Theme 2: Data Visualisation: ggplot2 Essentials - Assignments

Eemil Mitikka | IQDA Course 2023 | University of Helsinki

README

For this week's exercise, we will be using three different datasets:

- `lits_subset.csv`
- `levada_president_longer.csv`
- `vdem_subset_upd.csv`

`lits_subset.csv` is a subset of a survey called *Life in Transition III*. It was conducted in 2016 in various "transition economies". More information can be found here.

`levada_president_longer.csv` captures Putin's approval ratings as measured by the Russian pollster Levada-Center.

`vdem_subset_upd.csv` is the same dataset you used before, only with one new variable called `country_group`.

Packages

```
library(tidyverse)
library(here)
```

Data imports

Since our focus is not this time on data importing, I'll give the code needed to import the data below. Please consult Moodle materials in case of problems, and also reach out if you get stuck for a long time.

```
# Import LiTS data
lits_subset <- here("lits_subset.csv") %>%
  read.csv()

# Import Putin's approval ratings data
putins_approval <- here("levada_president_longer.csv") %>%
  read.csv() %>%
  mutate(date = ymd(date)) # Note: this converts date variable to date object <date> instead of the default chara
cter <chr>. ymd() function comes from lubridate package, and it converts properly formatted character variables w
ith year-month-date logic, hence the name "ymd()" -> "year", "month", "date". This is just for information at thi
s point

# Import updated V-Dem data
vdem_upd <- here("vdem_subset_upd.csv") %>%
  read.csv()
```

Assignment 1: Scatterplot with `geom_point()`

- Using the `lits_subset.csv` data, draw a scatter plot on the relationship between variables `household_income` and `age_pr` ***in Russia***. Note that you can subset the data using `filter()` function (recall the use of this function from Theme 1's exercises). Here, the plotted variables measure following things:
  - `household_income` = *"What is the total net monthly income of your household at present?"* (note: in local currency)
  - `age_pr` = Age of primary respondent of the survey.

Answer the following questions:

1. What can you say on the relationship between `age_pr` and `household_income` based on your scatter plot? Or can you say anything? ***There are no right or wrong answers***, just use your own words about your first impression. 🤗
2. Are there extreme values in the variables plotted? Extreme values are simply mean individual observations (here: points) that are far away from the rest of the observations. 🤗

```
rus_income_and_age <- lits_subset %>%
  filter(country == ___)

ggplot(rus_income_and_age) +
  aes(x = age_pr, y = household_income) +
  geom_point()
```

Assignment 2: Line plot with `geom_line()`

Tasks

- Using the `levada_president_longer.csv` data, draw a line graph of Vladimir Putin's approval ratings using `ggplot()` and `geom_line()` functions. 🤗
- Add the `approval` variable to your plot. Hint: good way to add categorical variables to `ggplot()`s (or plots in general) is to put categorical variable as `color` argument within the `aes()` function call. Check the `gapminder` plot example on Theme 2's lecture slides if you are unsure what I'm talking about. 🤗

```
ggplot(putins_approval) +
  aes(x = ___, y = ___, color = ___) +
  # correct geom_ goes here
```

Assignment 3: Bar plot with `geom_col()`

Tasks

- With the `lits_subset` data, draw a bar chart comparing ***mean trust in the Presidency*** between countries in the `lits_subset` data. Fill in the correct arguments to `x` and `y` within the `aes()` function and use `geom_col()` function to tell R you want to draw a bar chart. 🤗
- To calculate the mean trust in the Presidency by countries, recall how we calculated mean for variables in the Theme 1's assignment. Hint: you can do this using `group_by()` and `summarise()` functions. 🤗
- 😉 ***If you want*** the bars to appear in descending order, like in Theme 2 – Support Code and Materials example available on Moodle – you can use `y = reorder(y_axis_variable, +x_axis_variable)` within the `aes()` function call. Note that your variables are not called `y_axis_variable` or `x_axis_variable`, but something else – this just exemplifies the logic behind `reorder()`. In other words, the logic is this:

```
ggplot(dataframe) +
aes(x = x_axis_variable, y = reorder(y_axis_variable, +x_axis_variable)) +
geom_col()
```

Trust in the Presidency between LiTS countries

```
lits_president_trust <- lits_subset %>%
  drop_na(trust_president) %>%
  group_by(___) %>%
  summarise(mean_trust_president = mean(___))

ggplot(lits_president_trust) +
  aes(y = reorder(___, +___),
      x = ___) +
  # Correct geom
```

Assignment 4: Histogram with `geom_histogram()`, and Density plot with `geom_density()`

Tasks

- Using the `lits_subset` data frame, draw a histogram and a density plot on `household_income` and `age_pr` in Russia. Recall the use of the `filter()` function from the previous exercise (relevant variable for filtering is called `country` in these data). Exclude the missing observations (i.e. rows) before plotting the data. 🤗
- Look at the resulting plots and answer the following questions: 🤗
1. How do the distribution of `household_income` and `age_pr` look like in Russia according to this survey? Are they normally distributed?
2. Are there extreme values (very low or extremely high monthly incomes or ages)? You can answer this question by looking at the graph by the "naked eye" or calculate use `summarise()` function as in Theme 1's exercise to calculate minimum and maximum values for these variables.

```
rus_income_and_age <- lits_subset %>%
  filter(country == "Russia") %>%
  select(country, household_income, age_pr) %>%
  drop_na()

# Histogram of household income
ggplot(___) +
  aes(x = ___) +
  geom_histogram()

# Histogram of age
ggplot(___) +
  aes(x = ___) +
  # Correct geom here

# Density plot of household income: draw below


# Histogram of household income: draw below
```

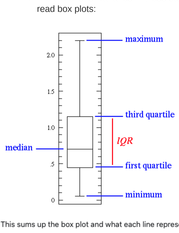
💰 BONUS TASK

- If you find plotting the household income levels and age distributions very easy and want some extra tasks, you can filter for the extreme values and try to plot these data again to see if these plots yield more telling results. **NOTE:** This task is not mandatory, but an extra task!

Assignment 5: Boxplot with `geom_boxplot()`

Tasks

- Using the updated V-Dem dataset `vdenm_subset_upd.csv`, create a box plot comparing political corruption `v2x_corr` between countries (variable `country_group`). Use `geom_boxplot()`. Filter the data to include only ex-USSR countries and observations from `year` 2022 (code for this is given below). 🤗
- Interpret the results briefly by answering the following questions: 🤗
  - Which country groups have the highest and lowest levels of political corruption according to V-Dem estimates?
  - Are there lot of variance in these data according to your box plot? The figure below illustrates how to read box plots:



This sums up the box plot and what each line represents.

How to read box plot visualisation ⬆️

```
# Subset the data
vdem_boxplot_corruption <- vdem_upd %>%
  filter(year == "2022",
         !is.na(country_group))

ggplot(___) +
  aes(x = reorder(___, -___), y = ___) +
  # Correct geom here
```

Afterword (not officially part of Theme 2 assignments, just some extra stuff!)

In case these assignments are too easy to you, please play around with the data and feel free to include these in your weekly report. This course is a "safe playground" for practicing data analysis with R, and you can ask questions about things that interest you in theme.

If you want see how the original *Life in Transition III* dataset looks like, you can take a look at it (file named as "LiTS III.dta"). I included in the materials of this week's exercises for your reference. More information about this survey can be found on European Bank for Reconstruction and Development's website. However, note that the file is .dta format, which is used with the *Stata* software. R can handle these kind of data formats too, but you need to install and activate R package called **haven** to import .dta file to your R environment. Example code for how to import is given below:

```
library(here)
library(tidyverse)
library(haven) # NB: as always, you need to run install.packages("haven"), if you haven't installed the package y
et

lits_original <- here("LiTS III.dta") %>%
  read_dta(encoding = "latin1")
```