



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»

ИНСТИТУТ КИБЕРНЕТИКИ

КАФЕДРА ВЫСШЕЙ МАТЕМАТИКИ

Лабораторная работа 3

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

Тема: **Проверка статистических гипотез с помощью критерия χ^2**

Выполнил:
Студент 4-го курса
Жолковский Д.А.

Группа: КМБО-01-16

МОСКВА 2019

Лабораторная работа по Математической статистике № 3
«Проверка статистических гипотез с помощью критерия χ^2 »

Задание I. Проверка гипотезы о нормальном распределении.

Из файла ND в соответствии с номером варианта взять выборку $\{x_1, \dots, x_N\}$. Построить группированную выборку (интервальный вариационный ряд).

Найти:

- 1) оценку математического ожидания \tilde{a}
- 2) оценку дисперсии $\tilde{\sigma}^2$

Построить

- 1) таблицу 1 следующего 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^*
2	a_2	$\frac{a_2 - \tilde{a}}{\tilde{\sigma}}$	$\frac{1}{\tilde{\sigma}} \varphi\left(\frac{a_2 - \tilde{a}}{\tilde{\sigma}}\right)$	$\Phi\left(\frac{a_2 - \tilde{a}}{\tilde{\sigma}}\right)$	p_2^*
...
m-1	a_{m-1}	$\frac{a_{m-1} - \tilde{a}}{\tilde{\sigma}}$	$\frac{1}{\tilde{\sigma}} \varphi\left(\frac{a_{m-1} - \tilde{a}}{\tilde{\sigma}}\right)$	$\Phi\left(\frac{a_{m-1} - \tilde{a}}{\tilde{\sigma}}\right)$	p_{m-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^*

- 2) график плотности нормального распределения $N(\tilde{a}, \tilde{\sigma}^2)$, наложенный на гистограмму относительных частот;
- 3) таблицу 2 следующего 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^* $	$\frac{N(w_m - p_m^*)^2}{p_m^*}$
				$\max w_k - p_k^* $	$\sum_{k=1}^m \frac{N(w_k - p_k^*)^2}{p_k^*}$

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

.

Задание II. Проверка гипотезы о равномерном распределении.

Из файла UD-1 в соответствии с номером варианта взять выборку $\{x_1, \dots, x_N\}$ и значения a и b .

Построить:

- 1) таблицу 3 следующего 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^* $	$\frac{N(w_m - p_m^*)^2}{p_m^*}$
				$\max w_k - p_k^* $	$\sum_{k=1}^m \frac{N(w_k - p_k^*)^2}{p_k^*}$

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 $0,05$.

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

Краткие теоретические сведения

При построении группированной выборки (интервального вариационного ряда) число интервалов $[a_0, a_1], (a_1, a_2], \dots, (a_{m-1}, a_m]$ определяется по формуле Стерджеса $m = 1 + [\log_2 N]$, $a_0 = X_{(1)}$, $a_m = X_{(N)}$, $d = a_m - a_0$, $a_k - a_{k-1} = d/m$.

Интервальный ряд (группированная выборка) имеет вид:

$[a_{i-1}, a_i]$	$[a_0, a_1]$...	$(a_{m-1}, a_m]$
n_i	n_1	...	n_m
w_i	w_1	...	w_m

Ассоциированный статистический ряд:

x_i^*	x_1^*	...	x_m^*
n_i	n_1	...	n_m
w_i	w_1	...	w_m

$$x_i^* = \frac{a_{i-1} + a_i}{2} - \text{середина интервала } (a_{i-1}, a_i]$$

Математическое ожидание

$$\tilde{a} = \sum_{i=1}^m x_i^* w_i$$

Дисперсия

$$\tilde{\sigma}^2 = \sum_{i=1}^m (x_i^*)^2 w_i - (\tilde{a})^2 - \frac{h^2}{12}$$

Среднее квадратичное отклонение

$$\tilde{\sigma} = \sqrt{\tilde{\sigma}^2}$$

Выборочное значение критерия χ_B^2

$\chi_B^2 = \sum_{k=1}^m \frac{N(w_k - p_k^*)^2}{p_k^*}$, где, для нормального распределения, $p_1^* = \Phi\left(\frac{a_1 - \tilde{a}}{\tilde{\sigma}}\right)$, $p_k^* = \Phi\left(\frac{a_k - \tilde{a}}{\tilde{\sigma}}\right) - \Phi\left(\frac{a_{k-1} - \tilde{a}}{\tilde{\sigma}}\right)$, $k = 2, \dots, m-1$, $p_m^* = 1 - \Phi\left(\frac{a_{m-1} - \tilde{a}}{\tilde{\sigma}}\right)$.
А для равномерного $p_k^* = \frac{1}{m}$

Равномерное распределение на отрезке $[a, b]$

Характеристика	Значение
Плотность	$f(x) = \begin{cases} 0, x \notin (a, b) \\ \frac{1}{b-a}, x \in (a, b) \end{cases}$
Функция распределения	$F(x) = \begin{cases} 0, x \leq a \\ \frac{x-a}{b-a}, x \in (a, b) \\ 1, x \geq b \end{cases}$
Математическое ожидание	$b \frac{a+b}{2}$
Дисперсия	$\frac{(b-a)^2}{12}$
Среднее квадратичное отклонение	$\frac{b-a}{2\sqrt{3}}$

Нормальное распределение

Характеристика	Значение
Плотность	$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-a)^2}{2\sigma^2}}$
Функция распределения	$\Phi\left(\frac{x-a}{\sigma}\right), \Phi(x) = \int_{-\infty}^x \varphi(t) dt$
Математическое ожидание	a
Дисперсия	σ^2
Среднее квадратичное отклонение	σ

Найденное значение критерия χ_B^2 сравнивается с критическим значением $\chi_{кр,\alpha}^2(l)$ из таблицы, где α – уровень значимости, $\alpha = 0,05$, $l = m-1$ – число степеней свободы для равномерного распределения и $l = m-3$ – число степеней свободы для нормального распределения.

Если $\chi_B^2 \leq \chi_{кр,\alpha}^2(l)$, то гипотеза о соответствии выборки равномерному распределению на отрезке $[a, b]$ (или нормальному распределению) не противоречит экспериментальным данным (может быть принята) при уровне значимости $\alpha = 0,05$.

Если $\chi_B^2 > \chi_{кр,\alpha}^2(l)$, то гипотеза о соответствии выборки равномерному распределению на отрезке $[a, b]$ (или нормальному распределению) противоречит экспериментальным данным (не может быть принята) при уровне значимости $\alpha = 0,05$.

Таблица критических значений $\chi_{кр,\alpha}^2(l)$

l	4	5	6	7	8
$\chi_{кр,\alpha}^2(l)$	9.5	11.1	12.6	14.1	15.5

Средства языка Python, которые использованы в программе расчета

В программе расчёта используются следующие средства языка:

Функции:

- `numpy.sort()` – сортирует вектор в порядке возрастания.
- `int(x)` – возвращает целую часть числа x
- `scipy.stats.norm.pdf(x)` – возвращает функцию плотности стандартного

нормального распределения с аргументом x , т.е. $pdf(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$

- `scipy.stats.norm.cdf(x)` – возвращает функцию стандартного нормального распределения с аргументом x , т.е. $cdf(x) = \int_{-\infty}^x pdf(x)$

- `pystats.uniform.pdf(x, loc, scale)` – возвращает функцию плотности равномерного распределения на отрезке $[loc; loc + scale]$ с аргументом x .

- `pystats.uniform.cdf(x, loc, scale)` – возвращает функцию равномерного распределения на отрезке $[loc; loc + scale]$ с аргументом x .

Построение графиков и гистограмм было осуществлено с помощью библиотеки `matplotlib`. А построение таблиц было реализовано средствами библиотеки `docx`.

Результаты расчетов с комментариями

Задание 1)

Полученная выборка:

-3.77485	-0.77886	1.99200	-3.47613	0.79238	2.15419	0.98599	-0.85755	-0.62919	3.96032
-0.27336	0.15951	2.15332	1.15289	4.45739	0.64989	-2.60073	0.71441	1.34994	-1.82686
-1.26397	-1.37423	-1.36990	1.15059	-0.02140	-3.26987	-1.22980	1.18522	-1.98212	0.12386
-3.41938	1.16375	-0.48423	3.18687	1.65830	-0.91055	3.17200	0.57451	4.33148	-1.55506
-1.10570	0.01584	3.62201	0.43492	-0.27804	2.60946	-1.38345	-1.86311	-2.83698	2.54883
-2.96379	1.60007	0.76336	1.56464	1.51394	-0.99685	0.65567	-0.42172	0.10594	0.70173
1.44911	2.98624	-3.81508	-3.47485	-3.23239	1.52948	-2.71202	3.72059	0.32328	-2.54836
0.80319	-0.37975	1.97732	-0.45867	-1.26359	-1.56686	-1.35846	1.29475	-2.81496	1.69229
0.03771	0.04595	4.46792	1.18613	-1.19233	-0.75523	1.28561	0.91927	2.45968	-0.59971
-0.23860	-1.99965	-0.15561	-0.38043	0.69596	-2.04313	-1.03511	-2.11438	-4.26711	2.96899
1.19954	2.51308	1.85695	-2.36627	1.54859	-0.20508	-1.88209	1.26181	-1.98450	-2.01668
-1.05951	1.61224	-1.98562	3.82919	4.05208	1.41556	3.64918	-2.44670	0.36317	-2.04978
-3.26729	0.00486	2.30364	0.51686	0.98750	1.21405	0.53487	-0.20960	2.62371	1.24873
-3.79429	-1.65437	4.76518	-1.78814	-0.35574	0.10661	-0.75592	2.39940	2.43348	-0.96688
0.50165	7.46306	0.22037	-0.07852	-0.60370	1.16222	-0.78884	1.23366	-0.71186	0.68438
-0.07069	3.29687	-0.38676	-0.70831	2.08303	1.58235	5.95893	-1.18743	-1.35612	-0.00978
1.42044	0.28755	0.04978	0.94703	-1.29245	-2.15035	-0.44091	-0.10394	2.67289	-0.33502
1.24936	0.40473	0.88622	4.37701	0.68814	-3.65955	-1.47423	0.49579	2.98502	1.49159
-1.52323	-1.70732	-2.79224	-3.00831	-1.10247	-2.13875	-4.86603	-0.78863	2.47714	1.10348
1.97699	-0.90311	-2.99731	-0.95257	-1.15897	-3.36916	3.83449	0.58166	-0.28003	-3.59553
-0.59612	-3.00332	-2.05623	-1.93565	-0.32867	0.44716	2.62060	-0.19244	2.48821	-0.90512
0.15995	-1.43265	-1.62721	-0.36575	0.11759	-2.57789	1.82642	0.65731	1.71577	4.18185
1.12922	-1.79057	-1.50080	2.78135	-4.32864	-5.69638	2.54350	4.12742	0.95021	0.58230
-0.60391	1.43782	2.29356	2.96383	2.02109	-4.98954	-1.78194	1.88825	3.36258	-1.40659
1.89974	-2.92459	1.00484	0.60604	0.56484	0.72057	-0.30440	6.03645	0.25745	0.21788
1.78879	0.39895	1.56788	-1.19022	0.38916	1.25926	-1.84040	2.44751	2.76881	2.79080
-3.46407	1.14493	-2.11212	2.53613	1.06432	3.06702	2.10938	-1.16123	1.68686	1.71780

Упорядоченная выборка:

-5.69638	-4.98954	-4.86603	-4.32864	-4.26711	-3.81508	-3.79429	-3.77485	-3.65955	-3.59553
----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

-3.47613	-3.47485	-3.46407	-3.41938	-3.36916	-3.26987	-3.26729	-3.23239	-3.00831	-3.00332
-2.99731	-2.96379	-2.92459	-2.83698	-2.81496	-2.79224	-2.71202	-2.60073	-2.57789	-2.54836
-2.44670	-2.36627	-2.15035	-2.13875	-2.11438	-2.11212	-2.05623	-2.04978	-2.04313	-2.01668
-1.99965	-1.98562	-1.98450	-1.98212	-1.93565	-1.88209	-1.86311	-1.84040	-1.82686	-1.79057
-1.78814	-1.78194	-1.70732	-1.65437	-1.62721	-1.56686	-1.55506	-1.52323	-1.50080	-1.47423
-1.43265	-1.40659	-1.38345	-1.37423	-1.36990	-1.35846	-1.35612	-1.29245	-1.26397	-1.26359
-1.22980	-1.19233	-1.19022	-1.18743	-1.16123	-1.15897	-1.10570	-1.10247	-1.05951	-1.03511
-0.99685	-0.96688	-0.95257	-0.91055	-0.90512	-0.90311	-0.85755	-0.78884	-0.78863	-0.77886
-0.75592	-0.75523	-0.71186	-0.70831	-0.62919	-0.60391	-0.60370	-0.59971	-0.59612	-0.48423
-0.45867	-0.44091	-0.42172	-0.38676	-0.38043	-0.37975	-0.36575	-0.35574	-0.33502	-0.32867
-0.30440	-0.28003	-0.27804	-0.27336	-0.23860	-0.20960	-0.20508	-0.19244	-0.15561	-0.10394
-0.07852	-0.07069	-0.02140	-0.00978	0.00486	0.01584	0.03771	0.04595	0.04978	0.10594
0.10661	0.11759	0.12386	0.15951	0.15995	0.21788	0.22037	0.25745	0.28755	0.32328
0.36317	0.38916	0.39895	0.40473	0.43492	0.44716	0.49579	0.50165	0.51686	0.53487
0.56484	0.57451	0.58166	0.58230	0.60604	0.64989	0.65567	0.65731	0.68438	0.68814
0.69596	0.70173	0.71441	0.72057	0.76336	0.79238	0.80319	0.88622	0.91927	0.94703
0.95021	0.98599	0.98750	1.00484	1.06432	1.10348	1.12922	1.14493	1.15059	1.15289
1.16222	1.16375	1.18522	1.18613	1.19954	1.21405	1.23366	1.24873	1.24936	1.25926
1.26181	1.28561	1.29475	1.34994	1.41556	1.42044	1.43782	1.44911	1.49159	1.51394
1.52948	1.54859	1.56464	1.56788	1.58235	1.60007	1.61224	1.65830	1.68686	1.69229
1.71577	1.71780	1.78879	1.82642	1.85695	1.88825	1.89974	1.97699	1.97732	1.99200
2.02109	2.08303	2.10938	2.15332	2.15419	2.29356	2.30364	2.39940	2.43348	2.44751
2.45968	2.47714	2.48821	2.51308	2.53613	2.54350	2.54883	2.60946	2.62060	2.62371
2.67289	2.76881	2.78135	2.79080	2.96383	2.96899	2.98502	2.98624	3.06702	3.17200
3.18687	3.29687	3.36258	3.62201	3.64918	3.72059	3.82919	3.83449	3.96032	4.05208
4.12742	4.18185	4.33148	4.37701	4.45739	4.46792	4.76518	5.95893	6.03645	7.46306

Группированная выборка (интервальный вариационный ряд):

$(a_{i-1}, a_i]$	[-5.69638; -4.23422]	(-4.23422; -2.77206]	(-2.77206; -1.30990]	(-1.30990; 0.15226]	(0.15226; 1.61442]	(1.61442; 3.07658]	(3.07658; 4.53874]	(4.53874; 6.00090]	(6.00090; 7.46306]
n_i	5	21	41	66	74	42	17	2	2
w_i	0.01852	0.07778	0.15185	0.24444	0.27407	0.15556	0.06296	0.00741	0.00741

$$\sum_{i=1}^N w_i = 1.00000$$

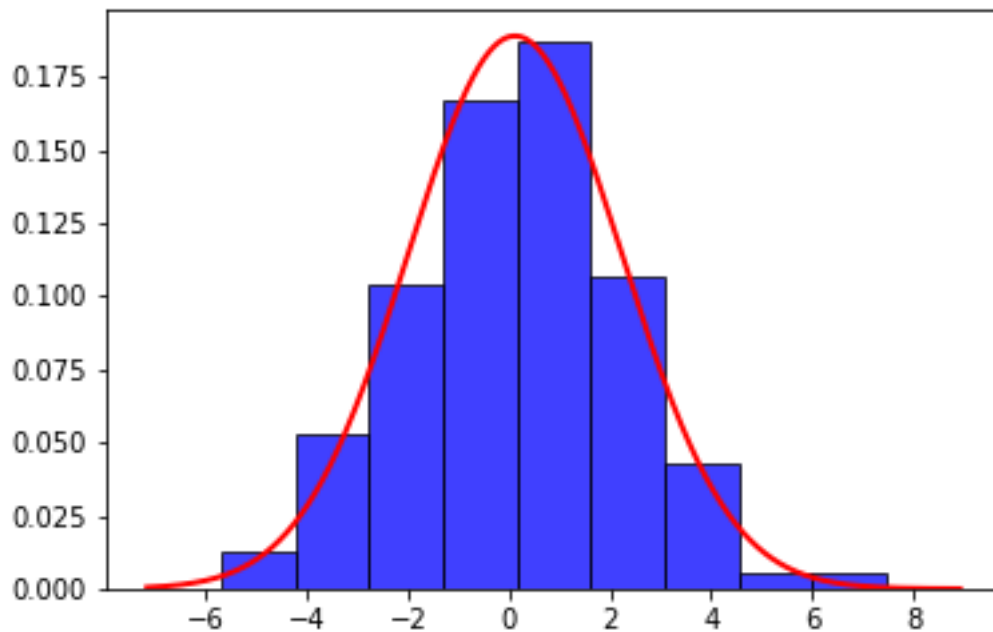
Математическое ожидание: $\tilde{a} = 0.11977$

Дисперсия: $\sigma^2 = 4.45689$

Среднеквадратическое отклонение: $\sigma = 2.11113$

k	a_k	$\frac{(a_k - \tilde{a})}{\sigma}$	$\frac{1}{\sigma} \varphi\left(\frac{(a_k - \tilde{a})}{\sigma}\right)$	$\Phi \frac{(a_k - \tilde{a})}{\sigma}$	p_k
0	-5.69638	-2.75499	0.00425	0.00293	-
1	-4.23422	-2.06239	0.02253	0.01959	0.01959
2	-2.77206	-1.36980	0.07395	0.08537	0.06579
3	-1.30990	-0.67720	0.15025	0.24914	0.16376
4	0.15226	0.01539	0.18895	0.50614	0.25700
5	1.61442	0.70799	0.14708	0.76052	0.25438
6	3.07658	1.40058	0.07087	0.91933	0.15881
7	4.53874	2.09317	0.02113	0.98183	0.06250
8	6.00090	2.78577	0.00390	0.99733	0.01550
9	7.46306	3.47836	0.00045	0.99975	0.00267

График плотности нормального распределения, наложенный на гистограмму относительных частот:



k	Интервал	w_k	p_k	$ w_k - p_k $	$\frac{N(w_k - p_k)^2}{p_k}$
1	$[-5.69638; -4.23422]$	0.01852	0.01959	0.00107	0.01568
2	$(-4.23422; -2.77206]$	0.07778	0.06579	0.01199	0.58979
3	$(-2.77206; -1.30990]$	0.15185	0.16376	0.01191	0.23393
4	$(-1.30990; 0.15226]$	0.24444	0.25700	0.01256	0.16565
5	$(0.15226; 1.61442]$	0.27407	0.25438	0.01969	0.41154
6	$(1.61442; 3.07658]$	0.15556	0.15881	0.00325	0.01798
7	$(3.07658; 4.53874]$	0.06296	0.06250	0.00046	0.00091

	4.53874]				
8	(4.53874; 6.00090]	0.00741	0.01550	0.00809	1.14011
9	(6.00090; 7.46306]	0.00741	0.00267	0.00474	2.26944
				$\max_k w_k - p_k = 0.01969$	$\sum_k \frac{N(w_k - p_k)^2}{p_k} = 4.84505$

$$\chi_B^2 = 4.84505$$

Задание 2)

a = 0.1

b = 6.1

Полученная выборка:

1.79902	2.03362	0.58672	4.75660	2.36932	5.44162	2.34982	2.05732	0.34228	2.53504
2.28112	2.03332	1.21168	5.27740	4.76254	3.80392	4.04998	3.76516	2.24800	2.16994
4.14712	2.06386	1.71880	5.95312	1.43110	1.83058	0.19774	4.96018	0.36544	2.80450
2.33662	2.50828	1.66630	5.38678	1.09030	1.86382	6.04234	5.41240	2.15200	3.09988
5.30374	4.79674	5.30938	2.80306	1.43584	5.19274	1.15480	4.89412	4.25122	5.47480
4.70332	1.11598	5.75686	0.97420	2.19292	3.46852	3.29038	0.65212	4.47232	0.18676
1.08514	0.68488	4.09126	0.68008	2.77966	0.40990	0.34426	1.57318	5.61748	1.81354
4.41940	4.12180	5.54404	0.88924	5.58634	2.77408	1.42672	4.64356	0.64354	1.91320
2.21542	2.63830	3.46264	2.51956	4.09936	1.99792	3.29908	1.86334	4.56148	4.37320
0.62272	4.25872	1.07404	0.11560	0.81292	0.95386	0.62530	0.15256	0.42262	1.10068
2.32864	2.82106	1.17418	3.95554	6.09910	3.98602	3.18064	1.02418	3.55810	0.49288
0.36400	4.84030	6.03586	1.49134	3.22066	5.96500	1.17592	4.29484	1.97266	3.71428
3.71926	2.51284	2.84590	1.71820	4.50520	0.90664	5.49016	5.82808	3.35500	5.62630
4.47220	4.41628	1.07926	3.58414	2.44366	3.59818	1.44844	2.94280	5.00674	1.94074
5.12962	3.26620	3.23302	4.36684	4.34326	1.29016	5.15362	5.43586	1.94104	2.42662
5.75650	4.29238	0.38548	1.50214	4.49188	3.47998	5.35588	2.88550	1.10440	2.10934
1.55656	1.19278	4.65334	1.37038	3.87436	1.20628	5.72878	2.75620	1.72354	2.87158
3.00286	3.35656	2.24884	1.48690	0.75106	1.35670	3.37606	1.32880	3.46642	4.16248
5.83198	4.97080	2.14660	6.01684	1.09186	5.60368	3.67120	2.50120	2.66248	3.46372
1.28386	0.33046	3.45934	3.00772	1.28230	1.08472	3.62392	4.54540	0.91384	0.45616
1.64404	5.84644	3.68578	0.72652	3.43978	2.44432	5.86966	0.73414	5.58532	1.47226
4.56148	5.77528	2.33824	4.68802	4.48900	3.98458	1.22998	0.20626	5.56666	0.85384
0.85060	0.99196	5.45374	2.70118	3.21154	2.25028	1.96192	2.71528	1.61818	1.70806
4.18402	4.04176	0.33556	5.46544	0.69028	0.54220	0.84628	2.61124	4.89088	5.33284

3.82330	4.70002	2.63884	2.85994	2.21944	0.13642	1.16926	1.26466	4.68154	1.15846
0.15004	4.33942	1.68562	1.37824	1.25956	0.52528	3.41962	1.67278	4.35226	2.29726
4.55758	3.34534	1.56112	3.42082	3.92314	5.35522	3.40930	1.90558	5.77156	4.67368

Упорядоченная выборка:

0.11560	0.13642	0.15004	0.15256	0.18676	0.19774	0.20626	0.33046	0.33556	0.34228
0.34426	0.36400	0.36544	0.38548	0.40990	0.42262	0.45616	0.49288	0.52528	0.54220
0.58672	0.62272	0.62530	0.64354	0.65212	0.68008	0.68488	0.69028	0.72652	0.73414
0.75106	0.81292	0.84628	0.85060	0.85384	0.88924	0.90664	0.91384	0.95386	0.97420
0.99196	1.02418	1.07404	1.07926	1.08472	1.08514	1.09030	1.09186	1.10068	1.10440
1.11598	1.15480	1.15846	1.16926	1.17418	1.17592	1.19278	1.20628	1.21168	1.22998
1.25956	1.26466	1.28230	1.28386	1.29016	1.32880	1.35670	1.37038	1.37824	1.42672
1.43110	1.43584	1.44844	1.47226	1.48690	1.49134	1.50214	1.55656	1.56112	1.57318
1.61818	1.64404	1.66630	1.67278	1.68562	1.70806	1.71820	1.71880	1.72354	1.79902
1.81354	1.83058	1.86334	1.86382	1.90558	1.91320	1.94074	1.94104	1.96192	1.97266
1.99792	2.03332	2.03362	2.05732	2.06386	2.10934	2.14660	2.15200	2.16994	2.19292
2.21542	2.21944	2.24800	2.24884	2.25028	2.28112	2.29726	2.32864	2.33662	2.33824
2.34982	2.36932	2.42662	2.44366	2.44432	2.50120	2.50828	2.51284	2.51956	2.53504
2.61124	2.63830	2.63884	2.66248	2.70118	2.71528	2.75620	2.77408	2.77966	2.80306
2.80450	2.82106	2.84590	2.85994	2.87158	2.88550	2.94280	3.00286	3.00772	3.09988
3.18064	3.21154	3.22066	3.23302	3.26620	3.29038	3.29908	3.34534	3.35500	3.35656
3.37606	3.40930	3.41962	3.42082	3.43978	3.45934	3.46264	3.46372	3.46642	3.46852
3.47998	3.55810	3.58414	3.59818	3.62392	3.67120	3.68578	3.71428	3.71926	3.76516
3.80392	3.82330	3.87436	3.92314	3.95554	3.98458	3.98602	4.04176	4.04998	4.09126
4.09936	4.12180	4.14712	4.16248	4.18402	4.25122	4.25872	4.29238	4.29484	4.33942
4.34326	4.35226	4.36684	4.37320	4.41628	4.41940	4.47220	4.47232	4.48900	4.49188
4.50520	4.54540	4.55758	4.56148	4.56148	4.64356	4.65334	4.67368	4.68154	4.68802
4.70002	4.70332	4.75660	4.76254	4.79674	4.84030	4.89088	4.89412	4.96018	4.97080
5.00674	5.12962	5.15362	5.19274	5.27740	5.30374	5.30938	5.33284	5.35522	5.35588
5.38678	5.41240	5.43586	5.44162	5.45374	5.46544	5.47480	5.49016	5.54404	5.56666
5.58532	5.58634	5.60368	5.61748	5.62630	5.72878	5.75650	5.75686	5.77156	5.77528
5.82808	5.83198	5.84644	5.86966	5.95312	5.96500	6.01684	6.03586	6.04234	6.09910

Группированная выборка (интервальный вариационный ряд):

(ai-1, ai]	[0.10000; 0.76667]	(0.76667; 1.43333]	(1.43333; 2.10000]	(2.10000; 2.76667]	(2.76667; 3.43333]	(3.43333; 4.10000]	(4.10000; 4.76667]	(4.76667; 5.43333]	(5.43333; 6.10000]
ni	31	40	34	32	27	27	33	18	28
wi	0.11481	0.14815	0.12593	0.11852	0.10000	0.10000	0.12222	0.06667	0.10370

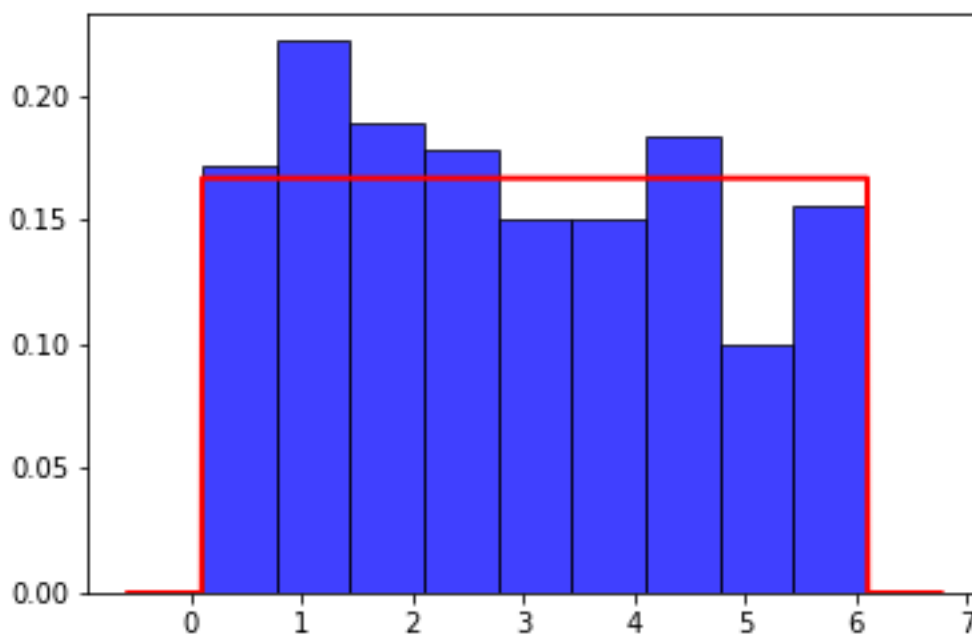
$$\sum_{i=1}^N w_i = 1.00000$$

Математическое ожидание: $\tilde{\alpha} = 2.89012$

Дисперсия: $\sigma^2 = 2.87036$

Среднеквадратическое отклонение: $\sigma = 1.69421$

График плотности равномерного распределения, наложенный на гистограмму относительных частот $\max_k |w_k - p_k| =$:



k	Интервал	w_k	p_k	$ w_k - p_k $	$\frac{N(w_k - p_k)^2}{p_k}$
1	[0.10000; 0.76667]	0.11481	0.11111	0.00370	0.03333
2	(0.76667; 1.43333]	0.14815	0.11111	0.03704	3.33333
3	(1.43333; 2.10000]	0.12593	0.11111	0.01481	0.53333
4	(2.10000; 2.76667]	0.11852	0.11111	0.00741	0.13333
5	(2.76667; 3.43333]	0.10000	0.11111	0.01111	0.30000
6	(3.43333; 4.10000]	0.10000	0.11111	0.01111	0.30000
7	(4.10000; 4.76667]	0.12222	0.11111	0.01111	0.30000
8	(4.76667; 5.43333]	0.06667	0.11111	0.04444	4.80000
9	(5.43333; 6.10000]	0.10370	0.11111	0.00741	0.13333
				$\max_k w_k - p_k = 0.04444$	$\sum_k \frac{N(w_k - p_k)^2}{p_k} = 9.86667$

$$\chi_B^2 = 9.86667$$

Анализ результатов и выводы

Таблица критических значений:

l	4	5	6	7	8
$\chi^2_{кр,\alpha}(l)$	9.5	11.1	12.6	14.1	15.5

1) Нормальное распределение

$$\chi_B^2 = 4.84505$$

$$\chi_{кр,\alpha}^2(6) = 12.6$$

Гипотеза о соответствии выборки нормальному распределению не противоречит экспериментальным данным (т.е. может быть принята) при уровне значимости $\alpha = 0,05$.

2) Равномерное распределение

$$\chi_B^2 = 9.86667$$

$$\chi_{кр,\alpha}^2(6) = 12.6$$

Гипотеза о соответствии выборки нормальному распределению не противоречит экспериментальным данным (т.е. может быть принята) при уровне значимости $\alpha = 0,05$.

Список использованной литературы

1. Лобузов А.А. Математическая статистика [Электронный ресурс]: Методические указания по выполнению лабораторных работ / под ред. Ю. И. Худака. Москва: Московский технологический университет (МИРЭА), 2017. 36 с.
2. Чернова Н. И. Математическая статистика: Учеб. пособие / Новосиб. гос. ун-т. Новосибирск, 2007. 148 с

Приложение (Листинг программы)

```
# coding: utf-8

# In[1]:

import math
import numpy

import matplotlib.mlab as mlab
import matplotlib.pyplot as plt

import scipy
import scipy.stats as pystats
from scipy.stats import norm as pynorm
from scipy.stats import uniform as pyuni

from docx import Document

# In[2]:

def strm(a):
    n = len(a)
    b = a[0]
    m = len(b)
    for i in range(n):
        for k in range(m):
            if type(a[i][k]) == int:
                a[i][k] = str(a[i][k])
            elif type(a[i][k]) != str:
                a[i][k] = str('%.5f'% a[i][k])

# In[3]:

def onetablein(docname, n):
    document = Document(docname)
    table = document.tables[n]
    array = []
    for i in range(0, len(table.rows)):
        row = table.rows[i]
        for cell in row.cells:
            array.append(float(cell.text.replace(',', '.')))
    return array

# In[4]:
```

```
krit_tbl = {
    '4' : 9.5,
    '5' : 11.1,
    '6' : 12.6,
    '7' : 14.1,
    '8' : 15.5}
```

```
# In[5]:
```

```
ln = onetablein('Data/tables_9.docx', 0)
```

```
lu = onetablein('Data/tables_9.docx', 1)
```

```
alfa = 0.05
```

```
a = 0.1
```

```
b = 6.1
```

```
# In[6]:
```

```
class stat(object):
```

```
    def __init__(self, array, m, ao = None, am = None):
        self.m = m
        if ao == None:
            self.ao = min(array)
        else:
            self.ao = ao
        if am == None:
            self.am = max(array)
        else:
            self.am = am
        self.h = (self.am - self.ao)/self.m
        self.num = [self.ao + (i + 0.5)*self.h for i in range(m)]
        self.interval = [self.ao + i*self.h for i in range(m+1)]
        self.cnt = [0 for i in range(m)]
        for i in array:
            for k in range(m):
                if i <= self.interval[k+1]:
                    break
            self.cnt[k] = self.cnt[k] + 1
        self.war = [i/len(array) for i in self.cnt]
```

```
# In[7]:
```

```

class NormSample(object):
    def __init__(self, array):
        self.array = [i for i in array]
        self.n = len(array)
        self.m = 1 + int(math.log2(self.n))
        self.stats = stat(array, self.m)

    def mean(self):
        s = 0
        for i in range(self.m):
            s = s + self.stats.war[i]*self.stats.num[i]
        return s

    def disp(self):
        s = 0
        for i in range(self.m):
            s = s + self.stats.war[i]*(self.stats.num[i]**2)
        return s - (self.stats.h**2)/12 - self.mean()**2

    def devi(self):
        return self.disp()**0.5

    def graf(self, hist = True, theory = True, filename =
'Data/graf_norm.png'):
        dx = 0.005
        xmin = self.stats.ao - self.stats.h
        xmax = self.stats.am + self.stats.h
        if hist:
            plt.bar(
                [i for i in self.stats.num],
                [i/self.stats.h for i in self.stats.war],
                color = 'blue', edgecolor = 'black',
                linewidth = 1, alpha = 0.75, width = self.stats.h
            )
        if theory:
            xlist = numpy.arange(xmin, xmax, dx) # mlab.frange(xmin, xmax,
dx)
            ylist = [pystats.norm.pdf(xz, loc = self.mean(), scale =
self.devi()) for xz in xlist]
            plt.plot(xlist, ylist, color = 'red', linewidth = 2)
            plt.savefig(filename)
            plt.clf()

    def out_1(self):
        l = []
        for k in range(len(self.stats.interval)):
            q = []
            ak = self.stats.interval[k]
            q.append(k)
            q.append(ak)
            q.append((ak - self.mean())/self.devi())

```

```

q.append(pynorm.pdf(q[2])/self.devi())
q.append(pynorm.cdf(q[2]))
if k == 0:
    q.append('-')
elif k == 1:
    q.append(q[4])
elif k == self.m:
    q.append(1 - l[k-1][4])
else:
    q.append(q[4] - l[k-1][4])
l.append(q)
return l

def out_2(self):
    out_1 = self.out_1()
    l = []
    f4 = 0
    f5 = 0
    for k in range(self.m):
        q = []
        q.append(k+1)
        s = ''
        if k == 0:
            s = s + '['
        else:
            s = s + '('
        s = s + str('%0.5f'% out_1[k][1]) + ';' + str('%0.5f'%
out_1[k+1][1]) + ']'
        q.append(s)
        q.append(self.stats.war[k])
        q.append(out_1[k+1][5])
        q.append(abs(q[2]-q[3]))
        q.append(self.n*(q[4]**2)/q[3])
        if q[4] > f4:
            f4 = q[4]
        f5 = f5 + q[5]
        l.append(q)
    l.append(['', '', '', '', f4, f5])
    return l

def hi2(self):
    s = 0
    n = [k for k in self.stats.cnt]
    p = [k[3] for k in self.out_2()]
    for k in range(self.m):
        s = s + ((n[k]-self.n*p[k])**2)/(self.n*p[k])
    return s

def krit(self, tbl):
    l = self.m - 3

```

```

    if self.hi2() > tbl[str(l)]:
        return False
    else:
        return True

```

```
# In[8]:
```

```

sn = NormSample(ln)
print(sn.mean())
print(sn.disp())
print(sn.devi(), '\n')
for i in sn.out_1():
    print(i)
print()
for i in sn.out_2():
    print(i)
print()
print(sn.hi2())
print(sn.krit(krit_tbl))
sn.graf()

```

```
# In[9]:
```

```

class UniSample(object):
    def __init__(self, array, a, b):
        self.array = [i for i in array]
        self.n = len(array)
        self.a = a
        self.b = b
        self.m = 1 + int(math.log2(self.n))
        self.stats = stat(array, self.m, ao = a, am = b)

    def mean(self):
        s = 0
        for i in range(self.m):
            s = s + self.stats.war[i]*self.stats.num[i]
        return s

    def disp(self):
        s = 0
        for i in range(self.m):
            s = s + self.stats.war[i]*(self.stats.num[i]**2)
        return s - (self.stats.h**2)/12 - self.mean()**2

    def devi(self):
        return self.disp()**0.5

```

```

def graf(self, hist = True, theory = True, filename =
'Data/graf_uni.png'):
    dx = 0.005
    xmin = self.stats.ao - self.stats.h
    xmax = self.stats.am + self.stats.h

    if hist:
        plt.bar(
            [i for i in self.stats.num],
            [i/self.stats.h for i in self.stats.war],
            color = 'blue', edgecolor = 'black',
            linewidth = 1, alpha = 0.75, width = self.stats.h
        )
    if theory:
        xlist = numpy.arange(xmin, xmax, dx) # mlab.frange(xmin, xmax,
dx)
        ylist = [pystats.uniform.pdf(xz, loc = self.a, scale = self.b -
self.a) for xz in xlist]
        plt.plot(xlist, ylist, color = 'red', linewidth = 2)
    plt.savefig(filename)
    plt.clf()

def out_3(self):
    l = []
    f4 = 0
    f5 = 0
    for k in range(self.m):
        q = []
        q.append(k+1)
        s = ''
        if k == 0:
            s = s + '['
        else:
            s = s + '('
        s = s + str('%5f'% self.stats.interval[k]) + ';' + str('%5f'%
self.stats.interval[k+1]) + ']'
        q.append(s)
        q.append(self.stats.war[k])
        q.append(1/self.m)
        q.append(abs(q[2]-q[3]))
        q.append(self.n*(q[4]**2)/q[3])
        l.append(q)
        if q[4] > f4:
            f4 = q[4]
        f5 = f5 + q[5]
    l.append(['', '', '', '', f4, f5])
    return l

def hi2(self):
    s = 0

```

```

        n = [k for k in self.stats.cnt]
        p = [k[3] for k in self.out_3()]
        for k in range(self.m):
            s = s + ((n[k]-self.n*p[k])**2)/(self.n*p[k])
        return s

    def krit(self, tbl):
        l = self.m - 3
        if self.hi2() > tbl[str(l)]:
            return False
        else:
            return True

# In[10]:

su = UniSample(lu,a,b)
print(su.mean())
print(su.disp())
print(su.devi())
print()
for k in su.out_3():
    print(k)
print()
print(su.hi2())
print(su.krit(krit_tbl))
su.graf()

# In[11]:

head_1_str = [
    'k',
    'ak',
    '(ak-a)/s',
    '1/s*f((ak-a)/s)',
    'F((ak-a)/s)',
    'pk'
]
head_2_str = [
    'k',
    'Интервал',
    'wk',
    'pk',
    '|wk-pk|',
    'N(wk-pk)^2/pk'
]

```

```
# In[12]:
```

```
def tabler(document, out, head = None):
    tl = []
    if head != None:
        tl.append(head)
    for i in out:
        tl.append(i)
    strm(tl)

    table = document.add_table(rows = len(tl), cols = len(tl[0]))
    for i in range(len(tl)):
        hdr_cells = table.rows[i].cells
        for k in range(len(tl[0])):
            if type(tl[i][k]) == str:
                hdr_cells[k].text = tl[i][k]
            else:
                pass #LaTeh
```

```
# In[13]:
```

```
def doc_task_1(document, sn):
    document.add_paragraph('Задание 1')

    document.add_paragraph('Полученная выборка:')
    l = [i for i in sn.array]
    tbl = []
    q = []
    for i in range(len(l)):
        q.append(l[i])
        if (len(q) == 10) or (i+1 == len(l)):
            while(len(q) != 10):
                q.append('')
            tbl.append(q)
            q = []
    tabler(document, tbl)

    document.add_paragraph('')
    document.add_paragraph('Упорядоченная выборка:')
    l.sort()
    tbl = []
    q = []
    for i in range(len(l)):
        q.append(l[i])
        if (len(q) == 10) or (i+1 == len(l)):
            while(len(q) != 10):
                q.append('')
```



```

tbl.append(q)
q = []
tabler(document, tbl)

document.add_paragraph('')
document.add_paragraph('Группированная выборка (интервальный вариационный
ряд):')
tbl = [['(ai-1, ai]'], ['ni'], ['wi']]
tbl[0].extend([i[1] for i in sn.out_2() if i[1] != ''])
tbl[1].extend(sn.stats.cnt)
tbl[2].extend(sn.stats.war)
tabler(document, tbl)
document.add_paragraph('<math>\sum_{i=0}^{\{N\}} w_i</math> = ' +
str('%5f' % sum(sn.stats.war)))

document.add_paragraph('Математическое ожидание: \tilde{a} = ' +
str('%5f' % sn.mean()))
document.add_paragraph('Дисперсия: \tilde{\sigma}^2 = ' + str('%5f' %
sn.disp()))
document.add_paragraph('Среднеквадратическое отклонение: \tilde{\sigma} =
' + str('%5f' % sn.devi()))
document.add_paragraph('')

tabler(document, sn.out_1(), head_1_str)
document.add_paragraph('График плотности нормального распределения,
наложенный на гистограмму относительных частот:')
sn.graf(filename = 'tskl_graf.png')
document.add_picture('tskl_graf.png')

tabler(document, sn.out_2(), head_2_str)
document.add_paragraph('')
document.add_paragraph('\chi_B^2 = ' + str('%5f' % sn.hi2()))
document.add_paragraph('')

# In[14]:

def doc_task_2(document, sn):
    document.add_paragraph('Задание 2')
    document.add_paragraph('')
    document.add_paragraph('a = ' + str(sn.a))
    document.add_paragraph('b = ' + str(sn.b))
    document.add_paragraph('')

    document.add_paragraph('Полученная выборка:')
    l = [i for i in sn.array]
    tbl = []
    q = []

```

```

for i in range(len(l)):
    q.append(l[i])
    if (len(q) == 10) or (i+1 == len(l)):
        while(len(q) != 10):
            q.append('')
        tbl.append(q)
        q = []
tabler(document, tbl)

document.add_paragraph('')
document.add_paragraph('Упорядоченная выборка:')
l.sort()
tbl = []
q = []
for i in range(len(l)):
    q.append(l[i])
    if (len(q) == 10) or (i+1 == len(l)):
        while(len(q) != 10):
            q.append('')
        tbl.append(q)
        q = []
tabler(document, tbl)

document.add_paragraph('')
document.add_paragraph('Группированная выборка (интервальный вариационный
ряд):')
tbl = [['(ai-1,ai]'], ['ni'], ['wi']]
tbl[0].extend([i[1] for i in sn.out_3() if i[1] != ''])
tbl[1].extend(sn.stats.cnt)
tbl[2].extend(sn.stats.war)
tabler(document, tbl)
document.add_paragraph('<math>\sum_{i=0}^{\{N\}} w_i</math> = ' +
str('%.5f'% sum(sn.stats.war)))

document.add_paragraph('Математическое ожидание: \tilde{a} = ' +
str('%.5f'% sn.mean()))
document.add_paragraph('Дисперсия: \tilde{\sigma}^2 = ' + str('%.5f'%
sn.disp()))
document.add_paragraph('Среднеквадратическое отклонение: \tilde{\sigma} =
' + str('%.5f'% sn.devi()))
document.add_paragraph('')

document.add_paragraph('График плотности равномерного распределения,
наложенный на гистограмму относительных частот:')
sn.graf(filename = 'tsk2_graf.png')
document.add_picture('tsk2_graf.png')

tabler(document, sn.out_3(), head_2_str)
document.add_paragraph('')
document.add_paragraph('\chi_B^2 = ' + str('%.5f'% sn.hi2()))

```

```

document.add_paragraph('')

# In[15]:

def doc_anal(document, sn, su, krit_table):
    NormResult = {
        True : 'Гипотеза о соответствии выборки нормальному распределению не
противоречит экспериментальным данным (т.е. может быть принята) при уровне
значимости alpha = 0,05. ',
        False : 'Гипотеза о соответствии выборки нормальному распределению
противоречит экспериментальным данным (т.е. не может быть принята) при уровне
значимости alpha = 0,05. '
    }
    UniResult = {
        True: 'Гипотеза о соответствии выборки равномерному распределению на
отрезке [a, b] не противоречит экспериментальным данным (т.е. может быть
принята) при уровне значимости alpha = 0,05. ',
        False: 'Гипотеза о соответствии выборки равномерному распределению на
отрезке [a, b] противоречит экспериментальным данным (т.е. не может быть
принята) при уровне значимости alpha = 0,05. '
    }

    document.add_paragraph('Анализ результатов и выводы')
    document.add_paragraph('')
    document.add_paragraph('Таблица критических значений:')#
\chi_{kp,\alpha}^2 (1)
    tbl = [['1'], ['chi']]#'\chi_{kp,\alpha}^2 (1) '
    tbl[0].extend([i for i in krit_table])
    tbl[1].extend([str(krit_table[i]) for i in krit_table])
    tabler(document, tbl)
    document.add_paragraph('')
    document.add_paragraph('1) Нормальное распределение')
    document.add_paragraph('\chi_B^2 = ' + str('%0.5f' % sn.hi2()))
    document.add_paragraph('chi(' + str(sn.m-3) + ') = ' +
str(krit_table[str(sn.m-3)]) )#'\chi_{kp,\alpha}^2 (1) '
    document.add_paragraph(NormResult[sn.krit(krit_table)])

    document.add_paragraph('')
    document.add_paragraph('2) Равномерное распределение')
    document.add_paragraph('\chi_B^2 = ' + str('%0.5f' % su.hi2()))
    document.add_paragraph('chi(' + str(su.m-3) + ') = ' +
str(krit_table[str(su.m-3)]) )#'\chi_{kp,\alpha}^2 (1) '
    document.add_paragraph(NormResult[su.krit(krit_table)])

# In[16]:

def doccreator(sn, su, krit_tbl, filename = 'final_doc.docx'):

```

```
document = Document()
doc_task_1(document, sn)
document.add_paragraph('')
doc_task_2(document, su)
document.add_paragraph('')
doc_anal(document, sn, su, krit_tbl)
document.save(filename)
print('Done!')
```

In[17]:

```
doccreator(sn, su, krit_tbl, filename = 'Data/final_doc.docx')
```