭА, 2017.
- 2) Гмурман В.Е. Теория вероятностей и математическая статистика. – М.:Юрайт, 2020.
- 3) Ивченко Г.И., Медведев Ю.И. Математическая статистика. – М.: URSS, 2020.

Приложение

```
1. import scipy.stats as sps
2. import numpy as np
3. import pandas as pd
4. import matplotlib.pyplot as plt
5. import pylab
6. import statistics
7. import cmath
8. import seaborn as sns
9. from matplotlib import ticker
10.     from math import factorial
11.     from math import exp
12.     from math import log
13.
14.     def Normrnd(a,sd,size):
15.         distr = sps.norm.rvs(loc=a, scale=sd, size=200)
16.         distr.sort()
17.         print (
18.             '\nПараметр a = ', a,
19.             '\nПараметр sd = ', sd,)
20.         print ('Нормальное распределение:\n',distr)
21.         return (distr)
22.
23.     def Exprnd(lm,size):
24.         distr = sps.expon.rvs(scale=1/lm, size=size)
25.         distr.sort()
26.         print (
27.             '\nПараметр lambda = ', lm,)
28.         print ('Показательное распределение:\n',distr)
29.         return (distr)
30.
31.     def Uniformrnd(a,b,size):
32.         distr = sps.uniform.rvs(loc=a, scale=6,size=size)
33.         distr.sort()
34.         print (
35.             '\nПараметр a = ', a,
36.             '\nПараметр b = ', b,)
37.         print ('Равномерное распределение:\n',distr)
38.         return (distr)
39.
40.
41.     def Xi(distr, size):
42.         xi = []
43.         for i in range(size):
44.             if distr[i]!=distr[i-1]:
45.                 xi.append(distr[i])
```

```

46.         return(xi)
47.
48.     def Freq(distr, xi, size):
49.         ni=[]
50.         for j in range(len(xi)):
51.             help = 0
52.             for i in range(size):
53.                 if distr[i]==xi[j]:
54.                     help += 1
55.             ni.append(help)
56.         return(ni)
57.
58.     def Rel_freq(ni, size):
59.         wi = []
60.         for i in range(len(ni)):
61.             wi.append(ni[i]/size)
62.         return(wi)
63.
64.     def koef_srfq(wi, k):
65.         S=0
66.         for i in range (k+1):
67.             S = S + wi[i]
68.         return(S)
69.
70.     def Sum_rfq(wi, k):
71.         sk = []
72.         for i in range (len(wi)):
73.             sk.append(koef_srfq(wi,i))
74.         return(sk)
75.
76.     def intervals(xi, size):
77.         m = 1+round(log(size,2))
78.         d = xi[-1] - xi[0]
79.         a = [xi[0]]
80.         for i in range (1,(m+1)):
81.             a.append(d/m+a[i-1])
82.         return a
83.
84.     def Freq_inter(distr, inter, size):
85.         ni=[]
86.         for j in range(1,len(inter)):
87.             count = 0
88.             h=0
89.             for i in range(size):
90.                 if j-1 == 0:
91.                     if distr[i]>=inter[j-1] and
distr[i]<=inter[j]:
92.                         count +=1

```

```

93.             elif j == len(inter)-1:
94.                 if distr[i]>inter[j-1] and
distr[i]<=inter[j]:
95.                     count += 1
96.                     if distr[i]>inter[j]:
97.                         count += 1
98.                     else:
99.                         if distr[i]>inter[j-1] and
distr[i]<=inter[j]:
100.                            count += 1
101.                            ni.append(count)
102.                            return(ni)
103.
104. def X_X(inter):
105.     x_x = []
106.     for i in range(len(inter)-1):
107.         x_x.append((inter[i]+inter[i+1])/2)
108.     return(x_x)
109.
110.
111. def Mean(xi ,wi):
112.     X = 0
113.     for i in range (len(xi)):
114.         help = xi[i]*wi[i]
115.         X += help
116.     return (X)
117.
118. def DispSh (inter,m,xi,wi,mean):
119.     h = (inter[-1]-inter[0])/m
120.     s=xi
121.     for i in range (len(xi)):
122.         s[i]=((xi[i]-mean)**2)*wi[i]-(h**2)/12
123.
124.     return (sum(s))
125.
126. def Mode(xi,wi):
127.     max_w = max(wi)
128.     h = (inter[-1]-inter[0])/m
129.     for k in range(len(wi_inter)-1):
130.         mode = xi[k]
131.         flag += 1
132.     return(round(mode, 5))
133.
134. def Median(xi,sk):
135.     for k in range(len(xi)):
136.         if k == 0:
137.             if sk[k] > 0.5:
138.                 md = round(xi[k], 5)

```

```

139.         return (md)
140.         if sk[k] == 0.5:
141.             md = (round(xi[k]+xi[k+1])/2, 5)
142.             return (md)
143.         if (sk[k] > 0.5) and (sk[k-1] < 0.5):
144.             md = round(xi[k], 5)
145.             return (md)
146.         if sk[k] == 0.5:
147.             md= (round(xi[k]+xi[k+1])/2, 5)
148.             return (md)
149.
150.     def Moment(xi,wi):
151.         M = []
152.         for i in range (len(xi)):
153.             help = 0
154.             for j in range (len(xi)):
155.                 help += (xi[j]**i)*wi[j]
156.             M.append(help)
157.         return (M)
158.
159.     def Asym_coef(M, sd):
160.         asym = (M[2]-3*M[1]*M[0]+2*pow(M[0], 3))/(pow(sd, 3))
161.         return (asym)
162.
163.     def Exe_coef(M, sd):
164.         help = M[3]-4*M[2]*M[0]+6*M[1]*pow(M[0], 2)-3*pow(M[0],
165.         4)
166.         exe = help/(pow(sd, 4))-3
167.         return (exe)
168.
169.     v = 46
170.     a_norm = round((-1)**v*0.1*v,2)
171.     sigma = 0.005*v+1
172.     size=200
173.     xi = []
174.     ni = []
175.     wi = []
176.     sk = []
177.     #xi - значение
178.     #ni - кол-во значения
179.     #sum(ni) = size
180.     #wi = ni/size
181.     #sk = sum(wj)
182.
183.     distr1 = Normrnd(a_norm,sigma,size)
184.     xi = Xi(distr1, size)
185.     ni = Freq(distr1, xi, size)
186.     wi= Rel_freq(ni, size)

```

```

186.     sk = Sum_rfq(wi, size)
187.
188.     #Distr2 = Geomerty(p,size)
189.     #Distr3 = Poisson(l,size)
190.
191.     inter = intervals(xi,size)
192.     inter
193.
194.     #inter = [[a[0],a[1]], ... , [a[n-1],a[n]]]
195.
196.     interval_all = [[inter[0], inter[1]],
197.                     [inter[1], inter[2]],
198.                     [inter[2], inter[3]],
199.                     [inter[3], inter[4]],
200.                     [inter[4], inter[5]],
201.                     [inter[5], inter[6]],
202.                     [inter[6], inter[7]],
203.                     [inter[7], inter[8]],
204.                     [inter[8], inter[9]],]
205.     print (interval_all)
206.
207.     ni_inter = Freq_inter(distr1, inter, size)
208.     print (ni_inter)
209.     print (sum(ni_inter))
210.
211.     wi_inter = Rel_freq(ni_inter,size)
212.     print (wi_inter)
213.     print (sum(wi_inter))
214.
215.     x_x = X_X(inter)
216.
217.     Table = pd.DataFrame({'center_Intervals':x_x, 'Ni':ni_inter,
218.                           'Wi':wi_inter})
219.     Table
220.     # график функции эмпирического распределения
221.     plt.figure(figsize=(12, 8))
222.     fig, sec = plt.subplots()
223.     sec.xaxis.set_major_locator(ticker.MultipleLocator(0.5))
224.     sec.yaxis.set_major_locator(ticker.MultipleLocator(0.1))
225.     for k in range(len(xi) - 1):
226.         sec.plot([xi[k], xi[k+1]], [sk[k], sk[k]])
227.     plt.title("Эмпирическая функция распределения")
228.     plt.grid(True)
229.     plt.show()
230.
231.     m = 1+round(log(size,2))
232.     plt.figure(figsize=(12, 7))

```

```

233. sns.distplot(distr1, bins = m)
234. plt.title('Гистограмма относительных частот')
235.
236. data = pd.DataFrame({'distr1': distr1})
237.
238. # Выборочное среднее (теор/эксп)
239. X_teor = a_norm
240. X = Mean(x_x,wi_inter)
241. print (X_teor, '\n', distr1.mean(), '\n')
242.
243. #Выборочный момент
244. M = Moment(x_x,wi_inter)
245.
246. # Выборочная дисперсия с поправкой Шепарда (теор/эксп)
247. D_teor = sigma**2
248. D = DispSh(inter,m,x_x,wi_inter,X)
249. print (D_teor, '\n', distr1.var(), '\n')
250.
251. #Выборочное среднеквадратическое откл. (теор/эксп)
252. SD_teor = sigma
253. print (SD_teor, '\n', distr1.std(), '\n')
254.
255. #Выборочная мода (теор/эксп)
256. mode_teor = a_norm
257. print (mode_teor, '\n', '\n')
258.
259. #Выборочная медиана (теор/эксп)
260. median_teor = a_norm
261. print (median_teor, '\n', data.describe(percentiles=
    [.50]).iloc[4,0], '\n')
262.
263. #Выборочный коэф.асимметрии (теор/эксп)
264. asym_teor = 0
265. print (asym_teor, '\n', data.skew()[0], '\n')
266.
267. #Выборочный коэф. эксцесса (теор/эксп)
268. exe_teor = 0
269. exe = Exe_coef(M, distr1.std())
270. print (exe_teor, '\n', data.kurtosis()[0], '\n')
271.
272. lm = 3 + (-1)**v * 0.01*v
273. xi2 = []
274. ni2 = []
275. wi2 = []
276. sk2 = []
277. #xi - значение
278. #ni - кол-во значения
279. #sum(ni) = size

```

```

280.     #wi = ni/size
281.     #sk = sum(wj)
282.
283.     distr2 = Exprnd(lm,size)
284.     xi2 = Xi(distr2, size)
285.     ni2 = Freq(distr2, xi2, size)
286.     wi2 = Rel_freq(ni2, size)
287.     sk2 = Sum_rfq(wi2, size)
288.
289.     inter2 = intervals(xi2,size)
290.     inter2
291.
292.     #inter = [[a[0],a[1]], ... , [a[n-1],a[n]]]
293.
294.     ni_inter2 = Freq_inter(distr2, inter2, size)
295.     print (ni_inter2)
296.     print (sum(ni_inter2))
297.
298.     wi_inter2 = Rel_freq(ni_inter2,size)
299.     print (wi_inter2)
300.     print (sum(wi_inter2))
301.
302.     # график функции эмпирического распределения
303.     plt.figure(figsize=(12, 8))
304.     fig, sec = plt.subplots()
305.     sec.xaxis.set_major_locator(ticker.MultipleLocator(0.5))
306.     sec.yaxis.set_major_locator(ticker.MultipleLocator(0.1))
307.     for k in range(len(xi2) - 1):
308.         sec.plot([xi2[k], xi2[k+1]], [sk2[k], sk2[k]])
309.     plt.title("Эмпирическая функция распределения")
310.     plt.grid(True)
311.     plt.show()
312.
313.     m = 1+round(log(size,2))
314.     plt.figure(figsize=(12, 7))
315.     sns.distplot(distr2, bins = m)
316.     plt.title('Гистограмма относительных частот')
317.
318.     data2 = pd.DataFrame({'distr2': distr2})
319.
320.     # Выборочное среднее (теор/эксп)
321.     X_teor2 = lm**(-1)
322.     X2 = Mean(x_x2,wi_inter2)
323.     print (X_teor2, '\n', distr2.mean(), '\n')
324.
325.     #Выборочный момент
326.     M2 = Moment(x_x2,wi_inter2)
327.

```



```

328.     # Выборочная дисперсия с поправкой Шепарда (теор/эксп)
329.     D_teor2 = lm**(-2)
330.     D2 = DispSh(inter2,m,x_x2,wi_inter2,X2)
331.     print (D_teor2, '\n', distr2.var(), '\n')
332.
333.     #Выборочное среднеквадратическое откл. (теор/эксп)
334.     SD_teor2 = lm**(-1)
335.     print (SD_teor2, '\n', distr2.std(), '\n')
336.
337.     #Выборочная мода (теор/эксп)
338.     mode_teor2 = 0
339.     print (mode_teor2, '\n', '\n')
340.
341.     #Выборочная медиана (теор/эксп)
342.     median_teor2 = log(2)/lm
343.     print (median_teor2, '\n', data2.describe(percentiles=
    [.50]).iloc[4,0], '\n')
344.
345.     #Выборочный коэф. асимметрии (теор/эксп)
346.     asym_teor2 = 2
347.     print (asym_teor2, '\n', data2.skew()[0], '\n')
348.
349.     #Выборочный коэф. эксцесса (теор/эксп)
350.     exe_teor2 = 6
351.     exe2 = Exe_coef(M2, distr2.std())
352.     print (exe_teor2, '\n', data2.kurtosis()[0], '\n')
353.
354.     a_unif = (-1)**v *0.02*v
355.     b_unif = a_unif + 6
356.     xi3 = []
357.     ni3 = []
358.     wi3 = []
359.     sk3 = []
360.     #xi - значение
361.     #ni - кол-во значения
362.     #sum(ni) = size
363.     #wi = ni/size
364.     #sk = sum(wj)
365.
366.     distr3 = Uniformrnd(a_unif,b_unif,size)
367.     xi3 = Xi(distr3, size)
368.     ni3 = Freq(distr3, xi3, size)
369.     wi3 = Rel_freq(ni3, size)
370.     sk3 = Sum_rfq(wi3, size)
371.
372.
373.     # график функции эмпирического распределения
374.     plt.figure(figsize=(12, 8))

```

```

375.     fig, sec = plt.subplots()
376.     sec.xaxis.set_major_locator(ticker.MultipleLocator(0.5))
377.     sec.yaxis.set_major_locator(ticker.MultipleLocator(0.1))
378.     for k in range(len(xi3) - 1):
379.         sec.plot([xi3[k], xi3[k+1]], [sk3[k], sk3[k]])
380.     plt.title("Эмпирическая функция распределения")
381.     plt.grid(True)
382.     plt.show()
383.
384.     m = 1+round(log(size,2))
385.     plt.figure(figsize=(12, 7))
386.     sns.distplot(distr3, bins = m)
387.     plt.title('Гистограмма относительных частот')
388.
389.     data3 = pd.DataFrame({'distr3': distr3})
390.
391.     # Выборочное среднее (теор/эксп)
392.     X_teor3 = (a_unif+b_unif)/2
393.     X3 = Mean(x_x3,wi_inter3)
394.     print (X_teor3, '\n', distr3.mean(), '\n')
395.
396.     #Выборочный момент
397.     M3 = Moment(x_x3,wi_inter3)
398.
399.     # Выборочная дисперсия с поправкой Шепарда (теор/эксп)
400.     D_teor3 = (b_unif - a_unif)**2/12
401.     D3 = DispSh(inter3,m,x_x3,wi_inter3,X3)
402.     print (D_teor3, '\n', distr3.var(), '\n')
403.
404.     #Выборочное среднеквадратическое откл. (теор/эксп)
405.     SD_teor3 = (b_unif - a_unif)/(2*3**(1/2))
406.     print (SD_teor3, '\n', distr3.std(), '\n')
407.
408.     #Выборочная мода (теор/эксп)
409.     mode_teor3 = (a_unif + b_unif)/2
410.     print (mode_teor3, '\n', '\n')
411.
412.     #Выборочная медиана (теор/эксп)
413.     median_teor3 = (a_unif + b_unif)/2
414.     print (median_teor3, '\n', data3.describe(percentiles=
        [.50]).iloc[4,0], '\n')
415.
416.     #Выборочный коэф.асимметрии (теор/эксп)
417.     asym_teor3 = 0
418.     print (asym_teor3, '\n', data3.skew()[0], '\n')
419.
420.     #Выборочный коэф. эксцесса (теор/эксп)
421.     exe_teor3 = -6/5

```

```
422.     exe3 = Exe_coef(M3, distr3.std())
423.     print (exe_teor3, '\n', data3.kurtosis()[0], '\n')
424.
```