

Задание.

- Для равномерного и нормального распределений выбрать параметры и построить графики плотности.
- Произвести выборку $n = 10^6$ из соответствующего распределения.
- Для $m = 100$ построить гистограмму.
- Объединить графики плотности и гистограмму на одном графике.
- Проверить гипотезу о виде распределения по критерию хи-квадрат.

Графики.

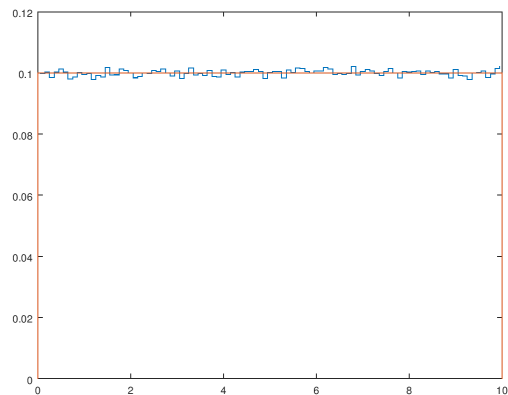


Рис. 1: Равномерное

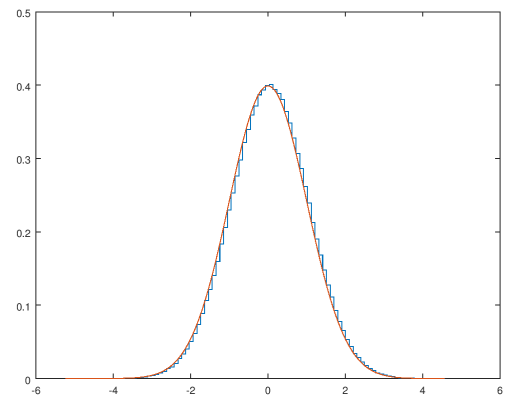


Рис. 2: Нормальное

- Нормальное, код:

```
1 pkg load statistics;
2
3 clc;
4 clear;
5
6 n = 1000000;
7 m = 100;
8 mu = 0;
9 sigma = 1;
10
11 x = normrnd(mu, sigma, n, 1);
12 [h_y, h_x] = hist(x, m);
13 a = min(x);
14 b = max(x);
15 h_y = h_y / n / ((b - a) / m);
16 [plot_x, plot_y] = stairs(h_x, h_y);
17 normp_x = a:0.05:b;
18 normp_y = normpdf(normp_x, mu, sigma);
19
20 plot(plot_x, plot_y, normp_x, normp_y);
```

- Равномерное, код:

```
1 m = 100;
2 left = 0;
3 right = 10;
4 p = 1 / (right - left);
5
6 x = unifrnd(left, right, n, 1);
```

```
7 [h_y, h_x] = hist(x, m);
8 a = min(x);
9 b = max(x);
10 h_y = h_y / n / ((b - a) / m);
11 [plot_x, plot_y] = stairs(h_x, h_y);
12 unifp_x = [a, a, b, b];
13 unifp_y = [0, p, p, 0];
14
15 plot(plot_x, plot_y, unifp_x, unifp_y);
```

- Нормальное

```
1 pkg load statistics;
2
3 clc;
4 clear;
5
6 n = 10000;
7 m = 22;
8 k = 1000;
9 mu = 0;
10 sigma = 1;
11
12
13 chi2 = zeros(1, k);
14
15 i = 1;
16 while (i <= k)
17     x = normrnd(mu, sigma, n, 1);
18     [h_y, h_x] = hist(x, m);
19     a = min(x);
20     b = max(x);
21     len = (b - a) / m;
22     j = 1;
23     while (j <= m)
24         p = normcdf(a + j * len, mu, sigma) - normcdf(a + (j - 1) * len, mu, sigma);
25         chi2(i) = chi2(i) + ((h_y(j) - n * p) ^ 2) / n / p;
26         j++;
27     endwhile
28     i++;
29 endwhile
30
31 err1 = mean(chi2 > chi2inv(0.95, m - 1));
32
33 function retval = do_tests(mu, sigma, n, m, k, shift)
34     chi2 = zeros(1, k);
35     segments = zeros(1, k);
36
37     i = 1;
38     while (i <= k)
39         x = normrnd(mu, sigma, n, 1);
40         [h_y, h_x] = hist(x, m);
41         a = min(x);
42         b = max(x);
43         len = (b - a) / m;
44         j = 0;
45         while (j < m)
46             cnt = 0;
47             p = -normcdf(a + j * len, mu + shift, sigma);
48             while (cnt < 6 && j < m)
49                 j++;
50                 cnt += h_y(j);
51             end
52             p += normcdf(a + j * len, mu + shift, sigma);
53             chi2(i) = chi2(i) + ((cnt - n * p) ^ 2) / n / p;
54             segments(i) = segments(i) + 1;
55         endwhile
```

```
56         i++;
57     endwhile
58
59     retval = mean(chi2 > chi2inv(0.95, segments - 1));
60 endfunction
61
62 err2 = do_tests(mu, sigma, n, m, k, 0);
63 err3 = 1 - do_tests(mu, sigma, n, m, k, 0.5);
64
65 printf("Type I error alpha: %f \n", err1);
66 printf("Type I error alpha (corrected): %f\n", err2);
67 printf("Type II error beta with delta=0.005: %f\n", 1 - do_tests(mu, sigma, n, m, k, 0.005));
68 printf("Type II error beta with delta=0.5: %f\n", err3);
```

Выход:

```
1 Type I error alpha: 0.113000
2 Type I error alpha (corrected): 0.072000
3 Type II error beta with delta=0.005: 0.920000
4 Type II error beta with delta=0.5: 0.000000
```

• Равномерное

```
1 pkg load statistics;
2
3 clc;
4 clear;
5
6 n = 10000;
7 m = 22;
8 k = 1000;
9 left = 0;
10 right = 10;
11
12 function retval = do_tests(left, right, n, m, k, shift)
13     p = 1 / (right - left + shift);
14     chi2 = zeros(1, k);
15
16     i = 1;
17     while (i <= k)
18         x = unifrnd(left, right, n, 1);
19         [h_y, h_x] = hist(x, m);
20         a = min(x);
21         b = max(x);
22         len = (b - a) / m;
23         j = 1;
24         while (j <= m)
25             pj = len * p;
26             chi2(i) = chi2(i) + ((h_y(j) - n * pj) ^ 2) / n / pj;
27             j++;
28         endwhile
29         i++;
30     endwhile
31
32     retval = mean(chi2 > chi2inv(0.95, m - 1));
33 endfunction
34
35
36
37 err1 = do_tests(left, right, n, m, k, 0);
38 err2 = 1 - do_tests(left, right, n, m, k, 0.01);
39 err3 = 1 - do_tests(left, right, n, m, k, 1); #a tyt mi shitaem bety. shakalim pri pomoshi
        shifta. i smotrim, skiolko on za istiny poschital. tam voobshe 0
40
41 printf("Type I error alpha = %f\n", err1);
42 printf("Type II error alpha with delta=0.005: %f\n", err2);
43 printf("Type II error alpha with delta=0.5: %f\n", err3);
```

Выход:

```
1  Type I error alpha = 0.052000
2  Type II error alpha with delta=0.005: 0.957000
3  Type II error alpha with delta=0.5: 0.000000
```

Вывод:

Построили гистограммы, получили эмпирическую оценку плотности. По графикам видно, что гистограмма достаточно хорошо приближает плотность.

Ошибки первого рода сходится с теоритическим значением по критерию хи-квадрат. Ошибка второго рода реагирует только на сильные изменения, поэтому можно утверждать, что она стремится к нулю при больших n .