Course of Study Bachelor Computer Science	Exercises Statistics WS 2021/22
Sheet VI - Solutions	

Probability Spaces and Basic Rules

1. Consider a random experiment of tossing two dice. Let A denote the event that the first die score is 1 and B the event that the sum of the scores is 7.

   (a) Give the sample space Ω and find $|\Omega|$.

   (b) Explicitly list the elements of the following events:

   $$A, B, A \cup B, A \cap B, A^c \cap B^c$$

   Answer:

   (a) $\Omega = \{(i,j)|i,j \in \{1,2,3,4,5,6\}\}$, $|\Omega| = 36$

   (b) $A = \{(1,1,), (1,2), (1,3), (1,4), (1,5), (1,6)\}$
   $B = \{(1,6,), (2,5), (3,4), (4,3), (5,2), (6,1)\}$
   $A \cap B = \{(1,6)\}$
   $A \cup B = \{(1,1,), (1,2), (1,3), (1,4), (1,5), (1,6), (2,5), (3,4), (4,3), (5,2), (6,1)\}$
   $A^c \cap B^c = \Omega \setminus (A \cup B)$

2. Suppose that A and B are events in an experiment with $P(A) = 1/3$, $P(B) = 1/4$, $P(A \cap B) = 1/10$. Express each of the following events verbally and find its probability:

   $$A \setminus B, A \cup B, A^c \cup B^c, A^c \cap B^c, A \cup B^c$$

   Answer:

   (a) $A \setminus B$: The event A but not B occures.
   $P(A \setminus B) = P(A \cap B^C) = P(A) - P(A \cap B) = \frac{1}{3} - \frac{1}{10} = \frac{7}{30}$

   (b) $A \cup B$: One or both of the events A and B occure.
   $P(A \cup B) = P(A) + P(B) - P(A \cap B) = \frac{1}{3} + \frac{1}{4} - \frac{1}{10} = \frac{29}{60}$

(c) $A^c \cup B^c$: Both events do not occure.
$P(A^c \cup B^c) = P(A^C \cup B^C) = P((A \cap B)^C) = 1 - P(A \cap B) = 1 - \frac{1}{10} = \frac{9}{10}$

(d) $A^c \cap B^c$: None of the events A and B occure.
$P(A^c \cap B^c) = P(A^C \cap B^C) = P((A \cup B)^C) = 1 - P(A \cup B) = 1 - \frac{29}{60} = \frac{31}{60}$

(e) $A \cup B^c$: only B does not occure ($A \cup B^c = (B \setminus A)^c$).
$P(A \cup B^c) = 1 - (P(B) - P(A \cap B)) = 1 - \frac{1}{4} + \frac{1}{10} = \frac{17}{20}$

3. Suppose that A, B, and C are events in an experiment with $P(A) = 0.3$, $P(B) = 0.2$, $P(C) = 0.4$, $P(A \cap B) = 0.04$, $P(A \cap C) = 0.1$, $P(B \cap C) = 0.1$, $P(A \cap B \cap C) = 0.01$. Express each of the following events in set notation and find its probability:

(a) At least one of the three events occurs.

(b) None of the three events occurs.

(c) Exactly one of the three events occurs.

(d) Exactly two of the three events occur.

Answer:

(a) $P(A \cup B \cup C) = P(A) + P(B) + P(C) - P(A \cap B) - P(A \cap C) - P(B \cap C) + P(A \cap B \cap C) = 0.3 + 0.2 + 0.4 - 0.04 - 0.1 - 0.1 + 0.01 = 0.67$

(b) $P((A \cup B \cup C)^c) = 1 - P(A \cup B \cup C) = 1 - 0.67 = 0.33$

(c) $P((A \cup B \cup C) \setminus ((A \cap B) \cup (A \cap C) \cup (B \cap C))) = P(A \cup B \cup C) - P(A \cap C) - P(A \cap B) - P(B \cap C) + 2P(A \cap B \cap C) = 0.67 - 0.1 - 0.04 - 0.1 + 2 \cdot 0.01 = 0.45$

(d) $P((A \cap B) \cup (A \cap C) \cup (B \cap C) \setminus (A \cap B \cap C)) = 0.04 + 0.1 + 0.1 - 3 \cdot 0.01 = 0.21$

4. **Law of Large Numbers**
The Law of Large Numbers says that the average (mean) approaches what it's estimating. As we flip a fair coin over and over, it average eventually converges to the true probability of a head (.5).

(a) To see this in action, create a R function coinPlot, which takes an integer n which is the number of coin tosses that will be simulated. As coinPlot does these coin flips it computes the cumulative sum (assuming heads are 1 and tails 0), but after each toss it divides the cumulative sum by the number of flips performed so far. It

then plots this value for each of the k=1...n tosses.
Hint: Use the function sample() to simulate tossing coins.

(b) Call coinPlot several times for n=10, 100 and 1000 and describe what you see.

Answer:

(a)
```
#######################################################
# Law of Large Numbers: Simulation of tossing coins
#
# File: tossing_coins.R
#
#######################################################

library(tidyverse)

# function coinPlot
# head = 1, tail = 0, n = number of tosses
coinPlot <- function(n) {
  # create a tibble containing the result of n tosses
  tosses <- tibble(
    # number of flip
    no = 1:n,
    # result
    coin = sample(x=c(0,1), size = n, replace = TRUE),
    # average
    avg.no = cumsum(coin)/no)
  # create a diagram
  ggplot(data = tosses) +
    # scatterplot of the points (no, avg.no)
    geom_point(mapping = aes(x=no,y=avg.no)) +
    # connecting the points
    geom_line(mapping = aes(x=no,y=avg.no)) +
    # horizontal line: probability of a head
    geom_hline(yintercept = 0.5) +
    # setting the bounds for y
    ylim(0,1) +
    # title and labels of the axis
    ggtitle(paste(n," tosses of a fair coin"),
            subtitle = "average number of head") +
    xlab("n") +
    ylab("mean") +
    # theme
    theme_classic()
}

# repeated calls of conPlot
coinPlot(10)
coinPlot(100)
coinPlot(1000)
```

(b) The output depends on R's random number generator, the plot differs from call to call and probably jumps around a bit. If you did this several times, your plots would vary quite a bit. If the number of flips is increasing the line approaches its asymptote of .5. This is the probability you expect since what we're plotting, the cumulative sum/number of flips, represents the probability of the coin landing on heads. As we know, this is 0.5.

5. **Urn Models**
A large number of discrete probability spaces can be traced back to so-called urn models. An urn contains n balls, which do not all have to be different. From these urns r balls are drawn with or without replacement. For the result of the drawing, the order or only the quantity of

the drawn balls can be of importance.
Here an urn with 10 balls is considered. 5 of them are red, 3 balls are blue and 2 balls are green. 3 balls are drawn. The following 4 cases should be distinguished:

I Drawing with replacement with respect to the order

II Drawing with replacement without observing the order

III Drawing without replacement with respect to the order

IV Drawing without replacement without observing the order

Solve the following tasks.

(a) Load the library gtools and inspect the commands combinations() and permutations(). Consider the bags $b_1 = \{a, b, c\}$ and $b_2 = \{a, a, b, c\}$ = and list all combinations and all permutations of order 2 if duplicated elements in the outpot are allowed or not allowed.

(b) Use the function sample() to determine the result of 10 random draws.

(c) Determine a suitable event space Ω and its size to describe the random experiment.
Note that depending on whether the order of the drawn balls is important or not, the result of a drawing is considered as a r-variation or as a r-combination of the n set of balls. With the help of the permutations() and combinations() functions of the R-package gtools, the corresponding r-variations or r-combinations can be determined.

(d) Determine the probabilities of all elementary events in Ω using a Laplace model, i.e. as a determination of the ratio of the number of favorable cases by the number of all cases. The probabilities are first determined by counting methods and then by using the R function permutations().
Hint: To determine the probabilities with R, assume that the n balls are numbered consecutively, i.e. they are distinguishable, and that the order is first observed in a drawing. Every r-variation of the numbers 1 to n is equally probable. Determine the set of all these drawings with permutations(). Then map each such drawing to the corresponding elementary event. By dividing the number of drawings belonging to an elementary event and the number of all drawings, you then obtain the corresponding probabilities.

Answer:

(a) The R function combinations() enumerates the possible combinations of a specified size from the elements of a vector and the R function permutations() enumerates the possible permutations. In both functions you decide via logical flags whether duplicates should be removed from the source and whether the output may include duplicated values.

(b) Samples of 10 random draws out of the bag $= \{r, r, r, r, r, b, b, b, g, g\}$ are

- without replacement: r, r, r, r, g, b, g, r, b, b
  Mention that the bag is empty after the 10 draws.
- with replacement: b, r, b, g, b, g, r, b, b, b
  Mention that in the sample can occure for example more blue balls than there are in the bag.

(c) Set of elementary events and their probabilities

i. Drawing with replacement with respect to the order

- $|\Omega| = 3^3 = 27$ number of 3-variations of a 3-set with possible repititions
- Every elementary event can be seen as a word of length 3 with 3 possible letters. The possible letters are given by the colours r, b, g of the balls in the urn. To calculate the probability of an elementary event we count how many r, b and g are in the word. If R, B resp. G are the numbers of the red, blue resp. green drawn balls, we get

$$P(R = i, B = j, G = k) = \frac{5^i \cdot 3^j \cdot 2^k}{10^3}$$

with $i + j + k = 3; i, j, k \geq 0$

	event	prob
1	b b b	0.027
2	b b g	0.018
3	b b r	0.045
4	b g b	0.018
5	b g g	0.012
6	b g r	0.030
7	b r b	0.045
8	b r g	0.030
9	b r r	0.075
10	g b b	0.018
11	g b g	0.012
12	g b r	0.030
13	g g b	0.012
14	g g g	0.008
15	g g r	0.020
16	g r b	0.030
17	g r g	0.020
18	g r r	0.050
19	r b b	0.045
20	r b g	0.030
21	r b r	0.075
22	r g b	0.030
23	r g g	0.020
24	r g r	0.050
25	r r b	0.075
26	r r g	0.050
27	r r r	0.125

You can use the R command permutations() to get the elementary events and their probabilities by counting all equally like 3 permutations of numbers 1 to 10. The balls are numbererd from 1 to 10 and the balls 1 to 5 are red, 6 to 8 are blue and 9, 10 are green.

ii. Drawing with replacement without respect to the order

- $|\Omega| = \binom{3+3-1}{3} = \binom{5}{3} = 10$ number of 3-multisets from a 3-set
- Every elementary event can be seen as a 3-multiset of the letters r, b, g. Such a 3-multiset is given by a set of words of length 3 with R r's, B b' and G g's. All words of length 3 with the same numbers of r, b and g are equally like. For example

$$\begin{aligned} P(\{b,b,g\}) &= P(\{R=0,B=2,G=1\}) \\ &= P(\{(b,b,g),(b,g,b),(g,b,b)\}) \\ &= P((b,b,g)) + P((b,g,b)) + P((g,b,b)) \\ &= \frac{3!}{0!2!1!} \cdot P((b,b,g)) \\ &= \frac{3!}{0!2!1!} \cdot \frac{5^0 \cdot 3^2 \cdot 2^1}{10^3} = 0.054 \end{aligned}$$

In general we get

$$P(\{R=i,B=j,G=k\}) = \frac{5^i \cdot 3^j \cdot 2^k}{10^3} \cdot \frac{3!}{i! \cdot j! \cdot k!}$$

with $i+j+k=3; i,j,k \geq 0$

	event	prob
1	b b b	0.027
2	b b g	0.054
3	b b r	0.135
4	b g g	0.036
5	b g r	0.180
6	b r r	0.225
7	g g g	0.008
8	g g r	0.060
9	g r r	0.150
10	r r r	0.125

As in (i) you can generate by applying the R command permutations() all possible ordered sampling results. Every ordered sampling corresponds to an unordered sampling result. Counting all rows with the same unordered sampling result and dividing by total number of ordered sampling results you get the probabilities of the elementary events.

iii. Drawing without replacement with respect to the order

- We must count the number of 3-variations with repititions and subtract the number of impossible events (here: 3 green balls)
$$|\Omega| = 3^3 - 1$$
- Regarding that now the drawn balls are not replaced we get analogousley to the case of drawing with replacement
$$P(R = i, B = j, G = k) = \frac{5^{\underline{i}} \cdot 3^{\underline{j}} \cdot 2^{\underline{k}}}{10^{\underline{3}}}$$
with $i + j + k = 3; i, j, k \geq 0$. Mention that $n^{\underline{m}} = n \cdot ... \cdot (n - m + 1)$ is the falling factorial and give the number of m-variations from a set of n distint elements.

	event	prob
1	b b b	0.008
2	b b g	0.017
3	b b r	0.042
4	b g b	0.017
5	b g g	0.008
6	b g r	0.042
7	b r b	0.042
8	b r g	0.042
9	b r r	0.083
10	g b b	0.017
11	g b g	0.008
12	g b r	0.042
13	g g b	0.008
14	g g r	0.014
15	g r b	0.042
16	g r g	0.014
17	g r r	0.056
18	r b b	0.042
19	r b g	0.042
20	r b r	0.083
21	r g b	0.042
22	r g g	0.014
23	r g r	0.056
24	r r b	0.083
25	r r g	0.056
26	r r r	0.083

As in the case drawing with replacement with respect to the order you can apply R to get the probabilities. But set the flag

repeats.allowed to FALSE in the R command permutations() to realise drawing without replacement.

iv. Drawing without replacement without respect to the order

- We must count the number of 2-multisets of a 3-set and subtract the number of impossible events (here: 3 green balls)

$$|\Omega| = \binom{3+3-1}{3} - 1 = 9$$

- Regarding that now the drawn balls are not replaced we get analogousley to case drawing with replacement

$$P(\{R=i, B=j, G=k\}) = \frac{\binom{5}{i} \cdot \binom{3}{j} \cdot \binom{2}{k}}{\binom{10}{3}}$$

with $i+j+k=3; i,j,k \geq 0$

	event	prob
1	b b b	0.008
2	b b g	0.050
3	b b r	0.125
4	b g g	0.025
5	b g r	0.250
6	b r r	0.250
7	g g r	0.042
8	g r r	0.167
9	r r r	0.083

As in the case drawing with replacement without respect to the order you can apply R to get the probabilities. But set the flag repeats.allowed to FALSE in the R command permutations() to realise drawing without replacement.

```
############################################################
# Urn Models
# Elementary events and their probabilities: Evaluation
# and determination applying the R commands combinations()
# and permutations() in the gtools package.
#
# file: prob_basics_urn_models.R
############################################################

library(gtools)
library(tidyverse)

# combinations enumerates the possible combinations of a specified
# size from the elements of a vector. permutations enumerates the
# possible permutations.
# Usage
# combinations(n, r, v=1:n, set=TRUE, repeats.allowed=FALSE)
# permutations(n, r, v=1:n, set=TRUE, repeats.allowed=FALSE)
# Arguments
# n size of the source vector
# r      size of the target vectors
# v source vector
# set    = logical flag indicating whether duplicates should be removed
#        from the source vector v.
# repeats.allowed = logical flag indicating whether the constructed
#                   vectors may include duplicated values.
#
# Value: Returns a matrix where each row contains a vector of length r.

# Examples Consider the bags
```

```
b1 <-  c("a","b","c")
b2 <- c("a","a","b","c")
# and list all combinations and all permutations of order 2 if duplicated
# elements in the outpot are allowd or not allowed.
combinations(n=3,r=2,v=b1)
combinations(n=3,r=2,v=b1,repeats.allowed = TRUE)

permutations(n=3,r=2,v=b1)
permutations(n=3,r=2,v=b1,repeats.allowed = TRUE)

# n = number of distinct elements of source vector, if set = TRUE!!
combinations(n=3,r=2,v=b2, set = TRUE)
combinations(n=4,r=2,v=b2, set = FALSE)
combinations(n=4,r=2,v=b2, set = FALSE, repeats.allowed = TRUE)

##############################################################################
# Bag with 10 balls: 5*red, 3*blue und 2*green
# 3 balls are drawn
bag <- rep(c("r","b","g"),c(5,3,2))
bag

##############################################################################
# Use the function sample() to determine the result of 10 random draws.

sample(x=bag, size=10) # without replacement
sample(x=bag, size=10, replace=TRUE) # with replacement

##############################################################################
# Drawing with replacement with respect to the order
##############################################################################
# Elementary events and their probabilities directly calculated
permutations(n=3, r=3, v=bag, set = TRUE, repeats.allowed = TRUE) %>%
  as_tibble(.name_repair = "universal") %>%
  # Treatment of problematic column names
  # .name_repair        = "universal": Make the names unique and syntactic
  mutate(
    V1 = ...1, V2 = ...2, V3 = ...3,
    prob = case_when(
      V1 == "r" ~ 0.5,
      V1 == "b" ~ 0.3,
      V1 == "g" ~ 0.2) *
      case_when(
        V2 == "r" ~ 0.5,
        V2 == "b" ~ 0.3,
        V2 == "g" ~ 0.2) *
      case_when(
        V3 == "r" ~ 0.5,
        V3 == "b" ~ 0.3,
        V3 == "g" ~ 0.2
      )
  ) %>% select(V1,V2,V3,prob) %>%
  # sort
  arrange(V1,V2,V3) -> M1
M1

# Elementary events and their probabilities determined by counting all
# equally like 3 permutations of numbers 1 to 10 (balls are numbererd
# from 1 to 10)
permutations(n=10, r=3, v=1:10, set = TRUE, repeats.allowed = TRUE) %>%
  as_tibble() %>%
  # zeilenweises Vorgehen
  rowwise() %>%
  mutate(
    # Übersetzung der 3-Permutation von 1 bis 10 in die Farben der
    # gezogenen Kugeln
    event = paste(c(bag[V1],bag[V2],bag[V3]), collapse = " ")) %>%
  # Zählen der Häufigkeiten der gezogenen Farben
  group_by(event) %>%
  summarise(count = n()) %>%
  # W. = Häufigkeit/Gesamtanzahl
  mutate(prob = count/sum(count)) %>%
  # sortieren
  arrange(event) -> M2

cbind(M1,M2)

# generate a latex table
library(xtable)
M2 %>% select(event, prob) %>% xtable(digits = 3)
M1

##############################################################################
```

```
##############################################################################
# Drawing with replacement without respect to the order
##############################################################################
# Elementary events and their probabilities directly calculated
# Multinomial distribution of the number of drawn coloured balls
library(stats)
combinations(n=3, r=3, v=bag, set = TRUE, repeats.allowed = TRUE) %>%
  as_tibble() %>%
  # zeilenweises Vorgehen
  rowwise() %>%
  mutate(
    anz_r = if_else(V1 == "r",1,0) +
      if_else(V2 == "r",1,0) + if_else(V3 == "r",1,0) ,
    anz_b = if_else(V1 == "b",1,0) +
      if_else(V2 == "b",1,0) + if_else(V3 == "b",1,0) ,
    anz_g = if_else(V1 == "g",1,0) +
      if_else(V2 == "g",1,0) + if_else(V3 == "g",1,0) ,
    prob = dmultinom(x = c(anz_r, anz_b, anz_g), size = 3,
                     prob = c(0.5, 0.3, 0.2))
  ) %>%
  select(V1, V2, V3, prob) %>%
  # sortieren
  arrange(V1, V2, V3) -> M2

# Elementary events and their probabilities determined by counting all
# equally like 3 combinations of numbers 1 to 10 (balls are numbererd
# from 1 to 10)
permutations(n=10, r=3, v=1:10, set = FALSE, repeats.allowed = TRUE) %>%
  as_tibble() %>%
  # zeilenweises Vorgehen
  rowwise() %>%
  mutate(
    # pro Zeile sortieren nach den Farben und Umwandlung des sortierten
    # Vektors in einen String
    event = paste(sort(c(bag[V1],bag[V2],bag[V3])),
      collapse = " ")) %>%
  # Zählen der Häufigkeiten der gezogenen Farbkombinationen
  group_by(event) %>%
  summarise(count = n()) %>%
  # W. = Häufigkeit/Gesamtanzahl
  mutate(prob = count/sum(count)) %>%
  # sortieren
  arrange(event) -> M22

# generate a latex table
M22 %>% select(-count) %>% xtable(digits = 3)

##############################################################################

##############################################################################
# Drawing without replacement with respect to the order
##############################################################################
# Elementary events and their probabilities determined by counting all
# equally like 3 permutations (without repetitions) of numbers 1 to 10
# (balls are numbererd from 1 to 10)
permutations(n=10, r=3, v=bag, set = FALSE, repeats.allowed = FALSE) %>%
  as_tibble() %>%
  # zeilenweises Vorgehen
  rowwise() %>%
  mutate(
    # Übersetzung der 3-Permutation von 1 bis 10 in die Farben der
    # gezogenen Kugeln
    event = paste(c(V1,V2,V3), collapse = " ")) %>%
  group_by(event) %>%
  # Zählen der Häufigkeiten der gezogenen Farben
  summarise(count = n()) %>%
  # W. = Häufigkeit/Gesamtanzahl
  mutate(prob = count/sum(count)) %>% arrange(event) -> M3

# generate a latex table
M3 %>% select(-count) %>% xtable(digits = 3)

# some theoretical results
# P((b,r,b)) = P(b) * P(r|b) * P(b|(b,r)) = 3/10 * 5/9 * 2/8
# P((g,b,b)) = P(g) * P(b|g) * P(b|(g,b)) = 2/10 * 3/9 * 2/8
# ...

# Let (x,y,z) be an elementary event. I's probability depends only on the
# number of drawn coloured balls:
# P((x,y,z)) = P(R=i,B=j,G=k) =
#   fallende.Faktorielle(5,i) * fallende.Faktorielle(3,j) * fallende.Faktorielle(2,k) / (10*9*8)
permutations(n=10, r=3, v=bag, set = FALSE, repeats.allowed = FALSE) %>%
  as_tibble() %>%
```

```
  rowwise() %>%
  mutate(
    anz_r = if_else(V1 == "r",1,0) +
      if_else(V2 == "r",1,0) + if_else(V3 == "r",1,0) ,
    anz_b = if_else(V1 == "b",1,0) +
      if_else(V2 == "b",1,0) + if_else(V3 == "b",1,0) ,
    anz_g = if_else(V1 == "g",1,0) +
      if_else(V2 == "g",1,0) + if_else(V3 == "g",1,0)) %>%
  unique() %>%
  mutate(
    prob = (factorial(5)/factorial(5-anz_r)) *
      (factorial(3)/factorial(3-anz_b)) *
      (factorial(2)/factorial(2-anz_g)) / (10*9*8)) %>%
  arrange(V1,V2,V3)
##############################################################################

##############################################################################
# Drawing without replacement without respect to the order
##############################################################################
# Ereignisraum mit theoretischen W.
combinations(n=10, r=3, v=bag, set = FALSE, repeats.allowed = FALSE) %>%
  as_tibble() %>%
#  mutate(
#    V1 = bag[V1],
#    V2 = bag[V2],
#    V3 = bag[V3]) %>%
  unique() %>%
  # Bestimmung der Wahrscheinlichkeiten: zeilenweise werden zunächst die
  # Anzahl der gezogenen Farben bestimmt und anschließend ergeben sich die
  # Wahrscheinlichkeiten durch Anwendung einer verallgemeinert hypergeo-
  # metrischen Verteilung
  rowwise() %>%
  mutate(
    event = paste(sort(c(V1,V2,V3)), collapse = " "),
    anz_r = if_else(V1 == "r",1,0) +
      if_else(V2 == "r",1,0) + if_else(V3 == "r",1,0) ,
    anz_b = if_else(V1 == "b",1,0) +
      if_else(V2 == "b",1,0) + if_else(V3 == "b",1,0) ,
    anz_g = if_else(V1 == "g",1,0) +
      if_else(V2 == "g",1,0) + if_else(V3 == "g",1,0) ,
    prob = choose(5,anz_r)*choose(3,anz_b)*choose(2,anz_g)/choose(10,3)) %>%
  # sortieren
  arrange(event)

# Bestimmung der Elementarereignisse und ihrer W. durch Zählen aller
# gleichwahrscheinlichen 3-Kombinationen (ohne Wiederholungen) der Zahlen
# 1 bis 10 (Kugeln sind von 1 bis nummeriert)
permutations(n=10, r=3, v=bag, set = FALSE, repeats.allowed = FALSE) %>%
  as_tibble() %>%
  # zeilenweise sortieren und zu einem String zusammenfassen
  rowwise() %>%
  mutate(
    event = paste(sort(c(V1,V2,V3)), collapse = " ")) %>%
  # Zählen der Häufigkeiten der unterscheidlichen 3-Kombinationen
  group_by(event) %>%
  summarise(count = n()) %>%
  mutate(prob = count/sum(count)) %>%
  arrange(event) -> M4

# generate a latex table
M4 %>% select(-count) %>% xtable(digits=3)
```

Independence and Conditional Probabilities

1. Suppose that A and B are events in an experiment with $P(A) = 1/3$, $P(B) = 1/4$, $P(A \cap B) = 1/10$. Find each of the following:

$$P(A|B), P(B|A), P(A^c|B), P(B^c|A), P(A^c|B^c)$$

Answer:

(a) $P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{1/10}{1/4} = \frac{4}{10} = 0,4$

(b) $P(B|A) = \frac{P(A \cap B)}{P(A)} = \frac{1/10}{1/3} = \frac{3}{10} = 0,3$

(c) $P(A^C|B) = 1 - P(A|B) = 1 - 0,4 = 0,6$

(d) $P(B^C|A) = 1 - P(B|A) = 1 - 0,3 = 0,7$

(e) $P(A^C|B^C) = \frac{P(A^C \cap B^C)}{P(B^C)} = \frac{1-P(A \cup B)}{1-P(B)} = \frac{1-(P(A)+P(B)-P(A \cap B))}{1-1/4} = \frac{1-\left(\frac{1}{3}+\frac{1}{4}-\frac{1}{10}\right)}{3/4} = \frac{31}{45} \approx 0,689$

2. In a certain population, 30% of the persons smoke and 8% have a certain type of heart disease. Moreover, 12% of the persons who smoke have the disease.

   (a) What percentage of the population smoke and have the disease?

   (b) What percentage of the population with the disease also smoke?

   (c) Are smoking and the disease positively correlated, negatively correlated, or independent?

   Answer: S=Smoker and D=Disease, i.e. $P(S) = 0.30$, $P(D) = 0.08$, und $P(D|S) = 0.12$,

   (a) $P(S \cap D) = P(S) \cdot P(D|S) = 0.30 \cdot 0.12 = 0.036$

   (b) $P(S|D) = \frac{P(S \cap D)}{P(D)} = \frac{0.036}{0.08} = 0.45$

   (c) We compare $P(S) \cdot P(D) = 0,30 \cdot 0,08 = 0,024$ and $P(S \cap D) = 0,036$ and get $P(S \cap D) > P(S) \cdot P(D)$ i.e. dependent.

3. Suppose that a bag contains 12 coins: 5 are fair, 4 are biased with probability of heads 1/3 and 3 are two-headed. A coin is chosen at random from the bag and tossed.

   (a) Find the probability that the coin shows head.

   (b) Given that the coin shows head, find the conditional probability of each coin type.

   Answer: We have $n = 12$ coins. 5 of them are fair coins with $P($ Head $) = P($ Tail $) = 0.50$ (=A-coin). 4 are manipulated with $P($ Head $) = 1/3$ and $P($ Tail $) = 2/3$ (=B-coins). The other 3 are manipulated $P($ Head $) = 1$ and $P($ Tail $) = 0$ (=C-coins).

   (a) $P($ Head $) = P($ Head $|$ A-coin $) \cdot P($ A-coin$) + P($ Head $|$ B-coin $) \cdot P($ B-coin$) + P($ Head $|$ C-coin $) \cdot P($ C-coin$) = \frac{1}{2} \cdot \frac{5}{12} + \frac{1}{3} \cdot \frac{4}{12} + 1 \cdot \frac{3}{12} = \frac{41}{72} \approx 0,56944$

(b) $P(\text{ A-coin} \mid \text{Head }) = \frac{P(\text{ A-coin} \cap \text{Head })}{P(\text{ Head })} = \frac{P(\text{ Head} \mid \text{A-coin }) \cdot P(\text{ A-coin })}{P(\text{ Head })} = \frac{\frac{1}{2} \cdot \frac{5}{12}}{\frac{41}{72}} = \frac{15}{41} \approx 0.3658$ and $P(\text{ B-coin} \mid \text{Head }) = \frac{P(\text{ B-coin} \cap \text{Head })}{P(\text{ Head })} = \frac{P(\text{ Head} \mid \text{B-coin }) \cdot P(\text{ B-coin })}{P(\text{ Head })} = \frac{\frac{1}{3} \cdot \frac{4}{12}}{\frac{41}{72}} = \frac{8}{41} \approx 0.1951$ and $P(\text{ C-coin} \mid \text{Head }) = \frac{P(\text{ C-coin} \cap \text{Head })}{P(\text{ Head })} = \frac{P(\text{ Head} \mid \text{C-coin }) \cdot P(\text{ C-coin })}{P(\text{ Head })} = \frac{1 \cdot \frac{3}{12}}{\frac{41}{72}} = \frac{18}{41} \approx 0.4390$

4. Suppose we know the accuracy rates of the test for both the positive case (positive result when the patient has HIV) and negative case (result when the patient doesn't have HIV). These are referred to as test sensitivity and specificity, respectively. Let "D" be the event that the patient has HIV, and let "+" indicate a positive test result and "-" a negative.

   (a) Describe test sensitivity and specifity by the above notation.

   (b) Suppose a person gets a positive test result. Express the probability that he really has HIV by the above notation.

   (c) Let the disease prevalence be .001, test sensitivity be 99.7% and test specificity be 98.5%. The probability that a person has the disease given his positive test result, i.e. $P(D|+)$. This quantity is called the positive predictive value. Similarly, $P(D^c|-)$, is called the negative predictive value, the probability that a patient does not have the disease given a negative test result. Apply Bayes Rule to evaluate both values.

   (d) The diagnostic likelihood ratio of a positive test, DLR_+, is the ratio of the two positive conditional probabilities, one given the presence of disease and the other given the absence, i.e. $DLR_+ = \frac{P(+|D)}{P(+|D^c)}$. Do you expect DLR_+ to be large or small? Evaluate DLR_+.

   (e) Similarly, the DLR_- is defined as a ratio. Do you expect DLR_- to be large or small? Evaluate DLR_-.

   (f) Show that $\frac{P(D|+)}{P(D^c|+)}$, i.e. the post-test odds of disease given a positive test result equals the pre-test odds of disease, i.e. $\frac{P(D)}{P(D^c)}$ times DLR_+ and evaluate it.

   (g) Similarly we define $\frac{P(D|-)}{P(D^c|-)}$ the post-test odds of disease given a negative test result. Show that it equals the pre-test odds of disease, i.e. $\frac{P(D)}{P(D^c)}$ times DLR_- and evaluate it.

   (h) Are post-test odds greater than pre-test odds or post-test odds are less than pre-test odds?

   Answer:

   (a) sensitivity: $P(+|D)$, specificity: $P(-|D^c)$

   (b) $P(D|+)$

(c) Bayes Rule:

$$\begin{aligned} P(D|+) &= \frac{P(D)\cdot P(+|D)}{P(D)\cdot P(+|D)+P(D^c)\cdot P(+|D^c)} \\ P(D^c|-) &= \frac{P(D^c)\cdot P(-|D^c)}{P(D^c)\cdot P(-|D^c)+P(D)\cdot P(-|D)} \end{aligned}$$

With $P(D)=0.001, P(+|D)=0.997, P(-|D^c)=0.985$ we get

$$\begin{aligned} P(D|+) &= \frac{0.001\cdot 0.997}{0.001\cdot 0.997+(1-0.001)\cdot(1-0.985)} = 0.06238268 \\ P(D^c|-) &= \frac{(1-0.001)\cdot 0.985}{(1-0.001)\cdot 0.985+0.001\cdot(1-0.997)} = 0.999997 \end{aligned}$$

(d) Mention that $P(+|D)$ and $P(-|D^c)$, are accuracy rates of a diagnostic test for the two possible results. They should be close to 1 because no one would take an inaccurate test.
We expect a large value, since $DLR_+ = \frac{P(+|D)}{P(+|D^c)}$ = value close to 1 divided by a value close to 0.
$DLR_+ = 66.46667$

(e) We expect a small value, since $DLR_- = \frac{P(-|D)}{P(-|D^c)}$ = value close to 0 diveded by a value close to 1.
$DLR_- = 0.003045685$

(f)

$$\begin{aligned} \frac{P(D|+)}{P(D^c|+)} &= \frac{P(D\cap+)/P(+)}{P(D^c\cap+)/P(+)} = \frac{P(D\cap+)}{P(D^c\cap+)} \\ &= \frac{P(+|D)\cdot P(D)}{P(+|D^c)\cdot P(D^c)} = \frac{P(+|D)}{P(+|D^c)}\cdot\frac{P(D)}{P(D^c)} \\ &= \text{pretest odds of disease}\cdot\text{odds of disease} \\ &= DLR_+\cdot\text{odds of disease} \\ &= 0.0665332 \end{aligned}$$

(g)

$$\begin{aligned} \frac{P(D|-)}{P(D^c|-)} &= \frac{P(D\cap-)/P(-)}{P(D^c\cap-)/P(-)} = \frac{P(D\cap-)}{P(D^c\cap-)} \\ &= \frac{P(-|D)\cdot P(D)}{P(-|D^c)\cdot P(D^c)} = \frac{P(-|D)}{P(-|D^c)}\cdot\frac{P(D)}{P(D^c)} \\ &= \text{pretest odds of disease}\cdot\text{odds of disease} \\ &= DLR_-\cdot\text{odds of disease} \\ &= 3.048734e-06 \end{aligned}$$

(h) Since $\frac{P(D)}{P(D^c)} < 1$ post-test odds are less than pre-test odds.

5. In a computer science course at an university we have the following data over a long time.
   10% of all students have attended the exercises in statistics regularly. 2% of the students who have failed the statistics exam have attended the exercises regularly. 5% of the students who have attended the exercises regularly have failed the statistics exam.

   (a) Find the probability to fail the exam in statistics if the exercises in statistics are not attended regularly.

   (b) What is the effect of attending the exercises regularly to passing the exam?

   Answer: From the data we could assume for the events A="student attends the exercises regularly", B="student fails exam" the following probabilities

$$P(A) = 0.1, P(A \mid B) = 0.02, P(B \mid A) = 0.05$$

$$\begin{aligned} P(B) &= \frac{P(B \mid A)P(A)}{P(A \mid B)} = \frac{0.05 \cdot 0.2}{0.02} = 0.25 \\ P(B \mid A^c) &= \frac{P(B)P(A^c \mid B)}{1 - P(A^c)} = P(B)\frac{1 - P(A \mid B)}{1 - P(A)} \\ &= 0.25 \cdot \frac{1 - 0.02}{1 - 0.1} \approx 0.272 \end{aligned}$$

   Since $P(B \mid A) = 0.05 < P(B \mid A^c) \approx 0.272$ is the probability to fail of an untrained student more than 5 times higher as the probability of a trained student.