

Лабораторная работа 4

Борис Шапошников

Виктор Шарепов

Рамазан Рахматуллин

Обратная связь: bshaposhnikov01@gmail.com

Часть 1

- 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.

Код и графики:

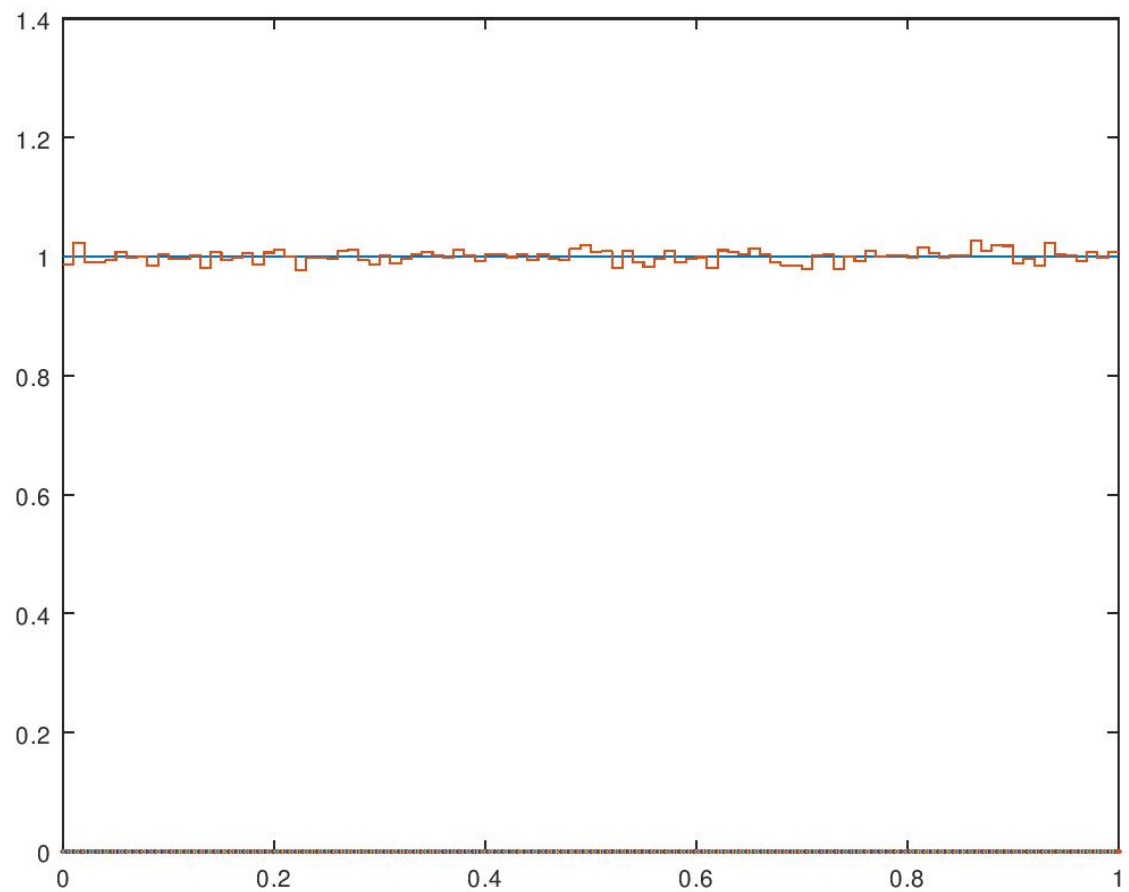
- Равномерное распределение

```
1 clc;  
2 clear;  
3  
4 pkg load statistics;  
5 n = 1e6;  
6 m = 1e2;  
7  
8 a = 0;  
9 b = 1;  
10 x = sort(unifrnd(a, b, n, 1));  
11 y = hist(x, m);  
12 y = y / n;  
13
```

```

14 step = (x(n) - x(1)) / m
15 f = y / step
16
17 [stx, sty] = stairs(x(1) : step : x(end), [f, f(end)]);
18 interval = a:0.001:b
19 plot(interval, unifpdf(interval, a, b), stx, sty, interval, 0)

```



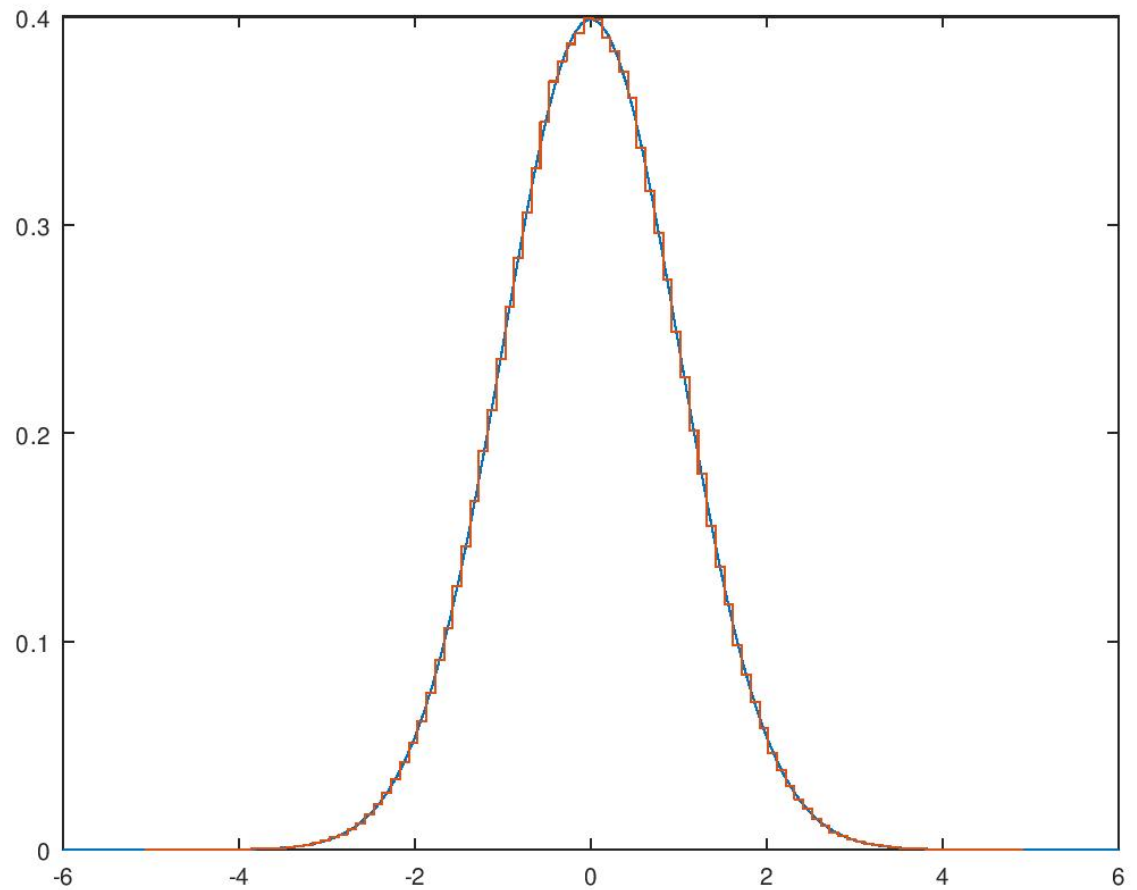
• Нормальное распределение

```

1 clc;
2 clear;
3
4 pkg load statistics;
5 n = 1e6;
6 m = 1e2;
7
8
9 x = sort(randn(1, n));
10 y = hist(x, m);

```

```
11 y = y / n;  
12  
13 step = (x(n) - x(1)) / m  
14 f = y / step  
15  
16 [stx, sty] = stairs(x(1) : step : x(end), [f, f(end)]);  
17 plot(-6:0.1:6, normpdf(-6:0.1:6), stx, sty)
```



Вывод из первой части

Построив гистограмму, мы получили эмпирическую оценку плотности распределения случайной величины. По графикам мы убедились, что гистограмма, построенная на основе выборки из распределения, достаточно хорошо приближает его плотность вероятности.

Часть 2

Проверка гипотез согласия с H_0 - тип распределения.

Цель работы - оценка ошибки первого рода.

1. Для $n = 10^6$ провести $m = 100$ проверок гипотез по критерию χ^2 .

Построенную статистику сравнить с порогом.

- Равномерное распределение

```

1  clc;
2  clear;
3
4  pkg load statistics;
5
6  n = 1e6;
7  m = 1e2;
8
9  a = 0;
10 b = 1;
11
12 errorsI = 0;
13 gamma = 0.95;
14
15 for i = 1:m
16     x = sort(unifrnd(a, b, n, 1));
17     mi = min(x);
18     ma = max(x);
19
20     y = hist(x, m);
21     step = (x(n) - x(1)) / m;
22
23     y1 = unifcdf(x(1) : step : x(end) - step, mi, ma);
24     y2 = unifcdf(x(1) + step : step : x(end), mi, ma);
25
26     p = y2 - y1;
27     hi2 = sum(((y - n * p) .^ 2) ./ (n * p));
28     threshold = chi2inv(gamma, m - 3);
29     if (hi2 >= threshold)
30         errorsI++;
31     endif
32 endfor
33

```

```

34 printf("Alpha: %d\n", 1 - gamma);
35 printf("Estimated alpha: %d\n", errorsI / m);

```

Результат

Alpha: 0.05

Estimated alpha: 0.05

● Нормальное распределение

```

1  clc;
2  clear;
3
4  pkg load statistics;
5
6  n = 1e6;
7  m = 1e2;
8
9  errorsI = 0;
10 gamma = 0.95;
11
12 for i = 1:m
13     x = sort(normrnd(1, 3, n, 1));
14
15     y = hist(x, m);
16     step = (x(n) - x(1)) / m;
17
18     E = mean(x);
19     sigma = std(x);
20
21     y1 = normcdf(x(1) : step : x(end) - step, E, sigma);
22     y2 = normcdf(x(1) + step : step : x(end), E, sigma);
23
24     p = y2 - y1;
25     hi2 = sum(((y - n * p) .^ 2) ./ (n * p));
26     threshold = chi2inv(gamma, m - 3);
27     if (hi2 >= threshold)
28         errorsI++;
29     endif
30 endfor
31
32 printf("Alpha: %d\n", 1 - gamma);
33 printf("Estimated alpha: %d\n", errorsI / m);

```

Результат

Alpha: 0.05

Estimated alpha: 0.19

Вывод

Для равномерного распределения оценка $\alpha = \frac{H_1 \text{count}}{m}$ очень близка к теоретической $\alpha = 1 - \gamma$. Для нормального распределения это не так. Для него не всегда выполнено требование $n_j \geq 6$ для близких к $a = \min(x)$ и $b = \max(x)$ интервалов. $\chi^2_{n,m-1} = \sum_{j=1}^m \frac{(n_j - n \cdot p_{j,0})^2}{n \cdot p_{j,0}}$

2. Для нормального распределения провести объединение соседних интервалов для выполнения $n_j \geq 6$.

```

1  clc;
2  clear;
3
4  pkg load statistics;
5
6  n = 1e6;
7  m = 1e2;
8
9  errorsI = 0;
10 gamma = 0.95;
11
12 for i = 1:m
13     x = sort(normrnd(1, 3, n, 1));
14
15     y = hist(x, m);
16     step = (x(n) - x(1)) / m;
17
18     E = mean(x);
19     sigma = std(x);
20
21     l = x(1)
22     r = x(1) + step
23
24     i = 1;
25     hi2 = 0;

```

```

26  seg_cnt = 0;
27  while (i <= m)
28      y_i = y(i);
29      while (y_i < 6 && i++ < m)
30          y_i += y(i);
31          r += step
32      endwhile
33      y1 = normcdf(l, E, sigma);
34      y2 = normcdf(r, E, sigma);
35      p = y2 - y1;
36      hi2 += (y_i - n * p) ^ 2 / (n * p);
37      l = r;
38      r += step;
39      i++;
40      seg_cnt++;
41  endwhile
42
43  threshold = chi2inv(gamma, seg_cnt - 3);
44  if (hi2 >= threshold)
45      errorsI++;
46  endif
47 endfor
48
49 printf("Alpha: %d\n", 1 - gamma);
50 printf("Estimated alpha: %d\n", errorsI / m);

```

Результат

Alpha: 0.05

Estimated alpha: 0.04

Вывод

При объединении соседних интервалов для выполнения $n_j \geq 6$ для нормального распределения оценка α близка к теоретической.

3. Испортить выборочные параметры и провести оценку β .

- Равномерное распределение

Делаем изменение вида $U(a - eps, b + eps)$ для различных eps из

0..0.005.

```
1  clc;
2  clear;
3
4  pkg load statistics;
5
6  n = 1e6;
7  m = 1e2;
8
9  a = 0;
10 b = 1;
11
12 gamma = 0.95;
13
14 for eps = 0 : 0.001 : 0.005
15     errorsII = 0;
16     for i = 1:m
17         x = sort(unifrnd(a, b, n, 1));
18         mi = min(x) - eps;
19         ma = max(x) + eps;
20
21         y = hist(x, m);
22         step = (x(n) - x(1)) / m;
23
24         y1 = unifcdf(x(1) : step : x(end) - step, mi, ma);
25         y2 = unifcdf(x(1) + step : step : x(end), mi, ma);
26
27         p = y2 - y1;
28         hi2 = sum(((y - n * p) .^ 2) ./ (n * p));
29         threshold = chi2inv(gamma, m - 3);
30         if (hi2 < threshold)
31             errorsII++;
32         endif
33     endfor
34
35     printf("Estimated beta for eps = %d: %d\n", eps, errorsII / m);
36 endfor
```

Результат

Estimated beta for eps = 0: 0.9

Estimated beta for eps = 0.001: 0.9

Estimated beta for eps = 0.002: 0.63

Estimated beta for eps = 0.003: 0.28

Estimated beta for eps = 0.004: 0

Estimated beta for eps = 0.005: 0

- Нормальное распределение

Делаем изменение вида $N(a + \text{eps}, \sigma)$ для различных eps из 0..0.05

```

1  clc;
2  clear;
3
4  pkg load statistics;
5
6  n = 1e6;
7  m = 1e2;
8
9  gamma = 0.95;
10
11 for eps = 0 : 0.01 : 0.05
12     errorsII = 0;
13
14     for i = 1:m
15         x = sort(normrnd(1, 3, n, 1));
16
17         y = hist(x, m);
18         step = (x(n) - x(1)) / m;
19
20         E = mean(x) + eps;
21         sigma = std(x);
22
23         y1 = normcdf(x(1) : step : x(end) - step, E, sigma);
24         y2 = normcdf(x(1) + step : step : x(end), E, sigma);
25
26         p = y2 - y1;
27         hi2 = sum(((y - n * p) .^ 2) ./ (n * p));
28         threshold = chi2inv(gamma, m - 3);
29         if (hi2 < threshold)
30             errorsII++;

```

```
31     endif
32   endfor
33
34   printf("Estimated beta for eps = %d: %d\n", eps, errorsII / m);
35 endfor
```

Результат

```
Estimated beta for eps = 0: 0.9
Estimated beta for eps = 0.01: 0.61
Estimated beta for eps = 0.02: 0.02
Estimated beta for eps = 0.03: 0
Estimated beta for eps = 0.04: 0
Estimated beta for eps = 0.05: 0
```

Вывод

Оценив ошибки второго рода для различных ϵ , убеждаемся в том, что при малых изменениях ошибка второго рода почти не меняется, однако при больших изменениях ϵ критерий χ^2 резко реагирует на это (ошибка второго рода с некоторого значения ϵ резко стремится к 0).