



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 3

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

ВАРИАНТ 46

Тема: Проверка статистических гипотез с помощью
критерия хи-квадрат и критерия Колмогорова

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

Группа: КМБО-03-19

МОСКВА – 2022

Задание

Задание 3-1. Проверка гипотезы о биномиальном распределении с помощью критерия χ^2 .

В соответствии с номером варианта взять из файла **MC_D_Binom** выборку $\{x_1, \dots, x_N\}$ и значение n . Построить статистический ряд (Таблица 1.1):

x_i	n_i	w_i
x_1^*	n_1	w_1
x_2^*	n_2	w_2
...
x_m^*	n_m	w_m
	$\sum_{i=1}^m n_i$	$\sum_{i=1}^m w_i$

$x_i^* < x_j^*$ при $i < j$, n_i – частота x_i^* ($n_i > 0$); $w_i = \frac{n_i}{N} > 0$.

Найти методом моментов оценку \tilde{p} параметра p биномиального распределения $Bi(n, p)$ при заданном значении n . Построить на одном рисунке график полигона относительных частот и график полигона соответствующих теоретических вероятностей. Построить Таблицу 1.2 вида:

x_i	w_i	p_i^*	$ w_i - p_i^* $	$\frac{N(w_i - p_i^*)^2}{p_i^*}$
x_1^*	w_1	p_1^*	$ w_1 - p_1^* $	$\frac{N(w_1 - p_1^*)^2}{p_1^*}$
...
x_m^*	w_m	p_m^*	$ w_m - p_m^* $	$\frac{N(w_m - p_m^*)^2}{p_m^*}$
	$\sum_{i=1}^m w_i$	$\sum_{i=1}^m p_i^*$	$\max w_i - p_i^* $	$\sum_{i=1}^m \frac{N(w_i - p_i^*)^2}{p_i^*}$

где p_i^* равны теоретическим вероятностям значений x_i при найденном значении \tilde{p} .

Проверить с помощью критерия χ^2 гипотезу о соответствии выборки биномиальному распределению $Bi(n, \tilde{p})$ при уровне значимости $0,05$.

Задание 3-3. Проверка гипотезы о нормальном распределении с помощью критерия χ^2 .

В соответствии с номером варианта взять из файла **MC_D_Norm** выборку $\{x_1, \dots, x_N\}$. Построить группированную выборку (Таблица 3.1). Найти оценки математического ожидания \tilde{a} и дисперсии $\tilde{\sigma}^2$.

Построить

1) Таблицу 3.2 вида

k	a_k	$\frac{a_k - \tilde{a}}{\tilde{\sigma}}$	$\frac{1}{\tilde{\sigma}} \varphi\left(\frac{a_k - \tilde{a}}{\tilde{\sigma}}\right)$	$\Phi\left(\frac{a_k - \tilde{a}}{\tilde{\sigma}}\right)$	p_k^*
0	a_0	$\frac{a_0 - \tilde{a}}{\tilde{\sigma}}$	$\frac{1}{\tilde{\sigma}} \varphi\left(\frac{a_0 - \tilde{a}}{\tilde{\sigma}}\right)$	$\Phi\left(\frac{a_0 - \tilde{a}}{\tilde{\sigma}}\right)$	—
1	a_1	$\frac{a_1 - \tilde{a}}{\tilde{\sigma}}$	$\frac{1}{\tilde{\sigma}} \varphi\left(\frac{a_1 - \tilde{a}}{\tilde{\sigma}}\right)$	$\Phi\left(\frac{a_1 - \tilde{a}}{\tilde{\sigma}}\right)$	p_1^*
...
m	a_m	$\frac{a_m - \tilde{a}}{\tilde{\sigma}}$	$\frac{1}{\tilde{\sigma}} \varphi\left(\frac{a_m - \tilde{a}}{\tilde{\sigma}}\right)$	$\Phi\left(\frac{a_m - \tilde{a}}{\tilde{\sigma}}\right)$	p_m^*
					$\sum_{k=1}^m p_k^*$

2) график плотности нормального распределения $N(\tilde{a}, \tilde{\sigma}^2)$, наложенный на гистограмму относительных частот;

3) Таблицу 3.3 вида

k	Интервал	w_k	p_k^*	$ w_k - p_k^* $	$\frac{N(w_k - p_k^*)^2}{p_k^*}$
1	$[a_0, a_1]$	w_1	p_1^*	$ w_1 - p_1^* $	$\frac{N(w_1 - p_1^*)^2}{p_1^*}$
2	$(a_1, a_2]$	w_2	p_2^*	$ w_2 - p_2^* $	$\frac{N(w_2 - p_2^*)^2}{p_2^*}$
...
m	$(a_{m-1}, a_m]$	w_m	p_m^*	$ w_m - p_m^* $	$\frac{N(w_m - p_m^*)^2}{p_m^*}$
		$\sum_{k=1}^m w_k$	$\sum_{k=1}^m p_k^*$	$\max w_k - p_k^* $	$\sum_{k=1}^m \frac{N(w_k - p_k^*)^2}{p_k^*}$

где значения p_k^* находятся в соответствии с указаниями к **Заданию 3-3**.

Проверить с помощью критерия χ^2 гипотезу о соответствии выборки нормальному распределению $N(\tilde{a}, \tilde{\sigma}^2)$ при уровне значимости 0,05.

**Задание 3-4. Проверка гипотезы о равномерном распределении
с помощью критерия χ^2 .**

В соответствии с номером варианта взять из файла **MC_D_Unif** выборку $\{x_1, \dots, x_N\}$ и значения a и b .

Построить:

1) группированную выборку (Таблица 4.1) и Таблицу 4.2 вида

k	Интервал	w_k	p_k^*	$ w_k - p_k^* $	$\frac{N(w_k - p_k^*)^2}{p_k^*}$
1	$[a, a_1]$	w_1	p_1^*	$ w_1 - p_1^* $	$\frac{N(w_1 - p_1^*)^2}{p_1^*}$
2	$(a_1, a_2]$	w_2	p_2^*	$ w_2 - p_2^* $	$\frac{N(w_2 - p_2^*)^2}{p_2^*}$
...
m	$(a_{m-1}, b]$	w_m	p_m^*	$ w_m - p_m^* $	$\frac{N(w_m - p_m^*)^2}{p_m^*}$
		$\sum_{k=1}^m w_k$	$\sum_{k=1}^m p_k^*$	$\max w_k - p_k^* $	$\sum_{k=1}^m \frac{N(w_k - p_k^*)^2}{p_k^*}$

где значения p_k^* находятся в соответствии с указаниями к **Заданию 3-4**;

2) график плотности равномерного распределения на отрезке $[a, b]$, наложенный на гистограмму относительных частот.

Проверить с помощью критерия χ^2 гипотезу о соответствии выборки равномерному распределению на отрезке $[a, b]$ при уровне значимости $0,05$.

**Задание 3-5. Проверка гипотезы о равномерном распределении
с помощью критерия Колмогорова.**

В соответствии с номером варианта взять из файла **MC_D_Unif** выборку $\{x_1, \dots, x_N\}$ и значения a и b .

Построить:

1) на одном рисунке график эмпирической функции распределения $F_N(x)$ данной выборки и график функции распределения $F(x)$ равномерного закона на отрезке $[a, b]$;

2) Таблицу 5.1 следующего вида

a	b	N	D_N	$D_N\sqrt{N}$	x^*	$F(x^*)$	$F_N(x^*)$	$F_N(x^* - 0)$

где $D_N = \max_{1 \leq j \leq N} (\max(|F_N(x_{(j)}) - F(x_{(j)})|, |F_N(x_{(j)} - 0) - F(x_{(j)})|))$,

$x^* = x_{(j)}$, если $D_N = \max(|F_N(x_{(j)}) - F(x_{(j)})|, |F_N(x_{(j)} - 0) - F(x_{(j)})|)$.

Проверить гипотезу о соответствии выборки равномерному распределению на отрезке $[a, b]$ при уровне значимости $0,05$ с помощью критерия Колмогорова.

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:

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

В разделе отчета **Анализ результатов** проводится сравнение рассчитанных в разделе отчета **Результаты расчетов** значений критериев и критических значений и

делается вывод о соответствии или несоответствии выборки соответствующему распределению. В этом разделе следует привести

таблицу критических значений $\chi_{кр,\alpha}^2(l)$ при уровне значимости $\alpha = 0,05$

l	4	5	6	7	8
$\chi_{кр,\alpha}^2(l)$	9,5	11,1	12,6	14,1	15,5

и таблицу критических значений распределения Колмогорова k_α

α	0,01	0,02	0,05	0,1	0,2
k_α	1,63	1,57	1,36	1,22	1,07

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/expon/uniform.cdf(x, loc, scale) — Функция распределения случайной непрерывной величины. Позволяет узнать значение функции в точке.;

sps.norm/expon/uniform.pdf(x, loc, scale) — Плотность распределения случайной непрерывной величины. Позволяет узнать значение плотности в точке;

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

Результаты расчётов

Задание 3-1

$$n = 9$$

Выборка из файла MC_D_Binom.pdf:

4	6	7	6	7	7	6	3	7	7
6	8	7	8	6	7	4	7	7	6
5	7	3	6	8	7	8	6	5	7
3	8	5	8	6	5	6	4	5	8
7	7	6	6	8	7	6	9	5	6
8	6	8	6	5	5	7	8	7	8
6	5	7	6	7	7	5	8	7	6
7	7	5	7	8	7	9	7	7	8
5	4	4	6	6	7	4	5	7	7
7	6	5	8	7	6	6	5	8	7
8	5	9	6	7	7	7	4	8	6
5	6	5	7	7	7	7	7	8	7
5	6	7	7	9	6	9	8	8	6
8	7	6	5	5	4	5	6	7	5
6	7	8	4	6	6	5	7	7	6
7	6	4	6	7	8	7	6	8	5
5	5	6	8	7	8	4	6	6	5
5	5	7	7	6	9	7	7	7	4
5	7	4	5	8	5	7	7	7	4
6	7	7	6	8	5	5	6	7	6

Сортированная выборка

3	3	3	4	4	4	4	4	4	4
4	4	4	4	4	4	4	5	5	5
5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5
5	5	6	6	6	6	6	6	6	6
6	6	6	6	6	6	6	6	6	6
6	6	6	6	6	6	6	6	6	6

6	6	6	6	6	6	6	6	6	6
6	6	6	6	6	6	6	6	6	7
7	7	7	7	7	7	7	7	7	7
7	7	7	7	7	7	7	7	7	7
7	7	7	7	7	7	7	7	7	7
7	7	7	7	7	7	7	7	7	7
7	7	7	7	7	7	7	7	7	7
7	7	7	7	7	7	7	7	7	7
7	7	7	7	7	7	7	7	7	7
7	7	7	7	8	8	8	8	8	8
8	8	8	8	8	8	8	8	8	8
8	8	8	8	8	8	8	8	8	8
8	8	8	8	9	9	9	9	9	9

Статистический ряд:

x_i	n_i	w_i
3	3	0,015
4	14	0,170
5	35	0,175
6	47	0,235
7	65	0,325
8	30	0,150
9	6	0,030
	200	1.0

Оценка $\tilde{p} = 0.70611$

Полигон относительных частот и график полигона соответствующих теоритических вероятностей



Таблица 1.2

X_i	W_i	P_i	$ W_i - P_i $	$\frac{N(W_i - P_i)^2}{P_i}$
3	0.015	0.01905	0.00405	0.17220
4	0.070	0.06867	0.00133	0.00515
5	0.175	0.16499	0.01001	0.12146
6	0.235	0.26428	0.02928	0.64880
7	0.325	0.27213	0.05287	2.05434
8	0.150	0.16346	0.01346	0.22167
9	0.030	0.04364	0.01364	0.85266
	1.0	0.99622	0.05287	4.07628

Оценка $\tilde{p} = 0.70611$

Выборочное значение $\chi^2 = 4.07628$

Число степеней свободы: $l = 5$

Задание 3-2

Выборка из файла MC_D_Exp.pdf:

0,49628	0,02259	0,00040	0,05806	1,63059	0,51029	1,23128	0,00755	0,04024	0,84819
1,55681	0,16015	0,06479	1,38866	0,56765	0,52571	1,12378	1,06742	0,72536	1,69262
0,37297	0,42906	2,11917	2,17256	0,69063	0,17531	0,08694	1,87670	0,69310	0,42116
0,10912	0,76509	0,92739	0,13592	0,20676	0,14595	1,19374	0,92776	0,21235	0,41517
0,25107	0,81523	0,92245	0,45171	0,51220	0,41253	1,18896	1,31295	2,91082	0,87170
0,46807	0,19552	0,00081	1,91352	1,05088	2,60738	1,11188	0,13929	1,91726	0,28394
0,24792	0,15739	1,34678	0,44408	0,09046	0,27818	0,74501	0,00174	0,43443	1,46321
1,71495	0,66364	1,12611	2,34382	0,48073	2,12724	0,23564	1,51726	1,70317	1,76708
0,84122	0,43739	0,14632	0,09811	1,58386	0,89172	0,55610	0,85295	0,07721	0,31931
0,91889	0,67610	1,36475	0,92720	0,41095	0,21642	0,34861	0,52992	0,82593	0,14207
0,67741	0,05637	0,02206	0,27342	0,40903	0,03090	0,01631	0,44665	0,18265	1,76193
0,26468	0,65431	0,65251	0,46023	2,99722	1,52676	1,37410	1,50322	2,24020	0,59787
2,20956	1,23387	0,57252	0,34066	1,09893	0,66105	1,28583	0,04749	0,13962	0,00689
0,71303	0,69219	1,40444	0,54301	1,35348	0,13559	0,20652	1,61129	0,52083	0,44850
0,84825	0,18711	1,27399	0,08482	2,43031	0,33862	0,47997	1,76315	1,40260	0,15558
0,54428	1,28525	0,46034	1,17596	0,01015	1,44526	0,17917	0,41267	0,17406	0,29945
0,25663	0,63074	0,73090	1,57545	0,27854	0,43471	1,60032	0,14453	1,81542	2,17253
1,40109	0,26168	0,94765	0,09979	0,00100	1,49746	0,20326	2,53646	0,05298	0,30559
1,06502	3,17029	0,08229	2,94829	0,69646	0,26311	0,07600	0,33859	0,78006	0,46522
0,19192	0,02046	0,43331	0,19490	1,62112	1,75990	0,38188	0,83528	0,14308	0,27799

Сортированная выборка:

0.0004	0.00081	0.001	0.00174	0.00689	0.00755	0.01015	0.01631	0.02046	0.02206
0.02259	0.0309	0.04024	0.04749	0.05298	0.05637	0.05806	0.06479	0.076	0.07721
0.08229	0.08482	0.08694	0.09046	0.09811	0.09979	0.10912	0.13559	0.13592	0.13929
0.13962	0.14207	0.14308	0.14453	0.14595	0.14632	0.15558	0.15739	0.16015	0.17406
0.17531	0.17917	0.18265	0.18711	0.19192	0.1949	0.19552	0.20326	0.20652	0.20676
0.21235	0.21642	0.23564	0.24792	0.25107	0.25663	0.26168	0.26311	0.26468	0.27342
0.27799	0.27818	0.27854	0.28394	0.29945	0.30559	0.31931	0.33859	0.33862	0.34066
0.34861	0.37297	0.38188	0.40903	0.41095	0.41253	0.41267	0.41517	0.42116	0.42906
0.43331	0.43443	0.43471	0.43739	0.44408	0.44665	0.4485	0.45171	0.46023	0.46034
0.46522	0.46807	0.47997	0.48073	0.49628	0.51029	0.5122	0.52083	0.52571	0.52992
0.54301	0.54428	0.5561	0.56765	0.57252	0.59787	0.63074	0.65251	0.65431	0.66105
0.66364	0.6761	0.67741	0.69063	0.69219	0.6931	0.69646	0.71303	0.72536	0.7309
0.74501	0.76509	0.78006	0.81523	0.82593	0.83528	0.84122	0.84819	0.84825	0.85295
0.8717	0.89172	0.91889	0.92245	0.9272	0.92739	0.92776	0.94765	1.05088	1.06502
1.06742	1.09893	1.11188	1.12378	1.12611	1.17596	1.18896	1.19374	1.23128	1.23387
1.27399	1.28525	1.28583	1.31295	1.34678	1.35348	1.36475	1.3741	1.38866	1.40109
1.4026	1.40444	1.44526	1.46321	1.49746	1.50322	1.51726	1.52676	1.55681	1.57545

1.58386	1.60032	1.61129	1.62112	1.63059	1.69262	1.70317	1.71495	1.7599	1.76193
1.76315	1.76708	1.81542	1.8767	1.91352	1.91726	2.11917	2.12724	2.17253	2.17256
2.20956	2.2402	2.34382	2.43031	2.53646	2.60738	2.91082	2.94829	2.99722	3.17029

Таблица 2.1 Группированная выборка

Интервалы	n_i	w_i
[0, 0.39664]	73	0.455
(0.39664, 0.79287]	50	0.290
(0.79287, 1.18911]	24	0.100
(1.18911, 1.58535]	24	0.075
(1.58535, 1.98158]	15	0.055
(1.98158, 2.37782]	7	0.005
(2.37782, 2.77405]	3	0.005
(2.77405, 3.17029]	4	0.015
	200	1.0

Оценка $\tilde{\lambda} = 1.27224$

Таблица 2.2

k	a_k	$f(a_k, \tilde{\lambda})$	$F(a_k, \tilde{\lambda})$	p_k^*
0	0	1.27224	0	0
1	0.39664	0.76809	0.39627	0.39627
2	0.79287	0.46397	0.63531	0.29905
3	1.18911	0.28026	0.77971	0.14440
4	1.58535	0.16929	0.86694	0.08722
5	1.98158	0.10226	0.91962	0.05269
6	2.37782	0.06177	0.95145	0.03183
7	2.77405	0.03791	0.97067	0.02503
8	3.17029	0.02254	0.98229	0.02352
				1.0

График плотности показательного распределения, наложенный на
гистограмму относительных частот

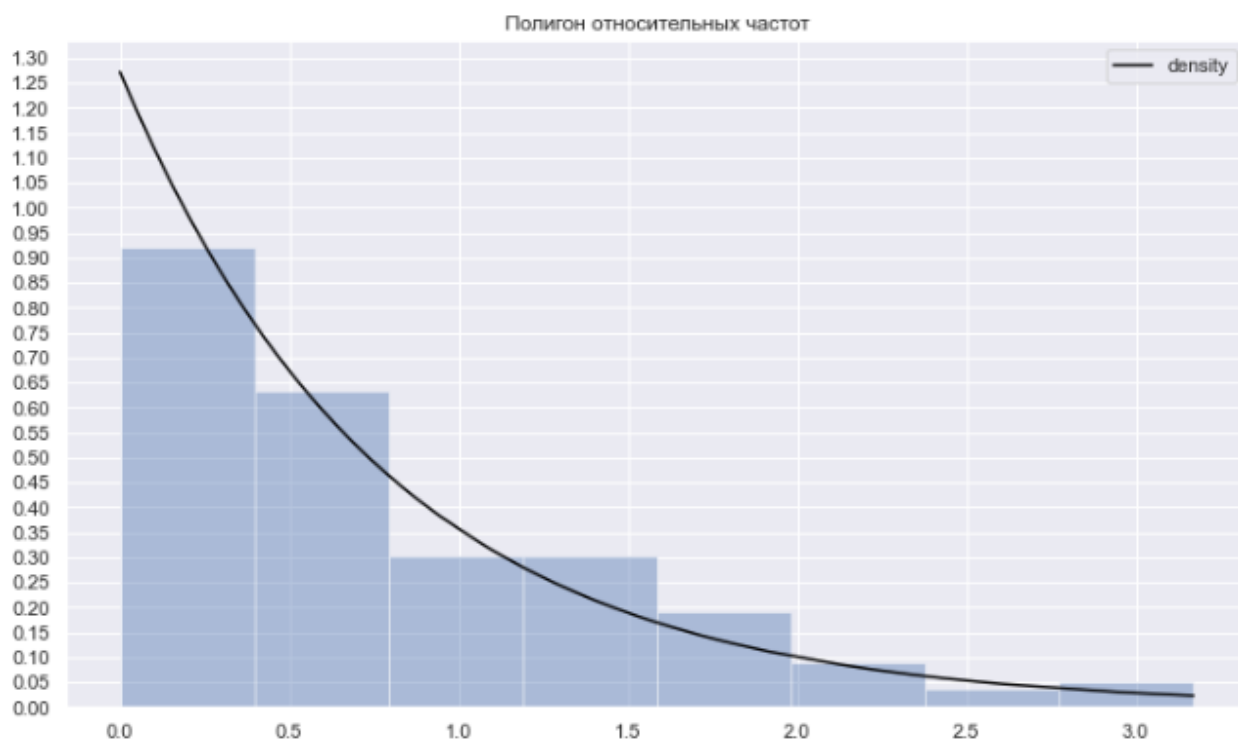


Таблица 2.3

k	Интервал	W_k	p_k^*	$ W_i - P_i $	$\frac{N(W_i - P_i)^2}{P_i}$
1	[0, 0.39664]	0.455	0.39627	0.03127	0.49336
2	(0.39664, 0.79287]	0.290	0.29905	0.01095	0.10034
3	(0.79287, 1.18911]	0.100	0.14440	0.02440	0.82461
4	(1.18911, 1.58535]	0.075	0.08722	0.03278	2.46327
5	(1.58535, 1.98158]	0.055	0.05269	0.02231	1.89016
6	(1.98158, 2.37782]	0.005	0.03183	0.00317	0.06333
7	(2.37782, 2.77405]	0.005	0.01922	0.00422	0.18559
8	(2.77405, 3.17029]	0.015	0.01772	0.00229	0.05897
		1.0	0.98839	0.03878	6.07963

Оценка $\tilde{\lambda} = 1.27224$

Выборочное значение $\chi^2 = 6.07963$

Число степеней свободы: $l = 6$

Задание 3-3

Выборка из файла MC_D_Norm.pdf

-1,35667	1,24143	1,11622	1,04492	1,21657	0,20674	0,65337	1,86252	2,58339	1,77909
1,21854	1,13874	1,23066	1,81092	2,11828	1,12695	0,12863	0,07847	0,53916	-0,22534
-0,26669	1,46225	2,50329	1,69518	3,26519	2,67786	-1,15486	2,95941	2,30745	3,65220
3,38956	2,45604	1,11439	3,03928	1,88113	0,32436	0,87917	0,92550	3,48448	2,88210
2,29158	1,37931	-0,40283	3,49580	0,89425	1,69221	1,90696	1,70863	2,65716	0,91428
1,70407	1,06835	-0,64476	1,63875	1,30391	2,74391	2,35218	2,96503	3,16807	1,82582
3,98761	0,88649	2,17991	-0,19314	2,85400	1,95464	0,73220	1,66247	-0,63438	1,98756
2,33429	1,69929	1,43285	0,95037	1,45812	0,22376	2,17701	3,63431	1,07051	3,72341
1,61555	1,86123	0,99148	2,76247	1,11735	1,38197	2,94673	2,84405	3,30726	0,96246
-0,94604	1,18851	1,39369	1,90391	1,32754	1,99268	2,89976	2,03392	3,22911	2,24807
0,65143	2,50811	3,18195	2,31193	2,03432	2,56746	0,94054	-0,77783	-0,60730	-0,44290
2,42170	1,85300	1,09450	0,52847	5,03770	0,21676	0,85696	-0,19580	1,03833	0,45559
2,09296	2,32374	1,52896	3,30723	2,01375	-0,88709	3,13175	1,16078	0,90832	1,72834
1,60963	1,24670	1,18159	1,27417	-0,17548	1,74392	2,13226	1,41718	2,31230	1,83965
2,46235	1,00487	-0,66618	3,40966	2,41535	0,65934	1,61713	1,25996	1,06796	1,94102
1,70226	3,09229	0,94100	1,65236	0,05914	1,69350	0,58079	2,62623	1,62008	1,15426
2,67259	1,15310	0,42525	1,36697	3,05925	1,50977	0,48056	2,10905	2,33621	1,61193
0,54910	1,28716	0,97138	1,71797	3,67019	0,96956	0,92596	0,76151	0,07952	0,78251
2,50061	2,11001	0,71519	2,51042	1,75756	-0,01124	2,44259	-0,79860	0,66993	0,71738
1,95591	1,16495	1,09041	2,85367	1,82497	2,78924	0,68828	1,00493	1,02521	2,54781

Сортированная выборка:

-1.35667	-1.15486	-0.94604	-0.88709	-0.7986	-0.77783	-0.66618	-0.64476	-0.63438	-0.6073
-0.4429	-0.40283	-0.26669	-0.22534	-0.1958	-0.19314	-0.17548	-0.01124	0.05914	0.07847
0.07952	0.12863	0.20674	0.21676	0.22376	0.32436	0.42525	0.45559	0.48056	0.52847
0.53916	0.5491	0.58079	0.65143	0.65337	0.65934	0.66993	0.68828	0.71519	0.71738
0.7322	0.76151	0.78251	0.85696	0.87917	0.88649	0.89425	0.90832	0.91428	0.9255
0.92596	0.94054	0.941	0.95037	0.96246	0.96956	0.97138	0.99148	1.00487	1.00493
1.02521	1.03833	1.04492	1.06796	1.06835	1.07051	1.09041	1.0945	1.11439	1.11622
1.11735	1.12695	1.13874	1.1531	1.15426	1.16078	1.16495	1.18159	1.18851	1.21657
1.21854	1.23066	1.24143	1.2467	1.25996	1.27417	1.28716	1.30391	1.32754	1.36697
1.37931	1.38197	1.39369	1.41718	1.43285	1.45812	1.46225	1.50977	1.52896	1.60963
1.61193	1.61555	1.61713	1.62008	1.63875	1.65236	1.66247	1.69221	1.6935	1.69518
1.69929	1.70226	1.70407	1.70863	1.71797	1.72834	1.74392	1.75756	1.77909	1.81092
1.82497	1.82582	1.83965	1.853	1.86123	1.86252	1.88113	1.90391	1.90696	1.94102
1.95464	1.95591	1.98756	1.99268	2.01375	2.03392	2.03432	2.09296	2.10905	2.11001
2.11828	2.13226	2.17701	2.17991	2.24807	2.29158	2.30745	2.31193	2.3123	2.32374
2.33429	2.33621	2.35218	2.41535	2.4217	2.44259	2.45604	2.46235	2.50061	2.50329
2.50811	2.51042	2.54781	2.56746	2.58339	2.62623	2.65716	2.67259	2.67786	2.74391

2.76247	2.78924	2.84405	2.85367	2.854	2.8821	2.89976	2.94673	2.95941	2.96503
3.03928	3.05925	3.09229	3.13175	3.16807	3.18195	3.22911	3.26519	3.30723	3.30726
3.38956	3.40966	3.48448	3.4958	3.63431	3.6522	3.67019	3.72341	3.98761	5.0377

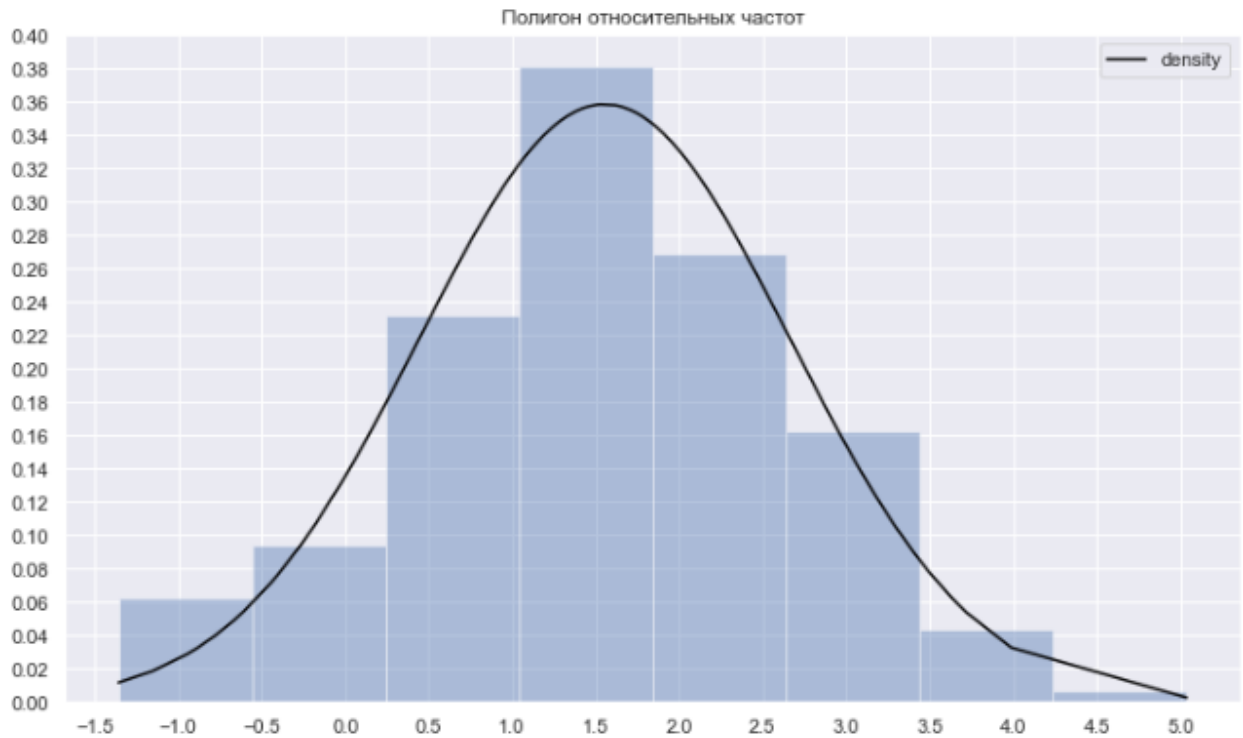
Таблица 3.1 Группированная выборка

Интервалы	n_i	w_i
$[-1.35667, -0.55737]$	10	0.050
$(-0.55737, 0.24192]$	15	0.075
$(0.24192, 1.04122]$	37	0.185
$(1.04122, 1.84052]$	61	0.305
$(1.84052, 2.63981]$	43	0.215
$(2.63981, 3.43911]$	26	0.130
$(3.43911, 4.22840]$	7	0.035
$(4.22840, 5.03770]$	1	0.005
	200	1.0

Оценка $a = 1.55347$, $\sigma^2 = 1.23681$

k	a_k	$\frac{a_k - \tilde{a}}{\sigma}$	$\frac{1}{\sigma} \varphi\left(\frac{a_k - \tilde{a}}{\sigma}\right)$	$\Phi\left(\frac{a_k - \tilde{a}}{\sigma}\right)$	p_k^*
0	-1.35667	-2.61675	0.01169	0.00444	-
1	-0.55737	-1.89803	0.05922	0.02885	0.02885
2	0.24192	-1.17932	0.17896	0.11913	0.09029
3	1.04122	-0.46060	0.32262	0.32254	0.20341
4	1.84052	0.25811	0.34697	0.60184	0.27930
5	2.63981	0.97682	0.22262	0.83567	0.23383
6	3.43911	1.69554	0.08521	0.95501	0.11934
7	4.22840	2.41425	0.01946	0.99212	0.03710
8	5.03770	3.13297	0.00265	0.99913	0.00788

График плотности нормального распределения, наложенный на гистограмму
относительных частот



k	Интервал	W_k	p_k^*	$ W_i - P_i $	$\frac{N(W_i - P_i)^2}{P_i}$
1	$[-1.35667, -0.55737]$	0.050	0.02885	0.05000	3.10102
2	$(-0.55737, 0.24192]$	0.075	0.09029	0.04615	0.51785
3	$(0.24192, 1.04122]$	0.185	0.20341	0.09471	0.33325
4	$(1.04122, 1.84052]$	0.305	0.27930	0.10159	0.47296
5	$(1.84052, 2.63981]$	0.215	0.23383	0.06430	0.30327
6	$(2.63981, 3.43911]$	0.130	0.11934	0.10383	0.19044
7	$(3.43911, 4.22840]$	0.035	0.03710	0.08434	0.02377
8	$(4.22840, 5.03770]$	0.005	0.00788	0.03210	0.21052
		1.0	1.0	0.10383	5.15308

Оценка $a = 1.55347$, $\sigma^2 = 1.23681$

Выборочное значение $\chi^2 = 5.15308$

Число степеней свободы: $l = 5$

Задание 3-4

$$a = 2.80; b = 9.22$$

Выборка из файла MC_D_Unif.pdf

7,53713	7,21191	6,54155	5,46763	3,80852	4,80675	7,96127	8,44730	3,61542	7,07555
4,82730	4,42786	6,10797	5,87239	3,40263	7,83918	8,40873	6,45319	8,67526	6,69085
3,59350	4,14326	4,03823	4,32327	4,92292	5,22313	7,72312	4,18622	6,12659	5,90653
5,56641	8,33367	7,72154	5,98299	7,64743	4,93666	7,20632	3,41044	7,80789	5,49988
3,57112	4,78963	3,72907	6,51599	5,26399	4,30711	8,08362	5,35017	7,89256	6,59192
8,55450	3,23254	5,72092	5,31765	6,74830	7,59140	6,88816	7,48597	6,99950	8,82651
3,32859	3,28731	5,46783	3,72166	3,87040	7,74392	8,22702	7,17477	6,10885	3,57052
7,52411	5,91301	5,95241	7,67310	9,03887	8,97141	2,96488	4,31540	7,33455	4,00579
8,06486	4,48665	7,51290	5,09658	8,88073	3,38889	5,54517	8,59340	6,08190	9,15989
7,60355	6,45877	5,65832	7,11989	4,32893	3,55518	4,63148	4,63172	5,81277	6,10399
3,60400	5,99701	3,70392	7,07217	7,83363	2,98799	8,98106	8,43733	5,32875	2,86499
4,40038	8,69548	7,85734	5,47684	3,75562	8,62093	7,51619	5,95630	6,74293	7,88239
5,16308	4,54473	4,82945	7,74756	5,03993	4,96901	3,07884	6,64324	3,25258	3,02852
9,18555	8,45236	8,95970	3,82579	2,88682	6,05582	6,82068	5,62362	2,85235	4,85316
5,77514	9,03091	8,84762	5,26837	5,45110	4,06294	6,31183	7,22077	6,84965	7,36376
6,38709	7,63516	6,10259	4,41490	8,55890	4,74235	8,84209	5,82502	4,48385	4,41361
3,21200	6,11752	6,78557	7,96075	3,18140	3,02000	5,52733	8,01132	4,94413	4,43032
4,52787	3,02694	7,85901	3,35438	3,54182	6,94069	8,71815	5,71557	8,08180	8,16893
5,02921	8,68781	3,43287	4,76188	8,24131	5,69803	2,83259	7,48577	4,11317	7,70866
3,25875	3,93375	3,75499	8,55144	2,89331	3,67115	8,19667	4,60069	3,11761	6,57222

Сортированная выборка:

2.83259	2.85235	2.86499	2.88682	2.89331	2.96488	2.98799	3.02	3.02694	3.02852
3.07884	3.11761	3.1814	3.212	3.23254	3.25258	3.25875	3.28731	3.32859	3.35438
3.38889	3.40263	3.41044	3.43287	3.54182	3.55518	3.57052	3.57112	3.5935	3.604
3.61542	3.67115	3.70392	3.72166	3.72907	3.75499	3.75562	3.80852	3.82579	3.8704
3.93375	4.00579	4.03823	4.06294	4.11317	4.14326	4.18622	4.30711	4.3154	4.32327
4.32893	4.40038	4.41361	4.4149	4.42786	4.43032	4.48385	4.48665	4.52787	4.54473
4.60069	4.63148	4.63172	4.74235	4.76188	4.78963	4.80675	4.8273	4.82945	4.85316
4.92292	4.93666	4.94413	4.96901	5.02921	5.03993	5.09658	5.16308	5.22313	5.26399
5.26837	5.31765	5.32875	5.35017	5.4511	5.46763	5.46783	5.47684	5.49988	5.52733
5.54517	5.56641	5.62362	5.65832	5.69803	5.71557	5.72092	5.77514	5.81277	5.82502
5.87239	5.90653	5.91301	5.95241	5.9563	5.98299	5.99701	6.05582	6.0819	6.10259
6.10399	6.10797	6.10885	6.11752	6.12659	6.31183	6.38709	6.45319	6.45877	6.51599
6.54155	6.57222	6.59192	6.64324	6.69085	6.74293	6.7483	6.78557	6.82068	6.84965

6.88816	6.94069	6.9995	7.07217	7.07555	7.11989	7.17477	7.20632	7.21191	7.22077
7.33455	7.36376	7.48577	7.48597	7.5129	7.51619	7.52411	7.53713	7.5914	7.60355
7.63516	7.64743	7.6731	7.70866	7.72154	7.72312	7.74392	7.74756	7.80789	7.83363
7.83918	7.85734	7.85901	7.88239	7.89256	7.96075	7.96127	8.01132	8.06486	8.0818
8.08362	8.16893	8.19667	8.22702	8.24131	8.33367	8.40873	8.43733	8.4473	8.45236
8.55144	8.5545	8.5589	8.5934	8.62093	8.67526	8.68781	8.69548	8.71815	8.82651
8.84209	8.84762	8.88073	8.9597	8.97141	8.98106	9.03091	9.03887	9.15989	9.18555

Таблица 4.1 Группированная выборка

Интервалы	n_i	w_i
[2.83259, 3.62672]	31	0.155
(3.62672, 4.42083]	23	0.115
(4.42083, 5.21495]	24	0.120
(5.21495, 6.00907]	29	0.145
(6.00907, 6.80319]	21	0.105
(6.80319, 7.59731]	21	0.105
(7.59731, 8.39143]	27	0.135
(8.39143, 9.18555]	24	0.120
	200	1.0

k	Интервал	W_k	p_k^*	$ W_i - P_i $	$\frac{N(W_i - P_i)^2}{P_i}$
1	[2.83259, 3.62672]	0.155	0.125	0.030	1.44
2	(3.62672, 4.42083]	0.115	0.125	0.010	0.16
3	(4.42083, 5.21495]	0.120	0.125	0.005	0.04
4	(5.21495, 6.00907]	0.145	0.125	0.020	0.64
5	(6.00907, 6.80319]	0.105	0.125	0.020	0.64
6	(6.80319, 7.59731]	0.105	0.125	0.020	0.64
7	(7.59731, 8.39143]	0.135	0.125	0.010	0.16
8	(8.39143, 9.18555]	0.120	0.125	0.005	0.04
		1.0	1.0	0.03	3.76

График плотности равномерного распределения, наложенный на
гистограмму относительных частот



Выборочное значение $\chi^2 = 3.76$

Число степеней свободы: $l = 7$

Задание 3-5

График Эмпирической функции распределения $F_N(x)$ и график функции распределения $F(x)$

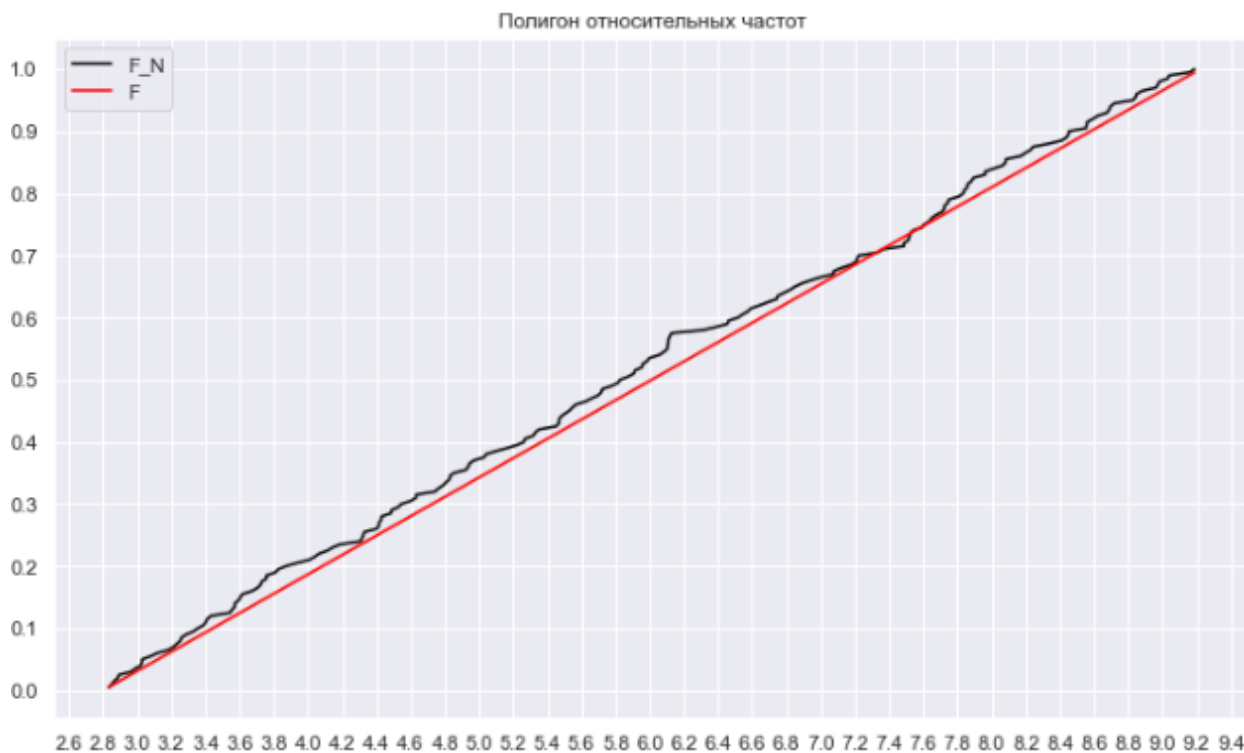


Таблица 5.1

a	b	N	D_N	$D_N\sqrt{N}$	x^*	$F(x^*)$	$F_N(x^*)$	$F_N(x^* - 0)$
2.90	9.22	200	0.05684	0.80383	6.12659	0.51816	0.575	0.57

Анализ результатов

Таблица критических значений $\chi^2_{кр,\alpha}(l)$ при уровне значимости $\alpha = 0.05$

l	1	2	3	4	5	6	7	8
$\chi^2_{кр,\alpha}(l)$	3,8	6,0	7,8	9,5	11,1	12,6	14,1	15,5

Таблица критических значений критерия Колмогорова k_α

α	0.01	0.02	0.05	0.1	0.2
k_α	1.63	1.57	1.36	1.22	1.07

Проверка гипотез с помощью критерия χ^2

Биномиальное распределение

$$\chi^2_{\text{в}} = 4.07628, l = 5, \chi^2_{\text{кр}, \alpha}(l) = 11,1$$

Так как **4.07628** \leq **11,1**, то гипотеза о соответствии выборки биномиальному распределению **не противоречит** экспериментальным данным при уровне значимости $\alpha=0.05$.

Показательное распределение

$$\chi^2_{\text{в}} = 6.07963, l = 6, \chi^2_{\text{кр}, \alpha}(l) = 12,6$$

Так как **6.07963** \leq **12,6**, то гипотеза о соответствии выборки показательному распределению **не противоречит** экспериментальным данным при уровне значимости $\alpha=0.05$.

Нормальное распределение

$$\chi^2_{\text{в}} = 5.15308, l = 5, \chi^2_{\text{кр}, \alpha}(l) = 11,1$$

Так как **5.15308** \leq **11,1**, то гипотеза о соответствии выборки нормальному распределению **не противоречит** экспериментальным данным при уровне значимости $\alpha=0.05$.

Равномерное распределение

$$\chi^2_{\text{в}} = 3.76, l = 7, \chi^2_{\text{кр}, \alpha}(l) = 14,1$$

Так как **3.76** \leq **14,1**, то гипотеза о соответствии выборки равномерному распределению на отрезке [2.80, 9.22] **не противоречит** экспериментальным данным при уровне значимости $\alpha=0.05$.

Проверка гипотез с помощью критерия Колмогорова

Равномерное распределение

$$D_N \sqrt{N} = 0.80383, k_{\alpha} = 1,36$$

Так как **0.80383** \leq **1,36**, то гипотеза о соответствии выборки равномерному распределению на отрезке [2.80, 9.22] **не противоречит** экспериментальным данным при уровне значимости $\alpha=0.05$.

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 math
9. import seaborn as sns
10. from matplotlib import ticker
11. from math import factorial
12. from math import exp
13. from math import log
14. from math import inf
15. from scipy import integrate
16.
17. def Xi(distr):
18.     xi = []
19.     for item in distr:
20.         if item not in xi:
21.             xi.append(item)
22.
23.     return(xi)
24.
25. def Freq(distr, xi, size):
26.     ni=[]
27.     for j in range(len(xi)):
28.         help = 0
29.         for i in range(size):
30.             if distr[i]==xi[j]:
31.                 help += 1
32.         ni.append(help)
33.     return(ni)
34.
35. def Rel_freq(ni, size):
36.     wi = []
37.     for i in range(len(ni)):
38.         wi.append(ni[i]/size)
39.     return(wi)
40.
41. def Mean_dis(xi ,wi):
42.     X = 0
43.     for i in range (len(xi)):
44.         help = xi[i]*wi[i]
45.         X += help
```

```

46.         return (X)
47.
48.     def Moment(xi,wi):
49.         M = []
50.         for i in range (len(xi)):
51.             help = 0
52.             for j in range (len(xi)):
53.                 help += pow(xi[j],i+1)*wi[j]
54.             M.append(help)
55.         return (M)
56.
57.
58.     def chi_square(wi, pi, size):
59.         chi = [i for i in range(len(wi))]
60.         for k in range (len(wi)):
61.             chi[k] = round((size*(wi[k]-pi[k])**2)/pi[k],5)
62.         return (chi)
63.
64.     def Disp(Moment):
65.         D = Moment[1]- pow(Moment[0], 2)
66.         return (D)
67.
68.     def intervals(distr, size):
69.         m = 1+int(log(size,2))
70.         d = distr[-1] - distr[0]
71.         a = [distr[0]]
72.         for i in range (1, (m+1)):
73.             a.append(d/m+a[i-1])
74.         return a
75.
76.     def Freq_inter(distr, inter, size):
77.         ni=[]
78.         for j in range(1,len(inter)):
79.             count = 0
80.             h=0
81.             for i in range(size):
82.                 if j-1 == 0:
83.                     if distr[i]>=inter[j-1] and distr[i]<=inter[j]:
84.                         count +=1
85.                 elif j == len(inter)-1:
86.                     if distr[i]>inter[j-1] and distr[i]<=inter[j]:
87.                         count += 1
88.                 if distr[i]>inter[j]:
89.                     count += 1
90.             else:
91.                 if distr[i]>inter[j-1] and distr[i]<=inter[j]:
92.                     count += 1
93.         ni.append(count)

```

```

94.         return(ni)
95.
96. def Mid_inter(inter):
97.     mid = []
98.     for i in range(len(inter)-1):
99.         mid.append((inter[i]+inter[i+1])/2)
100.    return(mid)
101.
102. def read_distr(files):
103.     with open(files) as file:
104.         array = [row.strip() for row in file]
105.    return (array)
106.
107. def Round_array(array):
108.     for i in range (len(array)):
109.         array[i] = round(array[i],5)
110.    return array
111.
112. X_kr = [0,0,0,0, 9.5, 11.1, 12.6, 14.1, 15.5]
113.
114. sns.set(rc={'figure.figsize':(12,7)})
115.
116. v3_1 = 46
117. n3_1 = 9
118. size = 200
119. distr3_1 = read_distr("binom.txt")
120. for i, item in enumerate(distr3_1):
121.     distr3_1[i] = int(item)
122. distr3_1.sort()
123. print(distr3_1)
124.
125. df_3_1 = pd.DataFrame({'distr3_1':distr3_1})
126.
127. xi3_1 = []
128. ni3_1 = []
129. wi3_1 = []
130. #xi - значение
131. #ni - кол-во значения
132. #sum(ni) = size
133. #wi = ni/size
134.
135. xi3_1 = Xi(distr3_1)
136. ni3_1 = Freq(distr3_1, xi3_1, size)
137. wi3_1 = Rel_freq(ni3_1, size)
138.
139. Table3_1_1 = pd.DataFrame({'Xi':xi3_1,
140.                             'Ni':ni3_1,
141.                             'Wi':wi3_1})

```

```

142. Table3_1_1
143.
144. # вычисление теор вероятности
145. M3_1 = Moment(xi3_1,wi3_1)
146. pi3_1 = [i for i in range(len(xi3_1))]
147. p3_1 = M3_1[0]/n3_1
148.
149. print('p =', round(p3_1, 5))
150.
151.
152. for k in range(len(xi3_1)):
153.     pi3_1[k] =
154.         round((factorial(n3_1)/(factorial(xi3_1[k])*factorial(n3_1-xi3_1[k])))*
155.             (p3_1**xi3_1[k])*(1-p3_1)**(n3_1-xi3_1[k]), 5)
156.
157. # Вычисление значения хи-квадрат
158. chi_square3_1 = chi_square(wi3_1, pi3_1, size)
159.
160.
161.
162. # Wi - Pi
163. diff_wp3_1 = []
164. for i in range(len(wi3_1)):
165.     diff_wp3_1.append(abs(wi3_1[i]-pi3_1[i]))
166.
167. Table3_1_2 = pd.DataFrame({'Xi':xi3_1,
168.                             'Wi':wi3_1,
169.                             'Pi':pi3_1,
170.                             'Wi-Pi':diff_wp3_1,
171.                             'chi_square':chi_square3_1})
172. Table3_1_2
173.
174. print ('Сумма wk:',round(sum(wi3_1),5))
175. print ('Сумма pk:',round(sum(pi3_1),5))
176. print('max |Wk-Pk|:',max(diff_wp3_1))
177. print('Хи квадрат =', sum(chi_square3_1))
178.
179. if (sum(chi_square3_1)<= X_kr[len(wi3_1)-2]):
180.     print ('''гипотеза о соответствии выборки биномиальному
181.         распределению
182. НЕ ПРОТИВОРЕЧИТ экспериментальным данным при уровне значимости
183.         0,05''')
184. else:
185.     print ('''гипотеза о соответствии выборки биномиальному
186.         распределению
187. ПРОТИВОРЕЧИТ экспериментальным данным при уровне значимости 0,05''')
188.

```

```

186. fig, fir = plt.subplots()
187. fir.xaxis.set_major_locator(ticker.MultipleLocator(1))
188. fir.yaxis.set_major_locator(ticker.MultipleLocator(0.02))
189. pi3_1 = [i for i in range(len(xi3_1))]
190.
191. for k in range(len(xi3_1)):
192.     pi3_1[k] = (factorial(n3_1)/(factorial(xi3_1[k])*factorial(n3_1-
193. xi3_1[k]))) * p3_1**xi3_1[k] * (1-p3_1)**(n3_1-xi3_1[k])
194. fir.plot(xi3_1, wi3_1, 'black', label='относительные частоты')
195. fir.plot(xi3_1, pi3_1, 'red', label='вероятности')
196. plt.title("Полигон относительных частот")
197. plt.grid(True)
198. plt.legend(loc='best')
199.
200. def Density3_2(inter, lam):
201.     foo=[i for i in range(len(inter))]
202.     for i in range (len(inter)):
203.         foo[i] = lam*exp(-lam*inter[i])
204.     return (foo)
205.
206. def func3_2(inter, lam):
207.     func=[i for i in range(len(inter))]
208.     for i in range (len(inter)):
209.         func[i] = 1-exp(-lam*inter[i])
210.     return (func)
211.
212. def Pi3_2(func):
213.     P=[i for i in range(len(func)-1)]
214.     P[-1] = 1 - func[-1]
215.     for i in range(len(func)-2):
216.         P[i] = func[i+1] - func[i]
217.     return P
218.
219. v3_2 = 46
220. size = 200
221. distr3_2 = read_distr("expon.txt")
222.
223. # делим из str - float
224. for i in range(len(distr3_2)):
225.     distr3_2[i] = float(str(distr3_2[i]).replace(",", "."))
226.
227. for i, item in enumerate(distr3_2):
228.     distr3_2[i] = float(item)
229.
230. distr3_2.sort()
231. print (distr3_2)
232.
233. df_3_2 = pd.DataFrame({'distr3_2':distr3_2})

```

```

234.
235. xi3_2 = []
236. ni3_2 = []
237. wi3_2 = []
238. sk3_2 = []
239. #xi - значение
240. #ni - кол-во значения
241. #sum(ni) = size
242. #wi = ni/size
243. #sk = sum(wj)
244.
245. xi3_2 = Xi(distr3_2)
246. ni3_2 = Freq(distr3_2, xi3_2, size)
247. wi3_2 = Rel_freq(ni3_2, size)
248.
249.
250. # Построение интервалов
251. inter3_2 = intervals(distr3_2, size)
252. inter3_2[0] = 0
253.
254. interval_all3_2 = [[inter3_2[0], inter3_2[1]],
255.                     [inter3_2[1], inter3_2[2]],
256.                     [inter3_2[2], inter3_2[3]],
257.                     [inter3_2[3], inter3_2[4]],
258.                     [inter3_2[4], inter3_2[5]],
259.                     [inter3_2[5], inter3_2[6]],
260.                     [inter3_2[6], inter3_2[7]],
261.                     [inter3_2[7], inter3_2[8]],
262.                     ]
263.
264. inter3_2_round = Round_array(inter3_2)
265. interval_all3_2_round = [
266.                     [inter3_2_round[0], inter3_2_round[1]],
267.                     [inter3_2_round[1], inter3_2_round[2]],
268.                     [inter3_2_round[2], inter3_2_round[3]],
269.                     [inter3_2_round[3], inter3_2_round[4]],
270.                     [inter3_2_round[4], inter3_2_round[5]],
271.                     [inter3_2_round[5], inter3_2_round[6]],
272.                     [inter3_2_round[6], inter3_2_round[7]],
273.                     [inter3_2_round[7], inter3_2_round[8]],
274.                     ]
275.
276. # интервальная частота
277. ni_inter3_2 = Freq_inter(distr3_2, inter3_2, size)
278.
279. # интервальная отн частота
280. wi_inter3_2 = Rel_freq(ni_inter3_2, size)
281.

```

```

282. # центры интервалов
283. x_x3_2 = Mid_inter(inter3_2)
284.
285. Table3_2_1 = pd.DataFrame({'Intervals':interval_all3_2_round,
286.                             'Ni':ni_inter3_2,
287.                             'Wi':wi_inter3_2})
288. Table3_2_1
289.
290. # находим плотность и функцию
291. f3_2 = Density3_2(inter3_2, lam3_2)
292. F3_2 = func3_2(inter3_2, lam3_2)
293. pi3_2 = Pi3_2(F3_2)
294.
295. func3_2_1 = [i for i in range(len(inter3_2))]
296. func3_2_1[0] = 0
297. for i in range (1,len(inter3_2)):
298.     func3_2_1[i] = pi3_2[i-1]
299.
300. Table3_2_2 = pd.DataFrame({'ak':inter3_2,
301.                             'f(ak, lam)':f3_2,
302.                             'F(ak, lam)':F3_2,
303.                             'pi':func3_2_1})
304. Table3_2_2
305.
306. diff_wp3_2 = []
307. for i in range(len(wi_inter3_2)):
308.     diff_wp3_2.append(abs(wi_inter3_2[i]-pi3_2[i]))
309.
310. chi_square3_2 = chi_square(wi_inter3_2, pi3_2, size)
311.
312. Table3_2_3 = pd.DataFrame({'Intervals':interval_all3_2_round,
313.                             'Wi':Round_array(wi_inter3_2),
314.                             'Pk':Round_array(pi3_2),
315.                             '|Wk-Pk|':Round_array(diff_wp3_2),
316.                             'X^2':chi_square3_2})
317. Table3_2_3
318.
319. fig, fir = plt.subplots()
320. fir.xaxis.set_major_locator(ticker.MultipleLocator(0.5))
321. fir.yaxis.set_major_locator(ticker.MultipleLocator(0.05))
322.
323. fir.plot(distr3_2, dens3_2, 'black', label='density')
324. sns.distplot(distr3_2, bins = 8, norm_hist=True, kde=False)
325. plt.title("Полигон относительных частот")
326. plt.grid(True)
327. plt.legend(loc='best')
328.
329. def PHI(x):

```



```

330.     return (sps.norm.cdf(x))
331.
332. def phi(t):
333.     return (sps.norm.pdf(t))
334.
335. def Pi3_3(inter,a,std):
336.     P=[i for i in range(len(inter))]
337.     P[0] = 0
338.     P[1] = PHI((inter[1]-a)/std)
339.     P[-1] = 1 - PHI((inter[-2]-a)/std)
340.     for i in range(2,len(inter)-1):
341.         P[i] = PHI((inter[i]-a)/std) - PHI((inter[i-1]-a)/std)
342.     return P
343.
344. distr3_3 = read_distr("norm.txt")
345. # делем из str - float
346. for i in range(len(distr3_3)):
347.     distr3_3[i] = float(str(distr3_3[i]).replace(",","."))
348.
349. for i, item in enumerate(distr3_3):
350.     distr3_3[i] = float(item)
351.
352. distr3_3.sort()
353. print(distr3_3)
354. df3_3 = pd.DataFrame({'df3_3':distr3_3})
355.
356. inter3_3 = intervals(distr3_3, size)
357.
358. interval_all3_3 = [
359.     [inter3_3[0], inter3_3[1]],
360.     [inter3_3[1], inter3_3[2]],
361.     [inter3_3[2], inter3_3[3]],
362.     [inter3_3[3], inter3_3[4]],
363.     [inter3_3[4], inter3_3[5]],
364.     [inter3_3[5], inter3_3[6]],
365.     [inter3_3[6], inter3_3[7]],
366.     [inter3_3[7], inter3_3[8]],
367. ]
368.
369. inter3_3_round = Round_array(inter3_3)
370. interval_all3_3_round = [
371.     [inter3_3_round[0], inter3_3_round[1]],
372.     [inter3_3_round[1], inter3_3_round[2]],
373.     [inter3_3_round[2], inter3_3_round[3]],
374.     [inter3_3_round[3], inter3_3_round[4]],
375.     [inter3_3_round[4], inter3_3_round[5]],
376.     [inter3_3_round[5], inter3_3_round[6]],
377.     [inter3_3_round[6], inter3_3_round[7]],

```

```

378.                [inter3_3_round[7], inter3_3_round[8]],
379.                ]
380.
381. # интервальная частота
382. ni_inter3_3 = Freq_inter(distr3_3, inter3_3, size)
383.
384. # интервальная отн частота
385. wi_inter3_3 = Rel_freq(ni_inter3_3, size)
386.
387. Table3_3_1 = pd.DataFrame({'Intervals':interval_all3_3_round,
388.                            'Ni':ni_inter3_3,
389.                            'Wi':wi_inter3_3})
390. Table3_3_1
391.
392. xi3_3 = []
393. ni3_3 = []
394. wi3_3 = []
395. #xi - значение
396. #ni - кол-во значения
397. #wi = ni/size
398.
399. xi3_3 = Xi(distr3_3)
400. ni3_3 = Freq(distr3_3, xi3_3, size)
401. wi3_3 = Rel_freq(ni3_3, size)
402.
403. # вычисление теор вероятности
404. M3_3 = Moment(xi3_3,wi3_3)
405.
406.
407. a3_3 = M3_3[0]
408. print('Мат. ожидание: ', a3_3)
409. #дисперсия
410. var3_3 = Disp(M3_3)
411. print('Дисперсия: ', var3_3)
412. #сред.квадратичное
413. std3_3 = var3_3**(1/2)
414. print('Ср.квадратичное: ',std3_3)
415.
416. # (Ak-a)/q
417. func3_3_1 = [i for i in range(len(inter3_3))]
418. for i in range (len(inter3_3)):
419.     func3_3_1[i] = (inter3_3[i]-a3_3)/std3_3
420.
421. #1/q*phi
422. func3_3_2 = [i for i in range(len(inter3_3))]
423. for i in range (len(inter3_3)):
424.     func3_3_2[i] = (1/std3_3)*phi(func3_3_1[i])
425.

```

```

426. #PHI
427. func3_3_3 = [i for i in range(len(inter3_3))]
428. for i in range (len(inter3_3)):
429.     func3_3_3[i] = PHI(func3_3_1[i])
430.
431. #Pk
432. pi3_3 = Pi3_3(inter3_3,a3_3,std3_3)
433. func3_3_4 = [i for i in range(len(inter3_3))]
434.
435.
436. Table3_3_2 = pd.DataFrame({'Ak':Round_array(inter3_3),
437.                             '(Ak-a)/q':Round_array(func3_3_1),
438.                             '1/q*phi':Round_array(func3_3_2),
439.                             'PHI':Round_array(func3_3_3),
440.                             'Pk':Round_array(pi3_3)})
441. Table3_3_2
442.
443. diff_wp3_3 = []
444. for i in range(len(wi_inter3_3)):
445.     diff_wp3_3.append(abs(wi_inter3_3[i]-pi3_3[i]))
446.
447. func3_3_5 = [i for i in range(len(pi3_3)-1)]
448. for i in range(len(pi3_3)-1):
449.     func3_3_5[i] = pi3_3[i+1]
450.
451. chi_square3_3 = chi_square(wi_inter3_3, func3_3_5, size)
452.
453. Table3_3_3 = pd.DataFrame({'Intervals':interval_all3_3_round,
454.                             'Wi':Round_array(wi_inter3_3),
455.                             'Pk':Round_array(func3_3_5),
456.                             '|Wk-Pk|':Round_array(diff_wp3_3),
457.                             'X^2':chi_square3_3})
458. Table3_3_3
459.
460. fig, fir = plt.subplots()
461. fir.xaxis.set_major_locator(ticker.MultipleLocator(0.5))
462. fir.yaxis.set_major_locator(ticker.MultipleLocator(0.02))
463.
464. fir.plot(distr3_3, dens3_3, 'black',label='density')
465. sns.distplot(distr3_3, bins = 8, norm_hist=True, kde=False)
466. plt.title("ПОЛИГОН ОТНОСИТЕЛЬНЫХ ЧАСТОТ")
467. plt.grid(True)
468. plt.legend(loc='best')
469.
470. def Pi3_4(inter):
471.     P=[i for i in range(len(inter)-1)]
472.     for i in range (len(inter)-1):
473.         P[i] = 1/(len(inter)-1)

```

```

474.     return P
475.
476. def Density3_4(x,a,b):
477.     if (a<=x and x<=b):
478.         return (1/(b-a))
479.     else:
480.         return 0
481.
482. a = 2.80
483. b = 9.22
484.
485. distr3_4 = read_distr("unif.txt")
486. # делим из str - float
487. for i in range(len(distr3_4)):
488.     distr3_4[i] = float(str(distr3_4[i]).replace(",","."))
489.
490. for i, item in enumerate(distr3_4):
491.     distr3_4[i] = float(item)
492.
493. distr3_4.sort()
494. print (distr3_4)
495.
496. inter3_4 = intervals(distr3_4, size)
497.
498. interval_all3_4 = [
499.     [inter3_4[0], inter3_4[1]],
500.     [inter3_4[1], inter3_4[2]],
501.     [inter3_4[2], inter3_4[3]],
502.     [inter3_4[3], inter3_4[4]],
503.     [inter3_4[4], inter3_4[5]],
504.     [inter3_4[5], inter3_4[6]],
505.     [inter3_4[6], inter3_4[7]],
506.     [inter3_4[7], inter3_4[8]],
507. ]
508.
509. inter3_4_round = Round_array(inter3_4)
510. interval_all3_4_round = [
511.     [inter3_4_round[0], inter3_4_round[1]],
512.     [inter3_4_round[1], inter3_4_round[2]],
513.     [inter3_4_round[2], inter3_4_round[3]],
514.     [inter3_4_round[3], inter3_4_round[4]],
515.     [inter3_4_round[4], inter3_4_round[5]],
516.     [inter3_4_round[5], inter3_4_round[6]],
517.     [inter3_4_round[6], inter3_4_round[7]],
518.     [inter3_4_round[7], inter3_4_round[8]],
519. ]
520.
521. # интервальная частота

```

```

522. ni_inter3_4 = Freq_inter(distr3_4, inter3_4, size)
523.
524. # интервальная отн частота
525. wi_inter3_4 = Rel_freq(ni_inter3_4, size)
526.
527. Table3_4_1 = pd.DataFrame({'Intervals':interval_all3_4_round,
528.                             'Ni':ni_inter3_4,
529.                             'Wi':wi_inter3_4})
530. Table3_4_1
531.
532. pi3_4 = Pi3_4(inter3_4)
533.
534. diff_wp3_4 = []
535. for i in range(len(wi_inter3_4)):
536.     diff_wp3_4.append(abs(wi_inter3_4[i]- pi3_4[i]))
537.
538. chi_square3_4 = chi_square(wi_inter3_4, pi3_4, size)
539.
540. Table3_4_2 = pd.DataFrame({'Intervals':interval_all3_4_round,
541.                             'Wi':Round_array(wi_inter3_4),
542.                             'Pk':Round_array(pi3_4),
543.                             '|Wk-Pk|':Round_array(diff_wp3_4),
544.                             'X^2':chi_square3_4})
545. Table3_4_2
546.
547. xi3_4 = Xi(distr3_4)
548. fig, fir = plt.subplots()
549. fir.xaxis.set_major_locator(ticker.MultipleLocator(0.5))
550. fir.yaxis.set_major_locator(ticker.MultipleLocator(0.01))
551.
552. fir.plot(xi3_4, dens3_4, 'black',label='density')
553. sns.distplot(distr3_4, bins = 8, norm_hist=True, kde=False)
554. plt.title("Полигон относительных частот")
555. plt.grid(True)
556. plt.legend(loc='best')
557.
558. distr3_5 = distr3_4
559. a = a
560. b = b
561. print (distr3_5)
562. df3_5 = pd.DataFrame({'data3_5':distr3_5})
563. df3_5.data3_5.round(5)[150:200]
564.
565. inter3_5 = intervals(distr3_5, size)
566.
567. interval_all3_5 = [
568.     [inter3_5[0], inter3_5[1]],
569.     [inter3_5[1], inter3_5[2]],

```

```

570.             [inter3_5[2], inter3_5[3]],
571.             [inter3_5[3], inter3_5[4]],
572.             [inter3_5[4], inter3_5[5]],
573.             [inter3_5[5], inter3_5[6]],
574.             [inter3_5[6], inter3_5[7]],
575.             [inter3_5[7], inter3_5[8]],
576.         ]
577.
578. inter3_5_round = Round_array(inter3_5)
579. interval_all3_5_round = [
580.             [inter3_5_round[0], inter3_5_round[1]],
581.             [inter3_5_round[1], inter3_5_round[2]],
582.             [inter3_5_round[2], inter3_5_round[3]],
583.             [inter3_5_round[3], inter3_5_round[4]],
584.             [inter3_5_round[4], inter3_5_round[5]],
585.             [inter3_5_round[5], inter3_5_round[6]],
586.             [inter3_5_round[6], inter3_5_round[7]],
587.             [inter3_5_round[7], inter3_5_round[8]],
588.         ]
589.
590. interval_all3_5
591.
592. def F_N(size):
593.     array = []
594.     for i in range(size):
595.         array.append((i+1)/size)
596.     return(array)
597.
598.
599. def F_N0(size):
600.     array = []
601.     for i in range(size):
602.         array.append((i)/size)
603.     return(array)
604.
605.
606. def F3_5(x,a,b):
607.     if (x<a):
608.         return 0
609.     elif (a<=x and x<=b):
610.         return (x-a)/(b-a)
611.     else:
612.         return 1
613.
614. def D_N(funcN, func0, func):
615.     Max = []
616.     for i in range(200):
617.         res1 = abs(funcN[i]-func[i])

```

```

618.         res2 = abs(func0[i]-func[i])
619.         if res1 > res2:
620.             Max.append(res1)
621.         else:
622.             Max.append(res2)
623.         for i in range(200):
624.             if max(Max) == Max[i]:
625.                 help1 = i
626.             print (help1)
627.         return (max(Max))
628.
629. func3_5_1 = F_N(size)
630. func3_5_2 = F_N0(size)
631.
632. func3_5_3 = []
633. for el in distr3_5:
634.     func3_5_3.append(F3_5(el, a, b))
635.
636. print('x*', distr3_5[114])
637. print('f(x*)', func3_5_3[114])
638. print('f_N(x*)', func3_5_1[114])
639. print('f_N0(x*)', func3_5_2[114])
640.
641. if ((dn * math.sqrt(size)) <= 1.36):
642.     print ('''гипотеза о соответствии выборки биномиальному
распределению
643. НЕ ПРОТИВОРЕЧИТ экспериментальным данным при уровне  значимости
0,05''')
644. else:
645.     print ('''гипотеза о соответствии выборки биномиальному
распределению
646. ПРОТИВОРЕЧИТ экспериментальным данным при уровне  значимости 0,05''')
647.
648. xi3_5 = Xi(distr3_5)
649. fig, fir = plt.subplots()
650. fir.xaxis.set_major_locator(ticker.MultipleLocator(0.2))
651. fir.yaxis.set_major_locator(ticker.MultipleLocator(0.1))
652.
653. fir.plot(xi3_5, func3_5_1, 'black',label='F_N')
654. fir.plot(xi3_5, func3_5_3, 'red',label='F')
655. plt.title("Полигон относительных частот")
656. plt.grid(True)
657. plt.legend(loc='best')
658.
659.
660.

```