Course: R Language in Computational Probability and Statistics

Lecture 12: Descriptive statistics

Lecturer: Nataliia Kruhlova

Побудова статистичного ряду

Для перетворення числової змінної у факторну ми використовуємо функцію **cut()**:

cut(x, breaks...)

-**x** — числовий вектор, що перетворюється на фактор;

-**breaks**—цей параметр може бути числом, яке буде задавати кількість інтервалів розбиття (обов'язково більше за 2), або вектором, який містить межі інтервалів;

-**labels** — символьний вектор назв категорій;

-**include.lowest** — логічний параметр, який відповідає за включення до інтервалу елементу, що співпадає з нижньою/верхньою межею інтервалу;

-**right** — логічний параметр, який вказує, яку саме межу інтервалу треба включити в інтервал;

-**ordered_result** — логічний параметр, який відповідає за створення впорядкованого фактору.

Для перетворення вибірки на статистичний ряд можна використовувати функцію **table().**

Приклад. Із сімейним лікарем Здоровейко підписали декларацію 40 людей, віком від 5 до 91 року. Лікар сказав своїй медсестрі надати йому інформацію про частоту кожної вікової категорії пацієнтів. Здоровейко своїх пацієнтів ділить на наступні категорії за віком: 5-13 – діти, 14-20 – підлітки, 21-35 – молоді люди, 36-55 – люди середнього віку, 56-91 – люди старшого віку.

```
> age<-sample(5:91,40,replace=T)
> age
 [1] 54 44 69 29 36 54 39 46 61 73 15 11 71 33 14 69 81 69 26 85 27
 47 20 72 86
[26] 63 56 50 73 28 85 60 52  9 18  7 16 41 75 19
> x<-cut(age,breaks=c(5,13,20,35,55,91))
> x<-cut(age,breaks=c(5,13,20,35,55,91),labels=c("child","teenager",
"young","middle","old"))
> table(x)
x
   child teenager    young   middle      old
       3        6        5       10       16
```

Числові характеристики випадкових величин

Функція **mean()** обчислює вибіркове середнє вибірки, яка є єдиним обов'язковим аргументом цієї функції.

-trim – аргумент, який визначає долю елементів вибірки, що мають бути видалені від початку і від кінця вибірки перед визначенням середнього. Можливі значення – числа від 0 до 0.5.

-na.rm – дозволяє обчислювати середнє вибірки без врахування NA.

Функція **sd()** визначає корінь з виправленої вибіркової дисперсії (стандартне відхилення).

Функція **var()** обчислює виправлену вибіркову дисперсію для вибірки x, якщо x — числовий вектор, і коваріаційну матрицю, якщо x — матриця. В останньому випадку кожен стовпчик матриці вважається вибіркою. Якщо задано другий елемент y, і якщо x і y — вектори, то будується коваріаційна матриця 2 × 2. Якщо x і y — матриці або таблиці даних однакової розмірності, то будується коваріаційна матриця, елементами якої являються коваріації відповідних стовпчиків матриці x і матриці y.

Приклад. Згенеруємо 100 значень біноміальної випадкової величини з параметрами n=10, p=1/2. Для одержаної вибірки потрібно побудувати статистичний ряд, знайти середнє, дисперсію, стандартне відхилення.

```
> x<-rbinom(100,10,1/2)
> table(x)
x
 1  2  3  4  5  6  7  8
 1  4 15 17 27 21  9  6
> mean(x)
[1] 4.94
> sd(x)
[1] 1.55583
> var(x)
[1] 2.420606
```

Функція **weighted.mean(x, w)** використовується для знаходження середньозваженого середнього за вектором **x**; аргумент **w** вказує вагу елементів **x**.

Функція **rank()** визначає ранги елементів вибірки.

Функція **cov()** призначена для побудови коваріаційної матриці заданих вибірок, а функція **cor()** будує матрицю коефіцієнтів кореляції. Для цих функцій бажано, щоб один з аргументів **x** або **y** був матрицею.

Функція **cov2cor()** будує кореляційну матрицю на основі заданої коваріаційної матриці.

Аргументами цих функцій є:

• **x** і **y** — числові вектори, матриці або фрейми.

• **na.rm** — логічний аргумент, що дозволяє не розглядати пропущені значення NA.

• **method** — символьний аргумент, що визначає метод обчислення коваріації:

– «pearson» (по замовченню) визначається звичайний коефіцієнт коваріації чи кореляції.

– «kendall» і «spearman» — рангові коефіцієнти кореляції.

• **V** — додатно визначена симетрична числова матриця (аргумент функції **cov2cor()**.

Приклад. Для змінних **Wind** і **Temp** з набору даних **airquality** побудуйте коваріаційну матрицю, використовуючи метод Пірсона; побудуйте кореляційну матрицю, використовуючи метод Спірмена.

```
> new<-airquality[,c(3,4)]
> cov(new)
         Wind      Temp
Wind  12.41154 -15.27214
Temp -15.27214  89.59133
> cor(new,method="spearman")
          Wind       Temp
Wind  1.0000000 -0.4465408
Temp -0.4465408  1.0000000
```

Функція **quantile()** дозволяє знаходити квантилі вибірки заданого розміру α ($0 \le \alpha \le 1$).

Аргументами функції є:

• **x** — числовий вектор (вибірка). Не повинен містити пропущених значень.

• **probs** — числовий вектор (ймовірності).

По замовчуванню обчислюються квартилі.

• **na.rm** — логічний аргумент, який вказує, чи потрібно видаляти з вибірки пропущені значення.

• **names** — логічний аргумент, що вказує, чи потрібно привласнювати імена результатам (вказувати, які квантилі обчислено).

• **type** — додатне ціле число, що приймає значення від 1 до 9 і визначає алгоритм обчислення квантилів.

Приклад. Згенеруємо експоненціальну вибірку з параметром $\lambda = 5$ і обчислимо для неї квартилі; квантилі з кроком 0.1.

```
> x<-rexp(100,5)
> quantile(x)
         0%         25%         50%         75%        100%
0.0003832395 0.0407144175 0.1106848664 0.2608322718 0.9187865333
> quantile(x,seq(0,1,by=0.1))
         0%         10%         20%         30%         40%
0.0003832395 0.0151675008 0.0311414139 0.0575266951 0.0765031102
        50%         60%         70%         80%         90%
0.1106848664 0.1601318860 0.2283770655 0.3054185381 0.4197781501
       100%
0.9187865333
```

Функція **IQR()** обчислює міжквартільний розмах: **IQR(x) = quantile(x,3/4) - quantile(x,1/4)**.

Приклад. Для вибірки з попереднього прикладу знайдемо міжквартільний розмах.

```
> IQR(x)
[1] 0.2201179
```

Функція **median()** знаходить медіану вибірки.

Приклад. Знайдемо медіану для вибірки з експоненціального розподілу (див. вище).

```
> median(x)
[1] 0.1106849
```

Для моди немає спеціальної функції. Тому використовують наступну конструкцію.

Приклад.

```
> moda<- unique(x)[which.max(tabulate(match(x, unique(x))))]
> moda
[1] 0.0321643
```

Пакет **moments** дозволяє знаходити коефіцієнт асиметрії і ексцес.

Приклад.

```
> install.packages("moments")

>library(moments)

> kurtosis(x, na.rm=TRUE)

[1] 5.886437
> skewness(x, na.rm = TRUE)

[1] 1.665976
```

Функція **summary()** виводить основні характеристики вибірки: мінімум і максимум, медіану і середнє, перший і третій квартілі.

Приклад. Знайдемо основні характеристики змінної **Temp** з набору даних **airquality**.

```
> summary(airquality$Temp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  56.00   72.00   79.00   77.88   85.00   97.00
```

Спеціальні пакети та функції для описових зведень фреймів

Багато пакетів у R мають власні функції, аналогічні стандартній функції **summary()**, для виведення компактних описових зведень за таблицями даних. Нижче наведено декілька прикладів таких пакетів та функцій.

Пакет **Hmisc**, функція **describe().**

Приклад.

```
>describe(mtcars[,1:3])
mtcars[, 1:3]

 3  Variables      32  Observations
--------------------------------------------------------------------------------
mpg
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25
      32        0       25    0.999    20.09    6.796    12.00    14.34    15.43
     .50      .75      .90      .95
   19.20    22.80    30.09    31.30

lowest : 10.4 13.3 14.3 14.7 15.0, highest: 26.0 27.3 30.4 32.4 33.9
--------------------------------------------------------------------------------
cyl
       n  missing distinct     Info     Mean      Gmd
      32        0        3    0.866    6.188    1.948

Value          4     6     8
Frequency     11     7    14
Proportion 0.344 0.219 0.438
--------------------------------------------------------------------------------
disp
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25
      32        0       27    0.999    230.7    142.5    77.35    80.61   120.83
     .50      .75      .90      .95
  196.30   326.00   396.00   449.00

lowest :  71.1  75.7  78.7  79.0  95.1, highest: 360.0 400.0 440.0 460.0 472.0
--------------------------------------------------------------------------------
```

Пакет **pastecs**, функція **stat.desc()**.

Приклад.

```
> stat.desc(mtcars[,1:3])
                      mpg        cyl        disp
nbr.val      32.0000000  32.0000000 3.200000e+01
nbr.null      0.0000000   0.0000000 0.000000e+00
nbr.na        0.0000000   0.0000000 0.000000e+00
min          10.4000000   4.0000000 7.110000e+01
max          33.9000000   8.0000000 4.720000e+02
range        23.5000000   4.0000000 4.009000e+02
sum         642.9000000 198.0000000 7.383100e+03
median       19.2000000   6.0000000 1.963000e+02
mean         20.0906250   6.1875000 2.307219e+02
SE.mean       1.0654240   0.3157093 2.190947e+01
CI.mean.0.95  2.1729465   0.6438934 4.468466e+01
var          36.3241028   3.1895161 1.536080e+04
std.dev       6.0269481   1.7859216 1.239387e+02
coef.var      0.2999881   0.2886338 5.371779e-01
```

Пакет **psych**, функція **describeBy()** - розрахунок числових характеристик вибірок для кожного рівня деякого фактору.

Приклад.

```
>describeBy(mtcars[,c(1,3)], mtcars$cyl)

 Descriptive statistics by group

group: 4

    vars  n   mean    sd median trimmed   mad  min   max range skew kurtosis

mpg     1 11  26.66  4.51     26   26.44  6.52 21.4  33.9  12.5 0.26    -1.65

disp    2 11 105.14 26.87    108  104.30 43.00 71.1 146.7  75.6 0.12    -1.64

      se

mpg  1.36

disp 8.10

------------------------------------------------------------

group: 6

    vars n   mean    sd median trimmed   mad   min   max range  skew kurtosis

mpg     1 7  19.74  1.45   19.7   19.74  1.93  17.8  21.4   3.6 -0.16    -1.91

disp    2 7 183.31 41.56  167.6  183.31 11.27 145.0 258.0 113.0  0.80    -1.23

       se

mpg   0.55

disp 15.71

------------------------------------------------------------

group: 8

    vars  n  mean    sd median trimmed   mad   min   max range  skew kurtosis

mpg     1 14  15.1  2.56   15.2   15.15  1.56  10.4  19.2   8.8 -0.36    -0.57

disp    2 14 353.1 67.77  350.5  349.63 73.39 275.8 472.0 196.2  0.45    -1.26

       se

mpg   0.68

disp 18.11
```

Пакет **doBy**, функція **summaryBy()**.

Приклад.

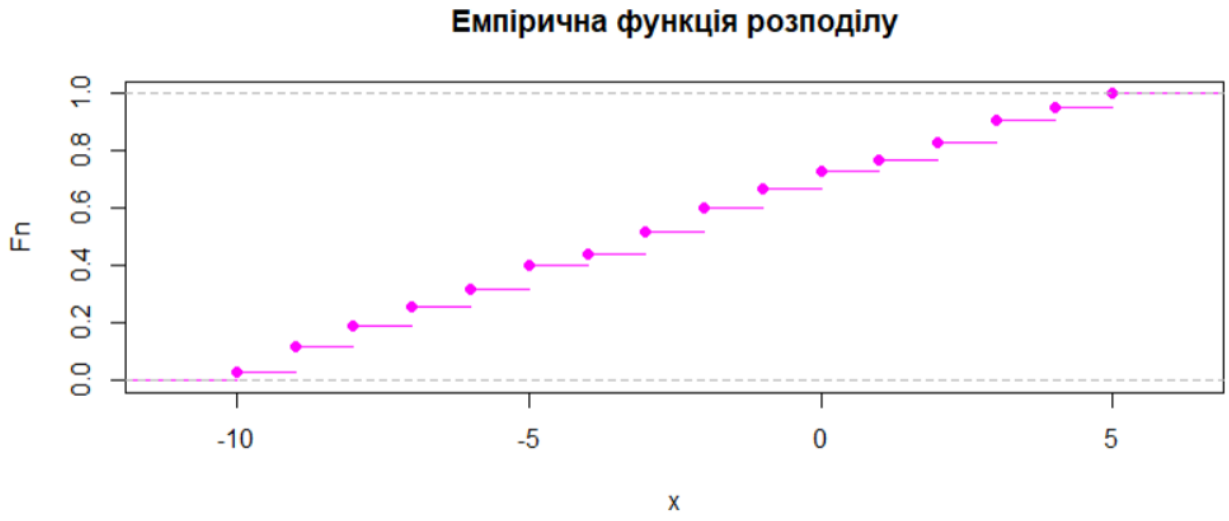
```
> summaryBy(mpg ~ cyl + vs, data = mtcars,
+           FUN = function(x) { c(m = median(x), v= var(x,na.rm=T)) } )
  cyl vs mpg.m      mpg.v
1   4  0 26.00         NA
2   4  1 25.85 22.5445556
3   6  0 21.00  0.5633333
4   6  1 18.65  2.6625000
5   8  0 15.20  6.5538462
```

Побудова емпіричної функції розподілу

Функція **ecdf()** будує емпіричну функцію розподілу вибірки.

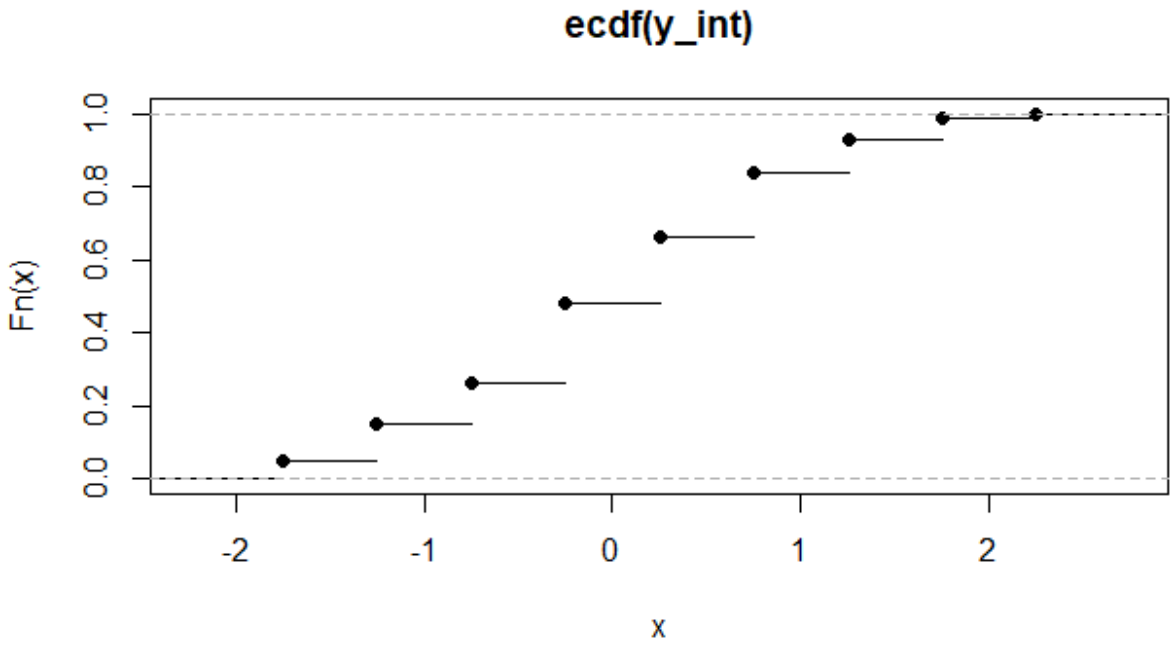
Приклад. Згенеруйте вибірку об’ємом100 і побудуйте для неї емпіричну функцію розподілу.

```
> x<-sample(-10:5,100,replace=T)
> plot(ecdf(x),col="magenta",ylab="Fn",main="Емпірична функція розподілу")
```



Приклад. Згенеруємо 100 значень нормальної стандартної випадкової величини. Поділимо вибірку на 7 інтервалів і побудуємо емпіричну функцію розподілу для такого інтервального розподілу.

```
> x<-rnorm(100)
> y<-hist(x,plot=F,breaks=7)
> y_int<-rep(y$mids,y$counts)
> plot(ecdf(y_int))
```



Всі малюнки в лекції створені в RStudio.