Template

Studentnames and studentnumbers here

2025-06-16

Set-up your environment

```
require(tidyverse)
```

```
## Loading required package: tidyverse
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

Title Page

Include your names

Include the tutorial group number

Include your tutorial lecturer's name

Part 1 - Identify a Social Problem

Use APA referencing throughout your document. Here's a link to some explanation.

1.1 Describe the Social Problem

Include the following:

- Why is this relevant?
- ...

Part 2 - Data Sourcing

2.1 Load in the data

Preferably from a URL, but if not, make sure to download the data and store it in a shared location that you can load the data in from. Do not store the data in a folder you include in the Github repository!

```
dataset <- midwest
```

midwest is an example dataset included in the tidyverse package

2.2 Provide a short summary of the dataset(s)

```
head(dataset)
```

```
## # A tibble: 6 x 28
##     PID county   state  area poptotal popdensity popwhite popblack popamerindian
##   <int> <chr>    <chr> <dbl>    <int>      <dbl>    <int>    <int>         <int>
## 1   561 ADAMS    IL    0.052    66090      1271.    63917     1702            98
## 2   562 ALEXAND~ IL    0.014    10626       759      7054     3496            19
## 3   563 BOND     IL    0.022    14991       681.    14477      429            35
## 4   564 BOONE    IL    0.017    30806      1812.    29344      127            46
## 5   565 BROWN    IL    0.018     5836       324.     5264      547            14
## 6   566 BUREAU   IL    0.05     35688       714.    35157       50            65
## # i 19 more variables: popasian <int>, popother <int>, percwhite <dbl>,
## #   percblack <dbl>, percamerindan <dbl>, percasian <dbl>, percother <dbl>,
## #   popadults <int>, perchsd <dbl>, percollege <dbl>, percprof <dbl>,
## #   poppovertyknown <int>, percpovertyknown <dbl>, percbelowpoverty <dbl>,
## #   percchildbelowpovert <dbl>, percadultpoverty <dbl>,
## #   percelderlypoverty <dbl>, inmetro <int>, category <chr>
```

In this case we see 28 variables, but we miss some information on what units they are in. We also don't know anything about the year/moment in which this data has been captured.

```
inline_code = TRUE
```

These are things that are usually included in the metadata of the dataset. For your project, you need to provide us with the information from your metadata that we need to understand your dataset of choice.

2.3 Describe the type of variables included

Think of things like:

- Do the variables contain health information or SES information?
- Have they been measured by interviewing individuals or is the data coming from administrative sources?

For the sake of this example, I will continue with the assignment...

Part 3 - Quantifying

3.1 Data cleaning

Say we want to include only larger distances (above 2) in our dataset, we can filter for this.

```
mean(dataset$percollege)
```

```
## [1] 18.27274
```

Please use a separate 'R block' of code for each type of cleaning. So, e.g. one for missing values, a new one for removing unnecessary variables etc.

3.2 Generate necessary variables

Variable 1

Variable 2

3.3 Visualize temporal variation

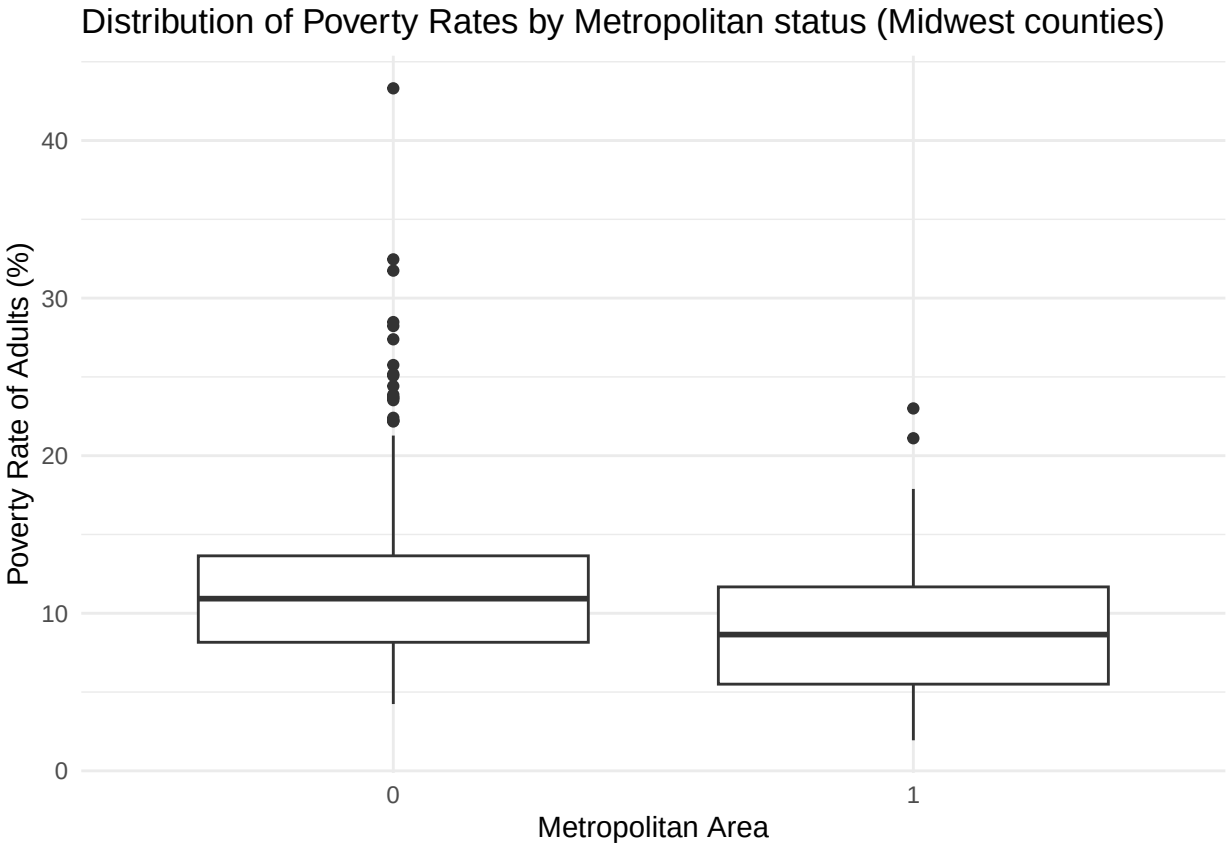
3.4 Visualize spatial variation

Here you provide a description of why the plot above is relevant to your specific social problem.

3.5 Visualize sub-population variation

What is the poverty rate by state?

```
dataset$inmetro <- dataset$inmetro %>% as.factor()
# Boxplot of poverty rate by state using the 'midwest' dataset
ggplot(dataset, aes(x = inmetro, y = percadultpoverty)) +
  geom_boxplot() +
  labs(
    title = "Distribution of Poverty Rates by Metropolitan status (Midwest counties)",
    x = "Metropolitan Area",
    y = "Poverty Rate of Adults (%)"
  ) +
  theme_minimal() +
  theme(
    legend.position = "right"
  )
```



Here you provide a description of why the plot above is relevant to your specific social problem.

3.6 Event analysis

Analyze the relationship between two variables.

Here you provide a description of why the plot above is relevant to your specific social problem.

Part 4 - Discussion

4.1 Discuss your findings

Part 5 - Reproducibility

5.1 Github repository link

Provide the link to your PUBLIC repository here: ...

5.2 Reference list

Use APA referencing throughout your document.

—- PACKAGES —-

install.packages(“tidyverse”) install.packages(“dplyr”) install.packages(“ggplot2”) install.packages(“scales”) install.packages(“sf”) install.packages(“rnaturalearth”) install.packages(“rnaturalearthdata”) install.packages(“viridis”) library(scales) library(tidyr) library(dplyr) library(readxl) library(ggplot2) library(tidyverse) library(sf) library(rnaturalearth) library(rnaturalearthdata) theme_set(theme_bw())

—- DATA CLEANING EN MERGING —-

#Dataframe maken van excel-bestand df1 <- read_excel(“Studieschuld_2011-2024(goeie).xlsx”, skip = 1) df2 <- read_excel(“England_figure_10.xlsx”) df3 <- read_excel(“Average Total Student Debt USA.xlsx”)

#Rijen en Kolommen die niet nodig zijn verwijderen df1 <- df1[-c(1, 2, 3),] df1 <- df1[, -c(2, 3, 6)] df1 <- df1[, -c(2)] df2 <- df2[, -c(3)]

#Veranderen van schooljaar naar kalenderjaar df2$‘Financial year’ <- 2007:2024

#Kolomnamen overal hetzelfde maken voor samenvoegen colnames(df1)[1] <- “Year” colnames(df1)[2] <- “Studieschuld_NL” colnames(df2)[1] <- “Year” colnames(df2)[2] <- “Studieschuld_UK” colnames(df3)[1] <- “Year” colnames(df3)[2] <- “Studieschuld_US”

#Year was blijkaar numeriek dus een ‘karakter’ van maken df2$Year <- -as.character(df2$Year) df3$Year <- -as.character(df3$Year)

#Stond een * bij 2023 en 24 bij NL waardoor hij dat als aparte variabele zag df1$Year[df1$Year == “2023”] <- 2023 df1$Year[df1$Year == ”2024”] <- 2024

df1,2,3 samenvoegen

df_merge <- full_join(df1, df2, by = “Year”) df_merge <- full_join(df_merge, df3, by = “Year”)

Sorteren van Oud naar Nieuw

df_sorted <- df_merge[order(df_merge$Year, decreasing = FALSE),]

Alles numeriek maken

df_sorted$Studieschuld_NL <- -as.numeric(df_sorted$Studieschuld_NL) df_sorted$Studieschuld_UK <- -as.numeric(df_sorted$Studieschuld_UK) df_sorted$Studieschuld_US <- -as.numeric(df_sorted$Studieschuld_US)

Nieuwe Variabele: Jaarlijkse groei berekenen (percentage)

df_sorted$Growth_NL <- -c(NA, diff(df_sorted$Studieschuld_NL) / head(df_sorted$Studieschuld_NL, -1) * 100)df_sorted$Growth_UK <- c(NA, diff(df_sorted$Studieschuld_UK)/head(df_sorted$Studieschuld_UK, -1) * 100) df_sorted$Growth_US <- -c(NA, diff(df_sorted$Studieschuld_US) / head(df_sorted$Studieschuld_US, -1) * 100)

#Kolommen omzetten in numeriek df_numeric <- df_sorted %>% mutate(across(c(Studieschuld_NL, Studieschuld_UK, Studieschuld_US, Growth_NL, Growth_UK, Growth_US), ~as.numeric(.)), Year = as.numeric(Year))

#Gemiddelde growth uitrekenen NL_Mean_Growth = mean(df_sorted$Growth_NL, na.rm = TRUE) UK_Mean_Growth = mean(df_sorted$Growth_UK, na.rm = TRUE) US_Mean_Growth = mean(df_sorted$Growth_US, na.rm = TRUE)

#Alles in Euros

df_numeric <- df_numeric %>% mutate(Studieschuld_US = Studieschuld_US * 0.86)

df_numeric <- df_numeric %>% mutate(Studieschuld_UK = Studieschuld_UK * 1.17)

—- EXTRA GRAFIEK ? —-

#Grafiek ggplot(df_numeric, aes(x = Year)) + geom_line(aes(y = Studieschuld_NL, color = "NL"), size = 1.5) + geom_line(aes(y = Studieschuld_UK, color = "UK"), size = 1.5) + geom_line(aes(y = Studieschuld_US, color = "US"), size = 1.5) + geom_point(aes(y = (Studieschuld_NL), color = "NL"), size = 3) + geom_point(aes(y = (Studieschuld_UK), color = "UK"), size = 3) + geom_point(aes(y = (Studieschuld_US), color = "US"), size = 3) + geom_vline(xintercept = 2020, linetype = "dashed", color = "darkred") + annotate("text", x = 2023, y = 12500, label = paste0("Mean growth NL:", round(NL_Mean_Growth, 2), "%"), color = "black", fontface = "bold", size = 3) + annotate("text", x = 2023, y = 58000, label = paste0("Mean growth UK:", round(UK_Mean_Growth, 2), "%"), color = "black", fontface = "bold", size = 3) + annotate("text", x = 2023, y = 47000, label = paste0("Mean growth US:", round(US_Mean_Growth, 2), "%"), color = "black", fontface = "bold", size = 3) + annotate("text", x = 2020.7, y = 2500, label = paste0(" Start Covid-19"), color = "black", fontface = "bold", size = 3) +

scale_x_continuous(limits = c(2014, NA), breaks = seq(2014, max(as.numeric(df_numeric$Year), na.rm = TRUE), by = 1) # alle jaren) + labs(title = "Growth in Student Debt per Country", x = "Year", y = "Student Debt", color = "Country") +

scale_y_continuous(limits = c(0, 60000), expand = c(0, 0), labels = label_dollar(prefix = "€", big.mark = ".", decimal.mark = ","))

theme_bw()

#Opslaan als PNG ggsave("Temporal_Visualization.png", width = 8, height = 5)

—- SUB POPULATION —-

data_selected <- df_numeric %>% select(Year, starts_with("Studieschuld"))

Data omvormen naar long format

data_long <- data_selected %>% pivot_longer(cols = -Year, names_to = c("variabele", "land"), names_pattern = "(Studieschuld)_(.*)") %>% rename(studieschuld = value)

Tijdsperiode labelen

data_long <- data_long %>% mutate(period = case_when(Year <= 2019 ~ "2014 - 2019", Year > 2019 ~ "2020 - 2024"))

Gemiddelde per land per periode berekenen

data_avg <- data_long %>% group_by(land, period) %>% summarise(studieschuld_mean = mean(studieschuld, na.rm = TRUE), .groups = “drop”)

Boxplot

ggplot(data_avg, aes(x = period, y = studieschuld_mean)) + geom_boxplot(fill = “lightblue”, color = “darkblue”) + labs(title = “Mean Student Debt per Period (All Countries)”, x = “Period”, y = “Mean Student Debt”) + scale_y_continuous(labels = label_dollar(prefix = “€”, big.mark = “.”, decimal.mark = “,”)) + theme_bw()

ggsave(“Sub_Population.png”, width = 8, height = 5)

```
# Alleen per periode groep maken (alle landen en jaren samen)

data_period <- data_long %>%

group_by(period) %>%

summarise(

mean_studieschuld = mean(studieschuld, na.rm = TRUE),

sd_studieschuld = sd(studieschuld, na.rm = TRUE),

min_studieschuld = min(studieschuld, na.rm = TRUE),

max_studieschuld = max(studieschuld, na.rm = TRUE),

.groups = "drop"

)



ggplot(data_long, aes(x = period, y = studieschuld)) +

geom_boxplot(fill = "lightblue", color = "darkblue") +

labs(title = "Student Debt Distribution per Period (All Countries and Years)",

x = "Period",

y = "Student Debt") +

scale_y_continuous(labels = scales::label_dollar(prefix = "€", big.mark = ".", decimal.mark = ",")) +

theme_bw()

—- SPATIAL VISUALIZATION —-
```


#Gemiddelden gemiddelde <- data_selected %>% summarise(Studieschuld_NL = mean(Studieschuld_NL, na.rm=TRUE), Studieschuld_UK = mean(Studieschuld_UK, na.rm=TRUE), Studieschuld_US =

mean(Studieschuld_US, na.rm=TRUE)) %>% pivot_longer(everything(), names_to = “land”, values_to = “gemiddelde_schuld”) %>% mutate(land = recode(land, Studieschuld_NL = “Netherlands”, Studieschuld_UK = “United Kingdom”, Studieschuld_US = “United States of America”))

Wereldkaart

wereldkaart <- ne_countries(scale = “medium”, returnclass = “sf”)

wereldkaart_met_data <- wereldkaart %>% left_join(gemiddelde, by = c(“admin” = “land”))

ggplot(wereldkaart_met_data) + geom_sf(aes(fill = gemiddelde_schuld), color = “black”) + scale_fill_gradient(low = “lightblue”, high = “darkblue”, na.value = “lightgrey”, # <- grijs voor landen zonder data name = “Mean Student Debt (€)”, labels = label_dollar(prefix = “€”, big.mark = “.”, decimal.mark = “,”)) + coord_sf(crs = 3857, xlim = c(-1.35e7, 0.4e6), # links (VS) tot rechts (NL) ylim = c(2.2e6, 8e6) # onder (VS zuid) tot boven (UK noord)) +

labs(title = “Mean Student Debt (2007–2024)”) +
theme_bw()

ggsave(“Spatial_Visualization.png”, width = 8, height = 5)

—- TEMPORAL VISUALIZATION —-

#Procentuele groei per jaar df_growth_long <- df_numeric %>% select(Year, Growth_NL, Growth_UK, Growth_US) %>% pivot_longer(cols = starts_with(“Growth_”), names_to = “land”, names_prefix = “Growth_”, values_to = “groei_percentage”)

df_growth_long %>% filter(Year >= 2015) %>% ggplot(aes(x = Year, y = groei_percentage, color = land)) + geom_line(size = 1.2) + geom_point() + geom_hline(yintercept = 0, linetype = “dashed”, color = “black”) + scale_y_continuous(labels = scales::percent_format(scale = 1)) + scale_x_continuous(breaks = 2015:2024) + theme_bw() + labs(title = “Procentual Growth of Student Debt (Since 2014)”, x = “Year”, y = “Growth compared to last year (%)”, color = “Country”)

ggsave(“Temporal_Visualization2.png”, width = 8, height = 5)

—- EVENT ANALYSIS —-

#GDP per capita data inlezen, cleanen en mergen gdp_per_capita_data <- read_xlsx(“gdp_per_capita2.xlsx”)

jaar_kolommen <- as.character(unlist(gdp_per_capita_data[3,]))

jaar_kolommen[is.na(jaar_kolommen)] <- paste0(“V”, which(is.na(jaar_kolommen))) df_gdp <- gdp_per_capita_data[4:6,] colnames(df_gdp) <- jaar_kolommen df_gdp <- df_gdp %>% select(`Country Name`, 2007:2023)

df_gdp_long <- df_gdp %>% pivot_longer(cols = -`Country Name`, names_to = “Year”, values_to = “GDP_PC”)

df_gdp_long$GDP_PC <- as.numeric(df_gdp_long$GDP_PC) df_gdp_long$Year <- as.integer(df_gdp_long$Year)

Vervolgens naar breed formaat

df_gdp_wide <- df_gdp_long %>% pivot_wider(names_from = `Country Name`, values_from = GDP_PC) %>% rename(GDP_PC_UK = `United Kingdom`, GDP_PC_NL = Netherlands, GDP_PC_US = `United States`) %>% arrange(Year)

#Omzetten van Dollars naar Euro's df_gdp_wide <- df_gdp_wide %>% mutate(GDP_PC_UK = GDP_PC_UK * 0.86)

df_gdp_wide <- df_gdp_wide %>% mutate(GDP_PC_NL = GDP_PC_NL * 0.86)

df_gdp_wide <- df_gdp_wide %>% mutate(GDP_PC_US = GDP_PC_US * 0.85)

#Samenvoegen met alle andere data en nieuwe variabele maken df_merged <- full_join(df_numeric, df_gdp_wide, by = "Year")

df_merged <- df_merged %>% mutate(Schuld_GDP_NL = 100 * Studieschuld_NL / GDP_PC_NL, Schuld_GDP_UK = 100 * Studieschuld_UK / GDP_PC_UK, Schuld_GDP_US = 100 * Studieschuld_US / GDP_PC_US)

#Plot maken df_long <- df_merged %>% select(Year, Schuld_GDP_NL, Schuld_GDP_UK, Schuld_GDP_US) %>% pivot_longer(cols = starts_with("Schuld_GDP"), names_to = "Land", values_to = "Schuld_GDP") %>% mutate(Land = case_when(Land == "Schuld_GDP_NL" ~ "NL", Land == "Schuld_GDP_UK" ~ "UK", Land == "Schuld_GDP_US" ~ "US"))

df_long %>% filter(Year >= 2015, Year <= 2023) %>% ggplot(aes(x = Year, y = Schuld_GDP, color = Land)) + geom_line(size = 1.2) + geom_point(size = 2) + geom_vline(xintercept = 2020, linetype = "dashed", color = "darkred") + labs(title = "Student Debt as percentage of GDP per capita", x = "Year", y = "Student Debt / GDP per capita (%)", color = "Country") + annotate("text", x = 2020.7, y = 6.250, label = paste0("Start Covid-19"), color = "black", fontface = "bold", size = 3) + scale_y_continuous(labels = scales::percent_format(scale = 1), limits = c(0, 150), expand = c(0, 0), breaks = seq(0,150,25)) + scale_x_continuous(breaks = 2014:2023) theme_bw()

ggsave("Event_Analysis.png", width = 8, height = 5)