Exam 1 SCR

The NZA Fee and Insurance Company Fees

the R-team

October 31, 2018

Exam Instructions

- You are allowed to consult the internet and all files on your computer, or your physical prepared written notes. However, keep in mind that any form of direct communication with others e.g. chatting, apping, facebook is not allowed and is considered **FRAUD**.
- Your exam answers need to be a neat `.R` script, including comments.
- Upload your `Lastname_ULCN.R` file only (e.g. `Tzortzakis_0730406.R` or `Kampert_002143.R`). Turn it in **BEFORE** 13:05 hours to **Blackboard**:
  - Go to Statistical Computing with R –> Exams & Assignments –> Exam 31 October 2018
  - Every minute later than 13:05 hours will cost you 10 out of 100 points: when you submit your exam at 13:10 hours spot on, it means you already lost 50 points. Exams submitted after 13:10 hours will not be graded.
- The exam consists of 3 tasks, each divided into five subtasks. Each subtask is worth an equal amount of points. To obtain a perfect score (100 out of 100), you'll need to obtain a perfect score on at least 10 subtasks. All subtasks will be graded.
- Your style of coding can affect your final exam grade. Your code does not need to be beautiful, the correct answer is the most important, but e.g. very complicated code where a simple known base `R` function will also work may cost you points.
- Adhere to consistent and neat programming style, for some examples:
  - https://google.github.io/styleguide/Rguide.xml
  - http://style.tidyverse.org
- Assume that you work in a similar working directory as the one in which you can find this `.Rmd` file. E.g. scriptlines such as `load("0_data/scr_exam_1_the-basics.Rdata")` should evaluate correctly.

Success!

the R-team.

The Exam

This exam is about the health insurances in the Netherlands, in particular, the extent to which your Health Insurer will reimburse the cost declared by your Healthcare provider in the Mental Healthcare sector. From the below background information please try and grasp what we mean by:

1. a DBC number
2. The NZA fee for a DBC number
3. The average contracted fee for a DBC number by a health insurance company

The (too) Simplified Background Information

Services and products offered by healthcare providers are identified by a three-digit number, the DBC number (Diagnosis and Treatment Combination). For example, 6 hours of therapy for Anxiety corresponds to DBC number 237. Invoices are issued by healthcare providers based solely on the DBC number corresponding to the service provided.

For each DBC number, the National Healthcare Authority (NZA) determines the maximum price that healthcare providers can ask for – this is referred to as the NZA fee. In general, Healthcare providers are not allowed to ask fees higher than the NZA fee.

Before the start of each year, the healthcare insurance companies negotiate the fees corresponding to each DBC number with selected healthcare providers. In principle, these fees need not be the same. For example, for DBC number 999, health insurance company X may have negotiated a fee of 100 euros with provider A, 110 euros with provider B, and 150 euros with provider C.

In this way, each insurer can compute an average fee for each DBC number, based on all its contracted fees for that specific DBC number. In the example above, insurance company X calculates an average fee of 120 euros for DBC 999, irregardless of the number of patients are treated by A, B or C.

Suppose you have a contract with health insurance company X. If you seek treatment corresponding to DBC 999, and receive treatment from providers A, B or C, you will be reimbursed the full cost of your treatment and you will not have to pay anything out of your own pocket. However, if for some reason you (have to) choose to receive a treatment from provider D, with which your insurance company X does not have a contract for DBC 999, company X will only reimburse a maximum of 75% of its computed average contracted fees for DBC 999, but you will be charged the full NZA fee by provider D.

The data

In this exam you will work with data that consists of the 2018 fees for DBC numbers in Mental Healthcare. For each DBC number, we have the NZA fee, and the average contracted fees of health insurance companies: `CZ`, `Menzis`, `VGZ`, Zorg & Zekerheid (`ZandZ`), and Zilveren Kruis (`ZK`).

The data and other variables and object you may need to use during the exam are stored in the file `healthcare_fees_2018.RData`, to be found in the `0_data` folder. Make sure that you can load this file with the following code:

```r
load("0_data/healthcare_fees_2018.RData")
```

The data is available in a long format (many rows, fewer columns):

```
str(fees2018_long)
```

```
## 'data.frame':    722 obs. of  7 variables:
##  $ dbc        : int  7 8 9 13 14 15 16 27 30 31 ...
##  $ group      : chr  "Diagnostics" "Diagnostics" "Diagnostics" "Crisis" ...
##  $ min_minutes: num  1 100 200 1 100 200 400 250 1800 3000 ...
##  $ max_minutes: num  99 199 399 99 199 ...
##  $ fee        : num  152 310 597 160 331 ...
##  $ insurance  : Factor w/ 6 levels "CZ","Menzis",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year       : num  2018 2018 2018 2018 2018 ...
```

We also have a wide format data set that can be seen here below:

```
str(fees2018_wide)
```

```
## 'data.frame':    121 obs. of  10 variables:
##  $ dbc        : int  7 8 9 13 14 15 16 27 30 31 ...
##  $ group      : chr  "Diagnostics" "Diagnostics" "Diagnostics" "Crisis" ...
##  $ min_minutes: num  1 100 200 1 100 200 400 250 1800 3000 ...
##  $ max_minutes: num  99 199 399 99 199 ...
##  $ NZA        : num  161 332 639 170 351 ...
##  $ CZ         : num  152 310 597 160 331 ...
##  $ Menzis     : num  146 302 581 NA NA ...
##  $ VGZ        : num  154 272 543 130 291 ...
##  $ ZandZ      : num  162 287 575 143 317 ...
##  $ ZK         : num  134 276 531 141 292 ...
```

In this wide format data set, the average contracted fee for each DBC number of every health insurance company is a variable on its own.

If we don’t tell in any of the subtasks which of the two data sets you need to use, then choose whichever of the data sets you prefer.

Task 1: Descriptives

1.1 Regarding missings in the data

a

Are there any fees missing (NA) in the long format data fees2018_long?

b

Are there any insurance companies that have missing fees (NA values) for certain DBC numbers? Use the wide format data fees2018_wide.

Hint: You may like to use the object representing the column names of the insurance companies only.

```
insurances <- c("CZ", "Menzis", "VGZ", "ZandZ", "ZK")
```

1.2

Are there any DBC numbers present in fees2018_long that are not present in fees2018_wide? If yes, show with R code which DBC numbers these are, and what kind of treatment comes with each specific DBC number that is absent in fees2018_wide.

1.3

According to the data in fees2018_wide we see that there is no data available for Menzis regarding crisis treatments and that there is no fee data available for VGZ and ZK for the Diagnostics treatment types where the minimum amount of minutes is at least 800 minutes. See the code:

```
fees2018_wide[rowSums(is.na(fees2018_wide[, as.character(insurances)])) > 0, ]
```

Could you corroborate these statements based on fees2018_long data for Menzis? And how about the fees for VGZ and ZK on the other hand?

1.4

Are the fees set by the NZA always higher than those from the health insurance companies? Show for which insurance companies this is the case, and for which companies this is not the case.

Hint: Using fees2018_wide over here makes life easier, but it's your choice.

1.5

a

Suppose you expect to end up with a non-contracted mental health-care provider, and you expect that the treatment sessions will **not** take longer than at most 20 hours in total. Then, the only informative rows in the fees2018_long data set are those for which the DBC numbers have time intervals that include 20 hours or less. Select these rows from the data, and assign them to a new data.frame variable with a name of your choice.

Show that your filtered `data.frame` is the same as the first 7 columns of `fees2018_long_ub20hrs`, a data set that also came with loading the file `healthcare_fees_2018.RData`.

b

In case you did not succeed in 1.5a, use the `fees2018_long_ub20hrs` data.

For each health insurance company, compute the average fee over all the DBC numbers, and show your results. Suppose this is the only information that you have. With which Healthcare insurancy company would you expect to pay the least when visiting a health care provider that is not part of your health insurance contract?

Task 2: Visualization

In this task we will mainly work with the data that consists of the fees for DBC numbers that represent treatment groups that do not take up more than 20 hours. Remember, these data sets are already loaded in yor workspace: the long format data is `fees2018_long_ub20hrs`, the wide format data is assigned the name `fees2018_wide_ub20hrs`.

You may also like to use the following colors for the NZA and each insurance company:

```r
insurance_colors <- c(
  NZA = "grey",
  CZ = "orange",
  Menzis = "blue4",
  VGZ = "green4",
  ZandZ = "cornflowerblue",
  ZK = "darkred"
)
```

2.1

Reorder the factor levels of the insurances factor variable in the data.frame `fees2018_long_ub20hrs` into the order as presented for the `insurance_colors`.

2.2

As was stated in the introduction of this exam. If you visit a health care provider that does not take part in your health insurance contract, then only a maximum of 75% of the fee of the health care insurance will be reimbursed to you. Moreover, the (mental) healthcare provider is in most cases binded to declare costs that are equal to the NZA fee.

Take a look at the variable `reimbursed` in the `fees2018_long_ub20hrs` data. This variable represents 75% of the fees of each of the health care insurance companies, but keeps 100% of the fees when the insurance variable value is `NZA`. Show that you can create this variable yourself too.

2.3

Take a look at the helpfile, or the examples of the function `split()`, and use this function to split the `reimbursed` variable of `fees2018_long_ub20hrs` into a `list` where each item in the list represents the reimbursed fee of either one of the insurance companies, or the NZA fee. Then, use this output to create the boxplots of Figure 1 while using the `fees2018_long_ub20hrs` data.

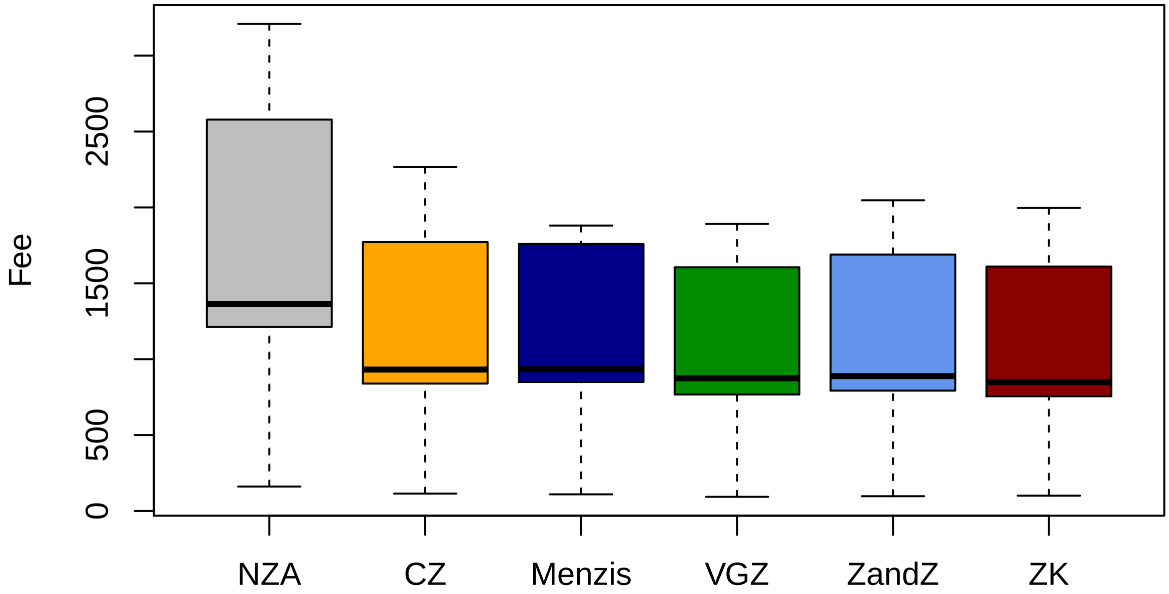

Figure 1: Boxplots of the distribution of the reimbursed fees of the health insurance companies and a boxplot of the distributon of the NZA fees (in grey).

Looking at these boxplots, what can you say about the median reimbursement of the health care insurances as compared to the NZA fee? Without showing any `R` code, how could you reason whether the medians in each boxplot are lower or higher than their corresponding mean?

NB: Note that the order of the boxplots may depend on your answer of subtask 2a. As long as the labels and colors are correct, we will not penalize a wrong order of the boxplots.

2.4

These boxplots from Figure 1 don't really show the real gap that remains to be payed, but merely the distribution of the reimbursed fees of every health insurance company, and the distribution of the NZA fees. The amount that you'll need to pay yourself for any of the DBC numbers is the NZA fee from which 75% of the Health insurance company fee is substracted, e.g. in code this could be

```
MyPay_for_ZK <- NZA_fee - 0.75 * ZK
```

Use the `fees2018_wide_ub20hrs` data to create a `data.frame` where each column is a variable representing how much you need to pay for each DBC number based on having a health insurance with one of the Health Insurance companies. Your data.frame should have the same values as `pay2018_wide_ub20hrs`. Here, the rows, representing the DBC numbers, in `pay2018_wide_ub20hrs` are ordered based on the NZA fee, from lowest to highest NZA fee.

2.5

Create Figure 2 by using the data set `pay2018_wide_ub20hrs` or your data created in subtask 2.4. Could you give in a few senteces a short interpretation of these two panels in the Figure? Do you notice something peculiar?

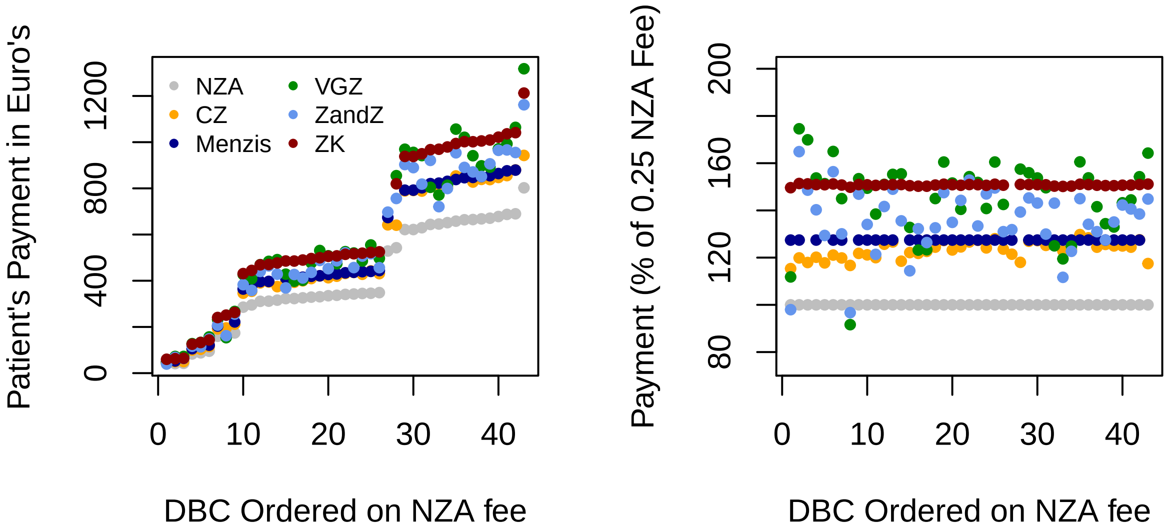

Figure 2: Patient's payment for each insurance company (in colors) for the rank numbers of the DBC codes that are ordered from low to high based on 0.25 * NZA Fee. In the left panel, the payment is expressed in Euros, the right panel the payment is expressed as a percentage of the 0.25 NZA Fee.

Task 3: Programming

In this task you'll create some of the functions that the instructors used to create the `fees2018_long` data from the `*.csv` files in the `0_data` folder.

3.1

Import each of the following files

- `NZA2018.csv`
- `ZK2018.csv`
- `CZ2018.csv`
- `MENZIS2018.csv`
- `ZorgZek2018.csv`

into a data.frame, and put them into a list. Each component (item) on the list will be a data.frame. Show that your answer is exactly the same as `fees_raw_list`.

Hint: A 100% score for this subtask can be obtained when you use an explit or implicit loop and not manually write down the file names in your `R` script. In particular, you may like to use the output of `list.files("0_data/", pattern = "2018.csv")`

3.2 Split one variable into three variables

Note that the `Group` variable of each data frame in `fees_raw_list` consists of a pattern from which we can extract three types of variables. The first variabe shows the type of the treatment group, the second variable shows the minimum amount of minutes, the third variable shows the maximum amount of minutes of the specific tariff group of that disorder.

Thus, the following value on the `Group` variable,

```
Diagnostics - from 1 up to and including 99 minutes
```

is a treatment group for `diagnostics` (= value of variable 1), for which the minimum time of the treatment is `1` minute (= value of variable 2), and the maximum set at `99` minutes (= value of variable 3).

Create a function that has as input one of the data sets, and that gives as output a data.frame with the three variables extracted from the `Group` variable. Show that the output of your function on the NZA data.frame in the list has the same values as those in `NZA_3vars`

*NB. Except for

`Attention Deficit - and behaviour - from 6000 up to and including 11999 minutes`

all the treatment groups are separated from the minutes with a hyphen "-". Replacing one of the two hyphens with something else might make it easy for you to split the name of each treatment group from the minimum and maximum number of minutes.

3.3

Without regular expressions, write a function that can split an input vector of the like

```r
chars <- c("Menzis2018", "ZK2017", "NZA2016")
```

into a `data.frame` with a `character` variable and a numeric variable, e.g.

```r
datfram_example <- data.frame(
  insurance = c("Menzis", "ZK", "NZA"),
  year = c(2018, 2017, 2016),
  stringsAsFactors = FALSE
)
```

Apply your function on `chars` and show that the output of your function is exactly the same as `datfram_example`.

3.4

All data sets in `fees_raw_list` have a column that represents the (average) fee of the specific health insurance company for each code of a treatment group.

For the `data.frame` in the list with the name `NZA`, this is `fee` column. The names of the correct column for the other data sets consist of either the characters `"X1"` or `"ontract"`.

Create a function that takes as input `fees_raw_list` from subtask 3.1. Inside the body of the function the specific fee column of each data set in the list is extracted such that it outputs a `fee` variable which is equal to the `fee` variable in `fees2018_long`. Don't forget to show that the output of your function on the `fees_raw_list` is exactly the same as the fee variable in `fees2018_long`.

3.5

Combine, and perhaps even reshape, the code from the previous subtasks to recreate the long format data set `fees2018_long`.

If you did not manage to answer all the previous subtasks, then make use of the `fees_raw_list`, `dats_3vars`, `datfram_ins_year` objects that are loaded in your workspace. With these objects you can also construct the data set `fees2018_long`.