Лабораторна робота

Простір елементарних подій. Події. Ймовірність події

ПРОСТІР ЕЛЕМЕНТАРНИХ ПОДІЙ

Деякі простори елементарних подій зустрічаються настільки часто, що для зручності роботи з ними в R були написані відповідні функції, які входять в пакет **“prob”**. Розглянемо випадковий експеримент з **підкиданням монети**. Результатами є **H** і **T** (**Г**ерб та **Р**ешка). Можна швидко побудувати простір елементарних подій за допомогою функції **tosscoin()**:

Підкидання монети 1 раз:

```
> library(prob)
> tosscoin(1)

  toss1
1     H
2     T
```

3 рази:

```
> tosscoin(3)

  toss1 toss2 toss3
1     H     H     H
2     T     H     H
3     H     T     H
4     T     T     H
5     H     H     T
6     T     H     T
7     H     T     T
8     T     T     T
```

Підкидання кубика (за замовчуванням – шестигранного):

```
> rolldie(1)

  X1
1  1
2  2
3  3
4  4
5  5
6  6
```

Для того, щоб змінити кількість граней потрібно задати інше значення параметра **nsides** функції **rolldie()**. Команда **rolldie(3, nsides=4)** буде використовуватися для підкидання кубика з 4-ма гранями 3 рази.

Вибір гральної карти з колоди:

```
> head(cards())

  rank suit
1    2 Club
2    3 Club
3    4 Club
4    5 Club
5    6 Club
6    7 Club
```

Тут функція **head()** використовується для того, щоб відобразити тільки декілька перших рядків об'єкта (велика таблиця даних).

Функція **cards()**, має необов'язкові параметри **jokers** (якщо потрібно, щоб в колоді були джокери – **jokers=TRUE**) і **makespace**, про яких буде написано дещо пізніше.

Вибір об'єкта (кульки) з урни.

Це, мабуть, найбільш фундаментальний тип випадкового експерименту. Ми маємо урну, що містить набір помічених об'єктів (кульок) всередині. Перемішуємо їх, вибираємо одну з кульок і дивимося. Це все. Але є різні варіації на цю тему. Можливо ми б хотіли вибрати не одну, а декілька кульок, наприклад дві. Які всеможливі результати експерименту у такому випадку? Все залежить від того, як ми це робимо. Ми могли б вибрати кульку, подивитися, покласти її назад, і вибрати знову. Інший варіант – це вибрати кульку, подивитися – але не класти її назад – і вибрати знову (що те ж саме, що вибрати зразу дві кульки). У першому випадку ми маємо вибір з поверненням, а у другому – без. Якщо, крім того, потрібно враховувати порядок, в якому вибираються кульки, то мова йде про впорядковані вибірки.

Варто звернути увагу на те, що цей загальний клас випадкових експериментів містить, як окремий випадок, всі експерименти, що ми розглядали до того. Підкидання монети два рази еквівалентно послідовному вибору двох кульок помічених як H і T з урни, із поверненням. Експеримент з кубиком еквівалентний вибору кульки з урни з шістьма елементами, поміченими цифрами від 1 до 6.

Пакет **"prob"** реалізує вибірку з урни за допомогою функції **urnsamples()**, яка має аргументи **x**, **size**, **replace**, і **order**. Аргумент **x** – це урна, з якої має бути зроблено вибірку, а **size** – кількість об'єктів, що відбираються. Параметри **replace** і **order** вказують на особливості проведення експерименту такі як: вибір з поверненням чи без; впорядкована чи невпорядкована вибірка.

Приклад 1. Нехай урна містить три кульки з мітками 1, 2 і 3 відповідно. Виймаються дві кульки з урни.

Впорядкована вибірка, з поверненням.

```
> urnsamples(1:3, size = 2, replace = TRUE, ordered = TRUE)
  X1 X2
1  1  1
2  2  1
3  3  1
4  1  2
5  2  2
6  3  2
7  1  3
8  2  3
9  3  3
```

Зауважимо, що цей експеримент еквівалентний підкиданню кубика з трьома гранями двічі (**rolldie(2, nsides=3)**).

Впорядкована вибірка, без повернень.

```
> urnsamples(1:3, size = 2, replace = FALSE, ordered = TRUE)
  X1 X2
1  1  2
2  2  1
3  1  3
4  3  1
5  2  3
6  3  2
```

Якщо числа 1, 2, і 3 замінити на "Фред", "Марія" і "Сью", то цей експеримент буде еквівалентний вибору двох осіб з трьох, на посади президента і віце-президента компанії, відповідно, а результат виконання функції показує всі можливі способи, якими це можна зробити.

Невпорядкована вибірка, без повернень

```
> urnsamples(1:3, size = 2, replace = FALSE, ordered = FALSE)
  X1 X2
1  1  2
2  1  3
3  2  3
```

Еквівалентно вибору пари кульок з урни (**urnsamples(1:3, 2)**).

Невпорядкована вибірка, з поверненням

```
> urnsamples(1:3, size = 2, replace = TRUE, ordered = FALSE)
  X1 X2
1  1  1
2  1  2
3  1  3
4  2  2
5  2  3
6  3  3
```

Кульки не обов’язково повинні бути помічені цифрами. Можна, наприклад, зробити наступним чином **x = c("Red","Blue","Green")**.

ПОДІЇ

Маючи таблицю даних (data frame) з елементарними подіями і ймовірностями їх настання ми можемо відібрати з неї окремі рядки, використовуючи оператор **[]**:

```
> S <- tosscoin(2, makespace = TRUE)
  toss1 toss2 probs
1     H     H  0.25
2     T     H  0.25
3     H     T  0.25
4     T     T  0.25

> S[1:3, ]
  toss1 toss2 probs
1     H     H  0.25
2     T     H  0.25
3     H     T  0.25


> S[c(2, 4), ]
  toss1 toss2 probs
2     T     H  0.25
4     T     T  0.25
```

Ми також можемо відібрати рядки, що задовольняють логічний вираз, використовуючи функцію **subset()**, наприклад

```
> S <- cards()
> subset(S, suit == "Heart")
   rank  suit
27    2 Heart
28    3 Heart
29    4 Heart
30    5 Heart
31    6 Heart
32    7 Heart
33    8 Heart
34    9 Heart
35   10 Heart
36    J Heart
37    Q Heart
38    K Heart
39    A Heart

> subset(S, rank %in% 7:9)
   rank    suit
6     7    Club
7     8    Club
8     9    Club
19    7 Diamond
20    8 Diamond
21    9 Diamond
32    7   Heart
33    8   Heart
34    9   Heart
45    7   Spade
46    8   Spade
47    9   Spade
```

Допускається використання математичних виразів:

```
> subset(rolldie(3), X1 + X2 + X3 > 16)
    X1 X2 X3
180  6  6  5
210  6  5  6
215  5  6  6
216  6  6  6
```

Функції для знаходження підмножин

1. Функція **%in%**

Перевіряє чи належить значення одного вектора іншому вектору.

```
> x <- 1:10
> y <- 8:12
> y %in% x
[1]  TRUE  TRUE  TRUE FALSE FALSE
```

2. Функція **isin()**

Використовується, якщо потрібно перевірити чи весь вектор *y* належить вектору *x*.

```
> isin(x, y)
[1] FALSE
```

```
> x <- 1:10
> y <- c(3, 3, 7)
> isin(x, y)
[1] FALSE
```

В останньому прикладі *x* містить 3, але не містить двох трійок.

Відмітимо, що функція **isin()** має необов'язковий параметр **ordered**, для врахування порядку елементів при перевірці:

```
> isin(x, c(3, 4, 5), ordered = TRUE)
[1] TRUE
```

```
> isin(x, c(3, 5, 4), ordered = TRUE)
[1] FALSE
```

Наведемо приклад знаходження підмножини простору елементарних подій з використанням функції **isin()**.

```
> S <- rolldie(4)
> subset(S, isin(S, c(2, 2, 6), ordered = TRUE))
     X1 X2 X3 X4
188   2  2  6  1
404   2  2  6  2
620   2  2  6  3
836   2  2  6  4
1052  2  2  6  5
1088  2  2  1  6
1118  2  1  2  6
1123  1  2  2  6
```

```
1124   2   2   2   6
1125   3   2   2   6
1126   4   2   2   6
1127   5   2   2   6
1128   6   2   2   6
1130   2   3   2   6
1136   2   4   2   6
1142   2   5   2   6
1148   2   6   2   6
1160   2   2   3   6
1196   2   2   4   6
1232   2   2   5   6
1268   2   2   6   6
```

Зауважимо, що функція **isin()** застосовується до кожного рядка таблиці даних *S*.

Об'єднання, перетин і різниця множин

Назва	Позначення	Функція в R
Об'єднання	$A \cup B$	union(A,B)
Перетин	$A \cap B$	intersect(A,B)
Різниця	$A \backslash B$	setdiff(A,B)

```
> S = cards()
> A = subset(S, suit == "Heart")
> B = subset(S, rank %in% 7:9)
> union(A, B)
   rank    suit
6     7    Club
7     8    Club
8     9    Club
19    7 Diamond
20    8 Diamond
21    9 Diamond
27    2   Heart
28    3   Heart
29    4   Heart
30    5   Heart
31    6   Heart
32    7   Heart
33    8   Heart
34    9   Heart
35   10   Heart
36    J   Heart
37    Q   Heart
38    K   Heart
39    A   Heart
45    7   Spade
46    8   Spade
47    9   Spade

> intersect(A, B)
   rank  suit
32    7 Heart
33    8 Heart
34    9 Heart

> setdiff(A, B)
   rank  suit
```

```
27     2 Heart
28     3 Heart
29     4 Heart
30     5 Heart
31     6 Heart
35    10 Heart
36     J Heart
37     Q Heart
38     K Heart
39     A Heart

> setdiff(B, A)
   rank    suit
6     7    Club
7     8    Club
8     9    Club
19    7 Diamond
20    8 Diamond
21    9 Diamond
45    7   Spade
46    8   Spade
47    9   Spade
```

ЙМОВІРНІСТЬ ПОДІЇ

У пакеті **"prob"** ймовірнісний простір є об'єктом результатів S і вектора ймовірностей (probs) із записами, які відповідають кожному результату в S. Якщо S є таблицею даних, ми можемо просто додати стовпець з назвою **probs** до S; ймовірнісним простором буде отримана таблиця даних, яку можна назвати S. У випадку, коли S є списком, ми можемо об'єднати результати експерименту і probs в більший список, який буде мати дві складові: результати і probs. Єдині вимоги: всі значення в probs повинні бути невід'ємними і їхня сума дорівнює одиниці.

Для побудови ймовірнісного простору в R можна використовувати функцію **probspace()**. Загальний синтаксис **probspace(x, probs)**, де x – простір елементарних подій, а probs – вектор (тієї ж довжини, що і x). Конкретний вибір probs залежить від контексту завдання.

Розглянемо приклад побудови ймовірнісного простору у випадку, коли простір елементарних подій складається з n елементів, і ймовірність настання кожної з них є однакова – 1/n.

Візьмемо, для прикладу n=6, тобто розглянемо експеримент з підкиданням шестигранного кубика. Побудуємо простір елементарних подій, згенеруємо вектор ймовірностей і об'єднаємо їх разом за допомогою функції **probspace()**.

```
> outcomes <- rolldie(1)
> outcomes

  X1
1  1
```

```
2   2
3   3
4   4
5   5
6   6
```

```
> p <- rep(1/6, times = 6)
> p
[1] 0.1666667 0.1666667 0.1666667 0.1666667 0.1666667 0.1666667
```

```
> probspace(outcomes, probs = p)
  X1     probs
1  1 0.1666667
2  2 0.1666667
3  3 0.1666667
4  4 0.1666667
5  5 0.1666667
6  6 0.1666667
```

Цього самого результату ми могли б досягнути наступним чином:

```
> probspace(1:6, probs = p)
  x     probs
1 1 0.1666667
2 2 0.1666667
3 3 0.1666667
4 4 0.1666667
5 5 0.1666667
6 6 0.1666667
```

або (для рівноможливих результатів експерименту)

```
> probspace(1:6)
  x     probs
1 1 0.1666667
2 2 0.1666667
3 3 0.1666667
4 4 0.1666667
5 5 0.1666667
6 6 0.1666667
```

І, нарешті, функція **rolldie()** має додатковий логічний аргумент **makespace**, який додасть стовпець probs з однаковими ймовірностями до згенерованого простору елементарних подій:

```
> rolldie(1, makespace = TRUE)
  X1     probs
1  1 0.1666667
2  2 0.1666667
3  3 0.1666667
4  4 0.1666667
5  5 0.1666667
6  6 0.1666667
```

Функції **tosscoin()** та **cards()** мають аналогічний аргумент.

Приклад 2. Розглянемо випадок експерименту з підкиданням монети, що не є ідеально збалансованою і ймовірність появи сторони «H» становить 0,7 замість 0,5. Ймовірнісний простір можна побудувати наступним чином:

```
> probspace(tosscoin(1), probs = c(0.7, 0.3))
  toss1 probs
1     H   0.7
2     T   0.3
```

Ймовірність в пакеті **“prob”** обчислюється за допомогої функції **Prob()**.

Приклад 3. Нехай виймається карта зі стандартної колоди гральних карт. Позначимо ймовірнісний простір, пов’язаний з експериментом через S, і нехай підмножини A і B визначаються так:

```
> S <- cards(makespace = TRUE)
> A <- subset(S, suit == "Heart")
> B <- subset(S, rank %in% 7:9)
```

Тоді ймовірності подій A і B обчислюються наступним чином:

```
> Prob(A)
[1] 0.25

> Prob(B)
[1] 0.2307692
```

Приклад 4. З урни, в якій 7 червоних і 3 зелені кульки виймають навмання 3. Знайти ймовірність того, що:

1) всі три кульки червоні;
2) дві червоні, одна зелена;
3) з урахуванням порядку (червона, зелена, червона).

R Скрипт буде мати наступний вигляд:

```
library(prob)

L <- rep(c("red", "green"), times = c(7, 3))

#  Простір елементарних подій
M <- urnsamples(L, size = 3, replace = FALSE, ordered = TRUE)

# Ймовірнісний простір
N <- probspace(M)

# Обчислення ймовірностей
p1 <- Prob(N, isrep(N, "red", 3))
cat("Три кульки червоні:", p1, "\n")

p2 <- Prob(N, isrep(N, "red", 2))
cat("Дві червоні, одна зелена:", p2, "\n")

p3 <- Prob(N, isin(N, c("red","green", "red"), ordered=TRUE))
cat("Червона -> зелена -> червона:", p3, "\n")

```

Результат виконання:

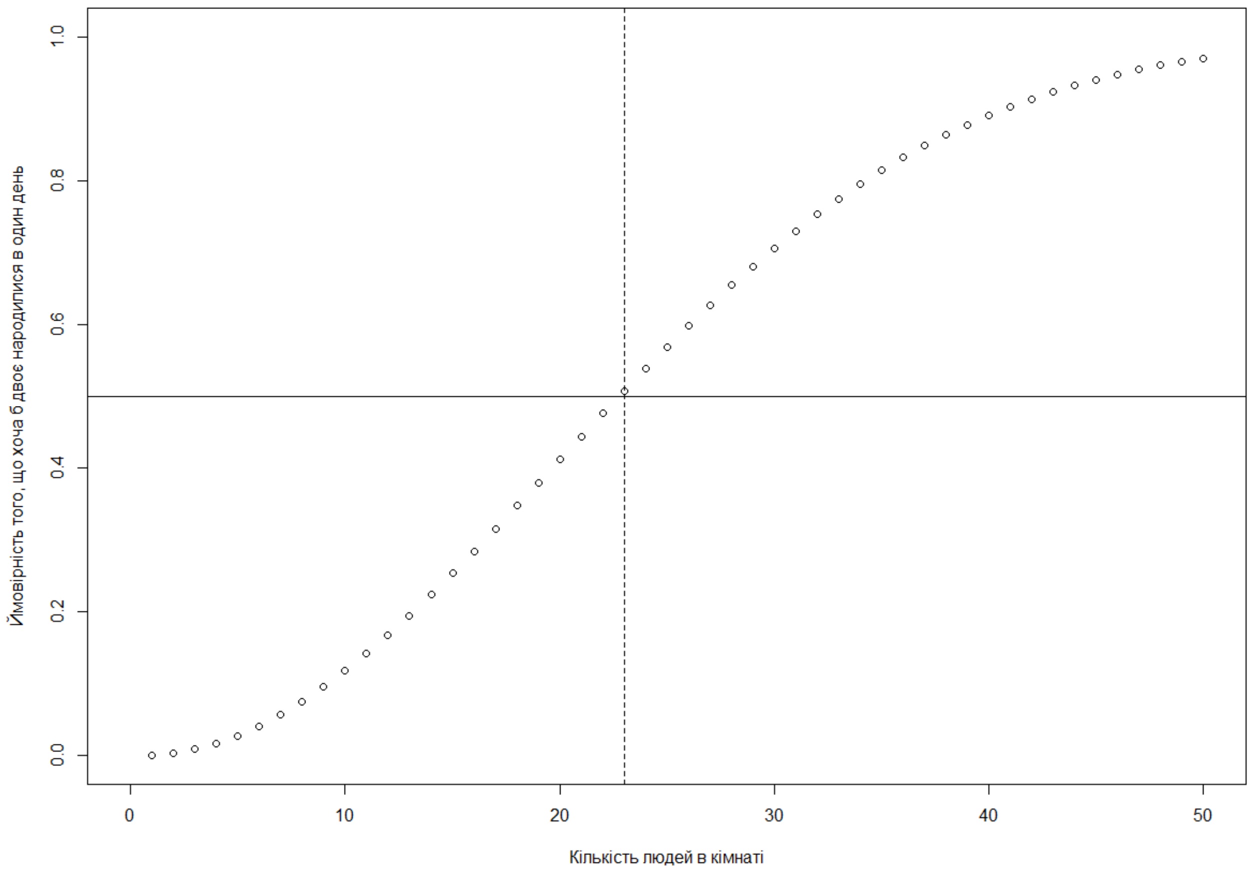
```
Три кульки червоні: 0.2916667
Дві червоні, одна зелена: 0.525
Червона -> зелена -> червона: 0.175
```

Приклад 5. В кімнаті знаходяться **n** людей. Яка ймовірність того, що хоча б двоє з них народилися в один день?

Розрахунок в R для **n** від 1 до 50:

```r
maxValue <- 50
plot(1, type="n",
     xlab="Кількість людей в кімнаті",
     ylab = "Ймовірність того, що хоча б двоє народилися в один день",
     xlim=c(0, maxValue), ylim=c(0, 1))
p <- NULL
for (count in 1:maxValue){
  p[count] <- 1-prod((365-count+1):365)/365^count
  points(count,p[count])
}
abline(h = 0.5)
abline(v = 23, lty = 2)

```



Індивідульні завдання

Всі індивідуальні завдання повинні бути виконані на аркуші паперу з використанням відповідних формул комбінаторики та теорії ймовірностей. Крім того, завдання 1 та 2 мають бути розв'язані з використанням функцій пакету "prob" в R.

Завдання 1

На окремих картках написано по одній літері (див. свій варіант). Навмання виймають **n** літер і розкладаються зліва направо. Яка ймовірність того, що отримається задане слово?

Варіант	**Картки з літерами та їх кількість**	**n**	**Слово**
1	К – 2; О – 4; Р – 2; П – 2	5	КОРОП
2	О – 6; Л – 1; В – 2	5	ОЛОВО
3	А – 5; Н – 2; С – 3	6	АНАНАС
4	К – 4; О – 3; У – 2	3	ОКО
5	Р – 2; О – 1; А – 4; П – 3	4	ПАРА
6	Г – 2; О – 3; Л – 3; С – 2	5	ГОЛОС
7	К – 3; О – 3; С – 3; М – 1	6	КОСМОС
8	К – 3; А – 3; Р – 2; Т – 2	5	КАРТА

Завдання 2

Двічі кидають шестигранний гральний кубик. Яка ймовірність того, що сума очок 1) дорівнюватиме 2) буде меншою 3) буде більшою, ніж **L + 2** ?

(**L** – номер варіанту)

Завдання 3

Варіант 1. Ймовірність своєчасної сплати податків для першого підприємства дорівнює 0,8, для другого – 0,6, для третього – 0,7. Знайдіть імовірність своєчасної сплати податків:

а) не більше ніж одним підприємством;

б) принаймні двома підприємствами.

Варіант 2. Аеропорт протягом дня виконує 3 рейси. Імовірність затримки першого рейсу за метеоумовами дорівнює 0,05, другого – 0,1, третього 0,15. Знайдіть імовірність того, що:

а) лише 2 рейси будуть відправлені з затримкою;

б) всі рейси будуть відправлені своєчасно.

Варіант 3. На кожному з трьох станків виготовлено по одній деталі. Імовірність одержання браку на першому станку дорівнює 0,05, на другому – 0,07, на третьому – 0,1. Знайдіть імовірність того, що виготовлена:

а) лише одна бракована деталь;

б) принаймні одна бракована деталь.

Варіант 4. Для повідомлення про пожежу у готелі встановлено 2 незалежно працюючі сигналізатори. Ймовірність того, що при пожежі спрацює перший сигналізатор, дорівнює 0,95; другий – 0,9. Знайдіть ймовірність того, що при пожежі надійде сигнал:

а) хоча б від одного сигналізатора;

б) тільки від одного сигналізатора.

Варіант 5. Екзаменаційний білет містить три питання. Ймовірності того, що студент дасть відповідь на перше і друге питання, однакові та дорівнюють 0,8; на третє – 0,9. Знайдіть ймовірність того, що студент складе іспит, якщо для цього необхідно відповісти:

а) на всі питання;

б) принаймні на два питання.

Варіант 6. На станції спостереження встановлені 3 радіолокатори різних конструкцій, які виявляють об’єкт незалежно один від одного. Імовірність виявлення об’єкта першим локатором дорівнює 0,86, другим – 0,9, третім – 0,95. Знайдіть імовірність виявлення об’єкта:

а) тільки одним локатором;

б) принаймні одним локатором.

Варіант 7. Три контролери незалежно один від одного оцінюють якість виробу. Імовірність того, що виріб буде прийнятий першим контролером становить 0,95, другим – 0,9, третім – 0,98. Знайдіть імовірність того, що виріб буде прийнятий:

а) принаймні одним контролером;

б) всіма контролерами.

Варіант 8. Радіостанція аеропорту надсилає 3 повідомлення для екіпажу літака. Імовірність прийому першого повідомлення дорівнює 0,6, другого – 0,65, третього – 0,7. Знайдіть імовірність того, що екіпаж прийме:

а) тільки два повідомлення;

б) всі три повідомлення.