Solutions to the Exercises

Stephan.Huber@hs-fresenius.de

March 6, 2024

This document contains the solutions to the exercises of the lecture notes found here.

1 Links to R scripts

- exe_duplicates.R
- exe_import_covid.R
- exe_genanddrop.R
- exe_base_pipe.R
- exe_subset.R
- exe_data_transformation.R
- exe_poser.R
- exe_datasauRus.R
- exe_convergence.R
- exe_un_gdp_ger_fra.R
- exe_hortacsu_figure_3.R
- exe_regress_lecture.R
- exe_calories.R
- exe_bundesliga.R
- exe_okun_solution.R
- exe_zipf_solution.R

2 Output of the solutions

2.1 exe_duplicates.R

```
# Find duplicates

# set working directory
# setwd("~/Dropbox/hsf/test/initial_script")
```

```r
# clear environment
rm(list = ls())

# load packages
if (!require(pacman)) install.packages("pacman")
```

```
Loading required package: pacman
```

```r
pacman::p_load(tidyverse, janitor, babynames, stringr)

load(url("https://github.com/hubchev/courses/raw/main/dta/df_names.RData"))

# Remove all objects except df_2022 and df_2022_error
rm(list = setdiff(ls(), c("df_2022_error", "df_2022")))

# Re-order the data so that surname, name, and age appears first.
# Save the changed data in a tibble called `df`.
df <- df_2022 |>
  relocate(surname, name, age)

# Sort the data according to surname, name, and age.
df <- df |>
  arrange(surname, name, age)

# Inspect df_2022 and df_2022_error
df
```

```
# A tibble: 1,018 x 8
   surname name        age sex      cm  time error error_desc
   <chr>   <chr>     <dbl> <chr> <dbl> <dbl> <dbl> <chr>
 1 Adams   Adonnis      30 M       192  2022     0 <NA>
 2 Adams   Adonnis      30 M       192  2022     1 duplicate
 3 Adams   Aila         79 F       157  2022     0 <NA>
 4 Adams   Avenelle     69 F       157  2022     0 <NA>
 5 Adams   Brysan       39 M       192  2022     0 <NA>
 6 Adams   Eona         84 F       157  2022     0 <NA>
 7 Adams   Eveline      42 F       157  2022     0 <NA>
 8 Adams   Faithe       17 F       172.  2022     0 <NA>
 9 Adams   Ineisha      47 F       157  2022     0 <NA>
10 Adams   Kloeigh      31 F       157  2022     0 <NA>
# i 1,008 more rows
```

```r
dim(df)
```

```
[1] 1018    8
```

```r
head(df)
```

```
# A tibble: 6 x 8
  surname name       age sex      cm  time error error_desc
  <chr>   <chr>    <dbl> <chr> <dbl> <dbl> <dbl> <chr>
1 Adams   Adonnis     30 M       192  2022     0 <NA>
2 Adams   Adonnis     30 M       192  2022     1 duplicate
3 Adams   Aila        79 F       157  2022     0 <NA>
4 Adams   Avenelle    69 F       157  2022     0 <NA>
5 Adams   Brysan      39 M       192  2022     0 <NA>
6 Adams   Eona        84 F       157  2022     0 <NA>
```

```r
tail(df)
```

```
# A tibble: 6 x 8
  surname name       age sex      cm  time error error_desc
  <chr>   <chr>    <dbl> <chr> <dbl> <dbl> <dbl> <chr>
1 Young   Leiliana    54 F     157    2022     0 <NA>
2 Young   Shamar      23 M     192    2022     0 <NA>
3 Young   Tajanay      1 F      81.5  2022     0 <NA>
4 huber   Stephan    186 M      41    2022     1 age/cm false, not capitalized ~
5 huber   Stephan     NA <NA>   NA    2022     1 wrong name
6 <NA>    Zita         6 <NA>  110    2022     2 surname missing, sex unspecifi~
```

```r
glimpse(df)
```

```
Rows: 1,018
Columns: 8
$ surname    <chr> "Adams", "Adams", "Adams", "Adams", "Adams", "Adams", "Adam~
$ name       <chr> "Adonnis", "Adonnis", "Aila", "Avenelle", "Brysan", "Eona",~
$ age        <dbl> 30, 30, 79, 69, 39, 84, 42, 17, 47, 31, 65, 80, 6, 5, 5, 20~
$ sex        <chr> "M", "M", "F", "F", "M", "F", "F", "F", "F", "F", "M", "F",~
$ cm         <dbl> 192.00000, 192.00000, 157.00000, 157.00000, 192.00000, 157.~
$ time       <dbl> 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022,~
$ error      <dbl> 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0,~
$ error_desc <chr> NA, "duplicate", NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
```

```r
summary(df)
```

```
   surname              name                age              sex           
 Length:1018        Length:1018        Min.   :  1.00   Length:1018       
 Class :character   Class :character   1st Qu.: 21.00   Class :character  
 Mode  :character   Mode  :character   Median : 43.00   Mode  :character  
                                       Mean   : 45.75                     
                                       3rd Qu.: 69.00                     
                                       Max.   :399.00                     
                                       NA's   :2                          
       cm             time          error          error_desc       
 Min.   : 41.0   Min.   :2022   Min.   :0.00000   Length:1018       
 1st Qu.:157.0   1st Qu.:2022   1st Qu.:0.00000   Class :character  
 Median :157.0   Median :2022   Median :0.00000   Mode  :character  
 Mean   :163.2   Mean   :2022   Mean   :0.02456                     
 3rd Qu.:192.0   3rd Qu.:2022   3rd Qu.:0.00000                     
 Max.   :295.0   Max.   :2022   Max.   :3.00000                     
 NA's   :4                                                          
```

```r
df_2022_error
```

```
# A tibble: 18 x 8
   sex   name    surname    age    cm  time error error_desc
   <chr> <chr>   <chr>    <dbl> <dbl> <dbl> <dbl> <chr>
 1 M     Savier  Campbell    72 192    2022     1 duplicate
 2 F     Tina    Adams        5  98.0  2022     1 duplicate
 3 F     Abery   Allen       79 157    2022     1 duplicate
 4 M     Adonnis Adams       30 192    2022     1 duplicate
 5 M     Stephan Maier       41 186    2022     1 wrong surname
 6 <NA>  Stephan huber       NA  NA    2022     1 wrong name
 7 M     stephan Huber      186  41    2022     1 age/cm false, not capitalized~
 8 M     Stephan huber      186  41    2022     1 age/cm false, not capitalized~
 9 M     Stephan Huber       41 186    2022     1 duplicate
10 M     Stephan Huber       41  NA    2022     1 duplicate, cm NA
11 F     Rosa    Huber        9  NA    2022     3 only age and sex given
12 <NA>  Rosa    Huber       NA 130    2022     3 age missing, sex unspecified
13 <NA>  Ignaz   Huber        7  NA    2022     2 cm missing, sex unspecified
14 <NA>  Zita    <NA>         6 110    2022     2 surname missing, sex unspecif~
15 <NA>  Alois   Huber        3 295    2022     2 cm not possible, sex unspecif~
16 F     Martina Huber      399 169    2022     2 age not possible
17 M     Stephan Huber       41 186    2022     0 no error
18 M     Stephan Huber       41 186    2022     1 duplicate
```

```r
# Make a variable that contains the year of birth. Name the variable `born`
# and new dataframe `df`.
df <- df_2022 |>
  mutate(born = time - age)

# Make a new variable that identifies each person by surname, name,
# and their birth born. Name the variable `id`.
df <- df |>
mutate(id = paste(surname, name, born, sep = "_"))

# How many different groups do exist?
df <- df |>
  group_by(id) |>
  mutate(id_num = cur_group_id()) |>
  ungroup()

max(df$id_num)
```

```
[1] 1011
```

```r
# Show groups that exist more than once.
df <- df |>
  group_by(id) |>
  mutate(
    dup_count = row_number(),
    dup_sum   = n()
  ) |>
  ungroup() |>
  arrange(id)

df |> filter(dup_sum > 1)
```

```
# A tibble: 12 x 13
   sex   name    surname    age    cm  time error error_desc  born id    id_num
   <chr> <chr>   <chr>    <dbl> <dbl> <dbl> <dbl> <chr>      <dbl> <chr>  <int>
 1 M     Adonnis Adams       30 192    2022     0 <NA>        1992 Adam~      1
 2 M     Adonnis Adams       30 192    2022     1 duplicate   1992 Adam~      1
 3 F     Tina    Adams        5  98.0  2022     1 duplicate   2017 Adam~     13
 4 F     Tina    Adams        5  98.0  2022     0 <NA>        2017 Adam~     13
 5 F     Abery   Allen       79 157    2022     0 <NA>        1943 Alle~     15
 6 F     Abery   Allen       79 157    2022     1 duplicate   1943 Alle~     15
 7 M     Savier  Campbell    72 192    2022     0 <NA>        1950 Camp~    100
 8 M     Savier  Campbell    72 192    2022     1 duplicate   1950 Camp~    100
 9 M     Stephan Huber       41 186    2022     1 duplicate   1981 Hube~    383
```

```
10 M     Stephan Huber       41 186    2022     0 no error     1981 Hube~    383
11 M     Stephan Huber       41 186    2022     1 duplicate    1981 Hube~    383
12 M     Stephan Huber       41  NA    2022     1 duplicate,~  1981 Hube~    383
# i 2 more variables: dup_count <int>, dup_sum <int>
```

```
df |> get_dupes(name, surname)
```

```
# A tibble: 18 x 14
   name  surname dupe_count sex     age    cm  time error error_desc  born id
   <chr> <chr>        <int> <chr> <dbl> <dbl> <dbl> <dbl> <chr>      <dbl> <chr>
 1 Step~ Huber            4 M        41 186    2022     1 duplicate   1981 Hube~
 2 Step~ Huber            4 M        41 186    2022     0 no error    1981 Hube~
 3 Step~ Huber            4 M        41 186    2022     1 duplicate   1981 Hube~
 4 Step~ Huber            4 M        41  NA    2022     1 duplicate~  1981 Hube~
 5 Abery Allen            2 F        79 157    2022     0 <NA>        1943 Alle~
 6 Abery Allen            2 F        79 157    2022     1 duplicate   1943 Alle~
 7 Adon~ Adams            2 M        30 192    2022     0 <NA>        1992 Adam~
 8 Adon~ Adams            2 M        30 192    2022     1 duplicate   1992 Adam~
 9 Merl~ Miller           2 F        12 153.   2022     0 <NA>        2010 Mill~
10 Merl~ Miller           2 F         2  99.9  2022     0 <NA>        2020 Mill~
11 Rosa  Huber            2 F         9  NA    2022     3 only age ~  2013 Hube~
12 Rosa  Huber            2 <NA>     NA 130    2022     3 age missi~    NA Hube~
13 Savi~ Campbe~          2 M        72 192    2022     0 <NA>        1950 Camp~
14 Savi~ Campbe~          2 M        72 192    2022     1 duplicate   1950 Camp~
15 Step~ huber            2 M       186  41    2022     1 age/cm fa~  1836 hube~
16 Step~ huber            2 <NA>     NA  NA    2022     1 wrong name    NA hube~
17 Tina  Adams            2 F         5  98.0  2022     1 duplicate   2017 Adam~
18 Tina  Adams            2 F         5  98.0  2022     0 <NA>        2017 Adam~
# i 3 more variables: id_num <int>, dup_count <int>, dup_sum <int>
```

```
# Make yourself familiar with the function `get_dupes()` from `janitor` package.
df |> get_dupes()
```

```
No variable names specified - using all columns.

No duplicate combinations found of: sex, name, surname, age, cm, time, error, error_desc,

# A tibble: 0 x 14
# i 14 variables: sex <chr>, name <chr>, surname <chr>, age <dbl>, cm <dbl>,
#   time <dbl>, error <dbl>, error_desc <chr>, born <dbl>, id <chr>,
#   id_num <int>, dup_count <int>, dup_sum <int>, dupe_count <int>
```

```r
df |> get_dupes(surname, name)
```

```
# A tibble: 18 x 14
   surname name  dupe_count sex     age    cm  time error error_desc  born id
   <chr>   <chr>      <int> <chr> <dbl> <dbl> <dbl> <dbl> <chr>      <dbl> <chr>
 1 Huber   Step~          4 M        41 186    2022     1 duplicate   1981 Hube~
 2 Huber   Step~          4 M        41 186    2022     0 no error    1981 Hube~
 3 Huber   Step~          4 M        41 186    2022     1 duplicate   1981 Hube~
 4 Huber   Step~          4 M        41  NA    2022     1 duplicate~  1981 Hube~
 5 Adams   Adon~          2 M        30 192    2022     0 <NA>        1992 Adam~
 6 Adams   Adon~          2 M        30 192    2022     1 duplicate   1992 Adam~
 7 Adams   Tina           2 F         5  98.0  2022     1 duplicate   2017 Adam~
 8 Adams   Tina           2 F         5  98.0  2022     0 <NA>        2017 Adam~
 9 Allen   Abery          2 F        79 157    2022     0 <NA>        1943 Alle~
10 Allen   Abery          2 F        79 157    2022     1 duplicate   1943 Alle~
11 Campbe~ Savi~          2 M        72 192    2022     0 <NA>        1950 Camp~
12 Campbe~ Savi~          2 M        72 192    2022     1 duplicate   1950 Camp~
13 Huber   Rosa           2 F         9  NA    2022     3 only age ~  2013 Hube~
14 Huber   Rosa           2 <NA>     NA 130    2022     3 age missi~    NA Hube~
15 Miller  Merl~          2 F        12 153.   2022     0 <NA>        2010 Mill~
16 Miller  Merl~          2 F         2  99.9  2022     0 <NA>        2020 Mill~
17 huber   Step~          2 M       186  41    2022     1 age/cm fa~  1836 hube~
18 huber   Step~          2 <NA>     NA  NA    2022     1 wrong name    NA hube~
# i 3 more variables: id_num <int>, dup_count <int>, dup_sum <int>
```

```r
df |> get_dupes(id)
```

```
# A tibble: 12 x 14
   id    dupe_count sex   name  surname   age    cm  time error error_desc  born
   <chr>      <int> <chr> <chr> <chr>   <dbl> <dbl> <dbl> <dbl> <chr>      <dbl>
 1 Hube~          4 M     Step~ Huber      41 186    2022     1 duplicate   1981
 2 Hube~          4 M     Step~ Huber      41 186    2022     0 no error    1981
 3 Hube~          4 M     Step~ Huber      41 186    2022     1 duplicate   1981
 4 Hube~          4 M     Step~ Huber      41  NA    2022     1 duplicate~  1981
 5 Adam~          2 M     Adon~ Adams      30 192    2022     0 <NA>        1992
 6 Adam~          2 M     Adon~ Adams      30 192    2022     1 duplicate   1992
 7 Adam~          2 F     Tina  Adams       5  98.0  2022     1 duplicate   2017
 8 Adam~          2 F     Tina  Adams       5  98.0  2022     0 <NA>        2017
 9 Alle~          2 F     Abery Allen      79 157    2022     0 <NA>        1943
10 Alle~          2 F     Abery Allen      79 157    2022     1 duplicate   1943
11 Camp~          2 M     Savi~ Campbe~    72 192    2022     0 <NA>        1950
12 Camp~          2 M     Savi~ Campbe~    72 192    2022     1 duplicate   1950
# i 3 more variables: id_num <int>, dup_count <int>, dup_sum <int>
```

```r
df_uni <- df |>
  arrange() |>
  distinct(id, .keep_all = TRUE)

df_uni_b <- df |>
  arrange(desc(dup_count)) |>
  distinct(id, .keep_all = TRUE)

anti_join(df, df_uni)
```

```
Joining with `by = join_by(sex, name, surname, age, cm, time, error,
error_desc, born, id, id_num, dup_count, dup_sum)`
```

```
# A tibble: 7 x 13
  sex   name     surname     age    cm  time error error_desc     born id    id_num
  <chr> <chr>    <chr>     <dbl> <dbl> <dbl> <dbl> <chr>         <dbl> <chr>  <int>
1 M     Adonnis  Adams        30 192    2022     1 duplicate      1992 Adam~      1
2 F     Tina     Adams         5  98.0  2022     0 <NA>           2017 Adam~     13
3 F     Abery    Allen        79 157    2022     1 duplicate      1943 Alle~     15
4 M     Savier   Campbell     72 192    2022     1 duplicate      1950 Camp~    100
5 M     Stephan  Huber        41 186    2022     0 no error       1981 Hube~    383
6 M     Stephan  Huber        41 186    2022     1 duplicate      1981 Hube~    383
7 M     Stephan  Huber        41  NA    2022     1 duplicate, ~   1981 Hube~    383
# i 2 more variables: dup_count <int>, dup_sum <int>
```

```r
anti_join(df, df_uni_b)
```

```
Joining with `by = join_by(sex, name, surname, age, cm, time, error,
error_desc, born, id, id_num, dup_count, dup_sum)`
```

```
# A tibble: 7 x 13
  sex   name     surname     age    cm  time error error_desc  born id       id_num
  <chr> <chr>    <chr>     <dbl> <dbl> <dbl> <dbl> <chr>      <dbl> <chr>     <int>
1 M     Adonnis  Adams        30 192    2022     0 <NA>        1992 Adams_~      1
2 F     Tina     Adams         5  98.0  2022     1 duplicate   2017 Adams_~     13
3 F     Abery    Allen        79 157    2022     0 <NA>        1943 Allen_~     15
4 M     Savier   Campbell     72 192    2022     0 <NA>        1950 Campbe~    100
5 M     Stephan  Huber        41 186    2022     1 duplicate   1981 Huber_~    383
6 M     Stephan  Huber        41 186    2022     0 no error    1981 Huber_~    383
7 M     Stephan  Huber        41 186    2022     1 duplicate   1981 Huber_~    383
# i 2 more variables: dup_count <int>, dup_sum <int>
# unload packages
suppressMessages(pacman::p_unload(tidyverse, janitor, babynames, stringr))
```

2.2 exe_import_covid.R

```
# Solution to excercise "Import data":

# load packages
if (!require(pacman)) install.packages("pacman")
pacman::p_load(tibble)

state <- c("BY", "NRW", "BW")
deaths <- c(4.92, 5.32, 3.69)
cases <- c(24111, 25466, 16145)
df_covid <- data.frame(state, deaths)
tbl_covid <- tibble(state, deaths)

suppressMessages(pacman::p_unload(tibble))
```

2.3 exe_genanddrop.R

```
# Generate and drop variables
# exe_genanddrop.R
# Stephan Huber; 2023-05-09

# setwd("/home/sthu/Dropbox/hsf/test")
rm(list=ls())

# load packages
if (!require(pacman)) install.packages("pacman")
pacman::p_load(datasets, tidyverse)

# a)
mtcars_new <- mtcars |>
  rownames_to_column(var = "car") |>
  as_tibble() |>
  mutate(d_cyl_6to8 = if_else(cyl > 6, 1, 0))
mtcars_new
```

```
# A tibble: 32 x 13
   car           mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
   <chr>       <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 Mazda RX4    21       6  160    110  3.9   2.62  16.5     0     1     4     4
 2 Mazda RX4 ~  21       6  160    110  3.9   2.88  17.0     0     1     4     4
 3 Datsun 710   22.8     4  108     93  3.85  2.32  18.6     1     1     4     1
 4 Hornet 4 D~  21.4     6  258    110  3.08  3.22  19.4     1     0     3     1
 5 Hornet Spo~  18.7     8  360    175  3.15  3.44  17.0     0     0     3     2
 6 Valiant      18.1     6  225    105  2.76  3.46  20.2     1     0     3     1
 7 Duster 360   14.3     8  360    245  3.21  3.57  15.8     0     0     3     4
 8 Merc 240D    24.4     4  147.    62  3.69  3.19  20       1     0     4     2
 9 Merc 230     22.8     4  141.    95  3.92  3.15  22.9     1     0     4     2
10 Merc 280     19.2     6  168.   123  3.92  3.44  18.3     1     0     4     4
# i 22 more rows
# i 1 more variable: d_cyl_6to8 <dbl>
```

```
# b)
mtcars_new <- mtcars_new |>
  mutate(posercar = if_else(cyl > 6 & mpg < 18, 1, 0))
mtcars_new
```

```
# A tibble: 32 x 14
   car           mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
   <chr>       <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 Mazda RX4    21       6  160    110  3.9   2.62  16.5     0     1     4     4
 2 Mazda RX4 ~  21       6  160    110  3.9   2.88  17.0     0     1     4     4
 3 Datsun 710   22.8     4  108     93  3.85  2.32  18.6     1     1     4     1
 4 Hornet 4 D~  21.4     6  258    110  3.08  3.22  19.4     1     0     3     1
 5 Hornet Spo~  18.7     8  360    175  3.15  3.44  17.0     0     0     3     2
 6 Valiant      18.1     6  225    105  2.76  3.46  20.2     1     0     3     1
 7 Duster 360   14.3     8  360    245  3.21  3.57  15.8     0     0     3     4
 8 Merc 240D    24.4     4  147.    62  3.69  3.19  20       1     0     4     2
 9 Merc 230     22.8     4  141.    95  3.92  3.15  22.9     1     0     4     2
10 Merc 280     19.2     6  168.   123  3.92  3.44  18.3     1     0     4     4
# i 22 more rows
# i 2 more variables: d_cyl_6to8 <dbl>, posercar <dbl>
```

```
# c)
mtcars_new <- mtcars_new |>
  select(-d_cyl_6to8)
mtcars_new
```

```
# A tibble: 32 x 13
   car           mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
   <chr>       <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 Mazda RX4    21       6  160    110  3.9   2.62  16.5     0     1     4     4
 2 Mazda RX4 ~  21        6  160    110  3.9   2.88  17.0     0     1     4     4
 3 Datsun 710   22.8      4  108     93  3.85  2.32  18.6     1     1     4     1
 4 Hornet 4 D~  21.4      6  258    110  3.08  3.22  19.4     1     0     3     1
 5 Hornet Spo~  18.7      8  360    175  3.15  3.44  17.0     0     0     3     2
 6 Valiant      18.1      6  225    105  2.76  3.46  20.2     1     0     3     1
 7 Duster 360   14.3      8  360    245  3.21  3.57  15.8     0     0     3     4
 8 Merc 240D    24.4      4  147.    62  3.69  3.19  20       1     0     4     2
 9 Merc 230     22.8      4  141.    95  3.92  3.15  22.9     1     0     4     2
10 Merc 280     19.2      6  168.   123  3.92  3.44  18.3     1     0     4     4
# i 22 more rows
# i 1 more variable: posercar <dbl>
```

```r
# unload packages
suppressMessages(pacman::p_unload(datasets, tidyverse))
```

2.4 exe_base_pipe.R

```r
# Base R or pipe
# exe_base_pipe.R
# Stephan Huber; 2023-05-08

# setwd("/home/sthu/Dropbox/hsf/test")
rm(list=ls())

# load packages
if (!require(pacman)) install.packages("pacman")
pacman::p_load(datasets, tidyverse)

# a)
# Using the pipe |>
# Select rows where cyl is 4 or 6 and wt is less than 3.5
df1 <- mtcars |>
  filter(cyl %in% c(4, 6) & wt < 3.5)
df1
```

```
                mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710     22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive 21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Valiant        18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Merc 240D      24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230       22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 280       19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Merc 280C      17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Fiat 128       32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Honda Civic    30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla 33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
Toyota Corona  21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Fiat X1-9      27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Porsche 914-2  26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Lotus Europa   30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Ferrari Dino   19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Volvo 142E     21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

```
# Without the pipe |>
# Select rows where cyl is 4 or 6 and wt is less than 3.5
df2 <- subset(mtcars, cyl %in% c(4, 6) & wt < 3.5)
df2
```

```
                mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710     22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive 21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Valiant        18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Merc 240D      24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230       22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 280       19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Merc 280C      17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Fiat 128       32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Honda Civic    30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla 33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
Toyota Corona  21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Fiat X1-9      27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Porsche 914-2  26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Lotus Europa   30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Ferrari Dino   19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Volvo 142E     21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

```
# Check if the resulting dataframe is identical to the expected output
identical(df1, df2)
```

```
[1] TRUE
```

```r
# b)
# Using the pipe |> and tidyverse (mutate)
df3 <- mtcars |>
  mutate(cyl_4_or_6 =
           if_else(cyl %in% c(4, 6) & wt < 3.5, TRUE, FALSE))
df3
```

```
                     mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4           21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag       21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710          22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive      21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout   18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Valiant             18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Duster 360          14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Merc 240D           24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230            22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 280            19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Merc 280C           17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Merc 450SE          16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
Merc 450SL          17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
Merc 450SLC         15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
Fiat 128            32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
                    cyl_4_or_6
Mazda RX4                 TRUE
Mazda RX4 Wag             TRUE
Datsun 710                TRUE
Hornet 4 Drive            TRUE
Hornet Sportabout        FALSE
Valiant                   TRUE
Duster 360               FALSE
Merc 240D                 TRUE
Merc 230                  TRUE
Merc 280                  TRUE
Merc 280C                 TRUE
Merc 450SE               FALSE
Merc 450SL               FALSE
Merc 450SLC              FALSE
Cadillac Fleetwood       FALSE
Lincoln Continental      FALSE
Chrysler Imperial        FALSE
Fiat 128                  TRUE
Honda Civic               TRUE
Toyota Corolla            TRUE
Toyota Corona             TRUE
Dodge Challenger         FALSE
AMC Javelin              FALSE
Camaro Z28               FALSE
Pontiac Firebird         FALSE
Fiat X1-9                 TRUE
Porsche 914-2             TRUE
Lotus Europa              TRUE
Ford Pantera L           FALSE
Ferrari Dino              TRUE
Maserati Bora            FALSE
Volvo 142E                TRUE
```

```r
# without pipe and with base R (transform)
df4 <- mtcars
df4$cyl_4_or_6 <- with(mtcars, cyl %in% c(4, 6) & wt < 3.5)

# Alternatively in one line:
df5 <- transform(mtcars, cyl_4_or_6 = cyl %in% c(4,6) & wt < 3.5)

# Check if the resulting dataframe is identical to the expected output
identical(df3, df4)
```

```
[1] TRUE
```

```r
identical(df3, df5)
```

```
[1] TRUE
```

```
# unload packages
suppressMessages(pacman::p_unload(datasets, tidyverse))
```

2.5 exe_subset.R

```
# Subsetting with \R
# exe_subset.R
# Stephan Huber; 2022-06-07

# setwd("/home/sthu/Dropbox/hsf/22-ss/dsda/work/")
rm(list=ls())




# 0
# load packages
if (!require(pacman)) install.packages("pacman")
pacman::p_load(tidyverse, dplyr, tibble)

# 1
mtcars
```

```
                     mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4           21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag       21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710          22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive      21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout   18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Valiant             18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Duster 360          14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Merc 240D           24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230            22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 280            19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Merc 280C           17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Merc 450SE          16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
Merc 450SL          17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
Merc 450SLC         15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
Fiat 128            32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

```
# 2
cars <- mtcars

# 3
class(cars)
```

```
[1] "data.frame"
```

```
# 4
dim(cars)
```

```
[1] 32 11
```

```
# Alternative
ncol(cars)
```

```
[1] 11
```

```
nrow(cars)
```

```
[1] 32
```

```
# 5
cars <- rename(cars, MPG = mpg)

# 6
cars <- rename_all(cars, toupper)
# if you like lower cases:
# cars <- rename_all(cars, tolower)
```

```r
# 7
cars <- rownames_to_column(mtcars, var = "car")

# 8
pvars <- select(cars, car, ends_with("p"))

# 9
carsSub <- select(cars, car, wt, qsec, hp)

# 10
dim(carsSub)
```

```
[1] 32  4
```

```r
# 11
carsSub <- rename_all(carsSub, toupper)

# 12
cars_mpg <- filter(cars, mpg > 20)
dim(cars_mpg)
```

```
[1] 14 12
```

```r
# 13
cars_whattever <- filter(cars, mpg < 16 & hp > 100)

# 14
carsSub <- filter(cars, cyl == 8)
carsSub <- select(carsSub, wt, qsec, hp, car)
dim(carsSub)
```

```
[1] 14  4
```

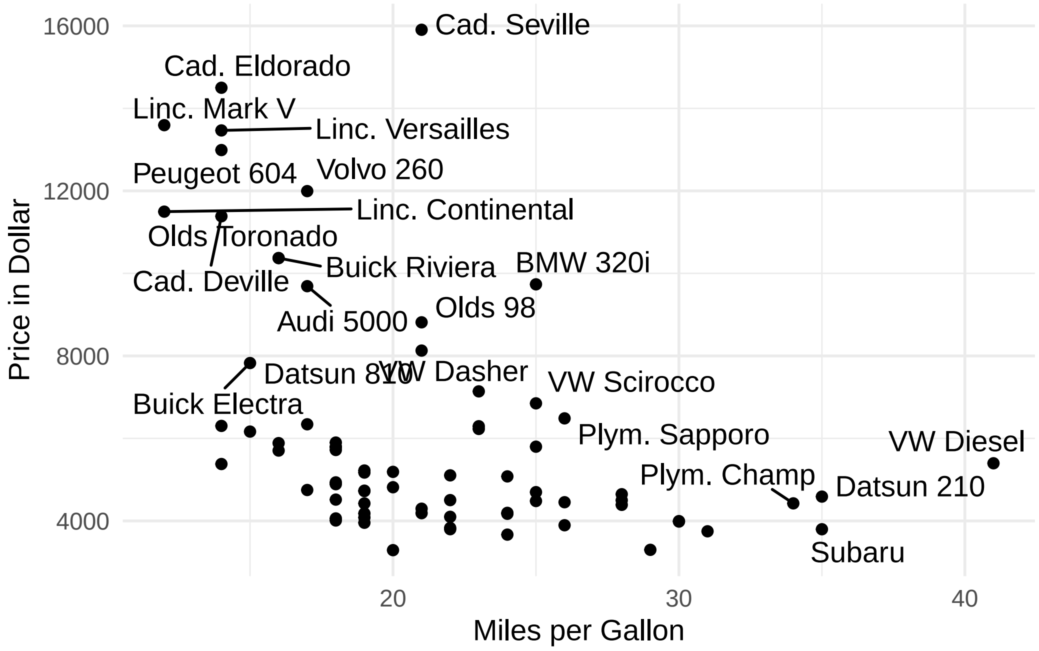
```r
# 15
# Alternative with pipe operator:
carsSub <- cars %>%
  filter(cyl == 8) %>%
  select(wt, qsec, hp, car)

# 16
carsSub <- arrange(carsSub, wt)
```

```r
# 17
carsSub <- carsSub %>%
  mutate(wt2 = wt^2)

# Alternatively you can put everything into one pipe:
carsSub2 <- cars %>%
  filter(cyl == 8) %>%
  select(wt, qsec, hp, car) %>%
  arrange(carsSub, wt) %>%
  mutate(wt2 = wt^2)


# unload packages
suppressMessages(pacman::p_unload(tidyverse, dplyr, tibble))
```

2.6 exe_poser.R

```r
# Load the required libraries
if (!require(pacman)) install.packages("pacman")
pacman::p_load(tidyverse, haven, ggrepel)

# setwd("~/Dropbox/hsf/23-ws/R_begin")

rm(list = ls())

# Read the Stata dataset
auto <- read_dta("http://www.stata-press.com/data/r8/auto.dta")


# Create a scatter plot of price vs. weight
scatter_plot <- ggplot(auto, aes(x = mpg, y = price, label = make)) +
  geom_point() +
  geom_text_repel() +
  xlab("Miles per Gallon") +
  ylab("Price in Dollar") +
  theme_minimal()

scatter_plot
```

```
Warning: ggrepel: 52 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```



```
# Save the scatter plot in different formats
ggsave("scatter_plot.png", plot = scatter_plot, device = "png")
```

```
Saving 5.5 x 3.5 in image

Warning: ggrepel: 52 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

```
ggsave("scatter_plot.pdf", plot = scatter_plot, device = "pdf")
```

```
Saving 5.5 x 3.5 in image

Warning: ggrepel: 52 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

```
# Create 'lp100km' variable for fuel consumption
n_auto <- auto %>%
  mutate(lp100km = (1/(mpg * 1.6/ 3.8))  * 100)

# Create 'larger6000' dummy variable
n_auto <- n_auto %>%
  mutate(larger6000 = ifelse(price > 6000, 1, 0))
```

```
n_auto <- n_auto |>
  filter(larger6000 == 0)

# Normalize variables

## Do it slowly
n_auto <- n_auto |>
  mutate(sprice = ( price - min(auto$price) )/( max(auto$price) - min(auto$price) ) )

## Do it with a self-written function
min_max_norm <- function(x) {
  (x - min(x, na.rm = TRUE)) / (max(x, na.rm = TRUE) - min(x, na.rm = TRUE))
}

n_auto <- n_auto |>
  mutate(smpg = min_max_norm(mpg)) |>
  mutate(sturn = min_max_norm(turn)) |>
  mutate(slp100km = min_max_norm(lp100km)) |>
  mutate(sprice = min_max_norm(price)) |>
  mutate(srep78 = min_max_norm(rep78))

## With a loop:

# vars_to_normalize <- c("mpg", "turn", "lp100km", "price", "rep78")
#
# # Loop through the selected variables and apply min_max_norm
# for (var in c("mpg", "turn", "lp100km", "price", "rep78")) {
#   auto <- auto |>
#     mutate(!!paste0("s", var) := min_max_norm(!!sym(var))) |>
#     select(make, starts_with("s"))
# }

## mpg and rep78 need to be changed because a SMALL value is poser-like
n_auto <- n_auto |>
  mutate(smpg = 1-smpg) |>
  mutate(srep78 = 1-srep78)

## create the poser (composite) indicator
n_auto <- n_auto |>
  mutate(poser = (sturn+smpg+sprice+srep78) / 4 )

## filter results
n_auto |>
  arrange(desc(poser)) |>
  select(make, poser) |>
  head(5)
```

```
# A tibble: 5 x 2
  make             poser
  <chr>            <dbl>
1 Dodge Magnum     0.888
2 Pont. Firebird   0.782
3 Merc. Cougar     0.763
4 Buick LeSabre    0.754
5 Pont. Grand Prix 0.720
```

```
df_poser <- n_auto |>
  filter(larger6000 == 0) |>
  arrange(desc(poser)) |>
  select(make, poser) |>
  na.omit()

# Five top poser cars
head(df_poser, 15)
```

```
# A tibble: 15 x 2
   make              poser
   <chr>             <dbl>
 1 Dodge Magnum      0.888
 2 Pont. Firebird    0.782
 3 Merc. Cougar      0.763
 4 Buick LeSabre     0.754
 5 Pont. Grand Prix  0.720
 6 Chev. Impala      0.702
 7 Dodge Diplomat    0.690
 8 Chev. Monte Carlo 0.684
 9 Pont. Catalina    0.678
10 Olds Cutl Supr    0.671
11 Plym. Volare      0.665
12 Buick Regal       0.663
13 Olds Cutlass      0.629
14 Olds Starfire     0.626
15 AMC Pacer         0.619
```

```
# Five top non-poser cars
tail(df_poser, 5)
```

```
# A tibble: 5 x 2
```

```
  make           poser
  <chr>          <dbl>
1 VW Diesel      0.261
2 Dodge Colt     0.227
3 Toyota Corolla 0.195
4 Datsun 210     0.195
5 Subaru         0.178
```

```
# unload packages
suppressMessages(pacman::p_unload(tidyverse, haven, ggrepel))
```

2.7 exe_datasauRus.R

```
# setwd("/home/sthu/Dropbox/hsf/23-ws/ds_mim/")
rm(list = ls())

# Load the packages datasauRus and tidyverse. If necessary, install these packages.

# load packages
if (!require(pacman)) install.packages("pacman")
pacman::p_load(datasauRus, tidyverse)

# The packagedatasauRus comes with a dataset in two different formats:
#   datasaurus_dozen and datasaurus_dozen_wide. Store them as ds and ds_wide.

ds <- datasaurus_dozen
ds_wide <- datasaurus_dozen_wide

# Open and read the R vignette of the datasauRus package.
#   Also open the R documentation of the dataset datasaurus_dozen.

??datasaurus

# Explore the dataset: What are the dimensions of this dataset? Look at the descriptive st

ds
```

```
# A tibble: 1,846 x 3
   dataset     x     y
   <chr>   <dbl> <dbl>
 1 dino     55.4  97.2
 2 dino     51.5  96.0
 3 dino     46.2  94.5
 4 dino     42.8  91.4
 5 dino     40.8  88.3
 6 dino     38.7  84.9
 7 dino     35.6  79.9
 8 dino     33.1  77.6
 9 dino     29.0  74.5
10 dino     26.2  71.4
# i 1,836 more rows
```

```
dim(ds)
```

```
[1] 1846    3
```

```
head(ds)
```

```
# A tibble: 6 x 3
  dataset     x     y
  <chr>   <dbl> <dbl>
1 dino     55.4  97.2
2 dino     51.5  96.0
3 dino     46.2  94.5
4 dino     42.8  91.4
5 dino     40.8  88.3
6 dino     38.7  84.9
```

```
glimpse(ds)
```

```
Rows: 1,846
Columns: 3
$ dataset <chr> "dino", "dino", "dino", "dino", "dino", "dino", "dino", "dino"~
$ x       <dbl> 55.3846, 51.5385, 46.1538, 42.8205, 40.7692, 38.7179, 35.6410,~
$ y       <dbl> 97.1795, 96.0256, 94.4872, 91.4103, 88.3333, 84.8718, 79.8718,~
```

```
view(ds)
summary(ds)
```

```
   dataset                x                y           
 Length:1846        Min.   :15.56   Min.   : 0.01512  
 Class :character   1st Qu.:41.07   1st Qu.:22.56107  
 Mode  :character   Median :52.59   Median :47.59445  
                    Mean   :54.27   Mean   :47.83510  
                    3rd Qu.:67.28   3rd Qu.:71.81078  
                    Max.   :98.29   Max.   :99.69468  
```

```
# How many unique values does the variable dataset of the tibble ds have?
#   Hint: The function unique() return the unique values of a variable and the
#   function length() returns the length of a vector, such as the unique elements.

unique(ds$dataset)
```

```
 [1] "dino"       "away"       "h_lines"    "v_lines"    "x_shape"
 [6] "star"       "high_lines" "dots"       "circle"     "bullseye"
[11] "slant_up"   "slant_down" "wide_lines"
```

```
unique(ds$dataset) |>
  length()
```

```
[1] 13
```

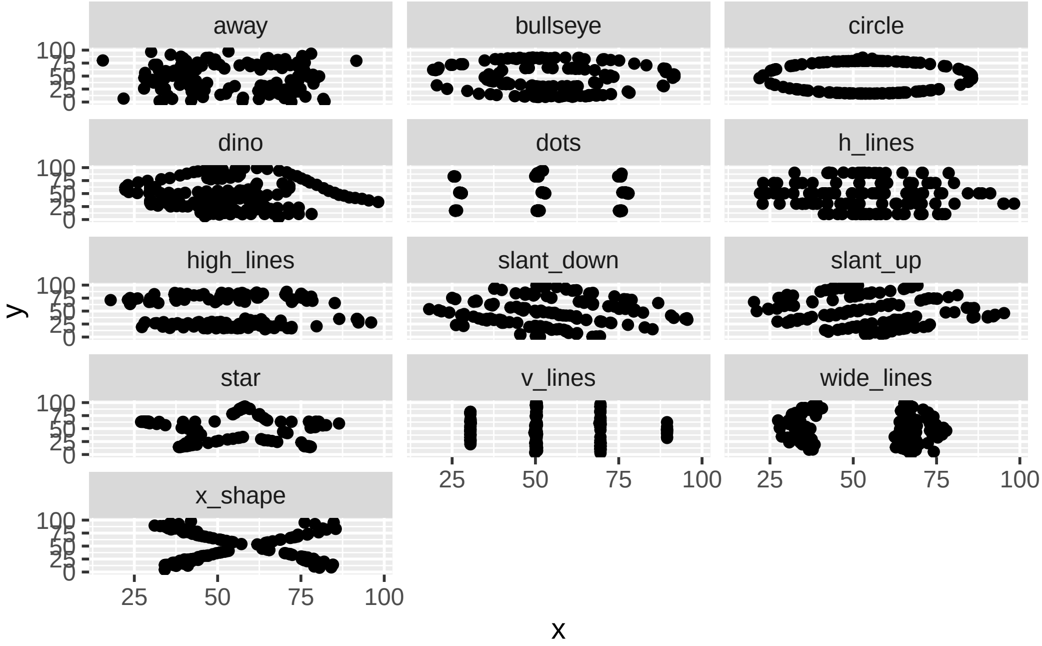
```
# Compute the mean values of the x and y variables for each entry in dataset.
#   Hint: Use the group_by() function to group the data by the appropriate column and
#   then the summarise() function to calculate the mean.

ds |>
  group_by(dataset) |>
  summarise(mean_x = mean(x),
            mean_y = mean(y))
```

```
# A tibble: 13 x 3
   dataset    mean_x mean_y
   <chr>       <dbl>  <dbl>
 1 away         54.3   47.8
 2 bullseye     54.3   47.8
 3 circle       54.3   47.8
 4 dino         54.3   47.8
 5 dots         54.3   47.8
 6 h_lines      54.3   47.8
 7 high_lines   54.3   47.8
 8 slant_down   54.3   47.8
 9 slant_up     54.3   47.8
10 star         54.3   47.8
11 v_lines      54.3   47.8
12 wide_lines   54.3   47.8
13 x_shape      54.3   47.8
```

```
# Compute the standard deviation, the correlation, and the median in the same way. Round t

ds |>
  group_by(dataset) |>
  summarise(mean_x = round(mean(x),2),
            mean_y = round(mean(y),2),
            sd_x = round(sd(x),2),
            sd_y = round(sd(y),2),
            med_x = round(median(x),2),
            med_y = round(median(y),2),
            cor = round(cor(x,y), digits = 4))
```

```
# A tibble: 13 x 8
   dataset    mean_x mean_y  sd_x  sd_y med_x med_y     cor
   <chr>       <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>   <dbl>
 1 away         54.3   47.8  16.8  26.9  53.3  47.5 -0.0641
 2 bullseye     54.3   47.8  16.8  26.9  53.8  47.4 -0.0686
 3 circle       54.3   47.8  16.8  26.9  54.0  51.0 -0.0683
 4 dino         54.3   47.8  16.8  26.9  53.3  46.0 -0.0645
 5 dots         54.3   47.8  16.8  26.9  51.0  51.3 -0.0603
 6 h_lines      54.3   47.8  16.8  26.9  53.1  50.5 -0.0617
 7 high_lines   54.3   47.8  16.8  26.9  54.2  32.5 -0.0685
 8 slant_down   54.3   47.8  16.8  26.9  53.1  46.4 -0.069
 9 slant_up     54.3   47.8  16.8  26.9  54.3  45.3 -0.0686
10 star         54.3   47.8  16.8  26.9  56.5  50.1 -0.063
11 v_lines      54.3   47.8  16.8  26.9  50.4  47.1 -0.0694
12 wide_lines   54.3   47.8  16.8  26.9  64.6  46.3 -0.0666
13 x_shape      54.3   47.8  16.8  26.9  47.1  39.9 -0.0656
```

```
# What can you conclude?
#   --> The standard deviation, the mean, and the correlation are basically the
#   same for all datasets. The median is different.

# Plot all datasets of ds. Hide the legend. Hint: Use the facet_wrap() and the theme() fun

ggplot(ds, aes(x = x, y = y)) +
  geom_point() +
  facet_wrap(~ dataset, ncol = 3) +
  theme(legend.position = "none")
```



```r
# Create a loop that generates separate scatter plots for each unique datatset of the tibb
#   Export each graph as a png file.

# Assuming uni_ds is a vector of unique values for the 'dataset' variable
uni_ds <- unique(ds$dataset)

# Create the 'pic' folder if it doesn't exist
if (!dir.exists("pic")) {
  dir.create("pic")
}

for (uni_v in uni_ds) {
  # Select data for the current value
  subset_ds <- ds |>
    filter(dataset == uni_v) %>%
    select(x, y)

  # Make plot
  graph <- ggplot(subset_ds, aes(x = x, y = y)) +
    geom_point() +
    labs(title = paste("Dataset:", uni_v),
         x = "X",
         y = "Y") +
    theme_bw()

  # Save the plot as a PNG file
  filename <- paste0("pic/", "plot_ds_", uni_v, ".png")
  ggsave(filename, plot = graph)
}
```

```
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image
```

```
# unload packages
suppressMessages(pacman::p_unload(datasauRus, tidyverse))
```

2.8 exe_convergence.R

```
# Convergence

# set working directory
# setwd("/home/sthu/Dropbox/hsf/github/courses/")


# clear the environment
rm(list = ls())

# some packages needed install.packages(...) and load packages library(...)

# Let us do the following:
  # 1. check if a package is installed
  # 2. if not installed the package should be installed and loaded
  # 3. if installed the package should be loaded
# I like to do it with a function that is part of pacman package:


# load packages
```

```
if (!require(pacman)) install.packages("pacman")
pacman::p_load(haven, tidyverse, vtable, gtsummary, pastecs, Hmisc,
               sjlabelled, tis, ggpubr, sjPlot, psych)

# an alternative is to install and load it like that
# install.packages(c("haven", "tidyverse", "vtable", "gtsummary", "pastecs"))
# library(c("haven", "tidyverse", "vtable", "gtsummary", "pastecs"))

# import data
data <- read_dta("https://github.com/hubchev/courses/raw/main/dta/convergence.dta")

# inspect data
names(data)
```

```
 [1] "country" "gdppc60" "gdppc65" "gdppc70" "gdppc75" "gdppc80" "gdppc85"
 [8] "gdppc90" "gdppc95" "africa"  "asia"    "weurope" "growth"
```

```
str(data)
```

```
tibble [107 x 13] (S3: tbl_df/tbl/data.frame)
 $ country: chr [1:107] "Algeria" "Angola" "Argentina" "Australia" ...
  ..- attr(*, "format.stata")= chr "%24s"
 $ gdppc60: num [1:107] 2848 2642 7879 11436 7842 ...
  ..- attr(*, "label")= chr "real gdp per capita 1960"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ gdppc65: num [1:107] 3536 3072 8802 13192 9387 ...
  ..- attr(*, "label")= chr "real gdp per capita 1965"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ gdppc70: num [1:107] 3670 3558 9903 15842 11946 ...
  ..- attr(*, "label")= chr "real gdp per capita 1970"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ gdppc75: num [1:107] 3917 2230 10609 16716 14198 ...
  ..- attr(*, "label")= chr "real gdp per capita 1975"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ gdppc80: num [1:107] 5094 2059 11359 18300 16869 ...
  ..- attr(*, "label")= chr "real gdp per capita 1980"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ gdppc85: num [1:107] 5876 1988 9246 19669 17919 ...
  ..- attr(*, "label")= chr "real gdp per capita 1985"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ gdppc90: num [1:107] 5307 2081 7716 21446 21178 ...
  ..- attr(*, "label")= chr "real gdp per capita 1990"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ gdppc95: num [1:107] 4935 1339 10973 23827 22474 ...
```

```
  ..- attr(*, "label")= chr "real gdp per capita 1995"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ africa : num [1:107] 1 1 0 0 0 0 0 0 1 0 ...
  ..- attr(*, "label")= chr "=1 if in Africa"
  ..- attr(*, "format.stata")= chr "%8.0g"
 $ asia   : num [1:107] 0 0 0 0 0 1 0 0 0 0 ...
  ..- attr(*, "label")= chr "=1 if in Asia"
  ..- attr(*, "format.stata")= chr "%8.0g"
 $ weurope: num [1:107] 0 0 0 0 0 0 0 1 0 0 ...
  ..- attr(*, "label")= chr "=1 if in Western Europe"
  ..- attr(*, "format.stata")= chr "%8.0g"
 $ growth : num [1:107] 0.55 -0.68 0.331 0.734 1.053 ...
  ..- attr(*, "format.stata")= chr "%9.0g"
```

```
data
```

```
# A tibble: 107 x 13
   country    gdppc60 gdppc65 gdppc70 gdppc75 gdppc80 gdppc85 gdppc90 gdppc95
   <chr>        <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
 1 Algeria      2848.   3536.   3670.   3917.   5094.   5876.   5307.   4935.
 2 Angola       2642.   3072.   3558.   2230.   2059.   1988.   2081.   1339.
 3 Argentina    7879.   8802.   9903.  10609.  11359.   9246.   7716.  10973.
 4 Australia   11436.  13192.  15842.  16716.  18300.  19669.  21446.  23827.
 5 Austria      7842.   9387.  11946.  14198.  16869.  17919.  21178.  22474.
 6 Bangladesh   1130.   1164.   1181.   1030.   1040.   1245.   1366.   1568.
 7 Barbados     3632.   4632.   6456.   8827.  10911.  11090.  14411.  14636.
 8 Belgium      8314.  10454.  12980.  15024.  17451.  18109.  21246.  22356.
 9 Benin        1140.   1188.   1170.   1048.   1069.   1252.   1069.   1139.
10 Bolivia      2516.   2880.   2670.   3124.   3264.   2718.   2615.   2795.
# i 97 more rows
# i 4 more variables: africa <dbl>, asia <dbl>, weurope <dbl>, growth <dbl>
```

```
head(data)
```

```
# A tibble: 6 x 13
  country gdppc60 gdppc65 gdppc70 gdppc75 gdppc80 gdppc85 gdppc90 gdppc95 africa
  <chr>     <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>  <dbl>
1 Algeria   2848.   3536.   3670.   3917.   5094.   5876.   5307.   4935.      1
2 Angola    2642.   3072.   3558.   2230.   2059.   1988.   2081.   1339.      1
3 Argent~   7879.   8802.   9903.  10609.  11359.   9246.   7716.  10973.      0
4 Austra~  11436.  13192.  15842.  16716.  18300.  19669.  21446.  23827.      0
5 Austria   7842.   9387.  11946.  14198.  16869.  17919.  21178.  22474.      0
6 Bangla~   1130.   1164.   1181.   1030.   1040.   1245.   1366.   1568.      0
# i 3 more variables: asia <dbl>, weurope <dbl>, growth <dbl>
```

```r
tail(data)
```

```
# A tibble: 6 x 13
  country gdppc60 gdppc65 gdppc70 gdppc75 gdppc80 gdppc85 gdppc90 gdppc95 africa
  <chr>     <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>  <dbl>
1 United~  10341.  11633.  12917.  14072.  15302.  16878.  19585.  20963.      0
2 United~  13118.  15697.  17478.  19284.  22806.  25251.  28281.  30366.      0
3 Uruguay   6279.   5936.   6553.   6949.   8580.   6625.   7763.   9399.      0
4 Venezu~   8381.  10618.  11253.   8815.   8516.   7274.   7431.   7582.      0
5 Zambia    1290.   1564.   1427.   1446.   1324.   1167.   1091.    870.      1
6 Zimbab~   1317.   1539.   2303.   2694.   2816.   2923.   3115.   2832.      1
# i 3 more variables: asia <dbl>, weurope <dbl>, growth <dbl>
```

```r
summary(data)
```

```
   country             gdppc60           gdppc65           gdppc70       
 Length:107         Min.   :  407.8   Min.   :  513.6   Min.   :  354.5  
 Class :character   1st Qu.: 1153.2   1st Qu.: 1364.5   1st Qu.: 1488.0  
 Mode  :character   Median : 2484.7   Median : 2884.4   Median : 3072.2  
                    Mean   : 3634.3   Mean   : 4367.5   Mean   : 5128.4  
                    3rd Qu.: 4354.0   3rd Qu.: 5873.3   3rd Qu.: 6994.6  
                    Max.   :16010.3   Max.   :18928.9   Max.   :22030.9  
    gdppc75           gdppc80           gdppc85           gdppc90       
 Min.   :  617.9   Min.   :  473.6   Min.   :  542.3   Min.   :  527.7  
 1st Qu.: 1480.7   1st Qu.: 1708.6   1st Qu.: 1598.8   1st Qu.: 1829.0  
 Median : 3741.7   Median : 4306.2   Median : 4200.7   Median : 4034.0  
 Mean   : 5759.1   Mean   : 6553.6   Mean   : 6900.3   Mean   : 7775.1  
 3rd Qu.: 8355.8   3rd Qu.: 9968.6   3rd Qu.:10037.2   3rd Qu.:11716.2  
 Max.   :21808.9   Max.   :23860.1   Max.   :25251.4   Max.   :28744.1  
    gdppc95            africa            asia           weurope      
 Min.   :  499.3   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
 1st Qu.: 1673.7   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000  
 Median : 4467.9   Median :0.0000   Median :0.0000   Median :0.0000  
 Mean   : 8468.2   Mean   :0.3738   Mean   :0.1308   Mean   :0.1402  
 3rd Qu.:13627.8   3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:0.0000  
 Max.   :36741.1   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000  
     growth       
 Min.   :-0.6888  
 1st Qu.: 0.2458  
 Median : 0.6587  
 Mean   : 0.6345  
 3rd Qu.: 1.0505  
 Max.   : 2.3493  
```

```r
view(data)

#library(vtable)
# vtable(data, missing=TRUE)

# library(pastecs)
stat.desc(data)
```

```
         country      gdppc60      gdppc65      gdppc70      gdppc75
nbr.val       NA 1.070000e+02 1.070000e+02 1.070000e+02 1.070000e+02
nbr.null      NA 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
nbr.na        NA 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
min           NA 4.078180e+02 5.135667e+02 3.545075e+02 6.178639e+02
max           NA 1.601025e+04 1.892888e+04 2.203095e+04 2.180892e+04
range         NA 1.560243e+04 1.841531e+04 2.167644e+04 2.119105e+04
sum           NA 3.888715e+05 4.673224e+05 5.487424e+05 6.162241e+05
median        NA 2.484720e+03 2.884388e+03 3.072176e+03 3.741725e+03
mean          NA 3.634313e+03 4.367500e+03 5.128433e+03 5.759103e+03
SE.mean       NA 3.314566e+02 4.021934e+02 4.736475e+02 5.272377e+02
CI.mean       NA 6.571449e+02 7.973875e+02 9.390523e+02 1.045300e+03
var           NA 1.175539e+07 1.730827e+07 2.400459e+07 2.974381e+07
std.dev       NA 3.428613e+03 4.160321e+03 4.899448e+03 5.453789e+03
coef.var      NA 9.434006e-01 9.525635e-01 9.553499e-01 9.469857e-01
              gdppc80      gdppc85      gdppc90      gdppc95       africa
nbr.val  1.070000e+02 1.070000e+02 1.070000e+02 1.070000e+02 107.00000000
nbr.null 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00  67.00000000
nbr.na   0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00   0.00000000
min      4.735793e+02 5.422725e+02 5.277151e+02 4.993415e+02   0.00000000
max      2.386009e+04 2.525136e+04 2.874414e+04 3.674105e+04   1.00000000
range    2.338651e+04 2.470909e+04 2.821642e+04 3.624171e+04   1.00000000
sum      7.012400e+05 7.383373e+05 8.319308e+05 9.061030e+05  40.00000000
median   4.306217e+03 4.200733e+03 4.034010e+03 4.467940e+03   0.00000000
mean     6.553645e+03 6.900349e+03 7.775054e+03 8.468253e+03   0.37383178
SE.mean  6.018749e+02 6.552251e+02 7.711596e+02 8.456513e+02   0.04699273
CI.mean  1.193276e+03 1.299048e+03 1.528899e+03 1.676586e+03   0.09316766
var      3.876112e+07 4.593724e+07 6.363152e+07 7.651850e+07   0.23628990
std.dev  6.225843e+03 6.777701e+03 7.976937e+03 8.747486e+03   0.48609659
coef.var 9.499817e-01 9.822259e-01 1.025965e+00 1.032974e+00   1.30030838
                 asia      weurope      growth
nbr.val  107.00000000 107.00000000 107.0000000
nbr.null  93.00000000  92.00000000   0.0000000
nbr.na     0.00000000   0.00000000   0.0000000
min        0.00000000   0.00000000  -0.6887722
max        1.00000000   1.00000000   2.3493433
range      1.00000000   1.00000000   3.0381155
```

```
sum       14.00000000  15.00000000  67.8899760
median     0.00000000   0.00000000   0.6586871
mean       0.13084112   0.14018692   0.6344858
SE.mean    0.03275433   0.03372119   0.0601857
CI.mean    0.06493865   0.06685553   0.1193240
var        0.11479457   0.12167166   0.3875881
std.dev    0.33881347   0.34881465   0.6225657
coef.var   2.58950297   2.48821120   0.9812131
```

```
# library(Hmisc)
describe(data)
```

```
         vars   n    mean      sd  median trimmed     mad    min      max
country*    1 107   54.00   31.03   54.00   54.00   40.03   1.00   107.00
gdppc60     2 107 3634.31 3428.61 2484.72 3032.19 2027.76 407.82 16010.25
gdppc65     3 107 4367.50 4160.32 2884.39 3673.42 2579.50 513.57 18928.88
gdppc70     4 107 5128.43 4899.45 3072.18 4370.29 2854.11 354.51 22030.95
gdppc75     5 107 5759.10 5453.79 3741.72 4977.54 3708.25 617.86 21808.92
gdppc80     6 107 6553.64 6225.84 4306.22 5707.40 4476.29 473.58 23860.09
gdppc85     7 107 6900.35 6777.70 4200.73 5929.46 4382.44 542.27 25251.36
gdppc90     8 107 7775.05 7976.94 4034.01 6660.00 4258.37 527.72 28744.14
gdppc95     9 107 8468.25 8747.49 4467.94 7235.12 4935.09 499.34 36741.05
africa     10 107    0.37    0.49    0.00    0.34    0.00   0.00     1.00
asia       11 107    0.13    0.34    0.00    0.05    0.00   0.00     1.00
weurope    12 107    0.14    0.35    0.00    0.06    0.00   0.00     1.00
growth     13 107    0.63    0.62    0.66    0.63    0.59  -0.69     2.35
            range skew kurtosis     se
country*   106.00 0.00    -1.23   3.00
gdppc60  15602.43 1.53     1.55 331.46
gdppc65  18415.31 1.41     1.16 402.19
gdppc70  21676.44 1.29     0.74 473.65
gdppc75  21191.05 1.15     0.09 527.24
gdppc80  23386.51 1.07    -0.11 601.87
gdppc85  24709.09 1.14     0.01 655.23
gdppc90  28216.42 1.13    -0.10 771.16
gdppc95  36241.71 1.14     0.12 845.65
africa       1.00 0.51    -1.75   0.05
asia         1.00 2.16     2.69   0.03
weurope      1.00 2.04     2.20   0.03
growth       3.04 0.15     0.07   0.06
```

```
# library(gtsummary)
tbl_summary(data)
```

```
Table printed with `knitr::kable()`, not {gt}. Learn why at
```

```
https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html
To suppress this message, include `message = FALSE` in code chunk header.
```

Characteristic	**N = 107**
country	
Algeria	1 (0.9%)
Angola	1 (0.9%)
Argentina	1 (0.9%)
Australia	1 (0.9%)
Austria	1 (0.9%)
Bangladesh	1 (0.9%)
Barbados	1 (0.9%)
Belgium	1 (0.9%)
Benin	1 (0.9%)
Bolivia	1 (0.9%)
Botswana	1 (0.9%)
Brazil	1 (0.9%)
Burkina Faso	1 (0.9%)
Burundi	1 (0.9%)
Cameroon	1 (0.9%)
Canada	1 (0.9%)
Cape Verde	1 (0.9%)
Central African Republic	1 (0.9%)
Chad	1 (0.9%)
Chile	1 (0.9%)
China	1 (0.9%)
Colombia	1 (0.9%)
Comoros	1 (0.9%)
Congo, Republic of	1 (0.9%)
Costa Rica	1 (0.9%)
Cote d’lvoire	1 (0.9%)
Cyprus	1 (0.9%)
Denmark	1 (0.9%)
Dominican Republic	1 (0.9%)
Ecuador	1 (0.9%)
Egypt	1 (0.9%)
El Salvador	1 (0.9%)
Ethiopia	1 (0.9%)
Fiji	1 (0.9%)
Finland	1 (0.9%)
France	1 (0.9%)
Gabon	1 (0.9%)
Gambia, The	1 (0.9%)
Ghana	1 (0.9%)
Greece	1 (0.9%)
Guatemala	1 (0.9%)
Guinea	1 (0.9%)
Guinea-Bissau	1 (0.9%)
Guyana	1 (0.9%)
Honduras	1 (0.9%)
Hong Kong	1 (0.9%)
Iceland	1 (0.9%)
India	1 (0.9%)
Indonesia	1 (0.9%)
Iran	1 (0.9%)
Ireland	1 (0.9%)
Israel	1 (0.9%)
Italy	1 (0.9%)

```r
# check the assignments of countries to continents
data %>%
  select(country, africa, asia, weurope) %>%
  view()

data <- mutate(data, x_1 = africa + asia + weurope)

data %>%
  filter(x_1==0) %>%
  select(africa, asia, weurope, country) %>%
  view()

# correct the assignment manually
data$weurope[data$country == "Austria"] <- 1
data$weurope[data$country == "Greece"] <- 1
data$weurope[data$country == "Cyprus"] <- 1

filter(data, data$weurope==1) # check changes
```

```
# A tibble: 18 x 14
   country       gdppc60 gdppc65 gdppc70 gdppc75 gdppc80 gdppc85 gdppc90 gdppc95
   <chr>           <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
 1 Austria         7842.   9387.  11946.  14198.  16869.  17919.  21178.  22474.
 2 Belgium         8314.  10454.  12980.  15024.  17451.  18109.  21246.  22356.
 3 Cyprus          3178.   4261.   5638.   4827.   8302.  10228.  13798.  17169.
 4 Denmark        11745.  14749.  17143.  17750.  19558.  21596.  23308.  25293.
 5 Finland         8007.   9851.  12198.  14884.  16621.  18585.  21667.  20084.
 6 France          8364.  10497.  13186.  14951.  17335.  18429.  21403.  21502.
 7 Greece          4454.   6549.   9022.  11121.  12672.  12287.  12794.  13332.
 8 Iceland         8786.  11403.  11678.  15235.  19440.  20414.  22502.  21901.
 9 Ireland         5490.   6413.   7760.   9064.  10649.  11641.  15133.  18456.
10 Italy           7364.   9097.  12072.  13386.  16286.  17518.  20638.  21691.
11 Luxembourg     12510.  14019.  16163.  17384.  19089.  21414.  28744.  36741.
12 Netherlands     9883.  11702.  14237.  15803.  17339.  17974.  20823.  22320.
13 Norway          8808.  10478.  11959.  14873.  17977.  20630.  21855.  25538.
14 Portugal        3665.   4866.   6730.   7951.   9667.   9847.  13155.  13924.
15 Spain           4956.   7459.   9701.  11970.  12294.  12583.  15475.  17434.
16 Sweden         10870.  13552.  15850.  17588.  18348.  20001.  22219.  22122.
17 Switzerland    16010.  18929.  22031.  21809.  23860.  24844.  27931.  26227.
18 United Kingd~  10341.  11633.  12917.  14072.  15302.  16878.  19585.  20963.
# i 5 more variables: africa <dbl>, asia <dbl>, weurope <dbl>, growth <dbl>,
#   x_1 <dbl>
# In the following, I do the same with a loop
# c_europe <- c("Austria","Greece","Cyprus")
# sum(data$weurope)                       # check changes
# for (i in c_europe){
#   print(i)
#   data$weurope[data$country == i] <- 1
#   }
# sum(data$weurope)                       # check changes
# data$weurope[data$country == "Austria"] # check changes

# create a category for the remaining countries
# use ifelse -- ifelse(condition, result if TRUE, result if FALSE)
data$rest <- ifelse(data$africa == 0 & data$asia == 0 & data$weurope == 0, 1, 0)
data$rest <- set_label(data$rest, label = "=1 if not in Africa, W.Europe, or Asia")

# create table with means across country groups
table_gdp <- data %>%
  group_by(africa, asia, weurope) %>%
  summarise_at(vars(gdppc60:gdppc95), list(name = mean))

data %>%
  group_by(africa, asia, weurope) %>%
  select(gdppc60:gdppc95) %>%
  summarise_all(mean)
```

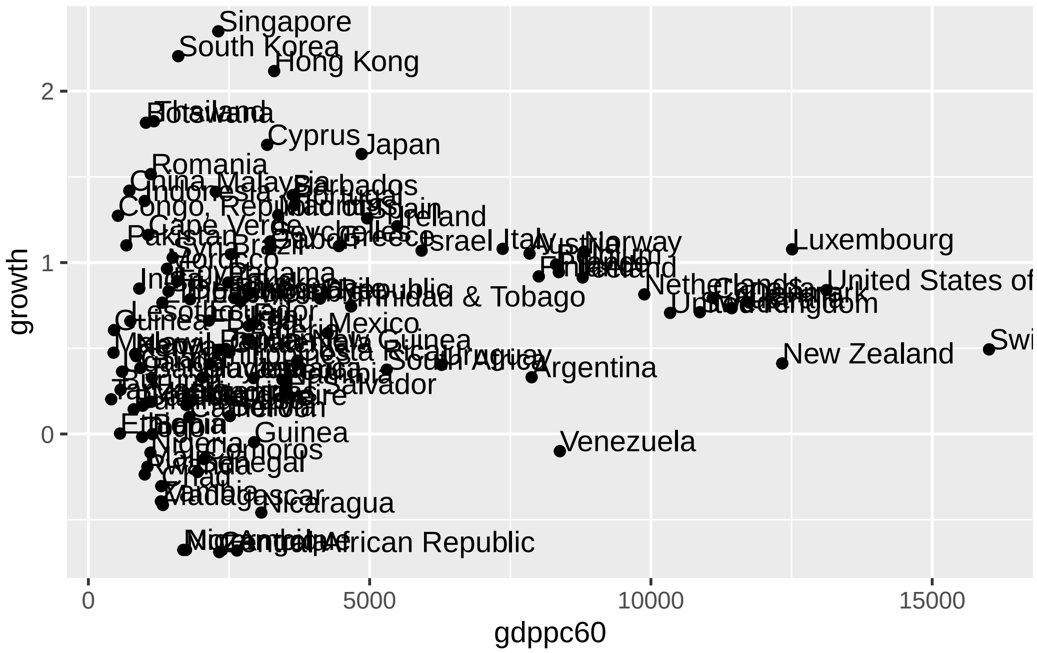
```
Adding missing grouping variables: `africa`, `asia`, `weurope`
```

```
# A tibble: 4 x 11
# Groups:   africa, asia [3]
  africa  asia weurope gdppc60 gdppc65 gdppc70 gdppc75 gdppc80 gdppc85 gdppc90
   <dbl> <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
1      0     0       0   4288.   5034.   5727.   6411.   7042.   7185.   7457.
2      0     0       1   8366.  10294.  12401.  13994.  16059.  17272.  20192.
3      0     1       0   1739.   2247.   3090.   3760.   4905.   5761.   7501.
4      1     0       0   1596.   1860.   2046.   2182.   2426.   2382.   2562.
# i 1 more variable: gdppc95 <dbl>
```

```
# create growth rate
data$gr1 <- (data$gdppc95 - data$gdppc60)/data$gdppc60
data$gr2 <- log(data$gdppc95) - log(data$gdppc60)
cor(data$gr1, data$gr2)
```

```
[1] 0.9008887
```

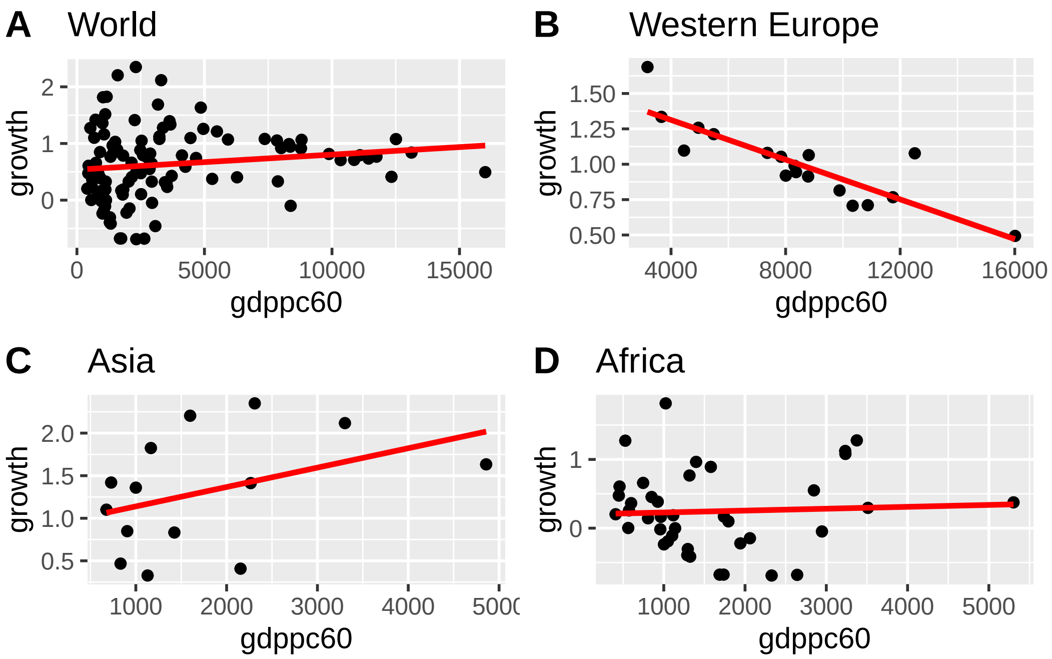
```
ggplot(data, aes(x = gdppc60, y = growth, label=country)) +
  geom_point() +
  geom_text(hjust=0, vjust=0)
```



```
p1 <- ggplot(data, aes(x = gdppc60, y = growth, label=country )) +
  geom_point() +
  stat_smooth(formula=y~x, method="lm", se=FALSE, colour="red", linetype=1) +
 # geom_text(hjust=0, vjust=0) +
  ggtitle("World")

p2 <- data %>%
  filter(weurope==1) %>%
  ggplot( aes(x = gdppc60, y = growth, label=country )) +
  geom_point() +
  stat_smooth(formula=y~x, method="lm", se=FALSE, colour="red", linetype=1) +
  #geom_text(hjust=0, vjust=0) +
  ggtitle("Western Europe")

p3 <- data %>%
  filter(asia==1) %>%
  ggplot( aes(x = gdppc60, y = growth, label=country )) +
  geom_point() +
  stat_smooth(formula=y~x, method="lm", se=FALSE, colour="red", linetype=1) +
 # geom_text(hjust=0, vjust=0) +
  ggtitle("Asia")
```

```r
p4 <- data %>%
  filter(africa==1) %>%
  ggplot( aes(x = gdppc60, y = growth, label=country )) +
  geom_point() +
  stat_smooth(formula=y~x, method="lm", se=FALSE, colour="red", linetype=1) +
#  geom_text( hjust=0, vjust=0) +
  ggtitle("Africa")

ggarrange(p1, p2, p3, p4 ,
          labels = c("A", "B", "C", "D"),
          ncol = 2, nrow = 2)
```

```
Warning: The following aesthetics were dropped during statistical transformation: label.
i This can happen when ggplot fails to infer the correct grouping structure in
  the data.
i Did you forget to specify a `group` aesthetic or to convert a numerical
  variable into a factor?
The following aesthetics were dropped during statistical transformation: label.
i This can happen when ggplot fails to infer the correct grouping structure in
  the data.
i Did you forget to specify a `group` aesthetic or to convert a numerical
  variable into a factor?
The following aesthetics were dropped during statistical transformation: label.
i This can happen when ggplot fails to infer the correct grouping structure in
  the data.
i Did you forget to specify a `group` aesthetic or to convert a numerical
  variable into a factor?
The following aesthetics were dropped during statistical transformation: label.
i This can happen when ggplot fails to infer the correct grouping structure in
  the data.
i Did you forget to specify a `group` aesthetic or to convert a numerical
  variable into a factor?
```



```r
# Regression analysis
m1  <-  lm(growth ~ gdppc60, data = data)
m2  <-  lm(growth ~ gdppc60, data = subset(data, weurope==1))
m3  <-  lm(growth ~ gdppc60, data = subset(data, asia==1))
m4  <-  lm(growth ~ gdppc60, data = subset(data, africa==1))

tab_model(m1, m2, m3, m4,
          p.style = "stars",
          p.threshold = c(0.2, 0.1, 0.05),
          show.ci = FALSE,
          show.se = FALSE,
          show.aic = TRUE,
          dv.labels = c("World", "W.Europe", "Asia", "Africa"))
```

	World	W.Europe	Asia	Africa
Predictors	Estimates	Estimates	Estimates	Estimates
(Intercept)	0.54 ***	1.59 ***	0.91 ***	0.20
real gdp per capita 1960	0.00 *	-0.00 ***	0.00 *	0.00
Observations	107	18	14	40
R^2 / R^2 adjusted	0.021 / 0.012	0.727 / 0.710	0.158 / 0.088	0.002 / -0.024
AIC	204.917	-14.237	31.220	76.318

* p<0.2 ** p<0.1 *** p<0.05

```
# reshape data (see: https://stackoverflow.com/questions/2185252/reshaping-data-frame-from
data_long <- gather(data, condition, measurement, gdppc60:gdppc95, factor_key=TRUE)
```

```
Warning: attributes are not identical across measure variables; they will be
dropped
```

```
data_long$year <- as.numeric(substr(data_long$condition, 6, 7))

data_long$gr_long <- data_long %>%
  select(country,measurement) %>%
  group_by(country) %>%
  mutate(gr = c(NA,diff(measurement))/lag(measurement, 1))

# erase all helping variables
data <- select(data, -starts_with("h_"))

# generate and remove variables in a dataframe
data <- mutate(data, Land = country)
data <- select(data, -country)




data %>%
  summarise(
    y65 = mean(gdppc65, na.rm = TRUE),
    y70 = mean(gdppc70, na.rm = TRUE),
    y75 = mean(gdppc75, na.rm = TRUE),
    y80 = mean(gdppc80, na.rm = TRUE),
    y85 = mean(gdppc85, na.rm = TRUE),
    y90 = mean(gdppc90, na.rm = TRUE),
    y95 = mean(gdppc95, na.rm = TRUE)
  )
```

```
# A tibble: 1 x 7
    y65   y70   y75   y80   y85   y90   y95
  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 4367. 5128. 5759. 6554. 6900. 7775. 8468.
```

```
suppressMessages(pacman::p_unload(haven, tidyverse, vtable, gtsummary, pastecs, Hmisc,
               sjlabelled, tis, ggpubr, sjPlot))
```

2.9 exe_un_gdp_ger_fra.R

```
# setwd("/home/sthu/Dropbox/hsf/exams/22-11/scr/")

rm(list=ls())

if (!require(pacman)) install.packages("pacman")
pacman::p_load(tidyverse, ggpubr, sjPlot)

load(url("https://github.com/hubchev/courses/raw/main/dta/forest.Rdata"))

head(df,8)
```

```
# A tibble: 8 x 11
# Groups:   country.x [1]
  country.x     date     gdp gdp_growth unemployment region income forest    pop
  <chr>        <dbl>   <dbl>      <dbl>        <dbl> <chr>  <chr>   <dbl>  <dbl>
1 United Arab~  1992 1.26e11     -2.48          1.84 Middl~ High ~   3.63 2.05e6
2 United Arab~  1993 1.27e11     -4.34          1.85 Middl~ High ~   3.72 2.17e6
3 United Arab~  1994 1.36e11      1.25          1.81 Middl~ High ~   3.81 2.29e6
4 United Arab~  1995 1.45e11      1.35          1.80 Middl~ High ~   3.90 2.42e6
5 United Arab~  1996 1.54e11      0.631         1.90 Middl~ High ~   3.99 2.54e6
6 United Arab~  1997 1.66e11      2.83          1.98 Middl~ High ~   4.08 2.67e6
7 United Arab~  1998 1.67e11     -4.77          2.14 Middl~ High ~   4.18 2.81e6
8 United Arab~  1999 1.72e11     -2.40          2.22 Middl~ High ~   4.27 2.97e6
# i 2 more variables: unemployment_dif <dbl>, gdppc <dbl>
```

```
tail(df,1)
```

```
# A tibble: 1 x 11
# Groups:   country.x [1]
  country.x  date        gdp gdp_growth unemployment region income forest    pop
  <chr>     <dbl>      <dbl>      <dbl>        <dbl> <chr>  <chr>   <dbl>  <dbl>
1 Zimbabwe   2020    1.94e10      -7.62         5.35 Sub-S~ Lower~   45.1 1.49e7
# i 2 more variables: unemployment_dif <dbl>, gdppc <dbl>
```

```
 # panel data set
 # date and country.x

observations_df <- dim(df)

df <- rename(df, nation=country.x)
df <- rename(df, year=date)
df <- df %>%
  select(nation, year, gdp, pop, gdppc, unemployment)

df <- df %>%
  mutate(gdp_pc = gdp/pop)

df <- df %>% filter(nation=="Germany" | nation=="France")

df  %>%
  group_by(nation) %>%
  summarise(mean(unemployment), mean(gdppc))
```

```
# A tibble: 2 x 3
  nation  `mean(unemployment)` `mean(gdppc)`
  <chr>                  <dbl>         <dbl>
1 France                  9.75        34356.
2 Germany                 7.22        36739.
```

```
df  %>%
  filter(year==2020) %>%
  group_by(nation) %>%
  summarise(mean(unemployment), mean(gdppc))
```

```
# A tibble: 2 x 3
  nation  `mean(unemployment)` `mean(gdppc)`
  <chr>                  <dbl>         <dbl>
1 France                  8.01        35786.
2 Germany                 3.81        41315.
```

```
df  %>%
  group_by(nation) %>%
  summarise(max(unemployment), max(gdppc))
```

```
# A tibble: 2 x 3
  nation  `max(unemployment)` `max(gdppc)`
  <chr>                 <dbl>        <dbl>
1 France                 12.6       38912.
2 Germany                11.2       43329.
```

```
df %>%
  group_by(nation) %>%
  summarise(sd(gdppc), sd(unemployment))
```

```
# A tibble: 2 x 3
  nation  `sd(gdppc)` `sd(unemployment)`
  <chr>         <dbl>              <dbl>
1 France        2940.               1.58
2 Germany       4015.               2.37
```

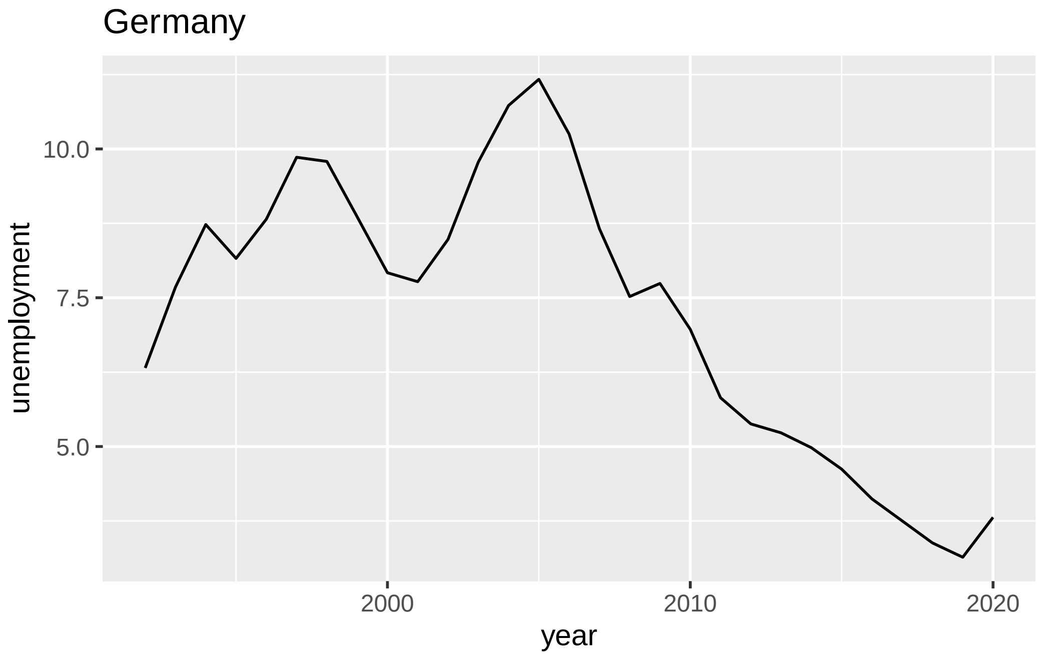
```
df %>%
  group_by(nation) %>%
  summarise(sd(unemployment), mean(unemployment), cov = sd(unemployment)/mean(unemployment
```

```
# A tibble: 2 x 4
  nation  `sd(unemployment)` `mean(unemployment)`   cov
  <chr>                <dbl>                <dbl> <dbl>
1 France                1.58                 9.75 0.162
2 Germany               2.37                 7.22 0.328
```

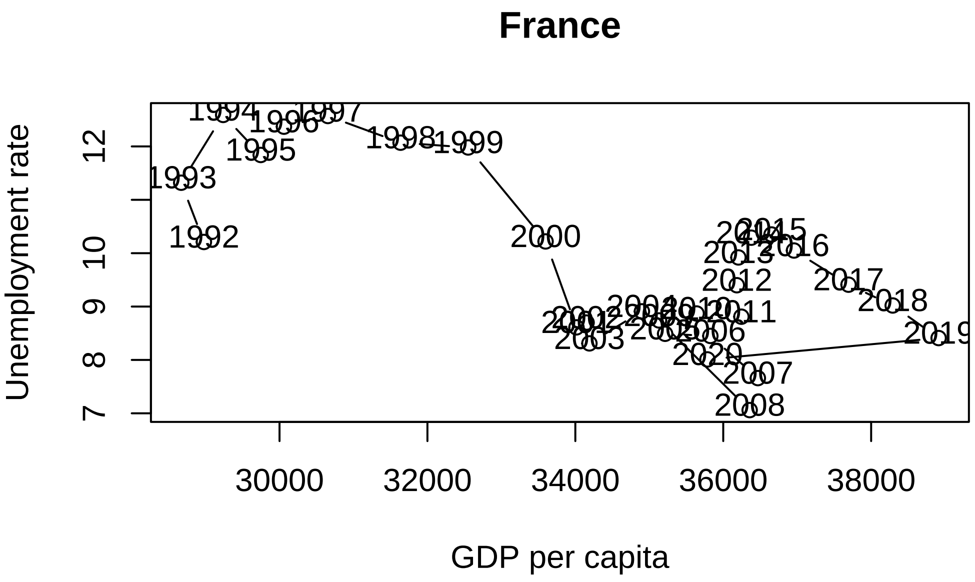
```
df %>%
  group_by(nation) %>%
  summarise(sd(gdppc),mean(gdppc), cov = sd(gdppc)/mean(gdppc))
```

```
# A tibble: 2 x 4
  nation  `sd(gdppc)` `mean(gdppc)`    cov
  <chr>         <dbl>         <dbl>  <dbl>
1 France        2940.        34356. 0.0856
2 Germany       4015.        36739. 0.109
```

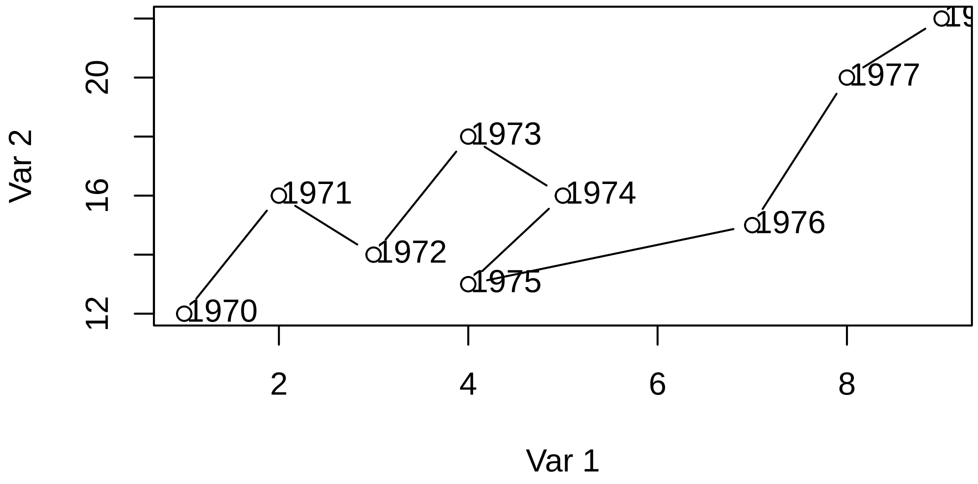
```
df %>%
  filter(nation == "Germany") %>%
  ggplot(aes(x = year, y = unemployment)) +
  geom_line() +
  ggtitle("Germany")
```



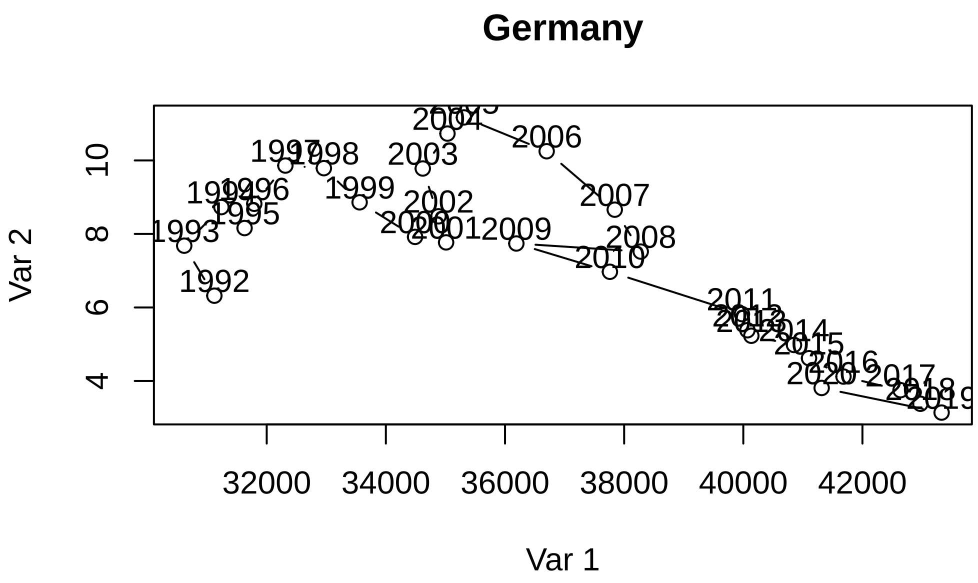
```
labels <- 1992:2020
dfra <- df %>% filter(nation == "France")
plot(dfra$gdppc, dfra$unemployment, type = "b",
     xlab = "GDP per capita", ylab = "Unemployment rate"); text(dfra$gdppc + 0.1, dfra$une
```



```r
# Data
x <- c(1, 2, 3, 4, 5, 4, 7, 8, 9)
y <- c(12, 16, 14, 18, 16, 13, 15, 20, 22)
labels <- 1970:1978

# Connected scatter plot with text
plot(x, y, type = "b", xlab = "Var 1", ylab = "Var 2"); text(x + 0.4, y + 0.1, labels)
```



```r
dfger <- df %>% filter(nation == "Germany")
labels <- 1992:2020
plot(dfger$gdppc, dfger$unemployment, type = "b",
     xlab = "Var 1", ylab = "Var 2"); text(dfger$gdppc + 0.7, dfger$unemployment + 0.4, la
```



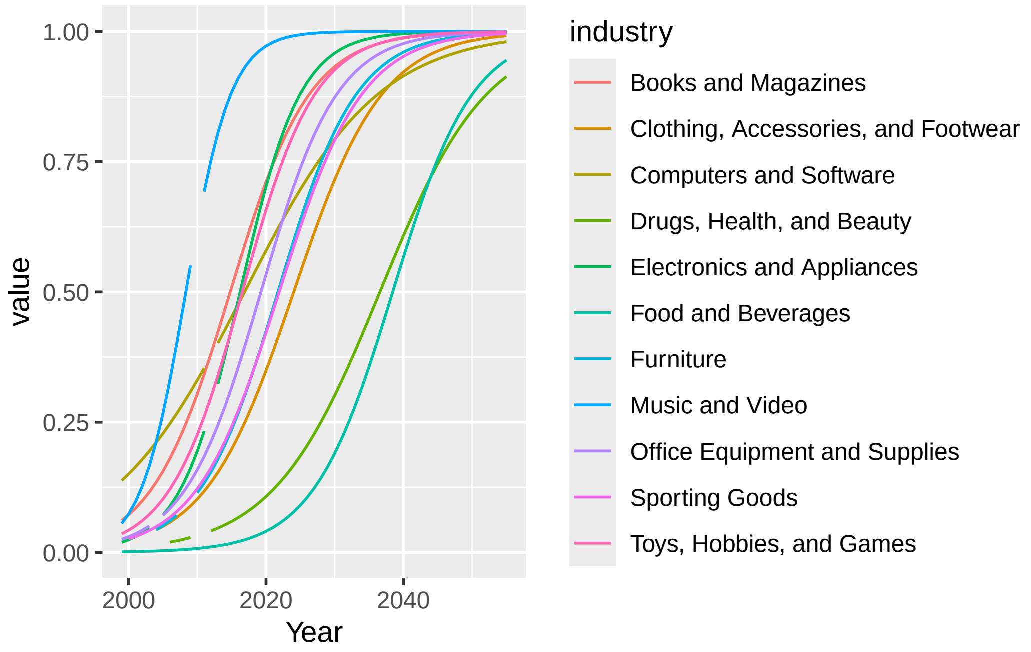
```
# rmarkdown::render("22-11_dsda_exam.Rmd", "all")

# knitr::purl(input = "22-11_dsda_exam.Rmd", output = "22-11_dsda_solution.R",documentatio

suppressMessages(pacman::p_unload(tidyverse, ggpubr, sjPlot))
```

2.10 exe_hortacsu_figure_3.R

```
# setwd("~/Dropbox/hsf/courses/Rlang/hortacsu")

rm(list = ls())


# install and load packages
if (!require(pacman)) install.packages("pacman")
pacman::p_load(tidyverse, readxl)


# Define the URL of the ZIP file
zipF <- "https://github.com/hubchev/courses/raw/main/dta/113962-V1.zip"

# Download the ZIP file
download.file(zipF, destfile = "113962-V1.zip")
```

```r
# Unzip the contents
unzip("113962-V1.zip")

df_curves <- read_excel("Hortacsu_Syverson_JEP_Retail/diffusion_curves_figure.xlsx",
                        sheet = "Data and Predictions", range = "N3:Y60")

df <- df_curves |>
  pivot_longer(
    cols = 'Music and Video':'Food and Beverages',
    names_to = "industry",
    values_to = "value"
  )

# Plot
df %>%
  ggplot( aes(x=Year, y=value, group=industry, color=industry)) +
  geom_line()
```

```
Warning: Removed 18 rows containing missing values or values outside the scale range
(`geom_line()`).
```



```r
#  unload packages
suppressMessages(pacman::p_unload(tidyverse, readxl))
```

2.11 exe_regress_lecture.R

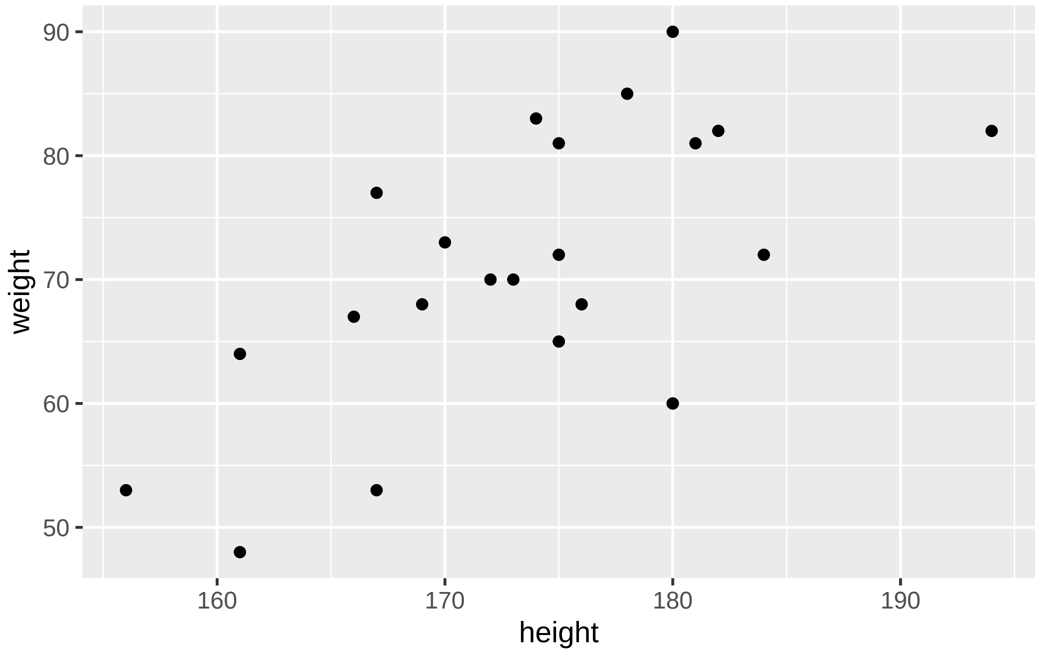
```
## ---- echo = TRUE--------------------------------------------------
# install and load packages
if (!require(pacman)) install.packages("pacman")
pacman::p_load(tidyverse, haven)

classdata <- read.csv("https://raw.githubusercontent.com/hubchev/courses/main/dta/classdat

head(classdata)
```

```
  id sex weight height siblings row
1  1   w     53    156        1   g
2  2   w     73    170        1   g
3  3   m     68    169        1   g
4  4   w     67    166        1   g
5  5   w     65    175        1   g
6  6   w     48    161        0   g
```

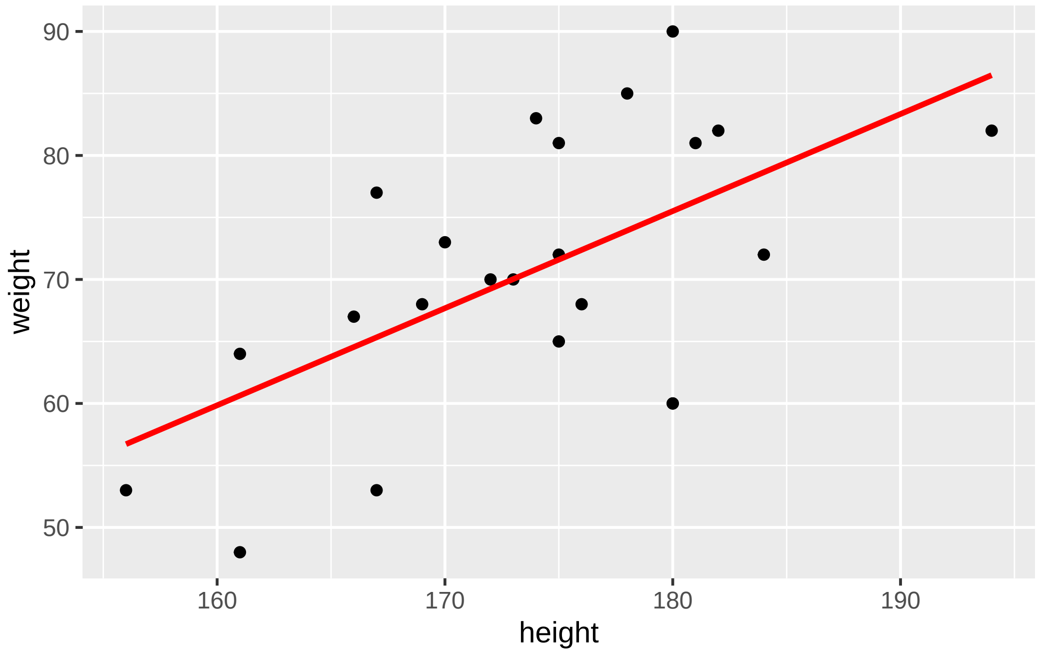
```
## ---- echo = TRUE--------------------------------------------------

summary(classdata)
```

```
       id           sex                weight          height     
 Min.   : 1.0   Length:23          Min.   :48.00   Min.   :156.0  
 1st Qu.: 6.5   Class :character   1st Qu.:64.50   1st Qu.:168.0  
 Median :12.0   Mode  :character   Median :70.00   Median :175.0  
 Mean   :12.0                      Mean   :70.61   Mean   :173.7  
 3rd Qu.:17.5                      3rd Qu.:81.00   3rd Qu.:180.0  
 Max.   :23.0                      Max.   :90.00   Max.   :194.0  
    siblings         row           
 Min.   :0.000   Length:23         
 1st Qu.:1.000   Class :character  
 Median :1.000   Mode  :character  
 Mean   :1.391                     
 3rd Qu.:2.000                     
 Max.   :4.000                     
```

```
## ----pressure, echo=TRUE-------------------------------------------
library("ggplot2")
ggplot(classdata, aes(x=height, y=weight)) + geom_point()
```



```
## ---- echo=TRUE-----------------------------------------------------
ggplot(classdata, aes(x=height, y=weight)) +
  geom_point() +
  stat_smooth(formula=y~x, method="lm", se=FALSE, colour="red", linetype=1)
```



```
## ---- echo=TRUE--------------------------------------------------------
## baseline regression  model
model  <- lm(weight ~ height + sex , data = classdata )
show(model)
```

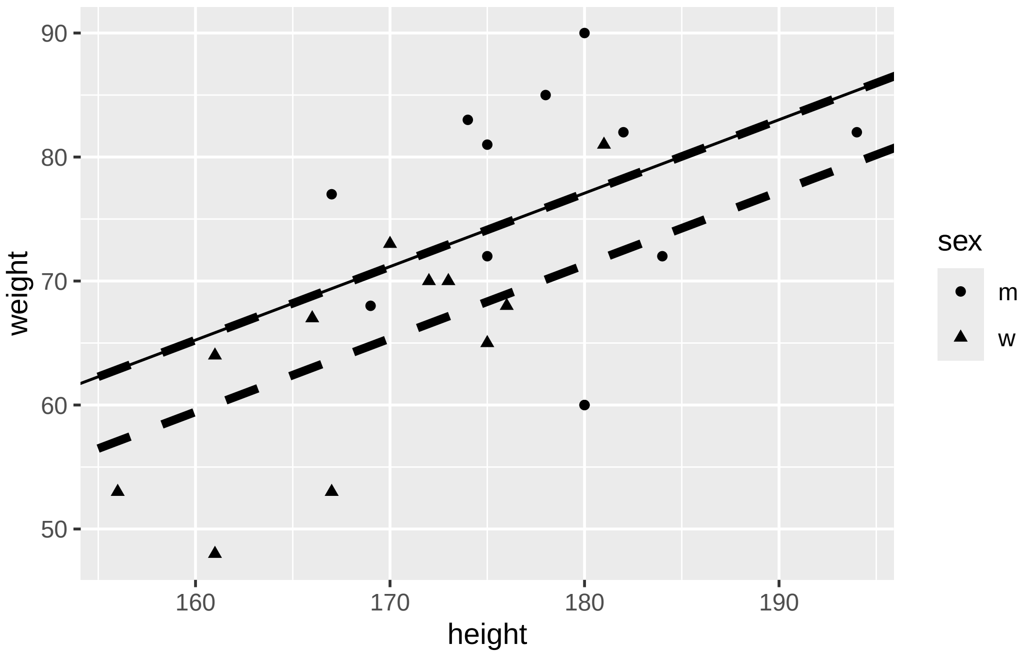
```
Call:
lm(formula = weight ~ height + sex, data = classdata)

Coefficients:
(Intercept)       height         sexw
   -29.5297       0.5923      -5.7894
```

```
interm <- model$coefficients[1]
slope  <- model$coefficients[2]
interw <- model$coefficients[1]+model$coefficients[3]

## ---- echo=TRUE--------------------------------------------------------
summary(model)
```

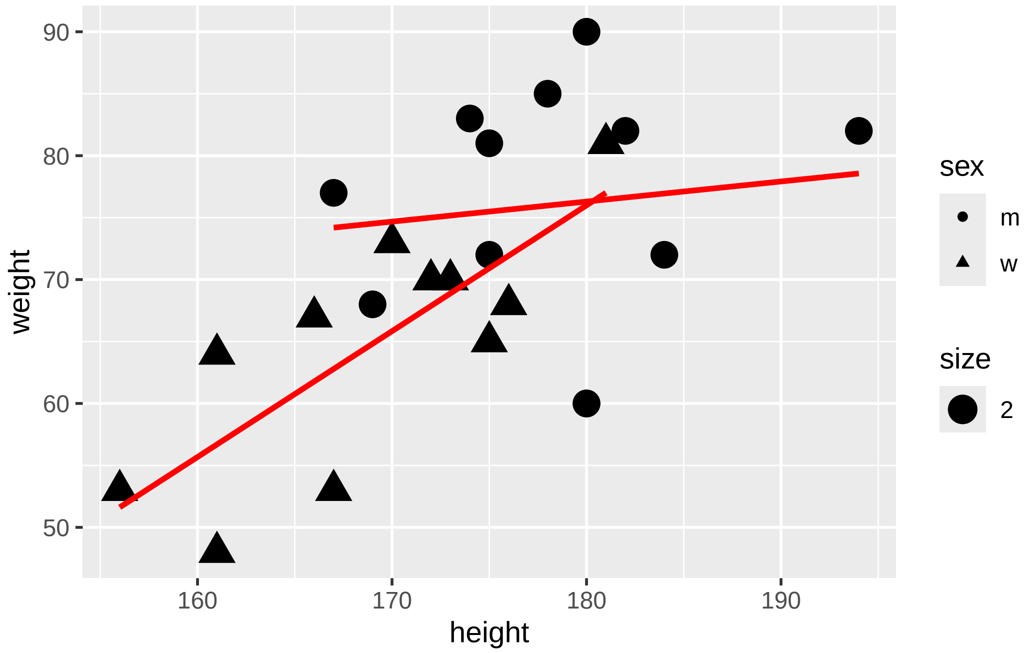
```
Call:
lm(formula = weight ~ height + sex, data = classdata)

Residuals:
    Min      1Q  Median      3Q     Max
-17.086  -3.730   2.850   7.245  12.914

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -29.5297    47.6606  -0.620   0.5425
height        0.5923     0.2671   2.217   0.0383 *
sexw         -5.7894     4.4773  -1.293   0.2107
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.942 on 20 degrees of freedom
Multiple R-squared:  0.4124,    Adjusted R-squared:  0.3537
F-statistic: 7.019 on 2 and 20 DF,  p-value: 0.004904
```

```
## ---- echo=TRUE--------------------------------------------------------
ggplot(classdata, aes(x=height, y=weight, shape = sex)) +
  geom_point() +
  geom_abline(slope = slope, intercept = interw, linetype = 2, size=1.5)+
  geom_abline(slope = slope, intercept = interm, linetype = 2, size=1.5) +
  geom_abline(slope = coef(model)[[2]], intercept = coef(model)[[1]])
```

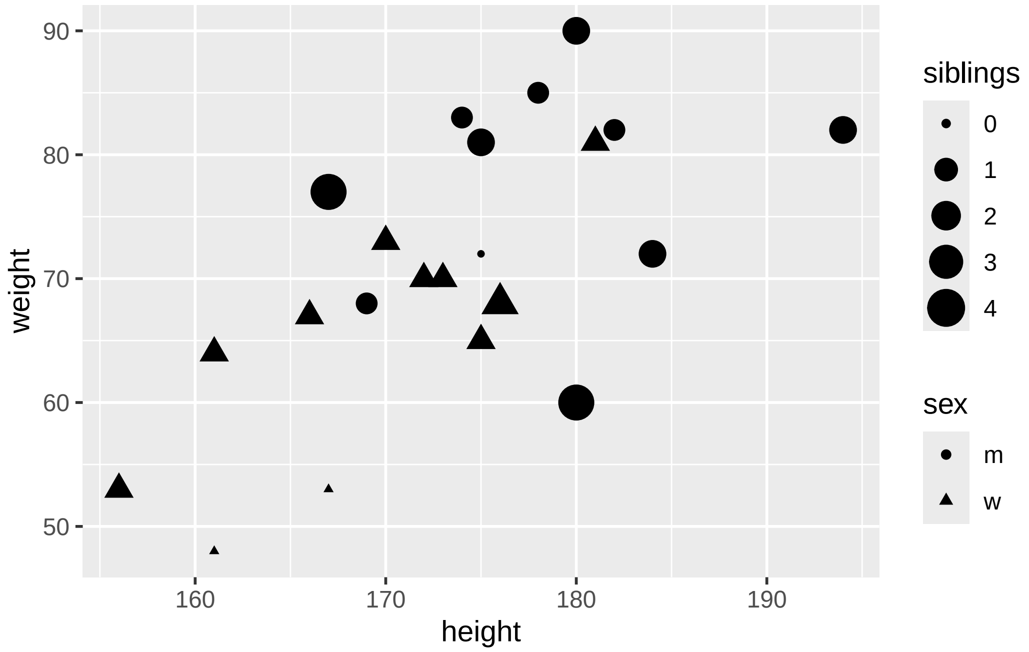
```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```



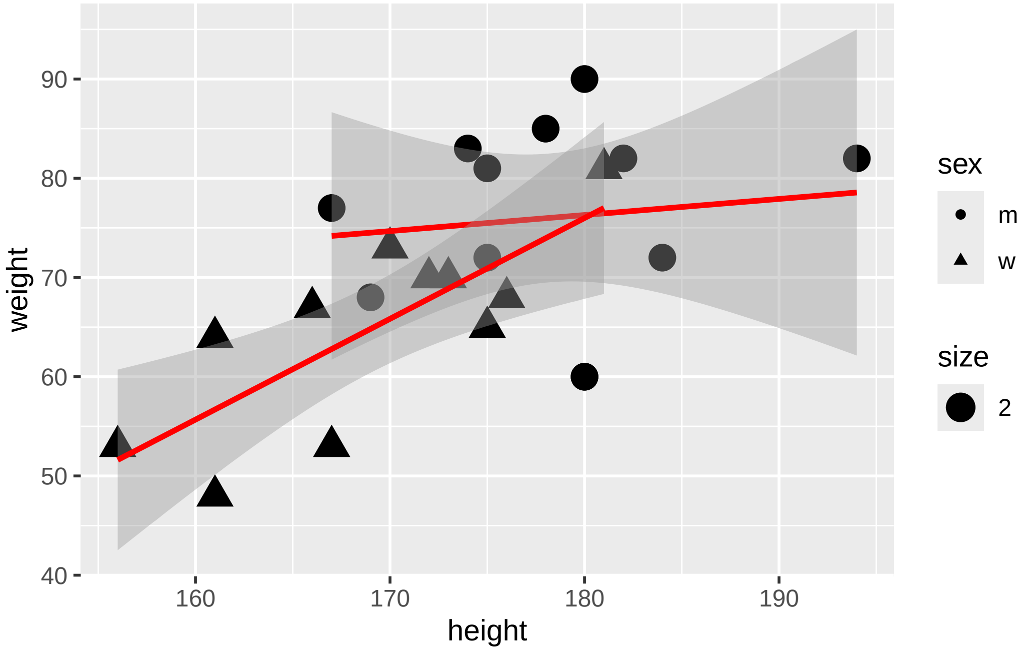
```
## ---- echo=TRUE--------------------------------------------------------

ggplot(classdata, aes(x=height, y=weight, shape = sex)) +
  geom_point( aes(size = 2)) +
  stat_smooth(formula = y ~ x,  method = "lm",
              se = FALSE, colour = "red", linetype = 1)
```



```
## ---- echo=TRUE------------------------------------------------------
ggplot(classdata, aes(x=height, y=weight, shape = sex)) +
  geom_point( aes(size = siblings))
```



```
## ---- echo=TRUE-----------------------------------------------------
## baseline model
model  <- lm(weight ~ height + sex , data = classdata )

ggplot(classdata, aes(x=height, y=weight, shape = sex)) +
  geom_point( aes(size = 2)) +
  stat_smooth(formula = y ~ x,
              method = "lm",
              se = T,
              colour = "red",
              linetype = 1)
```



```
## ---- echo=TRUE, results='hide'-------------------------------------

m1 <- lm(weight ~ height , data = classdata )
m2 <- lm(weight ~ height + sex , data = classdata )
m3 <- lm(weight ~ height + sex + height * sex , data = classdata )
m4 <- lm(weight ~ height + sex + height * sex + siblings , data = classdata )
m5 <- lm(weight ~ height + sex + height * sex , data = subset(classdata, siblings < 4 ))

library(sjPlot)
tab_model(m1, m2, m3, m4, m5,
          p.style = "stars",
          p.threshold = c(0.2, 0.1, 0.05),
          show.ci = FALSE,
          show.se = FALSE)
```

	weight	weight	weight	weight	weight
Predictors	Estimates	Estimates	Estimates	Estimates	Estimates
(Intercept)	-65.44 *	-29.53	47.14	50.27	27.69
height	0.78 ***	0.59 ***	0.16	0.16	0.28
sex [w]		-5.79	-153.96 **	-161.92 **	-134.51 *
height × sex [w]			0.85 *	0.89 *	0.74 *
siblings				-1.16	
Observations	23	23	23	23	21
R^2 / R^2 adjusted	0.363 / 0.333	0.412 / 0.354	0.487 / 0.407	0.496 / 0.385	0.572 / 0.497

* p<0.2   ** p<0.1   *** p<0.05

```
## ---- echo=FALSE---------------------------------------------------
tab_model(m1, m2, m3, m4,
          p.style = "stars",
          p.threshold = c(0.2, 0.1, 0.05),
          show.ci = FALSE,
          show.se = FALSE)
```

	weight	weight	weight	weight
Predictors	Estimates	Estimates	Estimates	Estimates
(Intercept)	-65.44 *	-29.53	47.14	50.27
height	0.78 ***	0.59 ***	0.16	0.16
sex [w]		-5.79	-153.96 **	-161.92 **
height × sex [w]			0.85 *	0.89 *
siblings				-1.16
Observations	23	23	23	23
R^2 / R^2 adjusted	0.363 / 0.333	0.412 / 0.354	0.487 / 0.407	0.496 / 0.385

* p<0.2   ** p<0.1   *** p<0.05

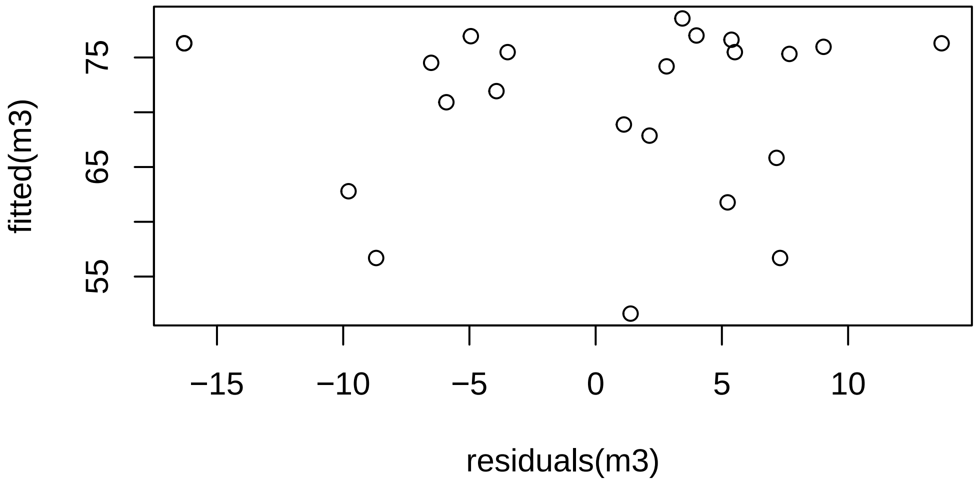
```
## ---- echo=FALSE---------------------------------------------------
tab_model(m3, m5,
          p.style = "stars",
          p.threshold = c(0.2, 0.1, 0.05),
          show.ci = FALSE,
          show.se = FALSE)
```

	weight	weight
Predictors	Estimates	Estimates
(Intercept)	47.14	27.69

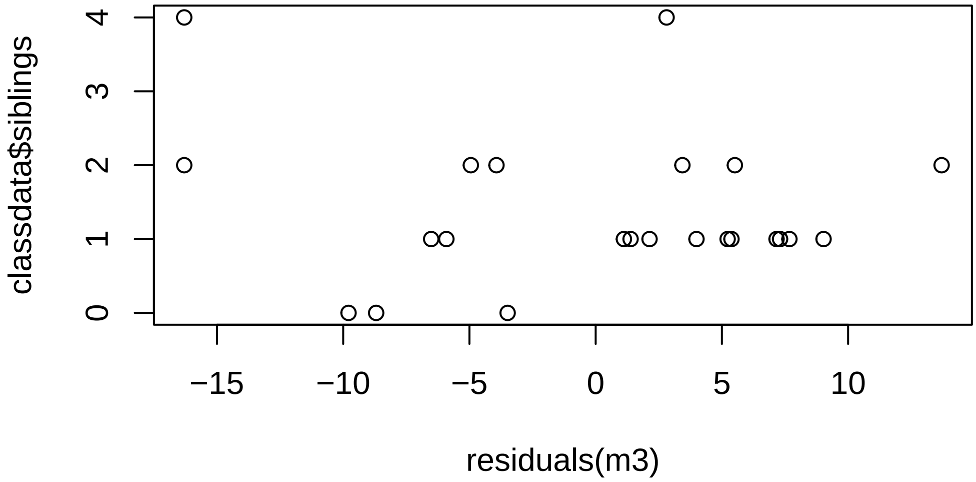
height	0.16	0.28
sex [w]	-153.96 **	-134.51 *
height × sex [w]	0.85 *	0.74 *
Observations	23	21
R^2 / R^2 adjusted	0.487 / 0.407	0.572 / 0.497

* p<0.2 ** p<0.1 *** p<0.05

```r
## ---- echo=T-----------------------------------------------------
plot(residuals(m3), fitted(m3))
```



```r
plot(residuals(m3), classdata$siblings)
```



```
## ----eval=FALSE-----------------------------------------------------
#  rmarkdown::render("regress_lecture.Rmd", "all")

#  unload packages
suppressMessages(pacman::p_unload(tidyverse, haven))
```

2.12 exe_calories.R

```
# 1
#Stephan Huber, 000, 2020-May-30

# 2
# setwd("/home/sthu/Dropbox/hsf/22-ss/dsb_bac/work/")

# 3
rm(list=ls())

# 4
if (!require(pacman)) install.packages("pacman")
pacman::p_load(tidyverse, haven)

# 5
# cross-section
# 6
sex <- c("f", "f", "f", "m", "m",  "m")
age <- c(21, 19, 23, 18, 20, 61)
weight <- c(48, 55,63,71,77,85)
calories <- c(1700,1800,2300,2000,2800,2500)
sport <- c(60,120,180,60,240,30)
df <- data.frame(sex, age, weight, calories, sport)

# write_csv(df, file = "/home/sthu/Dropbox/hsf/exams/21-04/stuff/df.csv")
# write_csv(df, file = "/home/sthu/Dropbox/hsf/github/courses/dta/df-calories.csv")
df <- read_csv("https://raw.githubusercontent.com/hubchev/courses/main/dta/df-calories.csv
```

```
Rows: 6 Columns: 5
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (1): sex
dbl (4): age, weight, calories, sport

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

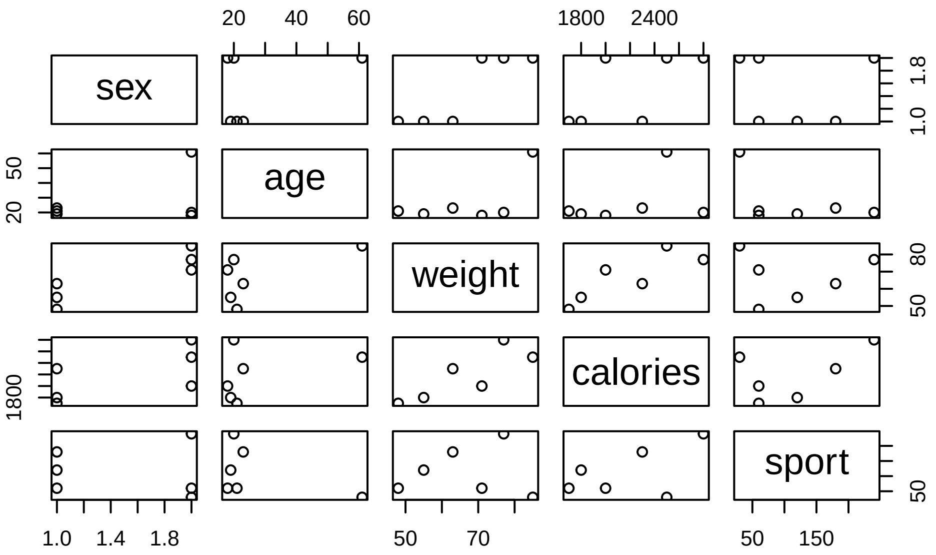
```
# 7
summary(df)
```

```
     sex                 age            weight        calories         sport    
 Length:6           Min.   :18.00   Min.   :48.0   Min.   :1700   Min.   : 30  
 Class :character   1st Qu.:19.25   1st Qu.:57.0   1st Qu.:1850   1st Qu.: 60  
 Mode  :character   Median :20.50   Median :67.0   Median :2150   Median : 90  
                    Mean   :27.00   Mean   :66.5   Mean   :2183   Mean   :115  
                    3rd Qu.:22.50   3rd Qu.:75.5   3rd Qu.:2450   3rd Qu.:165  
                    Max.   :61.00   Max.   :85.0   Max.   :2800   Max.   :240  
```

```
# 8
df  %>%
  group_by(sex) %>%
  summarise(mcal = mean(calories),
            sdcal = sd(calories),
            mweight = mean(weight),
            sdweight = sd(weight)
            )
```

```
# A tibble: 2 x 5
  sex    mcal sdcal mweight sdweight
  <chr> <dbl> <dbl>   <dbl>    <dbl>
1 f      1933.  321.    55.3     7.51
2 m      2433.  404.    77.7     7.02
```

```
# 9
# discussed in class

# 10
# Many things can be mentioned here such as the use of colors
# (red/blue is not a good choice for color blind people),
# the legend makes no sense as red and green both refer to \textit{sport},
# the label of `f' and `m' is not explained in the legend,
# rotating the labels of the y-axis would increase readability, and
# both axes do not start at zero which is hard to see.
# Also, it is a common to draw the variable you want to explain
# (here: calories) on the y-axis.

# 11
plot(df)
```



```
# 12
cor(df$calories, df$sport, method = c("pearson"))
```

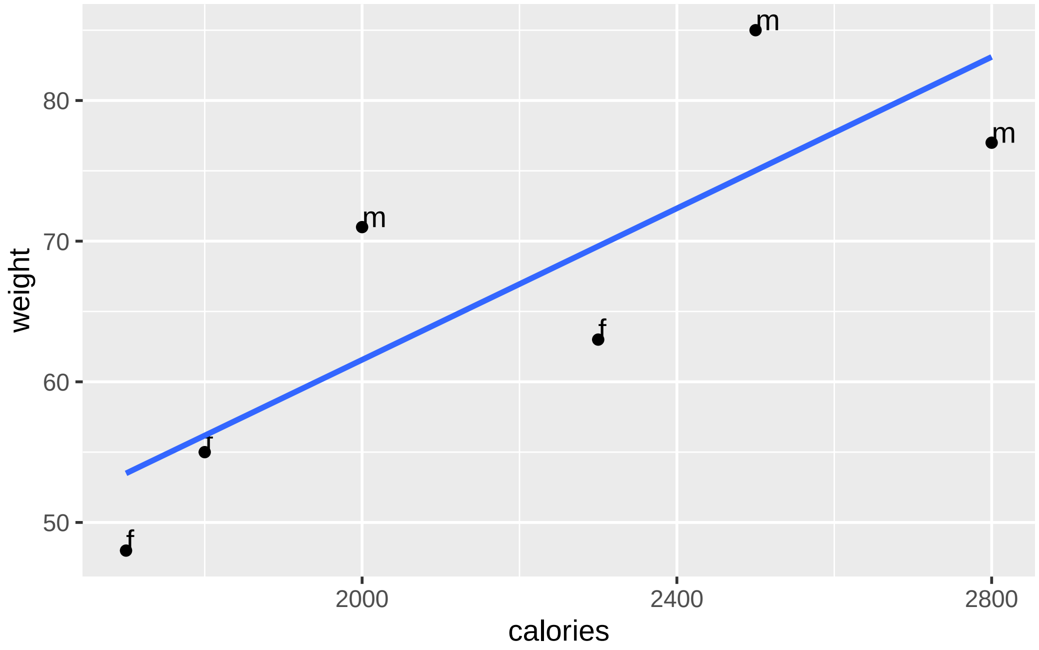
```
[1] 0.5330615
```

```
cor(df$weight, df$calories, method = c("pearson"))
```

```
[1] 0.8281972
```

```
# 13
ggplot(df, aes(x = calories, y = weight, label=sex )) +
  geom_point() +
  geom_text(hjust=0, vjust=0) +
  stat_smooth(formula=y~x, method="lm", se=FALSE)
```

```
Warning: The following aesthetics were dropped during statistical transformation: label.
i This can happen when ggplot fails to infer the correct grouping structure in
  the data.
i Did you forget to specify a `group` aesthetic or to convert a numerical
  variable into a factor?
```



```
# 14
reg_base <- lm(weight ~ calories, data = df)
summary(reg_base)
```

```

Call:
lm(formula = weight ~ calories, data = df)
```

```
Residuals:
     1      2      3      4      5      6 
-5.490 -1.182 -6.640  9.435 -6.099  9.976 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)  
(Intercept)  7.730275  20.197867   0.383   0.7214  
calories     0.026917   0.009107   2.956   0.0417 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.68 on 4 degrees of freedom
Multiple R-squared:  0.6859,    Adjusted R-squared:  0.6074 
F-statistic: 8.735 on 1 and 4 DF,  p-value: 0.04174
```

```r
# 15
# 1) An increase of 100 calories (taken on average on a daily basis) is associated
# - on average and ceteris paribus - with 2.69 more of kg the participants are
# pretended to weight.
# 2) The estimated coefficient $beta_1$ is statistically significantly different to zero
# on a significance level of 5%.
# 3) About 60 % of the variation of the weight is explained by the
# estimated coefficients of the empirical model.

# 16
# For omitted variable bias to occur, the omitted variable `Z` must satisfy
# two conditions:
#   1) The omitted variable is correlated with the included regressor
#   2) The omitted variable is a determinant of the dependent variable

# 17
# discussed in class


# unload packages
suppressMessages(pacman::p_unload(tidyverse, haven))
```

2.13 exe_bundesliga.R

```r
# In dfb.R I analyze German soccer results

# set working directory
# setwd("~/Dropbox/hsf/23-ws/dsda/scripts")

# clear environment
rm(list = ls())

# (Install and) load packagages
if (!require(pacman)) install.packages("pacman")
pacman::p_load(
  bundesligR,
  tidyverse
  )

# Read in the data as tibble
liga <- as_tibble(bundesligR)

# ---------------------------------------
# !!! ERRORS / ISSUES:
# "Borussia Moenchengladbach" is also entitled "Bor. Moenchengladbach"!
# Leverkusen is falsly entitled "SV Bayer 04 Leverkusen"
# Uerdingen has changed its name several times
# Stuttgarter Kickers are named differently

# How often is "Bor. Moenchengladbach" in the data?
sum(liga$Team == "Bor. Moenchengladbach")
```

```
[1] 2
```

```r
# show the entries
liga |>
  filter(Team == "Bor. Moenchengladbach")
```

```
# A tibble: 2 x 12
  Season Position Team         Played     W     D     L    GF    GA    GD Points
   <dbl>    <dbl> <chr>         <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
1   1989       15 Bor. Moench~     34    11     8    15    37    45    -8     41
2   1976        1 Bor. Moench~     34    17    10     7    58    34    24     61
# i 1 more variable: Pts_pre_95 <dbl>
```

```r
# Replace "Bor. Moenchengladbach" with "Borussia Moenchengladbach"
liga <- liga |>
  mutate(Team = ifelse(Team == "Bor. Moenchengladbach",
                       "Borussia Moenchengladbach",
                       Team)) |>
  mutate(Team = ifelse(Team == "SV Bayer 04 Leverkusen",
                       "TSV Bayer 04 Leverkusen",
                       Team)) |>
  mutate(Team = ifelse(Team == "FC Bayer 05 Uerdingen"
                       | Team == "Bayer 05 Uerdingen" ,
                       "KFC Uerdingen 05",
                       Team)) |>
  mutate(Team = ifelse(Team == "SV Stuttgarter Kickers",
                       "Stuttgarter Kickers",
                       Team))

# ------------------------------------

# Check for the data class
class(liga)
```

```
[1] "tbl_df"     "tbl"        "data.frame"
```

```
# view data
view(liga)

# Glimpse on the data
glimpse(liga)
```

```
Rows: 952
Columns: 12
$ Season     <dbl> 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015,~
$ Position   <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, ~
$ Team       <chr> "FC Bayern Muenchen", "Borussia Dortmund", "Bayer 04 Leverk~
$ Played     <dbl> 34, 34, 34, 34, 34, 34, 34, 34, 34, 34, 34, 34, 34, 34, 34,~
$ W          <dbl> 28, 24, 18, 17, 15, 14, 14, 12, 10, 11, 10, 9, 10, 9, 9, 9,~
$ D          <dbl> 4, 6, 6, 4, 7, 8, 8, 9, 13, 8, 10, 11, 8, 11, 10, 9, 6, 4, ~
$ L          <dbl> 2, 4, 10, 13, 12, 12, 12, 13, 11, 15, 14, 14, 16, 14, 15, 1~
$ GF         <dbl> 80, 82, 56, 67, 51, 46, 42, 47, 38, 40, 33, 42, 50, 38, 39,~
$ GA         <dbl> 17, 34, 40, 50, 49, 42, 42, 49, 42, 46, 42, 52, 65, 53, 54,~
$ GD         <dbl> 63, 48, 16, 17, 2, 4, 0, -2, -4, -6, -9, -10, -15, -15, -15~
$ Points     <dbl> 88, 78, 60, 55, 52, 50, 50, 45, 43, 41, 40, 38, 38, 38, 37,~
$ Pts_pre_95 <dbl> 60, 54, 42, 38, 37, 36, 36, 33, 33, 30, 30, 29, 28, 29, 28,~
```

```
# first and last observations
head(liga)
```

```
# A tibble: 6 x 12
```

```
  Season Position Team          Played     W     D     L    GF    GA    GD Points
   <dbl>    <dbl> <chr>          <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
1   2015        1 FC Bayern M~      34    28     4     2    80    17    63     88
2   2015        2 Borussia Do~      34    24     6     4    82    34    48     78
3   2015        3 Bayer 04 Le~      34    18     6    10    56    40    16     60
4   2015        4 Borussia Mo~      34    17     4    13    67    50    17     55
5   2015        5 FC Schalke ~      34    15     7    12    51    49     2     52
6   2015        6 1. FSV Main~      34    14     8    12    46    42     4     50
# i 1 more variable: Pts_pre_95 <dbl>
```

```
tail(liga)
```

```
# A tibble: 6 x 12
  Season Position Team          Played     W     D     L    GF    GA    GD Points
   <dbl>    <dbl> <chr>          <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
1   1963       11 Eintracht B~      30    11     6    13    36    49   -13     39
2   1963       12 1. FC Kaise~      30    10     6    14    48    69   -21     36
3   1963       13 Karlsruher ~      30     8     8    14    42    55   -13     32
4   1963       14 Hertha BSC        30     9     6    15    45    65   -20     33
5   1963       15 Preussen Mu~      30     7     9    14    34    52   -18     30
6   1963       16 1. FC Saarb~      30     6     5    19    44    72   -28     23
# i 1 more variable: Pts_pre_95 <dbl>
```

```
# summary statistics
summary(liga)
```

```
     Season        Position          Team               Played     
 Min.   :1963   Min.   : 1.000   Length:952         Min.   :30.00  
 1st Qu.:1976   1st Qu.: 5.000   Class :character   1st Qu.:34.00  
 Median :1989   Median : 9.000   Mode  :character   Median :34.00  
 Mean   :1989   Mean   : 9.486                      Mean   :33.95  
 3rd Qu.:2002   3rd Qu.:14.000                      3rd Qu.:34.00  
 Max.   :2015   Max.   :20.000                      Max.   :38.00  
       W               D                L               GF        
 Min.   : 2.00   Min.   : 2.000   Min.   : 1.00   Min.   : 15.00  
 1st Qu.: 9.75   1st Qu.: 7.000   1st Qu.:10.00   1st Qu.: 42.00  
 Median :12.00   Median : 9.000   Median :13.00   Median : 50.00  
 Mean   :12.61   Mean   : 8.733   Mean   :12.61   Mean   : 52.01  
 3rd Qu.:15.00   3rd Qu.:11.000   3rd Qu.:15.00   3rd Qu.: 61.00  
 Max.   :29.00   Max.   :18.000   Max.   :28.00   Max.   :101.00  
       GA             GD                Points        Pts_pre_95   
 Min.   :10.0   Min.   :-60.0000   Min.   :10.00   Min.   : 8.00  
 1st Qu.:43.0   1st Qu.:-13.0000   1st Qu.:38.00   1st Qu.:29.00  
 Median :51.0   Median : -2.0000   Median :44.00   Median :33.00  
 Mean   :51.7   Mean   :  0.3015   Mean   :46.56   Mean   :33.95
 3rd Qu.:60.0   3rd Qu.: 13.0000   3rd Qu.:55.00   3rd Qu.:39.00
 Max.   :93.0   Max.   : 80.0000   Max.   :91.00   Max.   :62.00
```

```
# How many teams have played in the league over the years?
table(liga$Season)
```

```
1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978
  16   16   18   18   18   18   18   18   18   18   18   18   18   18   18   18
1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994
  18   18   18   18   18   18   18   18   18   18   18   18   20   18   18   18
1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010
  18   18   18   18   18   18   18   18   18   18   18   18   18   18   18   18
2011 2012 2013 2014 2015
  18   18   18   18   18
```

```
# Which teams have played Bundesliga
unique(liga$Team)
```

```
 [1] "FC Bayern Muenchen"         "Borussia Dortmund"
 [3] "Bayer 04 Leverkusen"        "Borussia Moenchengladbach"
 [5] "FC Schalke 04"              "1. FSV Mainz 05"
 [7] "Hertha BSC"                 "VfL Wolfsburg"
 [9] "1. FC Koeln"                "Hamburger SV"
[11] "FC Ingolstadt 04"           "FC Augsburg"
[13] "Werder Bremen"              "SV Darmstadt 98"
[15] "TSG 1899 Hoffenheim"        "Eintracht Frankfurt"
[17] "VfB Stuttgart"              "Hannover 96"
[19] "SC Freiburg"                "SC Paderborn 07"
[21] "1. FC Nuernberg"            "Eintracht Braunschweig"
[23] "Fortuna Duesseldorf"        "SpVgg Greuther Fuerth"
[25] "1. FC Kaiserslautern"       "FC St. Pauli"
[27] "VfL Bochum"                 "Energie Cottbus"
[29] "Karlsruher SC"              "Arminia Bielefeld"
[31] "Hansa Rostock"              "MSV Duisburg"
[33] "Alemannia Aachen"           "TSV 1860 Muenchen"
[35] "SpVgg Unterhaching"         "SSV Ulm 1846"
[37] "KFC Uerdingen 05"           "Dynamo Dresden"
[39] "SG Wattenscheid 09"         "VfB Leipzig"
[41] "1. FC Saarbruecken"         "TSV Bayer 04 Leverkusen"
[43] "SV Werder Bremen"           "1. FC Dynamo Dresden"
[45] "Stuttgarter Kickers"        "FC Hansa Rostock"
[47] "SV Waldhof Mannheim"        "FC 08 Homburg"
```

```
[49] "FC Homburg"                "Blau-Weiss 90 Berlin"
[51] "Kickers Offenbach"         "Tennis Borussia Berlin"
[53] "Rot-Weiss Essen"           "Wuppertaler SV"
[55] "SC Fortuna Koeln"          "Rot-Weiss Oberhausen"
[57] "SC Rot-Weiss Oberhausen"   "Borussia Neunkirchen"
[59] "Meidericher SV"            "SC Tasmania 1900 Berlin"
[61] "Preussen Muenster"
```

```
# How many teams have played Bundesliga
n_distinct(liga$Team)
```

```
[1] 61
```

```
# How often has each team played in the Bundesliga
table(liga$Team)
```

```

     1. FC Dynamo Dresden      1. FC Kaiserslautern               1. FC Koeln
                        1                        44                        45
          1. FC Nuernberg        1. FC Saarbruecken           1. FSV Mainz 05
                       32                         5                        10
         Alemannia Aachen         Arminia Bielefeld       Bayer 04 Leverkusen
                        4                        17                        30
     Blau-Weiss 90 Berlin         Borussia Dortmund Borussia Moenchengladbach
                        1                        49                        48
     Borussia Neunkirchen            Dynamo Dresden    Eintracht Braunschweig
                        3                         3                        21
      Eintracht Frankfurt           Energie Cottbus             FC 08 Homburg
                       47                         6                         2
              FC Augsburg        FC Bayern Muenchen          FC Hansa Rostock
                        5                        51                         1
               FC Homburg          FC Ingolstadt 04             FC Schalke 04
                        1                         1                        48
             FC St. Pauli       Fortuna Duesseldorf              Hamburger SV
                        8                        23                        53
              Hannover 96             Hansa Rostock                Hertha BSC
                       28                        11                        33
            Karlsruher SC          KFC Uerdingen 05         Kickers Offenbach
                       24                        14                         7
           Meidericher SV              MSV Duisburg         Preussen Muenster
                        3                        25                         1
          Rot-Weiss Essen      Rot-Weiss Oberhausen          SC Fortuna Koeln
                        7                         3                         1
              SC Freiburg           SC Paderborn 07   SC Rot-Weiss Oberhausen
```

```
                      16                         1                         1
 SC Tasmania 1900 Berlin        SG Wattenscheid 09     SpVgg Greuther Fuerth
                       1                         4                         1
      SpVgg Unterhaching              SSV Ulm 1846       Stuttgarter Kickers
                       2                         1                         2
         SV Darmstadt 98       SV Waldhof Mannheim          SV Werder Bremen
                       3                         7                         1
  Tennis Borussia Berlin       TSG 1899 Hoffenheim         TSV 1860 Muenchen
                       2                         8                        20
 TSV Bayer 04 Leverkusen               VfB Leipzig             VfB Stuttgart
                       7                         1                        51
              VfL Bochum             VfL Wolfsburg             Werder Bremen
                      34                        19                        51
          Wuppertaler SV 
                       3 
```

```
# summary of variable Season only
summary(liga$Season)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   1963    1976    1989    1989    2002    2015 
```

```
# summary of numeric of variables (Team is a character)
liga |>
  select(Season, Position, Played, W, D, L, GF, GA, GD, Points, Pts_pre_95) |>
  summary()
```

```
     Season        Position          Played            W        
 Min.   :1963   Min.   : 1.000   Min.   :30.00   Min.   : 2.00  
 1st Qu.:1976   1st Qu.: 5.000   1st Qu.:34.00   1st Qu.: 9.75  
 Median :1989   Median : 9.000   Median :34.00   Median :12.00  
 Mean   :1989   Mean   : 9.486   Mean   :33.95   Mean   :12.61  
 3rd Qu.:2002   3rd Qu.:14.000   3rd Qu.:34.00   3rd Qu.:15.00  
 Max.   :2015   Max.   :20.000   Max.   :38.00   Max.   :29.00  
       D                L               GF               GA      
 Min.   : 2.000   Min.   : 1.00   Min.   : 15.00   Min.   :10.0  
 1st Qu.: 7.000   1st Qu.:10.00   1st Qu.: 42.00   1st Qu.:43.0  
 Median : 9.000   Median :13.00   Median : 50.00   Median :51.0  
 Mean   : 8.733   Mean   :12.61   Mean   : 52.01   Mean   :51.7  
 3rd Qu.:11.000   3rd Qu.:15.00   3rd Qu.: 61.00   3rd Qu.:60.0  
 Max.   :18.000   Max.   :28.00   Max.   :101.00   Max.   :93.0  
       GD                Points        Pts_pre_95   
 Min.   :-60.0000   Min.   :10.00   Min.   : 8.00  
 1st Qu.:-13.0000   1st Qu.:38.00   1st Qu.:29.00  
 Median : -2.0000   Median :44.00   Median :33.00  
 Mean   :  0.3015   Mean   :46.56   Mean   :33.95  
 3rd Qu.: 13.0000   3rd Qu.:55.00   3rd Qu.:39.00  
 Max.   : 80.0000   Max.   :91.00   Max.   :62.00  
```

```
# shorter alternative
liga |>
  select(Season, Position, Played:Pts_pre_95) |>
  summary()
```

```
     Season        Position          Played            W        
 Min.   :1963   Min.   : 1.000   Min.   :30.00   Min.   : 2.00  
 1st Qu.:1976   1st Qu.: 5.000   1st Qu.:34.00   1st Qu.: 9.75  
 Median :1989   Median : 9.000   Median :34.00   Median :12.00  
 Mean   :1989   Mean   : 9.486   Mean   :33.95   Mean   :12.61  
 3rd Qu.:2002   3rd Qu.:14.000   3rd Qu.:34.00   3rd Qu.:15.00  
 Max.   :2015   Max.   :20.000   Max.   :38.00   Max.   :29.00  
       D                L               GF               GA      
 Min.   : 2.000   Min.   : 1.00   Min.   : 15.00   Min.   :10.0  
 1st Qu.: 7.000   1st Qu.:10.00   1st Qu.: 42.00   1st Qu.:43.0  
 Median : 9.000   Median :13.00   Median : 50.00   Median :51.0  
 Mean   : 8.733   Mean   :12.61   Mean   : 52.01   Mean   :51.7  
 3rd Qu.:11.000   3rd Qu.:15.00   3rd Qu.: 61.00   3rd Qu.:60.0  
 Max.   :18.000   Max.   :28.00   Max.   :101.00   Max.   :93.0  
       GD                Points        Pts_pre_95   
 Min.   :-60.0000   Min.   :10.00   Min.   : 8.00  
 1st Qu.:-13.0000   1st Qu.:38.00   1st Qu.:29.00  
 Median : -2.0000   Median :44.00   Median :33.00  
 Mean   :  0.3015   Mean   :46.56   Mean   :33.95  
 3rd Qu.: 13.0000   3rd Qu.:55.00   3rd Qu.:39.00  
 Max.   : 80.0000   Max.   :91.00   Max.   :62.00  
```

```
# shortest alternative
liga |>
  select(-Team) |>
  filter(Season == 1999 |  Season == 2010) |>
  summary()
```

```
     Season        Position       Played         W               D         
 Min.   :1999   Min.   : 1.0   Min.   :34   Min.   : 4.00   Min.   : 3.000  
 1st Qu.:1999   1st Qu.: 5.0   1st Qu.:34   1st Qu.: 9.75   1st Qu.: 6.000  
 Median :2004   Median : 9.5   Median :34   Median :12.00   Median : 8.000  
 Mean   :2004   Mean   : 9.5   Mean   :34   Mean   :12.83   Mean   : 8.333  
 3rd Qu.:2010   3rd Qu.:14.0   3rd Qu.:34   3rd Qu.:14.25   3rd Qu.:10.250  
 Max.   :2010   Max.   :18.0   Max.   :34   Max.   :23.00   Max.   :15.000  
       L               GF              GA              GD        
 Min.   : 3.00   Min.   :31.00   Min.   :22.00   Min.   :-34.00  
 1st Qu.:10.75   1st Qu.:41.00   1st Qu.:44.00   1st Qu.:-10.25  
 Median :13.00   Median :47.00   Median :48.50   Median : -3.00  
 Mean   :12.83   Mean   :49.42   Mean   :49.42   Mean   :  0.00  
 3rd Qu.:16.00   3rd Qu.:54.25   3rd Qu.:59.00   3rd Qu.:  4.75  
 Max.   :21.00   Max.   :81.00   Max.   :71.00   Max.   : 45.00  
     Points        Pts_pre_95  
 Min.   :22.00   Min.   :18.00  
 1st Qu.:39.75   1st Qu.:29.75  
 Median :44.00   Median :32.00  
 Mean   :46.83   Mean   :34.00  
 3rd Qu.:50.75   3rd Qu.:37.50  
 Max.   :75.00   Max.   :52.00  
```

```r
# Most points ever received by a team
liga |>
  filter(Points == max(Points))
```

```
# A tibble: 1 x 12
  Season Position Team          Played     W     D     L    GF    GA    GD Points
   <dbl>    <dbl> <chr>         <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
1   2012        1 FC Bayern M~     34    29     4     1    98    18    80     91
# i 1 more variable: Pts_pre_95 <dbl>
```

```r
# Show only the team name
liga |>
  filter(Points == max(Points))|>
  select(Team) |>
  print()
```

```
# A tibble: 1 x 1
  Team              
  <chr>             
1 FC Bayern Muenchen
```

```r
# remove the variable `Pts_pre_95` from the data
liga_post95 <- liga |>
  select(-Pts_pre_95)

# rename W, D, and L to Win, Draw, and Loss
# additionally rename GF, GA, GD to Goals_shot, Goals_received, Goal_difference
liga_longnames <- liga |>
```

```
  rename(Win = W, Draw = D, Loss = L) |>
  rename(Goals_shot = GF, Goals_received = GA, Goal_difference = GD)

# Remove the variable `Pts_pre_95` from `liga`
# additionally remove all observations before the year 1996
liga_no3point <- liga |>
  select(-Pts_pre_95) |>
  filter(Season >= 1996)

# Remove the objects liga_post95, liga_longnames, and liga_no3point from the environment
rm(liga_post95, liga_longnames, liga_no3point)

# Rename all variables of `liga`to lower cases and store it as `dfb`
dfb <- liga |>
  rename_all(tolower)

# Show the winner and the runner up after 2010
# additionally show the points
dfb |>
  filter(season > 2010) |>
  group_by(season) |>
  arrange(desc(points)) %>%
  slice_head(n = 2) %>%
  select(team, points, position)
```

```
Adding missing grouping variables: `season`
```

```
# A tibble: 10 x 4
# Groups:   season [5]
   season team                points position
    <dbl> <chr>                <dbl>    <dbl>
 1   2011 Borussia Dortmund       81        1
 2   2011 FC Bayern Muenchen      73        2
 3   2012 FC Bayern Muenchen      91        1
 4   2012 Borussia Dortmund       66        2
 5   2013 FC Bayern Muenchen      90        1
 6   2013 Borussia Dortmund       71        2
 7   2014 FC Bayern Muenchen      79        1
 8   2014 VfL Wolfsburg           69        2
 9   2015 FC Bayern Muenchen      88        1
10   2015 Borussia Dortmund       78        2
```

```
# Create a variable that counts how often a team was ranked first
dfb <- dfb |>
```

```r
  group_by(team) |>
  mutate(meister_count = sum(position == 1))

# How often has each team played in the Bundesliga
table(liga$Team)
```

```
     1. FC Dynamo Dresden      1. FC Kaiserslautern               1. FC Koeln
                        1                        44                        45
          1. FC Nuernberg        1. FC Saarbruecken           1. FSV Mainz 05
                       32                         5                        10
         Alemannia Aachen         Arminia Bielefeld       Bayer 04 Leverkusen
                        4                        17                        30
     Blau-Weiss 90 Berlin         Borussia Dortmund Borussia Moenchengladbach
                        1                        49                        48
     Borussia Neunkirchen            Dynamo Dresden    Eintracht Braunschweig
                        3                         3                        21
      Eintracht Frankfurt           Energie Cottbus             FC 08 Homburg
                       47                         6                         2
              FC Augsburg        FC Bayern Muenchen          FC Hansa Rostock
                        5                        51                         1
               FC Homburg          FC Ingolstadt 04             FC Schalke 04
                        1                         1                        48
             FC St. Pauli       Fortuna Duesseldorf              Hamburger SV
                        8                        23                        53
              Hannover 96             Hansa Rostock                Hertha BSC
                       28                        11                        33
            Karlsruher SC          KFC Uerdingen 05         Kickers Offenbach
                       24                        14                         7
           Meidericher SV              MSV Duisburg         Preussen Muenster
                        3                        25                         1
          Rot-Weiss Essen      Rot-Weiss Oberhausen          SC Fortuna Koeln
                        7                         3                         1
              SC Freiburg           SC Paderborn 07   SC Rot-Weiss Oberhausen
                       16                         1                         1
  SC Tasmania 1900 Berlin        SG Wattenscheid 09     SpVgg Greuther Fuerth
                        1                         4                         1
       SpVgg Unterhaching              SSV Ulm 1846       Stuttgarter Kickers
                        2                         1                         2
          SV Darmstadt 98       SV Waldhof Mannheim          SV Werder Bremen
                        3                         7                         1
   Tennis Borussia Berlin       TSG 1899 Hoffenheim         TSV 1860 Muenchen
                        2                         8                        20
  TSV Bayer 04 Leverkusen               VfB Leipzig             VfB Stuttgart
                        7                         1                        51
               VfL Bochum             VfL Wolfsburg            Werder Bremen 
                       34                        19                       51 
           Wuppertaler SV 
                        3 
```

```
# Make a ranking
dfb |>
  group_by(team) |>
  summarise(appearances = n_distinct(season)) |>
  arrange(desc(appearances)) |>
  print(n = Inf)
```

```
# A tibble: 61 x 2
   team                       appearances
   <chr>                            <int>
 1 Hamburger SV                        53
 2 FC Bayern Muenchen                  51
 3 VfB Stuttgart                       51
 4 Werder Bremen                       51
 5 Borussia Dortmund                   49
 6 Borussia Moenchengladbach           48
 7 FC Schalke 04                       48
 8 Eintracht Frankfurt                 47
 9 1. FC Koeln                         45
10 1. FC Kaiserslautern                44
11 VfL Bochum                          34
12 Hertha BSC                          33
13 1. FC Nuernberg                     32
14 Bayer 04 Leverkusen                 30
15 Hannover 96                         28
16 MSV Duisburg                        25
17 Karlsruher SC                       24
18 Fortuna Duesseldorf                 23
19 Eintracht Braunschweig              21
20 TSV 1860 Muenchen                   20
21 VfL Wolfsburg                       19
22 Arminia Bielefeld                   17
23 SC Freiburg                         16
24 KFC Uerdingen 05                    14
25 Hansa Rostock                       11
26 1. FSV Mainz 05                     10
27 FC St. Pauli                         8
28 TSG 1899 Hoffenheim                  8
29 Kickers Offenbach                    7
30 Rot-Weiss Essen                      7
```

```
31 SV Waldhof Mannheim                  7
32 TSV Bayer 04 Leverkusen              7
33 Energie Cottbus                      6
34 1. FC Saarbruecken                   5
35 FC Augsburg                          5
36 Alemannia Aachen                     4
37 SG Wattenscheid 09                   4
38 Borussia Neunkirchen                 3
39 Dynamo Dresden                       3
40 Meidericher SV                       3
41 Rot-Weiss Oberhausen                 3
42 SV Darmstadt 98                      3
43 Wuppertaler SV                       3
44 FC 08 Homburg                        2
45 SpVgg Unterhaching                   2
46 Stuttgarter Kickers                  2
47 Tennis Borussia Berlin               2
48 1. FC Dynamo Dresden                 1
49 Blau-Weiss 90 Berlin                 1
50 FC Hansa Rostock                     1
51 FC Homburg                           1
52 FC Ingolstadt 04                     1
53 Preussen Muenster                    1
54 SC Fortuna Koeln                     1
55 SC Paderborn 07                      1
56 SC Rot-Weiss Oberhausen              1
57 SC Tasmania 1900 Berlin              1
58 SSV Ulm 1846                         1
59 SV Werder Bremen                     1
60 SpVgg Greuther Fuerth                1
61 VfB Leipzig                          1
```

```r
# Add a variable to `dfb` that contains the number of appearances of a team in the league
dfb <- dfb |>
  group_by(team) |>
  mutate(appearances = n_distinct(season))

# create a number that indicates how often a team has played Bundesliga in a given year
dfb <- dfb |>
  arrange(team, season) |>
  group_by(team) |>
  mutate(team_in_liga_count = row_number())

# Make a ranking with the number of titles of all teams that ever won the league
dfb |>
```

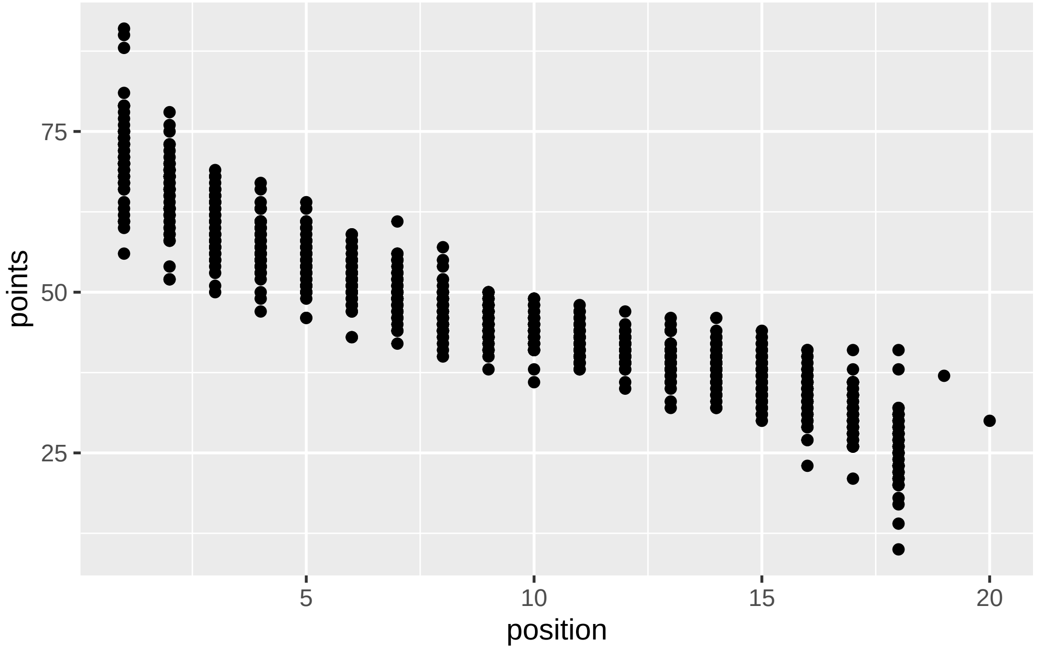
```
  filter(team_in_liga_count == 1) |>
  filter(meister_count != 0) |>
  arrange(desc(meister_count)) |>
  select(meister_count, team)
```

```
# A tibble: 12 x 2
# Groups:   team [12]
   meister_count team
           <int> <chr>
 1            25 FC Bayern Muenchen
 2             5 Borussia Dortmund
 3             5 Borussia Moenchengladbach
 4             4 Werder Bremen
 5             3 Hamburger SV
 6             3 VfB Stuttgart
 7             2 1. FC Kaiserslautern
 8             2 1. FC Koeln
 9             1 1. FC Nuernberg
10             1 Eintracht Braunschweig
11             1 TSV 1860 Muenchen
12             1 VfL Wolfsburg
```

```
# Create a numeric identifying variable for each team
dfb_teamid <- dfb |>
  mutate(team_id = as.numeric(factor(team)))

# When a team is in the league, what is the probability that it wins the league
dfb |>
  filter(team_in_liga_count == 1) |>
  mutate(prob_win = meister_count/appearances) |>
  filter(prob_win > 0) |>
  arrange(desc(prob_win)) |>
  select(meister_count, prob_win, team)
```

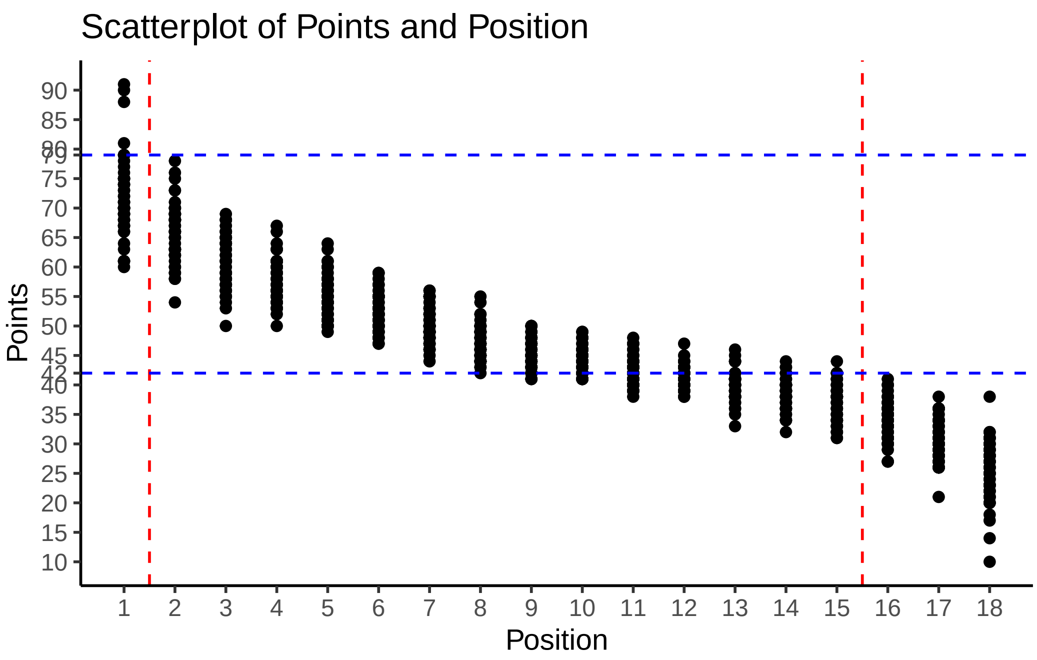
```
# A tibble: 12 x 3
# Groups:   team [12]
   meister_count prob_win team
           <int>    <dbl> <chr>
 1            25   0.490  FC Bayern Muenchen
 2             5   0.104  Borussia Moenchengladbach
 3             5   0.102  Borussia Dortmund
 4             4   0.0784 Werder Bremen
 5             3   0.0588 VfB Stuttgart
 6             3   0.0566 Hamburger SV
 7             1   0.0526 VfL Wolfsburg
 8             1   0.05   TSV 1860 Muenchen
 9             1   0.0476 Eintracht Braunschweig
10             2   0.0455 1. FC Kaiserslautern
11             2   0.0444 1. FC Koeln
12             1   0.0312 1. FC Nuernberg
```

```
# make a scatterplot with points on the y-axis and position on the x-axis
ggplot(dfb, aes(x = position, y = points)) +
  geom_point()
```



```
# Make a scatterplot with points on the y-axis and position on the x-axis.
# Additionally, only consider seasons with 18 teams and
# add lines that make clear how many points you needed to be placed
# in between rank 2 and 15.
dfb_18 <- dfb |>
  group_by(season) |>
  mutate(teams_in_league = n_distinct(team)) |>
  filter(teams_in_league == 18)

h_1 <- dfb_18 |>
  filter(position == 16) |>
  mutate(ma = max(points))

max_points_rank_16 <- max(h_1$ma) +1
```

```
h_2 <- dfb_18 |>
  filter(position == 2)  |>
  mutate(mb = max(points))

min_points_rank_2 <- max(h_2$mb) + 1

dfb_18 <- dfb_18 |>
  mutate(season_category = case_when(
    season < 1970 ~ 1,
    between(season, 1970, 1979) ~ 2,
    between(season, 1980, 1989) ~ 3,
    between(season, 1990, 1999) ~ 4,
    between(season, 2000, 2009) ~ 5,
    between(season, 2010, 2019) ~ 6,
    TRUE ~ 7  # Adjust this line based on the actual range of your data
  ))

ggplot(dfb_18, aes(x = position, y = points)) +
  geom_point() +
  labs(title = "Scatterplot of Points and Position",
       x = "Position",
       y = "Points") +
  geom_vline(xintercept = c(1.5, 15.5), linetype = "dashed", color = "red") +
  geom_hline(yintercept = max_points_rank_16, linetype = "dashed", color = "blue") +
  geom_hline(yintercept = min_points_rank_2, linetype = "dashed", color = "blue") +
  scale_y_continuous(breaks = c(min_points_rank_2, max_points_rank_16, seq(0, max(dfb_18$p
  scale_x_continuous(breaks = c(seq(0, max(dfb_18$points), by = 1))) +
  theme_classic()
```



```r
# Remove all objects except liga and dfb
rm(list=setdiff(ls(), c("liga", "dfb")))

# Rank "1. FC Kaiserslautern" over time
dfb_bal <- dfb |>
  select(season, team, position) |>
  as_tibble() |>
  complete(season, team)

table(dfb_bal$team)
```

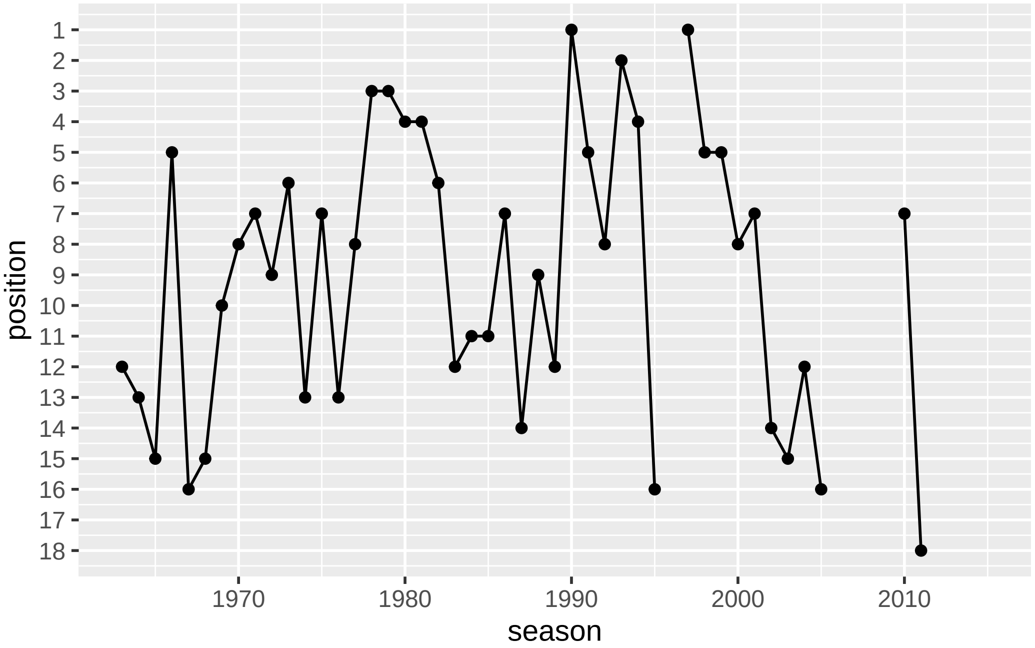
```
     1. FC Dynamo Dresden      1. FC Kaiserslautern               1. FC Koeln
                       53                        53                        53
          1. FC Nuernberg        1. FC Saarbruecken           1. FSV Mainz 05
                       53                        53                        53
         Alemannia Aachen         Arminia Bielefeld       Bayer 04 Leverkusen
                       53                        53                        53
     Blau-Weiss 90 Berlin         Borussia Dortmund Borussia Moenchengladbach
                       53                        53                        53
     Borussia Neunkirchen            Dynamo Dresden    Eintracht Braunschweig
                       53                        53                        53
      Eintracht Frankfurt           Energie Cottbus             FC 08 Homburg
                       53                        53                        53
              FC Augsburg        FC Bayern Muenchen          FC Hansa Rostock
```

```
                       53                        53                        53
               FC Homburg          FC Ingolstadt 04             FC Schalke 04
                       53                        53                        53
             FC St. Pauli       Fortuna Duesseldorf              Hamburger SV
                       53                        53                        53
              Hannover 96             Hansa Rostock                Hertha BSC
                       53                        53                        53
            Karlsruher SC          KFC Uerdingen 05         Kickers Offenbach
                       53                        53                        53
           Meidericher SV              MSV Duisburg         Preussen Muenster
                       53                        53                        53
          Rot-Weiss Essen      Rot-Weiss Oberhausen          SC Fortuna Koeln
                       53                        53                        53
              SC Freiburg           SC Paderborn 07   SC Rot-Weiss Oberhausen
                       53                        53                        53
  SC Tasmania 1900 Berlin        SG Wattenscheid 09     SpVgg Greuther Fuerth
                       53                        53                        53
       SpVgg Unterhaching              SSV Ulm 1846       Stuttgarter Kickers
                       53                        53                        53
          SV Darmstadt 98       SV Waldhof Mannheim          SV Werder Bremen
                       53                        53                        53
   Tennis Borussia Berlin       TSG 1899 Hoffenheim          TSV 1860 Muenchen
                       53                        53                        53
  TSV Bayer 04 Leverkusen               VfB Leipzig             VfB Stuttgart
                       53                        53                        53
               VfL Bochum             VfL Wolfsburg             Werder Bremen
                       53                        53                        53
           Wuppertaler SV
                       53
```

```r
dfb_fck <- dfb_bal |>
  filter(team == "1. FC Kaiserslautern")

ggplot(dfb_fck, aes(x = season, y = position)) +
  geom_point() +
  geom_line() +
  scale_y_reverse(breaks = seq(1, 18, by = 1))
```

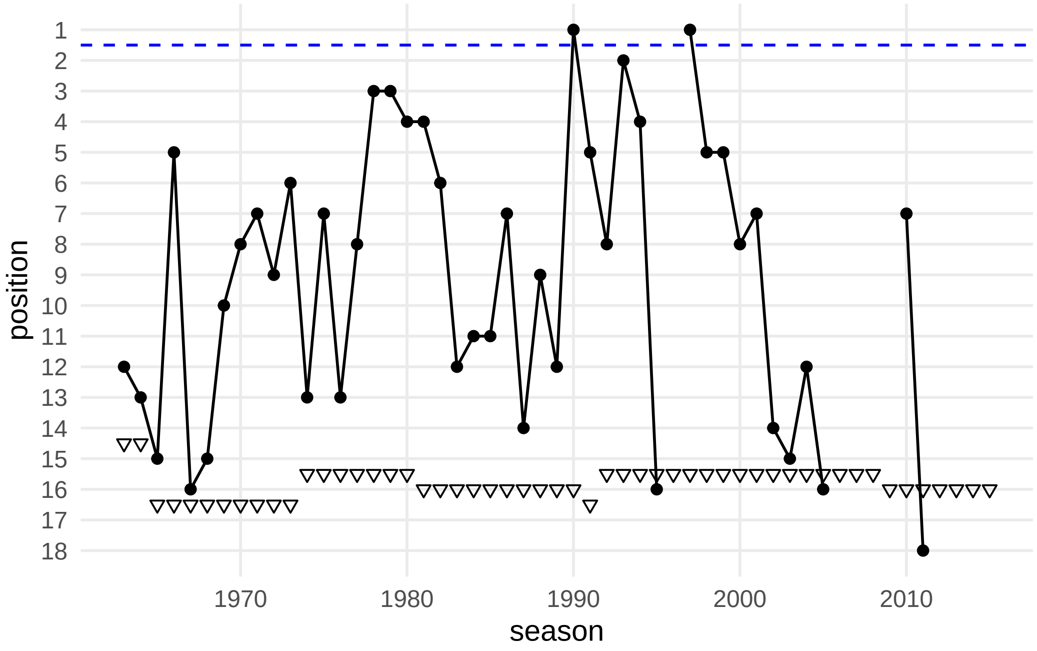
```
Warning: Removed 9 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 4 rows containing missing values or values outside the scale range
(`geom_line()`).
```



```
# Make the plot nice

# consider different rules for having to leave the league:
dfb_fck <- dfb_fck |>
  mutate(godown = ifelse(season <= 1964, 14.5, NA)) |>
  mutate(godown = ifelse(season > 1964 & season <= 1973, 16.5, godown)) |>
  mutate(godown = ifelse(season > 1973 & season <= 1980, 15.5, godown)) |>
  mutate(godown = ifelse(season > 1980 & season <= 1990, 16, godown)) |>
  mutate(godown = ifelse(season == 1991, 16.5, godown)) |>
  mutate(godown = ifelse(season > 1991 & season <= 2008, 15.5, godown)) |>
  mutate(godown = ifelse(season > 2008 , 16, godown))


ggplot(dfb_fck, aes(x = season)) +
  geom_point(aes(y = position)) +
  geom_line(aes(y = position)) +
  geom_point(aes(y = godown), shape = 25) +
  scale_y_reverse(breaks = seq(1, 18, by = 1)) +
  theme_minimal() +
  theme(panel.grid.minor = element_blank()) +
  geom_hline(yintercept = 1.5, linetype = "dashed", color = "blue")
```

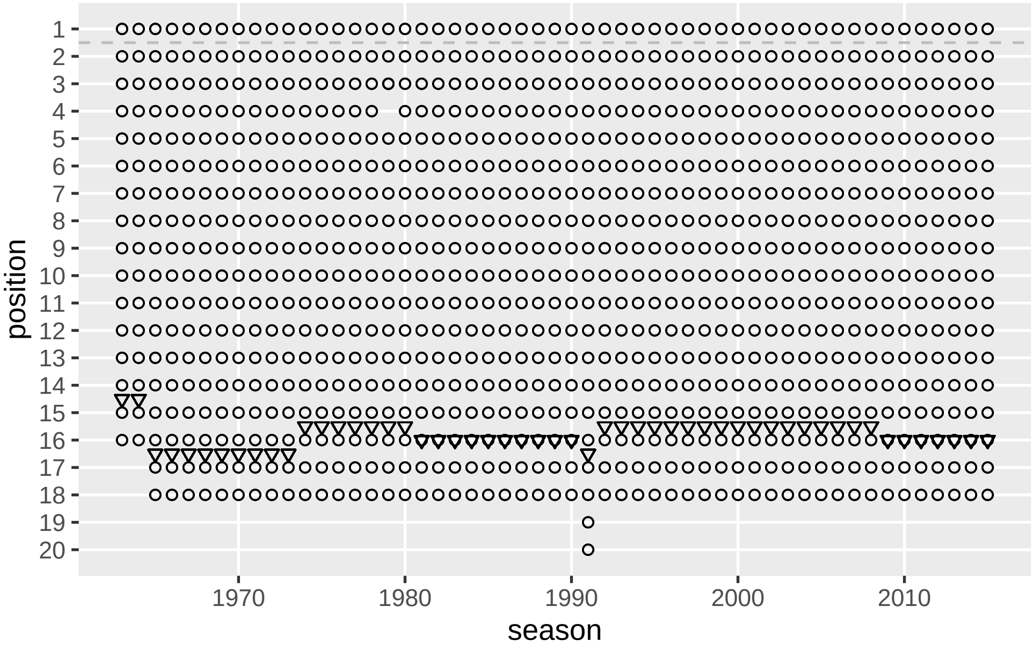
```
Warning: Removed 9 rows containing missing values or values outside the scale range
(`geom_point()`).
Removed 4 rows containing missing values or values outside the scale range
(`geom_line()`).
```



```
dfb_bal <- dfb_bal |>
  mutate(godown = ifelse(season <= 1964, 14.5, NA)) |>
  mutate(godown = ifelse(season > 1964 & season <= 1973, 16.5, godown)) |>
  mutate(godown = ifelse(season > 1973 & season <= 1980, 15.5, godown)) |>
  mutate(godown = ifelse(season > 1980 & season <= 1990, 16, godown)) |>
  mutate(godown = ifelse(season == 1991, 16.5, godown)) |>
  mutate(godown = ifelse(season > 1991 & season <= 2008, 15.5, godown)) |>
  mutate(godown = ifelse(season > 2008 , 16, godown)) |>
  mutate(inliga = ifelse(is.na(position), 0, 1))



rank_plot <- ggplot(dfb_bal, aes(x = season)) +
  geom_point(aes(y = position), shape = 1) +
  # geom_line(aes(y = position)) +
  geom_point(aes(y = godown), shape = 25) +
  scale_y_reverse(breaks = seq(1, 20, by = 1) , limits = c(20, 1)) +
  xlim(1963, 2015) +
  theme(panel.grid.minor = element_blank()) +
  geom_hline(yintercept = 1.5, linetype = "dashed", color = "gray") +
  geom_point(aes(y = position), shape = 1)

rank_plot
```

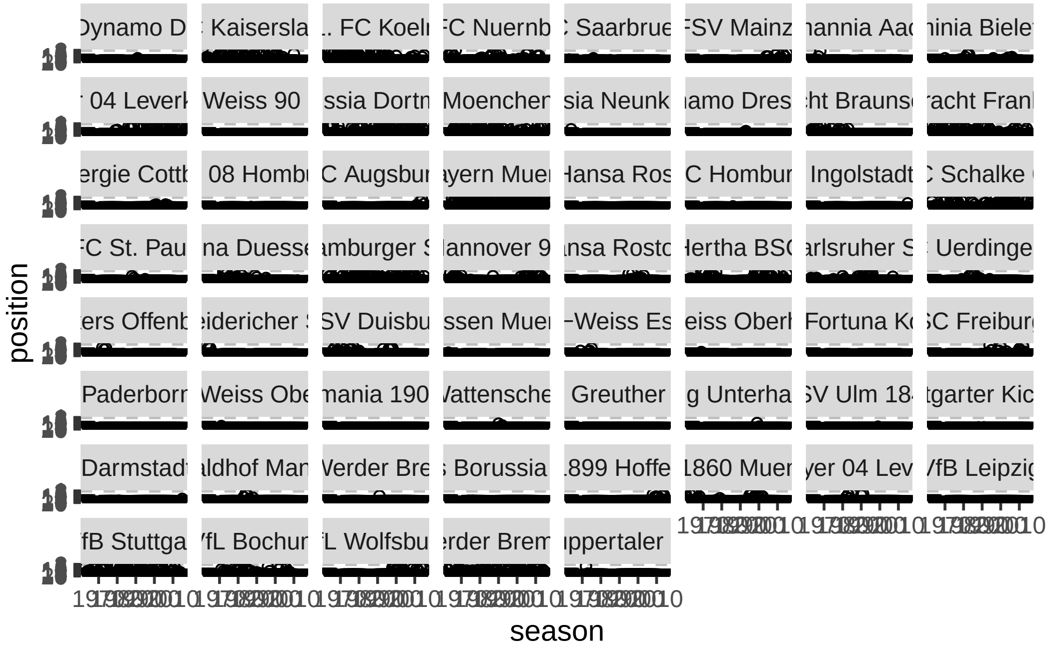
```
Warning: Removed 2281 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 2281 rows containing missing values or values outside the scale range
(`geom_point()`).
```



```
# !--> in 1979 is a gap! Error?
# No. Reason: two clubs shared the third place.

rank_plot +
  facet_wrap(~team)
```

```
Warning: Removed 2281 rows containing missing values or values outside the scale range
(`geom_point()`).
Removed 2281 rows containing missing values or values outside the scale range
(`geom_point()`).
```



```
# Create "test" directory if it doesn't already exist
if (!dir.exists("test")) {
  dir.create("test")
}


plots <- list()
for (club in unique(dfb_bal$team)) {
  dfb_subset <- subset(dfb_bal, team == club)

  p <- ggplot(dfb_subset, aes(x = season)) +
    geom_point(aes(y = position), shape = 15) +
    geom_line(aes(y = position)) +
    geom_point(aes(y = godown), shape = 25) +
    scale_y_reverse(breaks = seq(1, 20, by = 1) , limits = c(20, 1)) +
    xlim(1963, 2015) +
    theme(panel.grid.minor = element_blank()) +
    geom_hline(yintercept = 1.5, linetype = "dashed", color = "gray") +
    geom_point(aes(y = position), shape = 1) +
    labs(title = paste("Ranking History:", club))
  ggsave(filename=paste("test/r_",club,".png",sep=""))
  plots[[club]] <- p
}
```

```
Saving 5.5 x 3.5 in image
```

```
Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 9 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 4 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 9 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 8 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 8 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 21 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 21 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Saving 5.5 x 3.5 in image

Warning: Removed 48 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 23 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 48 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 43 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 41 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 43 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 49 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 13 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 49 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 36 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 14 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
Warning: Removed 36 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 23 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 16 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 23 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 4 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 4 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 5 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 5 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 49 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 32 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 32 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image
```

```
Warning: Removed 6 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 6 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 47 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 44 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 47 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 51 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 51 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 48 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 48 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 48 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image
```

```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).
Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).
Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 5 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 5 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 45 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 19 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 45 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 30 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 6 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 30 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image
Saving 5.5 x 3.5 in image

Warning: Removed 25 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 1 row containing missing values or values outside the scale range
(`geom_line()`).
```

```
Warning: Removed 25 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Saving 5.5 x 3.5 in image
```

```
Warning: Removed 42 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 40 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
Warning: Removed 42 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Saving 5.5 x 3.5 in image
```

```
Warning: Removed 20 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 20 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Saving 5.5 x 3.5 in image
```

```
Warning: Removed 29 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 7 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
Warning: Removed 29 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Saving 5.5 x 3.5 in image
```

```
Warning: Removed 39 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 32 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 39 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 46 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 37 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 46 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 28 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 11 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 28 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image
```

```
Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 46 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 42 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 46 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 49 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 37 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 31 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 37 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).
Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```

```
Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).
Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 49 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 49 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 49 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Saving 5.5 x 3.5 in image

Warning: Removed 51 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 51 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 51 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 51 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 49 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 51 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 15 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Saving 5.5 x 3.5 in image
```

```
Warning: Removed 46 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 46 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
Warning: Removed 46 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Saving 5.5 x 3.5 in image
```

```
Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```

```
Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Saving 5.5 x 3.5 in image
```

```
Warning: Removed 51 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
Warning: Removed 51 rows containing missing values or values outside the scale range
(`geom_point()`).
```

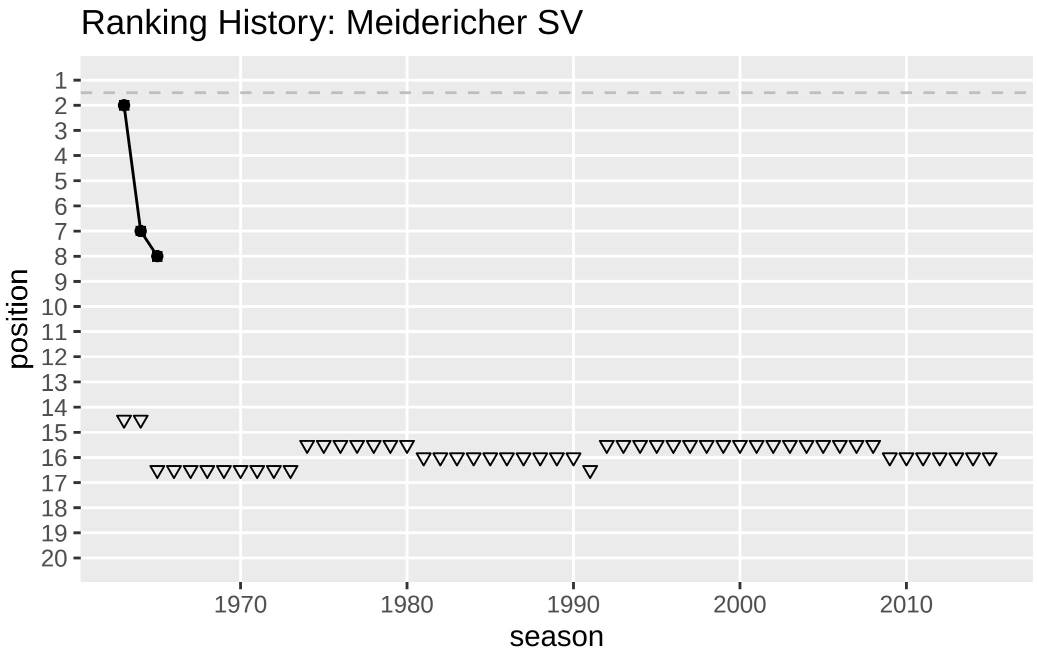
```
Saving 5.5 x 3.5 in image

Warning: Removed 45 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 45 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 45 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 33 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 12 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 33 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 46 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 42 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 46 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?

Warning: Removed 52 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 19 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 14 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 19 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 34 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 34 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 34 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).
```

```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).

Saving 5.5 x 3.5 in image

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).
```

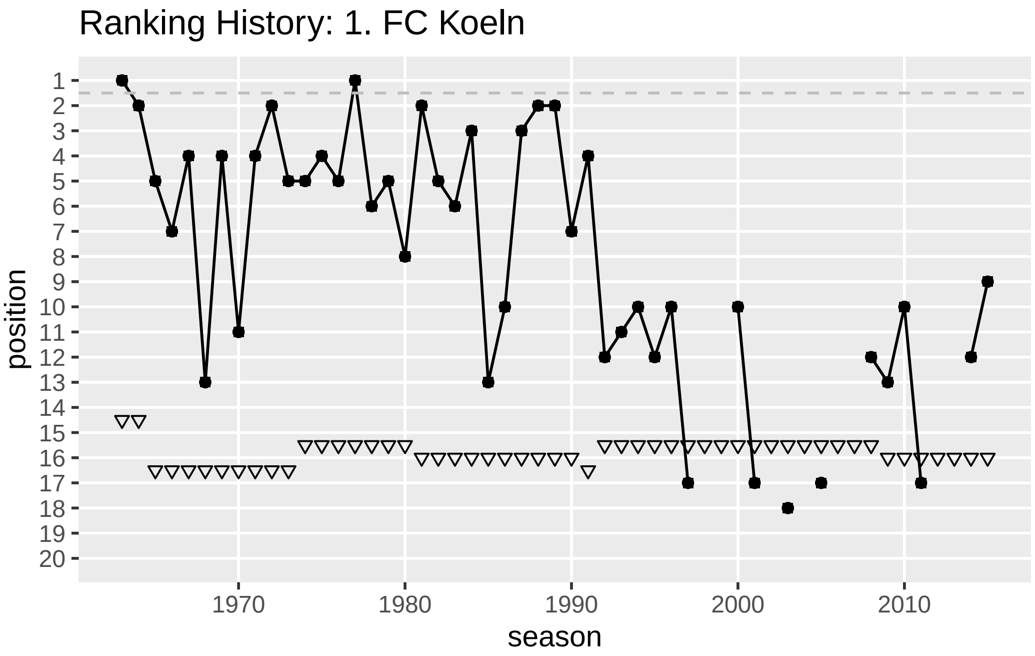
```
print(plots$`Meidericher SV`)
```

```
Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_line()`).

Warning: Removed 50 rows containing missing values or values outside the scale range
(`geom_point()`).
```



```
print(plots$`1. FC Koeln`)
```

```
Warning: Removed 8 rows containing missing values or values outside the scale range
(`geom_point()`).

Warning: Removed 8 rows containing missing values or values outside the scale range
(`geom_point()`).
```



```
# unload packages
suppressMessages(pacman::p_unload(
  bundesligR,
  tidyverse
))

# Remove the "test" directory and its contents after saving all graphs
unlink("test", recursive = TRUE)
```

2.14 exe_okun_solution.R

```
# setwd("/home/sthu/Dropbox/hsf/exams/22-11/scr/")

rm(list=ls())
```

```r
# load packages
if (!require(pacman)) install.packages("pacman")
pacman::p_load(tidyverse, ggpubr, sjPlot)

load(url("https://github.com/hubchev/courses/raw/main/dta/forest.Rdata"))

head(df,8)
```

```
# A tibble: 8 x 11
# Groups:   country.x [1]
  country.x     date     gdp gdp_growth unemployment region income forest    pop
  <chr>        <dbl>   <dbl>      <dbl>        <dbl> <chr>  <chr>   <dbl>  <dbl>
1 United Arab~  1992 1.26e11     -2.48          1.84 Middl~ High ~   3.63 2.05e6
2 United Arab~  1993 1.27e11     -4.34          1.85 Middl~ High ~   3.72 2.17e6
3 United Arab~  1994 1.36e11      1.25          1.81 Middl~ High ~   3.81 2.29e6
4 United Arab~  1995 1.45e11      1.35          1.80 Middl~ High ~   3.90 2.42e6
5 United Arab~  1996 1.54e11      0.631         1.90 Middl~ High ~   3.99 2.54e6
6 United Arab~  1997 1.66e11      2.83          1.98 Middl~ High ~   4.08 2.67e6
7 United Arab~  1998 1.67e11     -4.77          2.14 Middl~ High ~   4.18 2.81e6
8 United Arab~  1999 1.72e11     -2.40          2.22 Middl~ High ~   4.27 2.97e6
# i 2 more variables: unemployment_dif <dbl>, gdppc <dbl>
```

```r
tail(df,1)
```

```
# A tibble: 1 x 11
# Groups:   country.x [1]
  country.x  date        gdp gdp_growth unemployment region income forest    pop
  <chr>     <dbl>      <dbl>      <dbl>        <dbl> <chr>  <chr>   <dbl>  <dbl>
1 Zimbabwe   2020    1.94e10      -7.62         5.35 Sub-S~ Lower~   45.1 1.49e7
# i 2 more variables: unemployment_dif <dbl>, gdppc <dbl>
```

```r
 # panel data set
 # date and country.x

observations_df <- dim(df)

df <- rename(df, nation=country.x)
df <- rename(df, year=date)

df <- df %>%
  select(nation, year, gdp, pop, gdppc, unemployment)

df <- df %>%
  mutate(gdp_pc = gdp/pop)

df <- df %>% filter(nation=="Germany" | nation=="France")

df  %>%
  group_by(nation) %>%
  summarise(mean(unemployment), mean(gdppc))
```

```
# A tibble: 2 x 3
  nation  `mean(unemployment)` `mean(gdppc)`
  <chr>                  <dbl>         <dbl>
1 France                  9.75        34356.
2 Germany                 7.22        36739.
```

```
df  %>%
  filter(year==2020) %>%
  group_by(nation) %>%
  summarise(mean(unemployment), mean(gdppc))
```

```
# A tibble: 2 x 3
  nation  `mean(unemployment)` `mean(gdppc)`
  <chr>                  <dbl>         <dbl>
1 France                  8.01        35786.
2 Germany                 3.81        41315.
```

```
df  %>%
  group_by(nation) %>%
  summarise(max(unemployment), max(gdppc))
```

```
# A tibble: 2 x 3
  nation  `max(unemployment)` `max(gdppc)`
  <chr>                 <dbl>        <dbl>
1 France                 12.6       38912.
2 Germany                11.2       43329.
```

```
df %>%
  group_by(nation) %>%
  summarise(sd(gdppc), sd(unemployment))
```

```
# A tibble: 2 x 3
  nation  `sd(gdppc)` `sd(unemployment)`
  <chr>         <dbl>              <dbl>
1 France        2940.               1.58
2 Germany       4015.               2.37
```

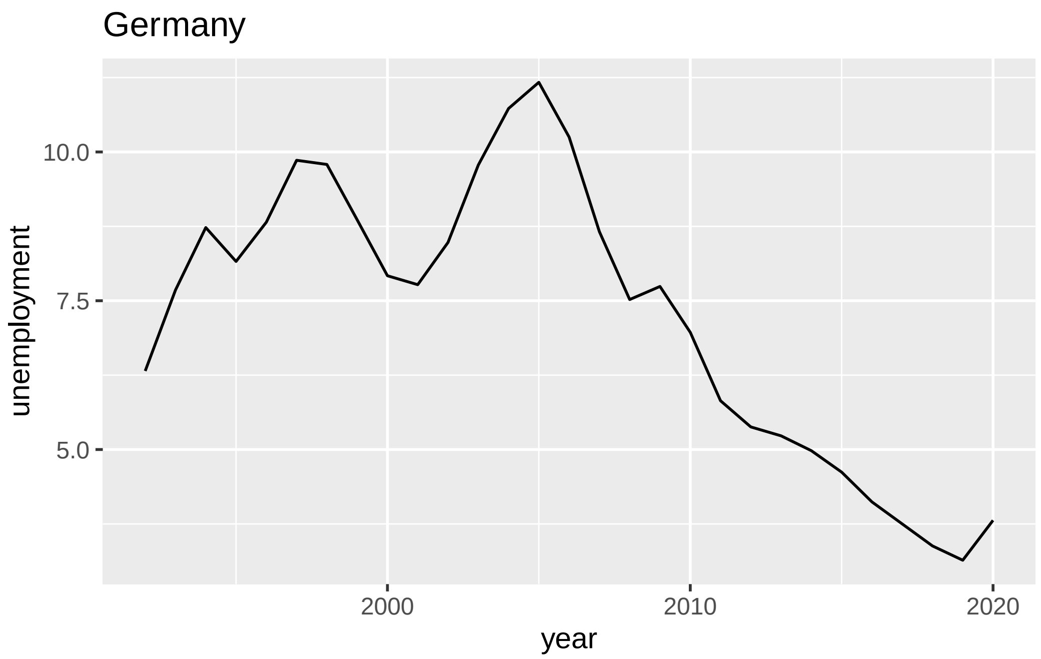
```
df %>%
  group_by(nation) %>%
  summarise(sd(unemployment), mean(unemployment), cov = sd(unemployment)/mean(unemployment
```

```
# A tibble: 2 x 4
  nation  `sd(unemployment)` `mean(unemployment)`   cov
  <chr>                <dbl>                <dbl> <dbl>
1 France                1.58                 9.75 0.162
2 Germany               2.37                 7.22 0.328
```

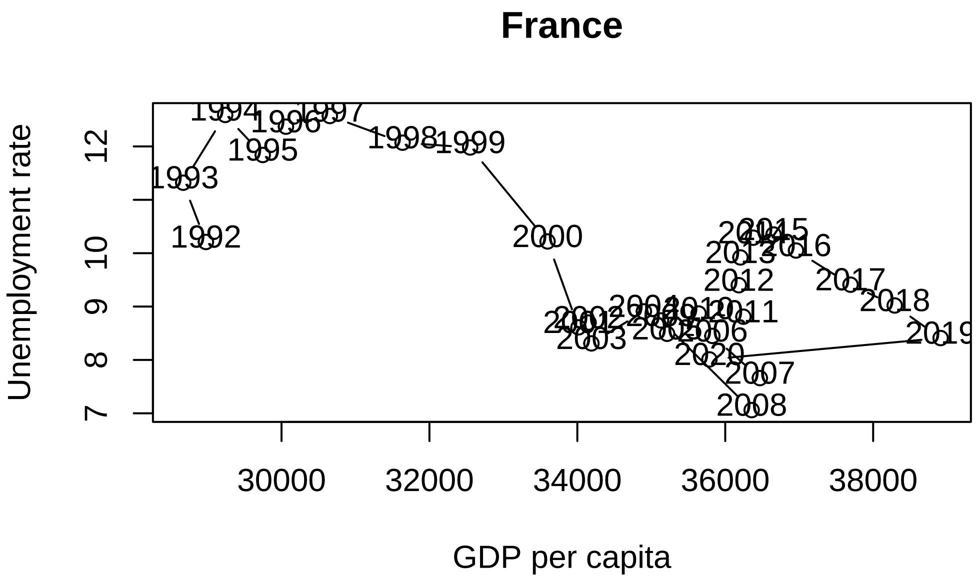
```
df %>%
  group_by(nation) %>%
  summarise(sd(gdppc),mean(gdppc), cov = sd(gdppc)/mean(gdppc))
```

```
# A tibble: 2 x 4
  nation  `sd(gdppc)` `mean(gdppc)`    cov
  <chr>         <dbl>         <dbl>  <dbl>
1 France        2940.        34356. 0.0856
2 Germany       4015.        36739. 0.109
```

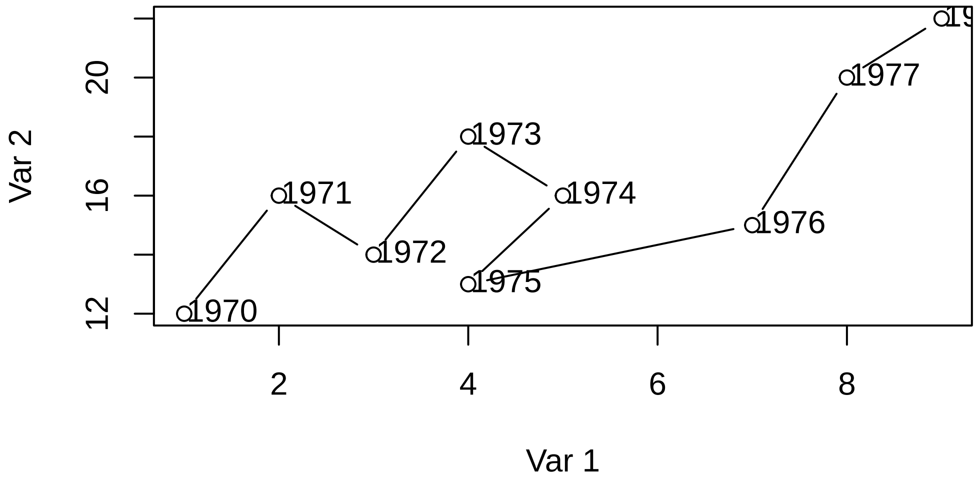
```
pger <- df %>%
  filter(nation=="Germany") %>%
  ggplot(.,aes(x=year, y=unemployment)) +
  geom_line() +
  ggtitle("Germany")
plot(pger)
```



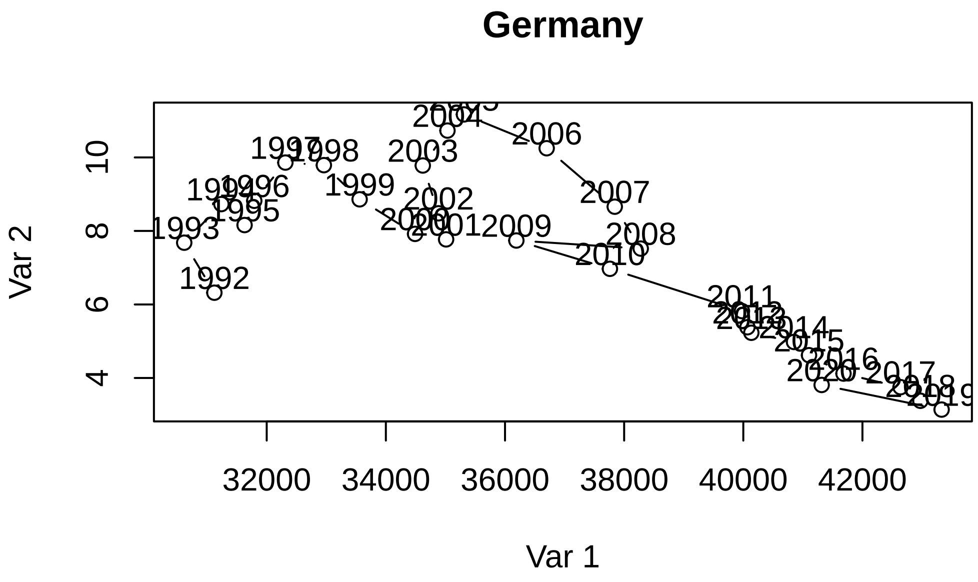
```
labels <- 1992:2020
dfra <- df %>% filter(nation == "France")
plot(dfra$gdppc, dfra$unemployment, type = "b",
     xlab = "GDP per capita", ylab = "Unemployment rate"); text(dfra$gdppc + 0.1, dfra$une
```



```
# Data
x <- c(1, 2, 3, 4, 5, 4, 7, 8, 9)
y <- c(12, 16, 14, 18, 16, 13, 15, 20, 22)
labels <- 1970:1978

# Connected scatter plot with text
plot(x, y, type = "b", xlab = "Var 1", ylab = "Var 2"); text(x + 0.4, y + 0.1, labels)
```



```
dfger <- df %>% filter(nation == "Germany")
labels <- 1992:2020
plot(dfger$gdppc, dfger$unemployment, type = "b",
     xlab = "Var 1", ylab = "Var 2"); text(dfger$gdppc + 0.7, dfger$unemployment + 0.4, la
```



```
# rmarkdown::render("22-11_dsda_exam.Rmd", "all")

# knitr::purl(input = "22-11_dsda_exam.Rmd", output = "22-11_dsda_solution.R",documentatio

suppressMessages(pacman::p_unload(tidyverse, ggpubr, sjPlot))
```

2.15 exe_zipf_solution.R

```
# load packages
if (!require(pacman)) install.packages("pacman")
suppressMessages(pacman::p_unload(all))
# setwd("~/Dropbox/hsf/exams/24-01/Rmd")

rm(list=ls())

pacman::p_load(tidyverse, haven, janitor, jtools)

df <- read_dta(
  "https://github.com/hubchev/courses/raw/main/dta/city.dta",
  encoding="latin1") |>
  as_tibble()

head(df)
```

```
# A tibble: 6 x 7
  stadt               status  state              pop1970 pop1987 pop2011 rankX
  <chr>               <chr>   <chr>                <dbl>   <dbl>   <dbl> <dbl>
1 Vohenstrauß         City    Bayern                7349    7059    7500  2069
2 Stockstadt a. Main Commune Bayern                6416    6615    7504  2068
3 Jesteburg           Commune Niedersachsen         4141    5818    7510  2067
4 Bordesholm          Commune Schleswig-Holstein    6011    6726    7513  2066
5 Herrieden           City    Bayern                5631    6250    7516  2065
6 Weida               City    Th_ringen               NA      NA    7522  2064
```

```
tail(df)
```

```
# A tibble: 6 x 7
  stadt              status                  state  pop1970 pop1987 pop2011 rankX
  <chr>              <chr>                   <chr>    <dbl>   <dbl>   <dbl> <dbl>
1 Frankfurt am Main City with County Rights Hessen  699297  618266  667925     5
2 Köln [Cologne]     City with County Rights Nordr~  994705  928309 1005775     4
3 München [Munich]   City with County Rights Bayern 1293599 1185421 1348335     3
4 Hamburg            City with County Rights Hambu~ 1793823 1592770 1706696     2
5 Berlin             City with County Rights Berlin 3210000 3260000 3292365     1
6 Perl               Commune                 Saarl~      NA      NA      NA    NA
```

```
dim(df)
```

```
[1] 2072    7
```

```
summary(df)
```

```
    stadt               status              state              pop1970
 Length:2072        Length:2072        Length:2072        Min.   :   1604
 Class :character   Class :character   Class :character   1st Qu.:   8149
 Mode  :character   Mode  :character   Mode  :character   Median :  11912
                                                           Mean   :  30504
                                                           3rd Qu.:  21318
                                                           Max.   :3210000
                                                           NA's   :355
    pop1987            pop2011             rankX
 Min.   :   4003   Min.   :   7500   Min.   :   1.0
 1st Qu.:   9194   1st Qu.:   9998   1st Qu.: 516.5
 Median :  13118   Median :  13937   Median :1034.0
 Mean   :  30854   Mean   :  30772   Mean   :1034.0
 3rd Qu.:  23074   3rd Qu.:  24096   3rd Qu.:1551.5
 Max.   :3260000   Max.   :3292365   Max.   :2069.0
 NA's   :248       NA's   :1         NA's   :1
```

```
df <- df |>
  rename(city = stadt)

df <- df |>
  select(-pop1970, -pop1987)

df  %>%
  group_by(state) %>%
  summarise( mean(pop2011),
             sum(pop2011)
  )
```

```
# A tibble: 17 x 3
   state                  `mean(pop2011)` `sum(pop2011)`
   <chr>                            <dbl>          <dbl>
 1 Baden-Wrttemberg                 7580            7580
 2 Baden-Württemberg               23680.        7837917
 3 Bayern                          23996.        7558677
 4 Berlin                        3292365         3292365
 5 Brandenburg                     18472.        1865632
 6 Bremen                         325432.         650863
 7 Hamburg                       1706696         1706696
 8 Hessen                          22996.        5036121
 9 Mecklenburg-Vorpommern          27034.         811005
10 Niedersachsen                   24107.        6219515
11 Nordrhein-Westfalen             47465.       18036727
12 Rheinland-Pfalz                 25644.        1871995
13 Saarland                           NA              NA
14 Sachsen                         27788.        2973351
15 Sachsen-Anhalt                  21212.        1993915
16 Schleswig-Holstein              24157.        1739269
17 Th_ringen                       29192.        1167692
```

```
df <- df %>%
  mutate(state = case_when(
    state == "Baden-Wrttemberg"  ~ "Baden-Württemberg",
    state == "Th_ringen" ~ "Thüringen",
    TRUE ~ state
  ))

df  %>%
  group_by(state) %>%
  summarise( mean(pop2011),
             sum(pop2011)
  )
```

```
# A tibble: 16 x 3
   state                  `mean(pop2011)` `sum(pop2011)`
   <chr>                            <dbl>          <dbl>
 1 Baden-Württemberg               23631.        7845497
 2 Bayern                          23996.        7558677
 3 Berlin                        3292365         3292365
 4 Brandenburg                     18472.        1865632
 5 Bremen                         325432.         650863
 6 Hamburg                       1706696         1706696
 7 Hessen                          22996.        5036121
 8 Mecklenburg-Vorpommern          27034.         811005
 9 Niedersachsen                   24107.        6219515
10 Nordrhein-Westfalen             47465.       18036727
11 Rheinland-Pfalz                 25644.        1871995
12 Saarland                           NA              NA
13 Sachsen                         27788.        2973351
14 Sachsen-Anhalt                  21212.        1993915
15 Schleswig-Holstein              24157.        1739269
16 Thüringen                       29192.        1167692
```

```
df |>
  filter(state == "Saarland") |>
  print(n = 100)
```

```
# A tibble: 47 x 5
   city                status  state    pop2011 rankX
   <chr>               <chr>   <chr>      <dbl> <dbl>
 1 Perl                Commune Saarland    7775  2003
 2 Freisen             Commune Saarland    8270  1894
 3 Großrosseln         Commune Saarland    8403  1868
 4 Nonnweiler          Commune Saarland    8844  1775
 5 Nalbach             Commune Saarland    9302  1678
 6 Wallerfangen        Commune Saarland    9542  1642
 7 Kirkel              Commune Saarland   10058  1541
 8 Merchweiler         Commune Saarland   10219  1515
 9 Nohfelden           Commune Saarland   10247  1511
10 Friedrichsthal      City    Saarland   10409  1489
11 Marpingen           Commune Saarland   10590  1461
12 Mandelbachtal       Commune Saarland   11107  1390
13 Kleinblittersdorf   Commune Saarland   11396  1354
14 Überherrn           Commune Saarland   11655  1317
15 Mettlach            Commune Saarland   12180  1241
16 Tholey              Commune Saarland   12385  1217
17 Saarwellingen       Commune Saarland   13348  1104
18 Quierschied         Commune Saarland   13506  1088
```

```
19 Spiesen-Elversberg  Commune Saarland   13509  1086
20 Rehlingen-Siersburg Commune Saarland   14526   996
21 Riegelsberg         Commune Saarland   14763   982
22 Ottweiler           City    Saarland   14934   969
23 Beckingen           Commune Saarland   15355   931
24 Losheim am See      Commune Saarland   15906   887
25 Schiffweiler        Commune Saarland   15993   882
26 Wadern              City    Saarland   16181   874
27 Schmelz             Commune Saarland   16435   857
28 Sulzbach/Saar       City    Saarland   16591   849
29 Illingen            Commune Saarland   16978   827
30 Schwalbach          Commune Saarland   17320   812
31 Eppelborn           Commune Saarland   17726   793
32 Wadgassen           Commune Saarland   17885   785
33 Bexbach             City    Saarland   18038   777
34 Heusweiler          Commune Saarland   18201   762
35 Püttlingen          City    Saarland   19134   718
36 Lebach              City    Saarland   19484   701
37 Dillingen/Saar      City    Saarland   20253   654
38 Blieskastel         City    Saarland   21255   601
39 St. Wendel          City    Saarland   26220   460
40 Merzig              City    Saarland   29727   392
41 Saarlouis           City    Saarland   34479   323
42 St. Ingbert         City    Saarland   36645   299
43 Völklingen          City    Saarland   38809   279
44 Homburg             City    Saarland   41502   247
45 Neunkirchen         City    Saarland   46172   206
46 Saarbrücken         City    Saarland  175853    43
47 Perl                Commune Saarland      NA    NA
```

```r
df <-  df |>
  filter(!(city=="Perl" & is.na(pop2011)) )

df |>
  filter(state == "Saarland") |>
  print(n = 100)
```

```
# A tibble: 46 x 5
   city                status  state    pop2011 rankX
   <chr>               <chr>   <chr>      <dbl> <dbl>
 1 Perl                Commune Saarland    7775  2003
 2 Freisen             Commune Saarland    8270  1894
 3 Großrosseln         Commune Saarland    8403  1868
 4 Nonnweiler          Commune Saarland    8844  1775
 5 Nalbach             Commune Saarland    9302  1678
 6 Wallerfangen        Commune Saarland    9542  1642
 7 Kirkel              Commune Saarland   10058  1541
 8 Merchweiler         Commune Saarland   10219  1515
 9 Nohfelden           Commune Saarland   10247  1511
10 Friedrichsthal      City    Saarland   10409  1489
11 Marpingen           Commune Saarland   10590  1461
12 Mandelbachtal       Commune Saarland   11107  1390
13 Kleinblittersdorf   Commune Saarland   11396  1354
14 Überherrn           Commune Saarland   11655  1317
15 Mettlach            Commune Saarland   12180  1241
16 Tholey              Commune Saarland   12385  1217
17 Saarwellingen       Commune Saarland   13348  1104
18 Quierschied         Commune Saarland   13506  1088
19 Spiesen-Elversberg  Commune Saarland   13509  1086
20 Rehlingen-Siersburg Commune Saarland   14526   996
21 Riegelsberg         Commune Saarland   14763   982
22 Ottweiler           City    Saarland   14934   969
23 Beckingen           Commune Saarland   15355   931
24 Losheim am See      Commune Saarland   15906   887
25 Schiffweiler        Commune Saarland   15993   882
26 Wadern              City    Saarland   16181   874
27 Schmelz             Commune Saarland   16435   857
28 Sulzbach/Saar       City    Saarland   16591   849
29 Illingen            Commune Saarland   16978   827
30 Schwalbach          Commune Saarland   17320   812
31 Eppelborn           Commune Saarland   17726   793
32 Wadgassen           Commune Saarland   17885   785
33 Bexbach             City    Saarland   18038   777
34 Heusweiler          Commune Saarland   18201   762
35 Püttlingen          City    Saarland   19134   718
36 Lebach              City    Saarland   19484   701
37 Dillingen/Saar      City    Saarland   20253   654
38 Blieskastel         City    Saarland   21255   601
39 St. Wendel          City    Saarland   26220   460
40 Merzig              City    Saarland   29727   392
41 Saarlouis           City    Saarland   34479   323
42 St. Ingbert         City    Saarland   36645   299
43 Völklingen          City    Saarland   38809   279
44 Homburg             City    Saarland   41502   247
45 Neunkirchen         City    Saarland   46172   206
46 Saarbrücken         City    Saarland  175853    43
```

```r
df  %>%
  filter(state == "Saarland") %>%
  summarise( mean(pop2011),
             sum(pop2011)
  )
```

```
# A tibble: 1 x 2
  `mean(pop2011)` `sum(pop2011)`
            <dbl>          <dbl>
1          20850.         959110
```

```
df |>
  group_by(city) |>
  mutate(unique_count = n()) |>
  arrange(city, state) |>
  filter(unique_count > 1) |>
  select(city, status, state, starts_with("pop"), unique_count) |>
  print(n = 100)
```

```
# A tibble: 23 x 5
# Groups:   city [11]
   city        status                  state               pop2011 unique_count
   <chr>       <chr>                   <chr>                 <dbl>        <int>
 1 Bonn        City with County Rights Nordrhein-Westfalen  305765            3
 2 Bonn        City with County Rights Nordrhein-Westfalen  305765            3
 3 Bonn        City with County Rights Nordrhein-Westfalen  305765            3
 4 Brühl       Commune                 Baden-Württemberg     13805            2
 5 Brühl       City                    Nordrhein-Westfalen   43568            2
 6 Erbach      City                    Baden-Württemberg     13024            2
 7 Erbach      City                    Hessen                13245            2
 8 Fürth       City with County Rights Bayern               115613            2
 9 Fürth       Commune                 Hessen                10481            2
10 Lichtenau   City                    Nordrhein-Westfalen   10473            2
11 Lichtenau   Commune                 Sachsen                7544            2
12 Münster     Commune                 Hessen                14071            2
13 Münster     City with County Rights Nordrhein-Westfalen  289576            2
14 Neunkirchen Commune                 Nordrhein-Westfalen   13930            2
15 Neunkirchen City                    Saarland              46172            2
16 Neuried     Commune                 Baden-Württemberg      9383            2
17 Neuried     Commune                 Bayern                 8277            2
18 Petersberg  Commune                 Hessen                14766            2
19 Petersberg  Commune                 Sachsen-Anhalt        10097            2
20 Senden      City                    Bayern                21560            2
21 Senden      Commune                 Nordrhein-Westfalen   19976            2
22 Staufenberg City                    Hessen                 8114            2
23 Staufenberg Commune                 Niedersachsen          7983            2
```

```
df |>
  group_by(city, state) |>
  mutate(unique_count = n()) |>
  arrange(city, state) |>
  filter(unique_count > 1) |>
  select(city, status, state, starts_with("pop"), unique_count) |>
  print(n = 100)
```

```
# A tibble: 3 x 5
# Groups:   city, state [1]
  city  status                  state               pop2011 unique_count
  <chr> <chr>                   <chr>                 <dbl>        <int>
1 Bonn  City with County Rights Nordrhein-Westfalen  305765            3
2 Bonn  City with County Rights Nordrhein-Westfalen  305765            3
3 Bonn  City with County Rights Nordrhein-Westfalen  305765            3
```

```
df <- df |>
  group_by(city, state) |>
  mutate(n_row = row_number() ) |>
  filter(n_row == 1) |>
  select(-n_row)

df |>
  group_by(city, state) |>
  mutate(unique_count = n()) |>
  arrange(city, state) |>
  filter(unique_count > 1) |>
  select(city, status, state, starts_with("pop"), unique_count) |>
  print(n = 100)
```

```
# A tibble: 0 x 5
# Groups:   city, state [0]
# i 5 variables: city <chr>, status <chr>, state <chr>, pop2011 <dbl>,
#   unique_count <int>
```

```
save(df, file = "city_clean.RData")

df <- df |>
  ungroup() |>
  arrange(desc(pop2011)) |>
  mutate(rank = row_number() )

df |>
```

```
  select(-rankX, -status, -state) |>
  head()
```

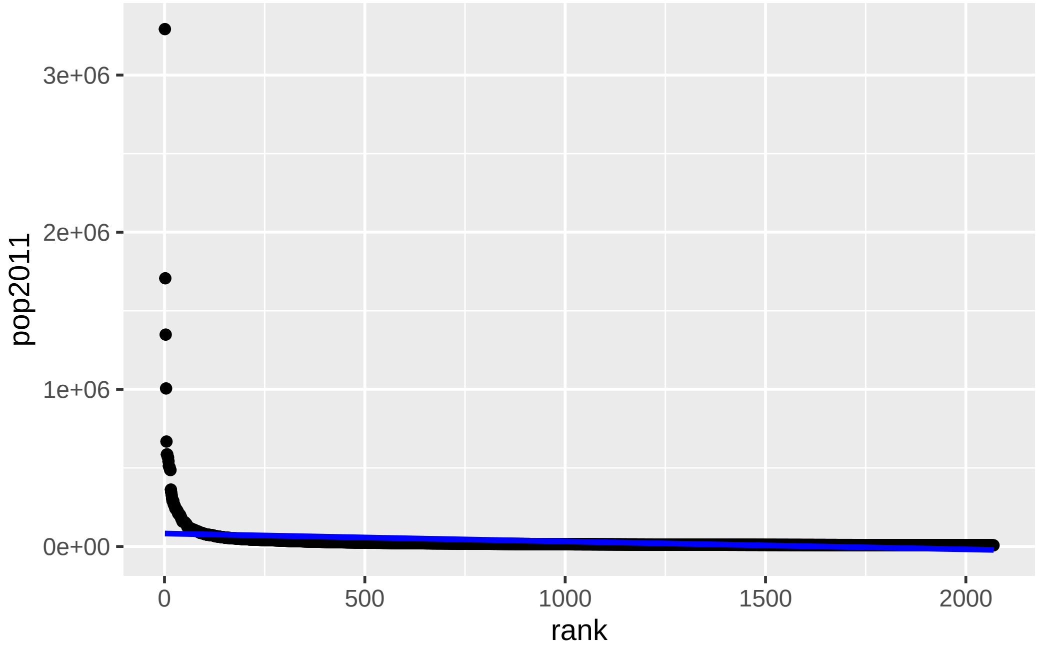
```
# A tibble: 6 x 3
  city                        pop2011  rank
  <chr>                         <dbl> <int>
1 Berlin                      3292365     1
2 Hamburg                     1706696     2
3 München [Munich]            1348335     3
4 Köln [Cologne]              1005775     4
5 Frankfurt am Main            667925     5
6 Düsseldorf [Dusseldorf]      586291     6
```

```
cor(df$pop2011, df$rank, method = c("pearson"))
```

```
[1] -0.2948903
```

```
ggplot(df, aes(x = rank, y = pop2011)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "blue")
```

```
`geom_smooth()` using formula = 'y ~ x'
```



```r
df <- df |>
  mutate(lnrank = log(rank) ) |>
  mutate(lnpop2011 = log(pop2011) )

df |>
  select(city, rank, lnrank, pop2011, lnpop2011) |>
  head()
```

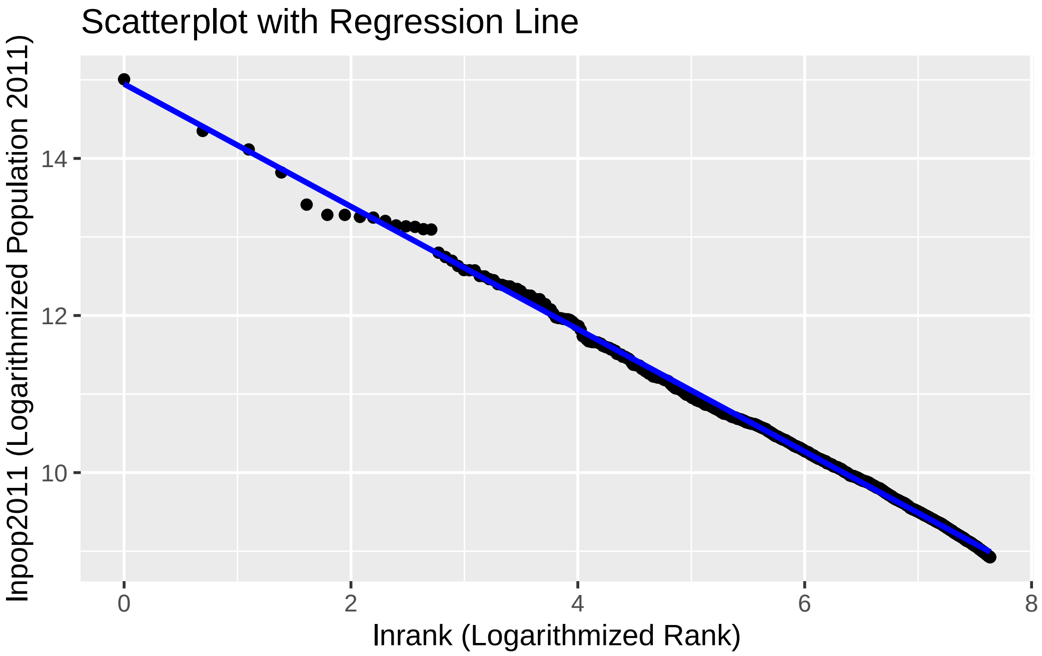
```
# A tibble: 6 x 5
  city                       rank lnrank pop2011 lnpop2011
  <chr>                     <int>  <dbl>   <dbl>     <dbl>
1 Berlin                        1  0     3292365      15.0
2 Hamburg                       2  0.693 1706696      14.4
3 München [Munich]              3  1.10  1348335      14.1
4 Köln [Cologne]                4  1.39  1005775      13.8
5 Frankfurt am Main             5  1.61   667925      13.4
6 Düsseldorf [Dusseldorf]       6  1.79   586291      13.3
```

```r
cor(df$lnpop2011, df$lnrank, method = c("pearson"))
```

```
[1] -0.9990053
```

```r
ggplot(df, aes(x = lnrank, y = lnpop2011)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "Scatterplot with Regression Line",
       x = "lnrank (Logarithmized Rank)",
       y = "lnpop2011 (Logarithmized Population 2011)")
```

```
`geom_smooth()` using formula = 'y ~ x'
```



```
zipf <- lm(lnpop2011 ~ lnrank, data = df)
summary(zipf)
```

```
Call:
lm(formula = lnpop2011 ~ lnrank, data = df)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.28015 -0.01879  0.01083  0.02005  0.25973 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 14.947859   0.005141    2908   <2e-16 ***
lnrank      -0.780259   0.000766   -1019   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.03454 on 2067 degrees of freedom
Multiple R-squared:  0.998, Adjusted R-squared:  0.998 
F-statistic: 1.038e+06 on 1 and 2067 DF,  p-value: < 2.2e-16
```

```
df <- df |>
  mutate(prediction = predict(zipf, newdata = df)) |>
  mutate(pred_pop = exp(prediction))
```

```
df |>
  select(city, pop2011, pred_pop) |>
  filter(city == "Regensburg")
```

```
# A tibble: 1 x 3
  city       pop2011 pred_pop
  <chr>        <dbl>    <dbl>
1 Regensburg  135403  134194.
```

```
suppressMessages(pacman::p_unload(tidyverse, haven, janitor, jtools))

# rmarkdown::render("24-01_dsda.Rmd", "all")

# knitr::purl(input = "24-01_dsda.Rmd", output = "24-01_dsda_solution.R",documentation = 0
```

```
Warning in file.copy(source_files, destination_folder, overwrite = TRUE):
problem copying ./exe_soutions.html to
/home/sthu/Dropbox/hsf/github/courses/rmd/exe_soutions.html: No such file or
directory
```

```
[1]  TRUE FALSE
```

```
Files copied to /home/sthu/Dropbox/hsf/github/courses/rmd/
```