Dr Isabella Gollini
Jose Camarena Brenes

Probability and Statistics

In this R lab you will:

- Perform set operations with R
- Use the function table for cross-tabulation
- Visualise the sample space

1 Set operations

We use the example presented in Section 2.1 of the lecture notes; the experiment consists in throwing a fair six-sided die, and observing the outcome.

The sample space is:

```r
S <- c(1, 2, 3, 4, 5, 6)
S
```

The event we are interested in is: “an odd number occurs” (A).

In order to find the odd numbers in R we have to use the modulo operation (%%):

```r
isA <- S %% 2 != 0
isA
```

isA is a vector of the same length of S, that identifies whether the corresponding element of S is odd.

The function subset finds a subset of S for which the condition isA holds:

```r
A <- subset(S, isA)
A
```

Complement The complement of A (A^c) is the event “A does not occur”.

```r
Ac <- subset(S, !isA)
Ac
```

The exclamation point (!) indicates the logical negation of the condition.

We introduce the new event B, “a score of 5 or less occurs”:

```
B <- 1:5
B
```

and the event C, “a score of 4 or 6 occurs”:

```
C <- c(4, 6)
C
```

Inclusion A is included in B if all the outcomes of A are also outcomes of B.

The function `%in%` allows to find out which elements of the first objects (in this case `A`) are elements of the second object (`B`). [In order to access the help files you can type `?"%in%"`]

```
AinB <- A %in% B
AinB
```

`AinB` is a vector of the same length of `A`, that identifies whether the corresponding element of `A` is an element of `B`.

The function `all` is needed to check if the condition is satisfied for all the elements of `A`.

```
all(AinB)
```

if `TRUE` A is included in B, since all the elements of A are elements of B.

Try

```
all(B %in% A)
```

does it return what you expect?

Union The union of events A and B, $A \cup B$, is the event that either A or B, or both occur

```
AuB <- union(A, B)
AuB
```

If we want to sort the values, we can use:

```
AuB <- sort(AuB)
AuB
```

Intersection The intersection of events A and B, $A \cap B$, is the event that both A and B occur

```
AiB <- intersect(A, B)
AiB
```

Mutual exclusion Two events A and C are mutually exclusives if they have no outcomes in common: $A \cap C = \emptyset$

```
AiC <- intersect(A, C)
AiC
```

The value `numeric(0)` represents the empty set.

2 Probability Tables

2.1 Example: College Loans

In a telephone survey of $1,000$ adults, respondents were asked about the expenses of a college education and the relative necessity of some form of financial assistance. The respondents were classified according to whether they currently had a child in college and whether they thought the loan burden for most college students is too high, the right amount, or too little. The number responding in each category is shown in the following table.

	Too High (A)	Right Amount (B)	Too Little (C)
Child in College (D)	0.35	0.08	0.01
No Child in College (E)	0.25	0.20	0.11

Suppose one respondent is chosen at random from this group.

1. What is the probability that the respondent has a child in college?
2. What is the probability that the respondent does not have a child in college?
3. What is the probability that the respondent has a child in college or thinks that the loan burden is too high or both?

Before start answering the questions we have to type the table into `R`:

```
college <- matrix(nrow = 2, ncol = 3)
colnames(college) <- c("A", "B", "C")
rownames(college) <- c("D", "E")
college
```

Now we have an empty 2×3 table, with the columns named A, B, C, and the rows named D, E.

We then fill in the values starting from the first row, and then the second row.

```
college[1,] <- c(.35, .08, .01)
college[2,] <- c(.25, .20, .11)
college
```

The table gives the probabilities for all the six possible outcomes. For this reason, it is good practice to check if it satisfies Axiom 2 of Probability (the sum of the probabilities of all the outcomes in the entire sample space must be equal to 1).

```
sum(college) == 1
```

If this equality is not satisfied check again the values you typed in!

The entry in the top-left corner of the table gives the probability that a respondent has a child in college (event D) *and* thinks the loan burden is too high (event A): $\Pr(D \cap A)$.
The second entry in the first row of the table gives the probability that a respondent has a child in college (event D) *and* thinks the loan burden is the right amount (event B): $\Pr(D \cap B)$.
and so on . . . .

1. The event that the respondent has a child in college (event D) is formed of the three outcomes present in the first row of the table. Since the outcomes are mutually exclusives:

$$\Pr(D) = \Pr(D \cap A) + \Pr(D \cap B) + \Pr(D \cap C)$$

that is the sum of the elements in the first row of the table.

It is good practice to give your objects meaningful names. That way you won't forget what they contain!

For this reason we call `PrD` the probability of D ($\Pr(D)$). Remember that `R` is case sensitive, and you cannot use spaces or special characters for object names.

```
PrD <- sum(college["D",])
```

notice the comma after `"D"`. This is because `D` is stored in a row of the table.

Type `PrD` to get $\Pr(D)$.

2. The event that the respondent does not have a child in college (event E) is formed of the three outcomes present in the second row of the table. Since the outcomes are mutually exclusives:

$$\Pr(E) = \Pr(E \cap A) + \Pr(E \cap B) + \Pr(E \cap C)$$

```r
PrE <- sum(college["E",])
PrE
```

The event that the respondent does not have a child in college is the complement of the event D denoted by D^c ($E = D^c$). So, we can also get it by using Lemma 1:

$$\Pr(D^c) = 1 - \Pr(D)$$

```r
1 - PrD
```

3. The event that the respondent has a child in college or thinks that the loan burden is too high or both corresponds to $\Pr(A \cup D)$. Using the Addition Rule:

$$\Pr(A \cup D) = \Pr(A) + \Pr(D) - \Pr(A \cap D)$$

Firstly we calculate the probability that he or she ranks the loan burden is too high.

```r
PrA <- sum(college[,"A"])
```

Notice the comma before `"A"`. This is because `A` is stored in a column of the table.

Then we calculate $\Pr(A \cap D)$. [I use `i` to indicate the intersection. In `R` you cannot use the symbol $\cap$ in a variable name].

```r
PrAiD <- college["D","A"]
PrAiD
```

Finally we can calculate $\Pr(A \cup D)$. [I use the letter `u` to indicate the union].

```r
PrAuD <- PrA + PrD - PrAiD
PrAuD
```

2.2 Example: Diamonds cut and color

We use again the dataset `diamonds` included in the `ggplot2` package, but this time we focus on the variables `cut` and `color`:

- The variable `cut` represents the quality of the cut: Fair, Good, Very Good, Premium, Ideal.
- The variable `color` represents the quality of the diamond color, from J (worst) to D (best)

```r
library(ggplot2)
data(diamonds)
attach(diamonds)
```

We want to create the probability table that summarises the information about the diamond quality given by the variables `cut` and `color`:

```r
quality <- table(cut, color) / nrow(diamonds)
quality
```

If we want to visualise the table rounded at the third digit after the decimal point:

```r
round(quality, 3)
```

We want to find the probability that a randomly selected diamond is of top quality.

To be of top quality the diamond must be simultaneously of Ideal quality of cut and have the best quality of color (color `D`).

```r
quality["Ideal", "D"]
```

What is the probability that a randomly selected diamond is of the lowest quality?

Write here your answer

What is the probability that a randomly selected diamond has quality color `F`?

Write here your answer

What is the probability that a randomly selected diamond has a "Good" quality of the cut?

Write here your answer

What is the probability that a randomly selected diamond has a "Premium" quality of the cut, or quality `G` of the color, or both?

Write here your answer

3 Cards

We go through the example of Section 2.3 of the lecture notes; consider drawing a card at random from a standard pack of 52 cards.
What is the probability that a spade or a king is drawn?

We create a vector `values` that contains all the values a card can take:

```r
values <- c("A", 2:10, "J", "Q", "K")
values
```

We create the vector `suits` that contains all the card suits:

```r
suits <- c("diamonds", "clubs", "spades", "hearts")
suits
```

The entire sample space can be created by using the function `expand.grid` that allows us to find all the possible combinations of values and suits.

```r
cards <- expand.grid(values = values, suits = suits)
cards
```

The number of rows in `cards` corresponds to $|S|$:

```r
nS <- nrow(cards)
nS
```

Let A be the event that a card of spades is drawn.

```r
isA <- cards[,"suits"] == "spades"
A <- subset(cards, isA)
A
```

Let `nA` be the number of outcomes in A ($|A|$).

```r
nA <- nrow(A)
nA
```

$\Pr(A)$ is:

```r
PrA <- nA / nS
PrA
```

Let B be the event that a king is drawn.

```r
isB <- cards[,"values"] == "K"
B <- subset(cards, isB)
B

nB <- nrow(B)
nB

PrB <- nB / nS
PrB
```

Now we have to find $A \cap B$, and $\Pr(A \cap B)$.

```r
isAandB <- isA & isB
AiB <- subset(cards, isAandB)
AiB

nAiB <- nrow(AiB)
nAiB

PrAiB <- nAiB / nS
PrAiB
```

The event that a spades or a king is drawn is $A \cup B$. Using the result of Theorem 4,

$$\Pr(A \cup B) = \Pr(A) + \Pr(B) - \Pr(A \cap B)$$

```r
PrAuB <- PrA + PrB - PrAiB
PrAuB
```

What is the probability that a face card is drawn? [A face card is a "J", "Q", or "K"]

Write here your answer

What is the probability that a red face card is drawn? [Hearts and Diamonds are red suits]

Write here your answer

4 Combinations

4.1 Football Kits

A group of friends decided to form a new football team. They did not agree on the colors for their jerseys, so they decides to go on a website which sells football kits, and randomly selects one shirt, one pair of shorts, and one pair of socks.

The possible options on the website are:

- Shirts of 5 different colors: red, green, blue, white, yellow.
- Shorts of 3 different colors: black, white, blue.
- Socks of 2 different colors: white, red.

We want to find out:

a. How many different outfits can be created?

b. What's the probability of wearing a red shirt?

c. What's the probability of a single colored outfit?

d. What's the probability of wearing the Arsenal F.C. traditional colors (red and white only)? [The order does not count]

```r
shirts <- c("red", "green", "blue", "white", "yellow")
shorts <- c("black", "white", "blue")
socks <- c("white", "red")

outfits <- expand.grid(shirts = shirts,
                       shorts = shorts,
                       socks = socks)
outfits
```

Hint to answer question (d.): The symbol & means that the conditions hold simultaneously.

```r
isC <- (outfits[,"shirts"] %in% c("red", "white")) &
  (outfits[,"shorts"] %in% c("red", "white")) &
  (outfits[,"socks"] %in% c("red", "white"))
```