Resolving Stepic problems

Valkrug

07/01/2022

This is my solutions for the problems published on study platform Stepic.org

Problem 1

Write a function that receives as an input a dataframe with one quantitative variable and any number of factor variables.

Factor variables will break all our observations into a certain number of groups.

You should create a new numeric variable in the data, which should be 1 if the observation in this row is an outlier in its group, and 0 if it is not.

```r
find_outliers <- function(t){
   t$name <- t[,sapply(t, is.numeric)]
   t %>%
     group_by_if(is.factor) %>%
     mutate(name, is_outlier= ifelse(name >
               mean(name)+2*sd(name) | name < mean(name)-2*sd(name), 1, 0)) %>%
     select(!name)
}
```

Problem 2

We have a data set (.csv) with errors. My task was to write the fix_data function, which receives a dataset as an input. Some variables of the data set have space between numbers added in some of the numeric variables. We need to delete this space and return the numeric variables to their numeric type (because now they are string). The function should return a converted dataset, in which all numeric variables will be converted to a numeric type, while those variables that really string do not need to be converted in any way.

```r
fix_data <- function(d){
   var_names <- names(d)
   fixed_data <- mutate_at(d, var_names, .funs = function(x)
      ifelse(stri_detect_regex(x, "[[:alpha:]]")==FALSE,
      as.numeric(stri_replace_all(x, regex = "[[:space:]]", "")), x))
}
```

Problem 3

Imagine that you have a clinical study, there during the seven days should be measured the temperature of the participants. Every participant of the study has their own id.

Participants had to come for an examination every day during a whole week. After the end of the study, it turned out that some participants were unable to attend all seven appointments. Someone after the first time no longer came to the examination, someone missed some days, so. For the purity of the study you need to write a get_id function that receives list of seven dataframes as an input (with variables: id of the participant and temperature). The function should return a new dataframe, there will be two variables "id" and "mean_temp" - the average temperature for a week only for those participants who attended all seven appointments, or in other words, the id of such participants is present in each of the seven dataframes.

```r
get_id <- function(data_list){
  new_data <- do.call(rbind, lapply(data_list, data.frame))
  new_data$id <- as.factor(new_data$id)
  new_data %>%
    group_by(id) %>%
    summarise(n=n(), mean_temp=mean(temp)) %>%
    filter(n==7) %>%
    select(-n)
}
```