Andreas Bender, Julia Niebisch
Institut für Statistik

Statistische Software (R) – Hausarbeit 4

Wintersemester 2022

Name: Mingyi Zhou

Immatrikulationsnummer: 12056620

Studiengang: Statistik und Data Science +VWL

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: 周明悦

Prüfungshinweise:

1. Überprüfen sie, ob die heruntergeladene Angabe vollständig ist. Sie sollte 2 Aufgaben mit je zwei Teilaufgaben 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 `AS4.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. 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.
9. Achten Sie darauf, dass alle Funktionen nach der Vorgabe in den Übungen dokumentiert sind.
10. Sollten Sie Verständnisfragen haben, nutzen Sie hierzu bitte das Forum, welches auf der Moodle Seite unter Forum zu finden ist.
11. 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!)

12. **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.**
13. **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.**
14. **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.**
15. **Zweifel an der eigenständigen Bearbeitung ihrer Abgabe führen zum Nichtbestehen der Prüfung und dem Einschalten des Prüfungsausschusses.**
16. Verwenden und laden Sie keine zusätzlichen R Pakete, außer die in den Teilaufgaben angegebenen Pakete.
17. Die Abgabe erfolgt bis zum 09.01.2023 um 17 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 *assignment4-janedoe*. Clone the repository to your local machine. Create a `rmarkdown` document in your repository folder with the title `AS4` and output format pdf. Save your file with the name `AS4.Rmd` in your local GitHub repository. **Sign this pdf on page 1 and put it in your repository in pdf, png or jpg format.** When you are done with this assignment, push your solution file `AS4.Rmd` and the signed pdf 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 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 replicate the plots created with the `ggplot2` package as exactly as possible. In each subtask, you have to perform some data preprocessing before you can recreate the plot. Points are given only on the replication of the plot and not on the individual preprocessing steps. They are however neccessary to get the plot right. For each deviation from the below shown plots, points are deducted. It does not matter if you do not have the exact same color of green or if the font size of the axis labels is not the same, however it should replicate the main features of the plot as exactly as possible.

Install and load the packages `ggplot2`, `reshape2` and `patchwork`. In your `Rmd` file, only keep the code for loading the packages using `library()`, not for installing them.

Aufgabe 1	10 Punkte

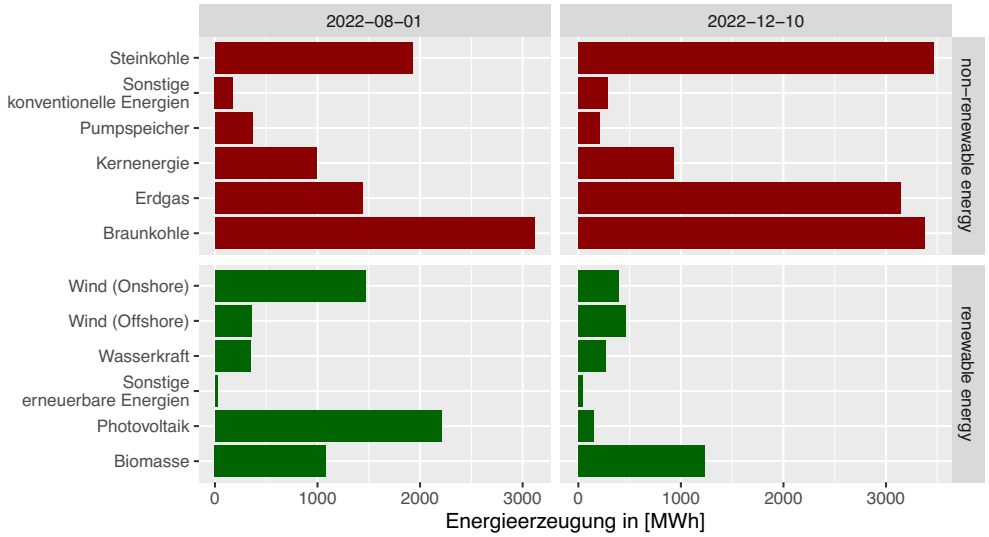
Read in the data from the file `Energieerzeugung_long_format.csv`, which is available in your repository. It contains the energy production from different sources over time (*Hint: See help page for function* `read.delim`). The column `date` is of type character, so R does not understand it as a date. Change the type to a Date format. Name the data set `dataraw` and use it for both subtasks in this exercise. The first 6 rows of the data are shown below:

```
##         date start   end   source energy
## 1 2022-01-01 00:00 00:15 Biomasse 1084.8
## 2 2022-01-01 00:15 00:30 Biomasse 1081.2
## 3 2022-01-01 00:30 00:45 Biomasse 1080.8
## 4 2022-01-01 00:45 01:00 Biomasse 1085.2
## 5 2022-01-01 01:00 01:15 Biomasse 1083.2
## 6 2022-01-01 01:15 01:30 Biomasse 1081.2
```

(a) Start with the data in `dataraw` and follow the following data preprocessing steps to recreate the plot using `ggplot2`.

- Subset the data so that it only contains the information for the two dates `2022-12-10` and `2022-08-01`.
- Create a new column `renewable` that indicates if the `source` of the energy is renewable or not. As can be seen in the plot, renewable energy sources are *Biomasse, Wasserkraft, Wind_Offshore, Wind_Onshore, Photovoltaik* and *Sonstige_Erneuerbare_Energien*. Some of the categories in the column `source` need to be replaced with nicer looking categories (for example *Wind_Onshore* becomes *Wind (Onshore)*).

Use the data to plot a bar chart that shows the mean energy production for each energy source on the two dates of interest for renewable and non-renewable energy sources. Try to replicate the plot as exactly as possible (see information above).

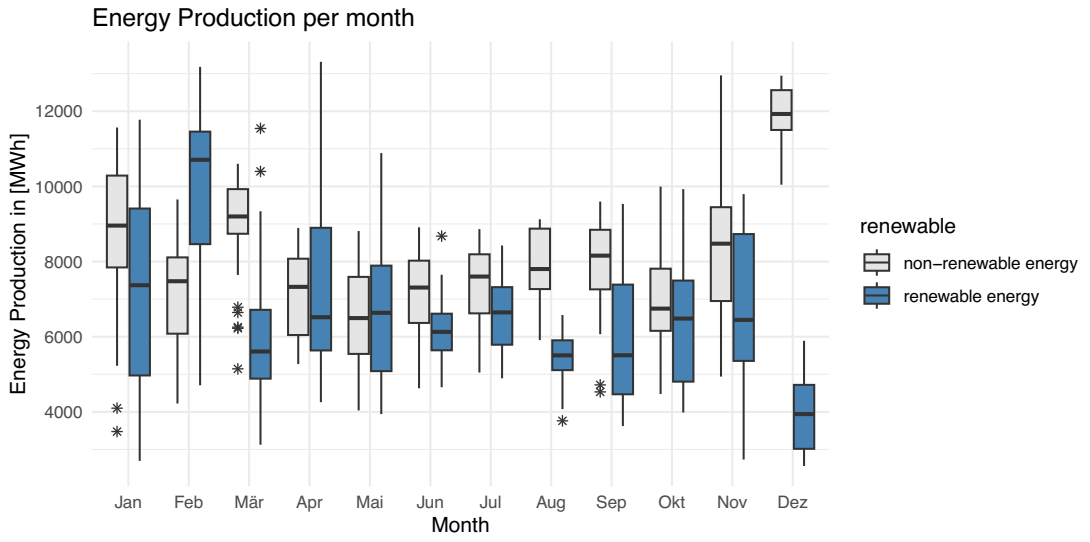


(b) Start with the data in `dataraw` and do the following data preprocessing steps to recreate the plot using `ggplot2`.

- Delete the columns `start` and `end`.
- Aggregate your data such that it contains the mean energy production per day for each energy source. After this step, your data consists of one row per `date` and `source`.
- Create a new column `renewable` that indicates if the `source` of the energy is renewable (= "renewable energy") or not ("non-renewable energy"), as in subtask (a).
- Aggregate your data again using the newly created column such that it contains the sum of energy productions for each date and category of the column `renewable`.

The first 6 rows of the new data could look as shown below (or similar). You may also include additional columns if that helps with the visualization. Use this data set to create boxplots that show the energy production per month of renewable and non-renewable energy sources. Try to replicate the plot as exactly as possible (see information above).

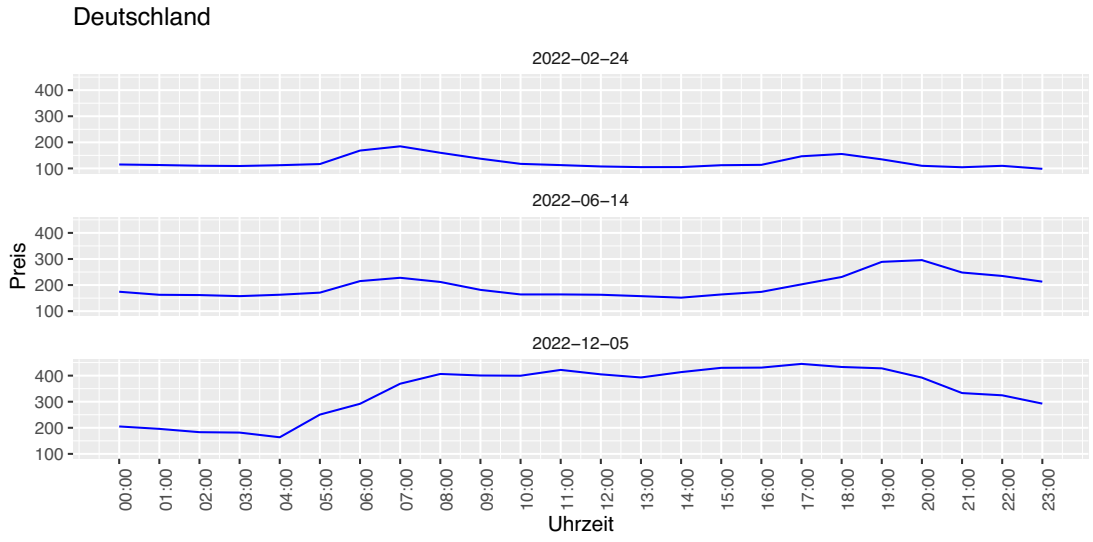
```
##         date            renewable energy
## 1 2022-01-01 non-renewable energy 4099.0
## 2 2022-01-02 non-renewable energy 3476.1
## 3 2022-01-03 non-renewable energy 5458.7
## 4 2022-01-04 non-renewable energy 8957.0
## 5 2022-01-05 non-renewable energy 7287.0
## 6 2022-01-06 non-renewable energy 9053.7
```



Aufgabe 2	10 Punkte

Read in the data from the file `Handelspreise_per_hour.csv`, which is available in your repository. It contains the prices for one specific energy source over time for different countries. The column `Date` is again of type character, so R does not understand it as a date. Change the type to a Date format. Name the data set `pricesraw` anduse it for both subtasks in this exercise.

(a) Start with the data in `pricesraw` and follow these data preprocessing steps to recreate the plot using `ggplot2`. Subset the data such that it only contains the three dates of interest for this plot: "2022-02-24", "2022-06-14" and "2022-12-05". The plot shows the development of the prices for the energy sources on the three days in Germany. For the x axis, the column `Start` is used, you may need to transform it a bit for this to work. Try to replicate the plot as exactly as possible (see information above).



(b) For this subtask, you need the package `patchwork` and the package `reshape2`. Start with the data in `pricesraw` and follow the data preprocessing steps to recreate the plot using `ggplot2`.
There are two columns for the country Denmark: `Dänemark_1` and `Dänemark_2`. Create a new column named `Dänemark` with the mean price from the two existing columns for Denmark and delete the columns `Dänemark_1` and `Dänemark_2` (*Hint: If on your computer you run into problems with special German characters, you may rename the countries according to their english names*). Aggregate the data such that it contains the maximum price per day for each country. Each row now represents a date. Use the function `melt()` from the package `reshape2` to transform your data set from wide format to narrow/long format (*Hint: Type `?melt.data.frame` for help on the `melt`-function and read `https://en.wikipedia.org/wiki/Wide_and_narrow_data` for information on wide and narrow/long data format*). If you prefer another way to do this transformation with base R, you can.
The first 6 rows as well as the dimensions of the processed data set are shown below. Use it to create the four plots below and arrange them with the `patchwork`-package. For the trend lines in all plots, use `geom_smooth(method = "loess")`.

```
dim(pricesmax)

## [1] 4816    3

head(pricesmax)

##         Date        Land  Preis
## 1 2022-01-01 Deutschland 149.97
## 2 2022-01-02 Deutschland  70.09
## 3 2022-01-03 Deutschland 148.49
## 4 2022-01-04 Deutschland 190.00
## 5 2022-01-05 Deutschland 148.09
## 6 2022-01-06 Deutschland 274.11
```