

Лабораторная работа 3

Борис Шапошников

Виктор Шарепов

Рамазан Рахматуллин

Обратная связь: bshaposhnikov01@gmail.com

Часть 1

Для случайной величины, распределенной по нормальному закону с параметрами (a, σ^2) и по равномерному распределению с параметрами (a, b) , выполнить следующие действия.

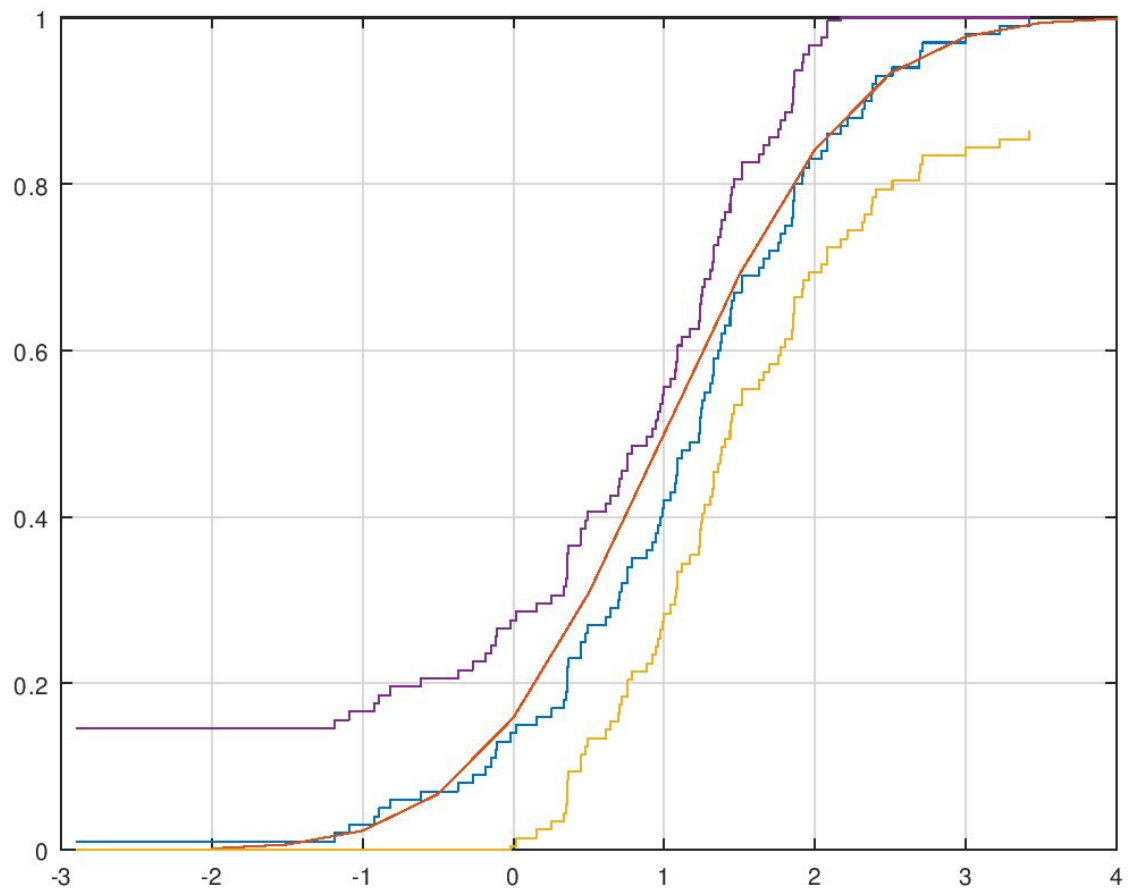
- Задать параметры распределения $X \sim N(a, \sigma^2)$, $X \sim U(a, b)$.
- Построить график $F_x(x)$, используя функцию `normcdf`.
- При $n=100$ построить выборку из генеральной совокупности X .
- По построенной выборке построить график эмпирической функции распределения $F_n(x)$, используя при построении встроенную функцию `[a,b]=stairs(x,y)` для построения кусочно- постоянной функции. Учесть при построении, что $F_n(x)$ изменяется на $1/n$ в каждой следующей точке выборки.
- Построить доверительную полосу надежности $\gamma = 0.95$; $u(\gamma) = 1.36$ (см. пособие стр.92-96).
- На этом же графике построить $F_n(x)$ и $F_x(x)$. Убедиться, что функция распределения попадает (?) в доверительную полосу.

Код и графики:

- Нормальное распределение

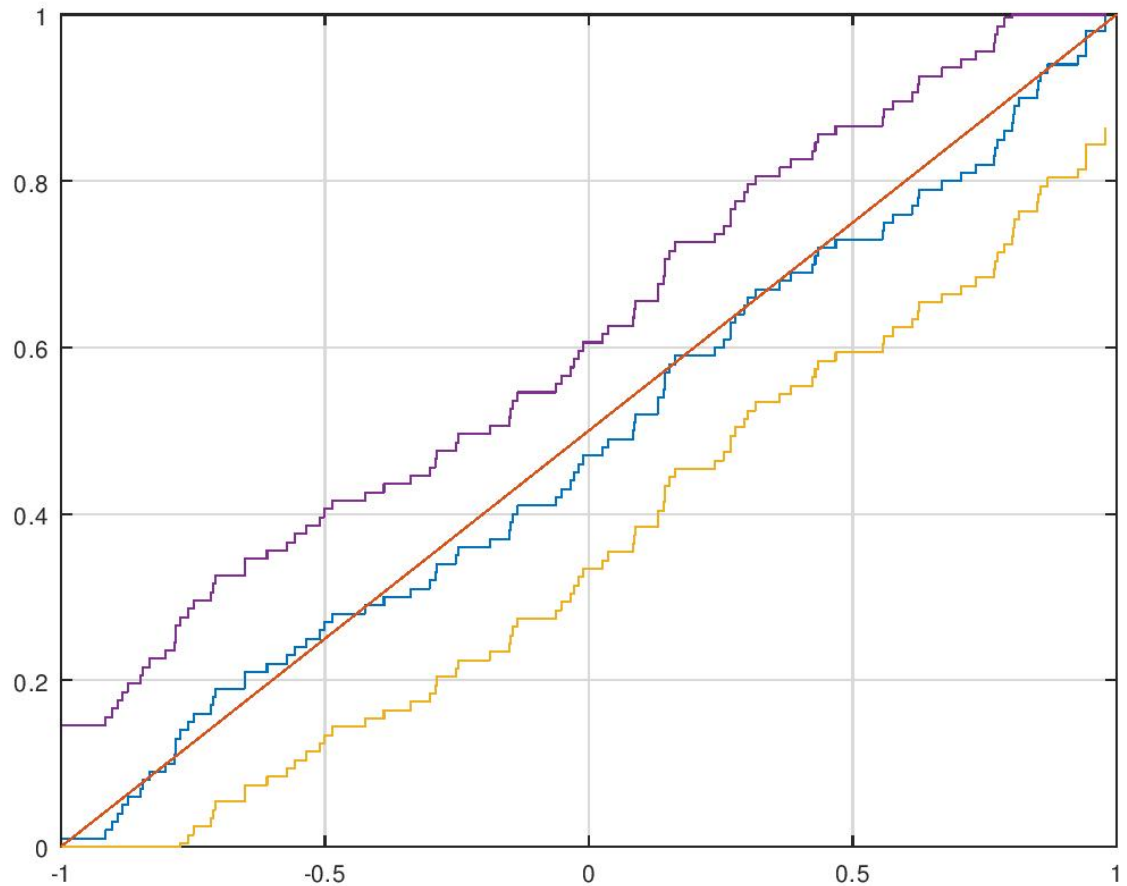
```
1 pkg load statistics;  
2 clc;  
3 clear;  
4
```

```
5 E = 1;
6 sigma = 1;
7
8 n = 100;
9 m = 1;
10
11 v = sort(normrnd(E, sigma, n, m));
12
13 Fn = 1 / n : 1 / n : 1;
14 [a, b] = stairs(v, Fn);
15
16 x = E - 3 * sigma : 0.5 : E + 3 * sigma;
17 F = normcdf(x, E, sigma);
18
19 d = 1.36 / sqrt(n);
20 b1 = max(0, b - d);
21 b2 = min(1, b + d);
22 plot(a, b, x, F, a, b1, a, b2), grid;
```



● Равномерное распределение

```
1 pkg load statistics;
2 clc;
3 clear;
4
5 x = -1;
6 y = 1;
7
8 n = 100;
9 m = 1;
10 v = sort(unifrnd(x, y, n, m));
11
12 Fn = 1 / n : 1 / n : 1;
13 Fx = -1 : 1 / n : 1;
14 F = unifcdf(Fx, x, y);
15 [a, b] = stairs(v, Fn);
16
17
18 d = 1.36 / sqrt(n);
19
20 b1 = max(0, b - d);
21 b2 = min(1, b + d);
22 plot(a, b, Fx, F, a, b1, a, b2), grid;
```



Вывод из первой части: реальная функция распределения попала в доверительную полосу.

Часть 2

На основе критерия Колмогорова (если сможете, и на основе критерия Смирнова) провести проверку гипотез при $n = 10^4$ и $n = 10^6$ (см. пособие стр. 96-97).

- Нормальное распределение

```

1 pkg load statistics;
2 clc;
3 clear;
4
5 E = 1;
6 sigma = 1;
7 n = 10^4;
8 Fn = 1 / n : 1 / n : 1;

```

```

9
10 fprintf("Selection: %d\nI type error\n\n", n);
11 for m = 100:100:1000
12     fprintf("m = %d\n", m);
13     v = sort(normrnd(E, sigma, n, m));
14     F = normcdf(v, E, sigma);
15     diff = F - Fn';
16     sup = max(max(abs(diff), abs(diff - 1 / n)));
17     kolmogorov = sqrt(n) * sup;
18     k = mean(kolmogorov > 1.36);
19
20     smirnov = 1 / (12*n) + sum((diff + 1 / (2*n)) .^ 2);
21     s = mean(smirnov > 0.46);
22     fprintf("Kolmogorov: %d\nSmirnov: %d\n\n", k, s);
23 endfor
24
25 fprintf("=====\n")
26 fprintf("Selection: %d\nII type error\n\n", n);
27 v = sort(normrnd(E, sigma, n, m));
28 for a = 0:0.01:0.1
29     fprintf("addition = %d\n", a);
30     F = normcdf(v, E, sigma + a);
31     diff = F - Fn';
32     sup = max(max(abs(diff), abs(diff - 1 / n)));
33     kolmogorov = sqrt(n) * sup;
34     k = 1 - mean(kolmogorov > 1.36);
35
36     smirnov = 1 / (12*n) + sum((diff + 1 / (2*n)) .^ 2);
37     s = 1 - mean(smirnov > 0.46);
38     fprintf("Kolmogorov: %d\nSmirnov: %d\n\n", k, s);
39 endfor
40
41 n = 10^6;
42 Fn = 1 / n : 1 / n : 1;
43
44 fprintf("=====\n")
45 fprintf("=====\n")
46 fprintf("Selection: %d\nI type error\n\n", n);
47 for m = 100:100:1000
48     fprintf("m = %d\n", m);
49     v = sort(normrnd(E, sigma, n, m));
50     F = normcdf(v, E, sigma);
51     diff = F - Fn';

```

```

52  sup = max(max(abs(diff), abs(diff - 1 / n)));
53  kolmogorov = sqrt(n) * sup;
54  k = mean(kolmogorov > 1.36);
55
56  smirnov = 1 / (12*n) + sum((diff + 1 / (2*n)) .^ 2);
57  s = mean(smirnov > 0.46);
58  fprintf("Kolmogorov: %d\nSmirnov: %d\n\n", k, s);
59 endfor
60
61 fprintf("=====\n")
62 fprintf("Selection: %d\nII type error\n\n", n);
63 v = sort(normrnd(E, sigma, n, m));
64 for a = 0:0.01:0.1
65     fprintf("addition = %d\n", a);
66     F = normcdf(v, E, sigma + a);
67     diff = F - Fn';
68     sup = max(max(abs(diff), abs(diff - 1 / n)));
69     kolmogorov = sqrt(n) * sup;
70     k = 1 - mean(kolmogorov > 1.36);
71
72     smirnov = 1 / (12*n) + sum((diff + 1 / (2*n)) .^ 2);
73     s = 1 - mean(smirnov > 0.46);
74     fprintf("Kolmogorov: %d\nSmirnov: %d\n\n", k, s);
75 endfor

```

Вывод программы см. в приложенном файле normoutput.txt.

- Равномерное распределение

```

1  pkg load statistics;
2  clc;
3  clear;
4
5  n = 10^4;
6  Fn = 1 / n : 1 / n : 1;
7  x = -1;
8  y = 1;
9  Fx = -1 : 2 / n : 1;
10
11 fprintf("Selection: %d\nI type error\n\n", n);
12 for m = 100:100:1000
13     fprintf("m = %d\n", m);
14     v = sort(unifrnd(x, y, n, m));
15     F = unifcdf(v, x, y);
16     diff = F - Fn';

```

```

17  sup = max(max(abs(diff), abs(diff - 1 / n)));
18  kolmogorov = sqrt(n) * sup;
19  k = mean(kolmogorov > 1.36);
20
21  smirnov = 1 / (12*n) + sum((diff + 1 / (2*n)) .^ 2);
22  s = mean(smirnov > 0.46);
23  fprintf("Kolmogorov: %d\nSmirnov: %d\n\n", k, s);
24  endfor
25
26  fprintf("=====\n")
27  fprintf("Selection: %d\nII type error\n\n", n);
28  v = sort(unifrnd(x, y, n, m));
29  for a = 0:0.01:0.1
30      fprintf("addition = %d\n", a);
31      F = unifcdf(v, x + 3 * a, y + 3 * a);
32      diff = F - Fn';
33      sup = max(max(abs(diff), abs(diff - 1 / n)));
34      kolmogorov = sqrt(n) * sup;
35      k = 1 - mean(kolmogorov > 1.36);
36
37      smirnov = 1 / (12*n) + sum((diff + 1 / (2*n)) .^ 2);
38      s = 1 - mean(smirnov > 0.46);
39      fprintf("Kolmogorov: %d\nSmirnov: %d\n\n", k, s);
40  endfor
41
42  n = 10^6;
43  Fn = 1 / n : 1 / n : 1;
44
45  fprintf("=====\n")
46  fprintf("=====\n")
47  fprintf("Selection: %d\nI type error\n\n", n);
48  for m = 100:100:1000
49      fprintf("m = %d\n", m);
50      v = sort(unifrnd(x, y, n, m));
51      F = unifcdf(v, x, y);
52      diff = F - Fn';
53      sup = max(max(abs(diff), abs(diff - 1 / n)));
54      kolmogorov = sqrt(n) * sup;
55      k = mean(kolmogorov > 1.36);
56
57      smirnov = 1 / (12*n) + sum((diff + 1 / (2*n)) .^ 2);
58      s = mean(smirnov > 0.46);
59      fprintf("Kolmogorov: %d\nSmirnov: %d\n\n", k, s);

```

```

60 endfor
61
62 fprintf("=====\n")
63 fprintf("Selection: %d\nII type error\n\n", n);
64 v = sort(unifrnd(x, y, n, m));
65 for a = 0:0.001:0.01
66     fprintf("addition = %d\n", a);
67     F = unifcdf(v, x + a, y + a);
68     diff = F - Fn';
69     sup = max(max(abs(diff), abs(diff - 1 / n)));
70     kolmogorov = sqrt(n) * sup;
71     k = 1 - mean(kolmogorov > 1.36);
72
73     smirnov = 1 / (12*n) + sum((diff + 1 / (2*n)) .^ 2);
74     s = 1 - mean(smirnov > 0.46);
75     fprintf("Kolmogorov: %d\nSmirnov: %d\n\n", k, s);
76 endfor

```

Вывод программы см. в приложенном файле unioutput.txt.

Вывод из второй части:

Эмпирическая функция распределения хорошо оценивает реальную функцию распределения. Реальная функция распределения попадает в доверительный интервал. Ошибки первого рода (гипотеза, верна, но отвергается), как показывают тесты, для эмпирической функции распределения, случаются редко и при совпадении параметров основной функции распределения и той, из которой делаем выборку, верная гипотеза почти никогда не отвергается. При изменении параметров функции распределения, из которой делаем выборку, относительно основной, критерий согласия Колмогорова и Смирнова начинают резко отвергать гипотезу.