Andreas Bender, Julia Niebisch
Institut für Statistik

Statistische Software (R) – Hausarbeit 5

Wintersemester 2022

Name: Olha Kosach

Immatrikulationsnummer: 12658180

Studiengang: Statistik und Data Science

Hiermit bestätige ich, dass ich die Anweisungen auf diesem Blatt gelesen und verstanden habe. Ich bestätige, dass die abgegebene Lösung vollständig und alleinig von mir bearbeitet und erstellt worden ist, ohne Hilfe von anderen in Anspruch zu nehmen. Ich bestätige, dass ich über die Vorlesungsmaterialien hinausgehende Quellen wie Bücher oder Internetseiten im Code angegeben und falls zutreffend verlinkt sind.

Unterschrift: Kosach

Prüfungshinweise:

1. Überprüfen sie, ob die heruntergeladene Angabe vollständig ist. Sie sollte 2 Aufgaben beinhalten.
2. Insgesamt können 20 Punkte (+1 Bonuspunkt) erreicht werden.
3. Die Lösung soll in Form von einer einzelnen `.Rmd` Datei (Rmarkdown) abgegeben werden. Benennen Sie diese Datei `AS5.Rmd`. Ihre Lösung muss Ihren vollständigen Namen und Ihre Immatrikulationsnummer beinhalten. Idealerweise oben in den Metainformationen des Dokuments unter "author".
4. Setzen Sie die `code chunk options` so, dass der Code sowie der R-Output im output file zu sehen sind.
5. Achten Sie darauf, dass Ihre `.Rmd` Datei kompilierbar ist, es dürfen keine Fehler im Code sein, die das kompilieren unmöglich machen.
6. Laden Sie die unterschriebene Angabe zusätzlich als pdf, jpg oder png Datei hoch. Wenn die Datei zum Zeitpunkt der Abgabe fehlt oder nicht unterzeichnet ist, werden sofort 0 Punkte eingetragen und keine Ausnahmen gemacht.
7. Die Abgabe erfolgt über Github Classrooms **oder** über Moodle. Für eine Abgabe mit Github Classrooms gibt es 1 Bonuspunkt. Bei einer Abgabe mit Github Classrooms zählt immer das aktuellste commit innerhalb der deadline.
8. Entscheiden Sie sich für eine der beiden Abgabenformen. Sobald Sie die GitHub Classroom Einladung annehmen und mindestens 1 mal etwas committed und gepusht haben, ist dies eine Abgabe mit GitHub Classrooms.
9. Machen Sie Beginn und Ende einzelner Probleme, Aufgaben und Teilaufgaben kenntlich. Ist die Zugehörigkeit von Code zu einer der (Teil-)Aufgaben nicht eindeutig deklariert, kann es passieren, dass Sie dafür keine Punkte bekommen.
10. Achten Sie darauf, dass alle Funktionen nach der Vorgabe in den Übungen dokumentiert sind.
11. Sollten Sie Verständnisfragen haben, nutzen Sie hierzu bitte das Forum, welches auf der Moodle Seite unter Forum zu finden ist.

12. Sollten Sie technische oder andere Schwierigkeiten haben, kontaktieren Sie bitte die Kursleiter. E-mail: andreas.bender@stat.uni-muenchen.de, julia.niebisch@stat.uni-muenchen.de. (Bitte die Emails an alle gelisteten Personen schicken!)
13. **Die Aufgaben müssen alle eigenständig bearbeitet werden. Insbesondere sind keine Arbeitsgruppen erlaubt und sonstige Diskussion der Aufgaben und Lösungen mit anderen Personen (egal ob diese Statistik studieren oder nicht) nicht zulässig.**
14. **Das Internet kann passiv genutzt werden. D.h. es dürfen Internetseiten oder Foren aufgerufen und gelesen werden, das aktive Stellen von Fragen, die relevant zur Lösung der Aufgaben sind, ist allerdings nicht zulässig. Ebenso dürfen keine Aufgaben oder Lösungsvorschläge und andere Hinweise im Internet gepostet oder per Chat, Email und anderen Kommunikationswegen diskutiert oder verteilt werden.**
15. **Sollte der Verdacht auf Plagiat, Betrug oder anderweitig unzulässiges Verhalten bestehen, können zusätzliche (mündliche) Prüfungen einberufen werden um die eigenständige Bearbeitung der Aufgaben zu prüfen. Wir nutzen Neuronale Netze um Unterschleif (teil-)automatisiert zu prüfen.**
16. **Zweifel an der eigenständigen Bearbeitung ihrer Abgabe führen zum Nichtbestehen der Prüfung und dem Einschalten des Prüfungsausschusses.**
17. Verwenden und laden Sie keine zusätzlichen R Pakete, außer die in den Teilaufgaben angegebenen Pakete.
18. Die Abgabe erfolgt bis zum 29.01.2023 um 23:50 Uhr.

Click on the GitHub Classroom invitation URL on moodle and accept the assignment. A repository under your GitHub name will automatically be created. For example, if your GitHub name is `janedoe`, your repository will be named *assignment5-janedoe*. Clone the repository to your local machine. Create a `rmarkdown` document in your repository folder with the title `AS5` and output format pdf. Save your file with the name `AS5.Rmd` in your local GitHub repository. **Sign this pdf on page 1 and put the first page in your repository in pdf, png or jpg format.** When you are done with this assignment, push your solution file `AS5.Rmd` and the signed pdf (or png/jpg) to the remote GitHub repository within the assignment deadline. Check the remote repository to see if it worked. If the solution file and the pdf (or jpg/png) show up in your remote GitHub repository, you are done with this assignment. It is also advisable to push your solution file after each subtask or after each day you work on it. That way, you get used to the GitHub workflow and you can make sure everything works. (1 Bonuspoint)

In this Assignment, you have to use the package `dplyr` and/or `tidyr` for each subtask. This means you have to use the pipe-operator `%>%` as well as package specific functions. If you solve the tasks with base R code only, there will be no points awarded. You may use base R for code for auxiliary tasks.

Install and load the packages `tidyr`, `dplyr`, `readr` and `ggplot2`. In your `Rmd` file, only keep the code for loading the packages using `library()`, not for installing them.

Aufgabe 1	5 Punkte

In this task, you have to transform three untidy data frames into tidy data (see lecture slides). You can find the untidy data in your repository and on moodle. All transformations should be done with calls to functions from package `tidyr` without additional calls to other functions (outside the calls of `tidyr` functions). Make sure that all columns have the appropriate data type after the transformation. Also, print the resulting tidy data at the end of each subtask such that it appears in the rendered PDF document.

(a) Read in the file `table1.Rds` and transform it into a tidy data set. (1P)

(b) Read in the file `table2.Rds` and transform it into a tidy data set. (2P)

(c) Read in the file `table3.Rds` and transform it into a tidy data set. (2P)

Aufgabe 2	15 Punkte

Read in the data from the file `auto_verbrauch.Rds`, which is available in your repository and on moodle. Name the data `auto`. It contains information on three different cars regarding gas consumption and travelled distances. For each date and type of car, the data set contains mileage (`km`), distance travelled since last refuelling the tank (`kmdiff`), liters of fuel added (`liter`) and total cost of refuelling (`euro`). The columns exist for each car separately, indicated by the prefixes `BMW_`, `volvo_` and `seat_`.

In the following tasks, overwrite the `auto` data in each sub task unless it is explicitly mentioned not to do so.

(a) Turn the `auto` data into a tidy data set and reassign it to `auto`. The column names of the transformed data should be `type`, `date`, `km`, `kmdiff`, `liter` and `euro`, where `type` stores the information about the car, i.e. has values BMW, volvo or seat. The columns should appear in the order given above and the data should be sorted by `type` and `date`. Print the first 8 lines of the transformed data. (3P)

(b) Extract and print the first and last observation per type. Do not overwrite the data (print only). (1P)

(c) Calculate three new columns from the existing columns in the `auto` data and add them to the data set, namely

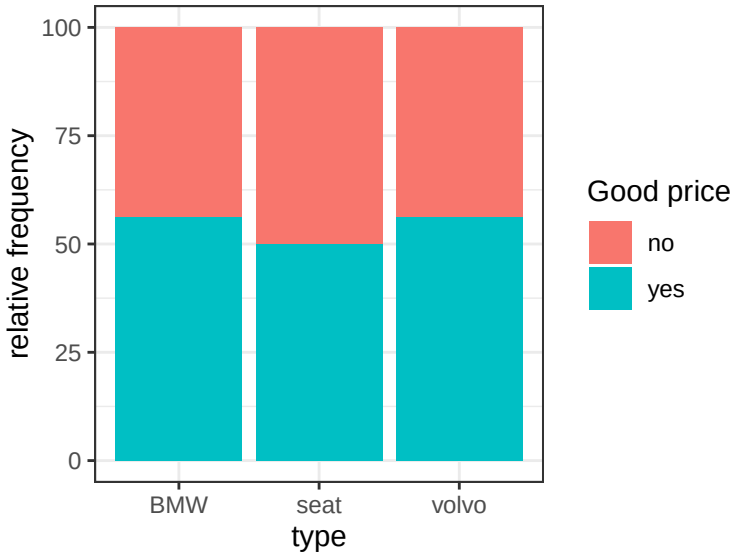
- price per liter (`price_per_l`),
- the price per kilometer (`price_per_km`) and
- the liter consumption per 100 km (`liter_per_100`)

(1P)

(d) Read in the file `gas_prices.Rds`, which is available in your repository and on moodle. It contains the average gas prices for each day between March and November. Join the data to your `auto` data set such that it contains the average gas price on each day with observations from the three cars. After the join, the `auto` data should have the same number of rows as before the join and one additional column. Create a new column `good_price` of type factor that contains `yes` if the `price_per_l` is below or equal to the average price and `no` otherwise. (2P)

(e) Create a table that indicates how often each car (`type`) was fueled at a below/above average price. The resulting table should appear in the rendered document. Do not overwrite the data (print only). (1P)

(f) Create a stacked bar plot that shows the relative frequency of times each car `type` was fueled at a below and above average price respectively using `ggplot2` as shown below. (2P)



(g) Use the `ggplot2` package to plot the liter consumption per 100 km from sub task (c) as a line over time for each of the three cars. Draw the three lines in one plot with different colors for each car. (1P)

(h) There are some unrealistic values in the column `kmdiff` for the BMW. With a full tank of fuel it can drive about 700 km at most. It also shows in the plot from sub task (g), because the column for the liter consumption per 100 km was calculated using `kmdiff`. To correct this, perform the following steps:

- Select the BMW data and calculate the median value of `kmdiff`.
- Select all rows where `kmdiff` is greater or equal to the median value but less than 700 excluding the unrealistic values, again only for the BMW.
- Calculate the mean `kmdiff` for this subset and replace the unrealistic values above 700 km with the calculated mean value.
- Calculate the column `liter_per_100` anew and plot it using your code from sub task (g).

If you did not solve the task, output the BMW data in your rendered file. Leave the other cars obervations untouched, but reassign the `auto` data such that it contains your transformations from this subtask. (3P)

(i) The `auto` data now contains all transformations from the tasks above. For each car, calcualate and output the kilometers driven in total using the column `km`, the total cost for fuel and the mean consumption per 100 kilometers over the observation period. (1P)