Physiotherapeutic data-driven perspective on respiratory diseases

UUN S2616861

01 December, 2023

Table of Contents

[1 Introduction, context, and audience 1](#_Toc152321914)

[2 Data wrangling and coding 2](#_Toc152321915)

[3 Data visualisation 14](#_Toc152321916)

[4 Discussion and analysis 30](#_Toc152321917)

[5 Conclusion and Recommendations 30](#_Toc152321918)

[6 Reflective practice 30](#_Toc152321919)

[References 31](#_Toc152321920)

## Libraries  
# Loading packages  
pacman::p\_load(tidyverse, openai, gridExtra, gt, webshot2, janitor, here)  
  
## Set variables  
OPENAI\_API\_KEY = Sys.getenv("OPENAI\_API\_KEY") # Setting API key for OpenAI  
YEAR = 2019 # Setting year for analysis  
CITY = "Milano" # Setting city for analysis  
  
# List files in working directory  
list.files("./")

## [1] "Presentation.R" "Report.docx" "Report.R" "Report.Rmd"

# 1 Introduction, context, and audience

**Note:** *The report builds on the presentation for the management, but looks at more factors influencing possible respiratory diseases and adds a country-specific view.*

This is data-driven report about respiratory diseases from a physiotherapeutic perspective. It is stored in a public [**Github Repository**](https://github.com/DrBenjamin/Assessment).

The purpose of this report is to show the present audience, Physiotherapists of the **Good Hope Clinic** in Milano, the **importance of physiotherapeutic measures** in cases of respiratory diseases as this health restriction is widespread long before COVID-19. For instance, China is facing a rise in respiratory diseases after lifting zero-COVID restrictions (Hawkings, A., 2023).

Another major public health problem is obesity. In view of increasing life expectancy, the prevalence of obesity is rising steadily in older age groups (Jura, M. & Kozak, L. P., 2016). World Health Organization (2023) defines overweight as a body mass index (BMI) of **25** to **29.9 kg/m2** and obesity as a BMI of **30 kg/m2** or higher. Obesity is a risk factor for many diseases, such as cardiovascular diseases, diabetes mellitus, cancer and musculoskeletal diseases.

Some researchers see a relation between high amount of **abdominal fat** and the function of the **respiratory system** (Rauch, E., 2015). From policy maker side, there are initiatives to reduce the occurrence of obesity, for instance Columbia introduced a tax for fast food (Daniels, J. P., 2023).

The aim of this study is to determine the degree of relevance of the topic “Obesity” for professionals working in the field of physiotherapy with clients that have a resp. disease.

**Is obesity a risk factor for respiratory diseases?**

# 2 Data wrangling and coding

The data was found on the *Organisation for Economic Cooperation and Development (OECD)* data repository as on different websites - for instance from *Statista*, a global data and business intelligence platform.

The data from OECD shows the total number of **respiratory diseases** per country and year.

## Processing OECD Health data of respiratory diseases  
{  
 # Source: https://stats.oecd.org/viewhtml.aspx?datasetcode=HEALTH\_PROC&lang=en#  
 oecd\_data\_raw <- read\_csv(here("raw\_data/", "HEALTH\_PROC\_10112023095631477.csv"))  
   
 # Eyeballing data  
 oecd\_data\_raw %>% glimpse() # Country names are in English language, `Türkiye` is used instead of `Turkey` see  
 # https://www.theguardian.com/world/2022/jun/03/turkey-changes-name-to-turkiye-as-other-name-is-for-the-birds  
 oecd\_data\_raw %>% skimr::skim()  
 oecd\_data\_raw %>% tabyl(Country, Year, show\_na = FALSE) %>% # Shows 2018 the highest number of observations  
 adorn\_totals("row") %>%  
 adorn\_percentages("row") %>%  
 adorn\_pct\_formatting(digits = 1) %>%  
 adorn\_ns %>%  
 adorn\_title %>%  
 head(5)  
   
 # Preparing the OECD data  
 oecd\_data\_tidy <- oecd\_data\_raw %>%  
 filter(Variable == "Diseases of the respiratory system") %>%  
 # Get the absolute number of resp. diseases  
 filter(Measure == "Number") %>%  
 select(Year, Country, Variable, Value) %>%  
 arrange(Year) %>%  
 pivot\_wider(names\_from = Variable, values\_from = Value) %>%   
 filter(Year == YEAR) %>%  
 select(Country, starts\_with("Diseases")) %>%  
 mutate(Country = Country %>%  
 factor())  
}

## Rows: 259,263  
## Columns: 11  
## $ VAR <chr> "ACATHEPB", "ACATHEPB", "ACATHEPB", "ACATHEPB", "ACATHEPB…  
## $ Variable <chr> "Immunisation: Hepatitis B", "Immunisation: Hepatitis B",…  
## $ UNIT <chr> "ENFANTTX", "ENFANTTX", "ENFANTTX", "ENFANTTX", "ENFANTTX…  
## $ Measure <chr> "% of children immunised", "% of children immunised", "% …  
## $ COU <chr> "AUS", "AUS", "AUS", "AUS", "AUS", "AUS", "AUS", "AUS", "…  
## $ Country <chr> "Australia", "Australia", "Australia", "Australia", "Aust…  
## $ YEA <dbl> 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 201…  
## $ Year <dbl> 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 201…  
## $ Value <dbl> 91.6, 91.6, 91.8, 91.0, 91.2, 92.8, 94.2, 94.7, 94.5, 94.…  
## $ `Flag Codes` <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, "E", …  
## $ Flags <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, "Esti…

On a Github repository data about **air pollution** is available. This data is captured for countries and cities of Europe. Several air pollution indicators are present. In this report the *PM2.5* (particulate matter with a diameter size of 2.5 micrometers or smaller) is used.

As the air pollution data is also available per city, it will be filtered by the CITY variable, which is set to Milano, as this report is adopted to Physiotherapists in a clinic in Milan.

## Processing air pollution data  
{  
 # Source: https://github.com/dw-data/edjnet-pm2p5)  
 # Reading data   
 air\_pollution\_data\_raw <- read\_csv(here("raw\_data/", "CAMS-Europe-Renalaysis-Countries-Yearly-2018-2022.csv"))  
 air\_pollution\_cities\_data\_raw <- read\_csv(here("raw\_data/", "CAMS-Europe-Renalysis-Yearly-2018-2022.csv"))  
   
 # Eyeballing data  
 air\_pollution\_data\_raw %>% glimpse() # Country names are each in their local language  
 air\_pollution\_data\_raw %>% skimr::skim()  
 air\_pollution\_data\_raw %>% tabyl(`Name (latin characters)`, Year, show\_na = FALSE) %>% # Shows for all years (2018 - 2022) the same amount of observations  
 adorn\_totals("row") %>%  
 adorn\_percentages("row") %>%  
 adorn\_pct\_formatting(digits = 1) %>%  
 adorn\_ns %>%  
 adorn\_title %>%  
 head(5)  
 air\_pollution\_cities\_data\_raw %>% glimpse() # Country names are each in their short code  
 air\_pollution\_cities\_data\_raw %>% skimr::skim()  
 air\_pollution\_cities\_data\_raw %>% tabyl(`Name (latin characters)`, Day, show\_na = FALSE) %>% # Shows for all years (2018 - 2022) the same amount of observations  
 adorn\_totals("row") %>%  
 adorn\_percentages("row") %>%  
 adorn\_pct\_formatting(digits = 1) %>%  
 adorn\_ns %>%  
 adorn\_title %>%  
 head(5)  
   
 # Preparing the air pollution data  
 air\_pollution\_data\_tidy <- air\_pollution\_data\_raw %>%  
 select(Year, "Name (latin characters)", "Yearly PM 2.5 average (µg/m³)", "Population estimate (GHSL 2020)") %>%  
 rename(Country = "Name (latin characters)", PM2.5 = "Yearly PM 2.5 average (µg/m³)", Pop = "Population estimate (GHSL 2020)") %>%  
 filter(Year == YEAR) %>%  
 select(Country, PM2.5, Pop) %>%  
 mutate(Country = Country %>%  
 factor())  
   
 # Preparing the air pollution data for cities  
 air\_pollution\_cities\_data\_tidy <- air\_pollution\_cities\_data\_raw %>%  
 select(Day, `Country code`, "Name (latin characters)", "Daily PM 2.5 average (µg/m³)") %>%  
 rename(Code = `Country code`, City = "Name (latin characters)", PM2.5 = "Daily PM 2.5 average (µg/m³)") %>%  
 filter(Day == paste0(YEAR, "-12-31")) %>%  
 select(City, Code, PM2.5) %>%  
 mutate(City = City %>%  
 factor(),  
 Code = Code %>%  
 factor())  
 # Storing PM2.5 value for the selected city  
 city\_pm <- air\_pollution\_cities\_data\_tidy %>%  
 filter(City == CITY) %>%  
 select(PM2.5) %>%  
 as.numeric()  
}

## Rows: 205  
## Columns: 19  
## $ Column1 <dbl> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11…  
## $ Year <dbl> 2018, 2019, 2020, 2021, 2022, 2018, …  
## $ `NUTS ID` <chr> "FR", "FR", "FR", "FR", "FR", "HR", …  
## $ `Country code` <chr> "FR", "FR", "FR", "FR", "FR", "HR", …  
## $ `Name (latin characters)` <chr> "France", "France", "France", "Franc…  
## $ `Yearly PM 2.5 average (µg/m³)` <dbl> 8.115917, 8.010881, 7.143126, 7.5490…  
## $ `Population estimate (GHSL 2020)` <dbl> 63026742, 63026742, 63026742, 630267…  
## $ `0–5µg/m³ - population` <dbl> 89203, 43901, 201984, 164288, 11839,…  
## $ `5–10µg/m³ - population` <dbl> 38248345, 39335096, 57924834, 498049…  
## $ `10–15µg/m³ - population` <dbl> 24689193, 23647745, 4899923, 1305745…  
## $ `15–20µg/m³ - population` <dbl> 0, 0, 0, 0, 0, 913689, 1516356, 1361…  
## $ `20–25µg/m³ - population` <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
## $ `25+ µg/m³ - population` <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
## $ `0–5µg/m³ - percentage` <dbl> 0.001415322, 0.000696552, 0.00320474…  
## $ `5–10µg/m³ - percentage` <dbl> 0.606858999, 0.624101685, 0.91905170…  
## $ `10–15µg/m³ - percentage` <dbl> 0.39172568, 0.37520176, 0.07774355, …  
## $ `15–20µg/m³ - percentage` <dbl> 0.00000000, 0.00000000, 0.00000000, …  
## $ `20–25µg/m³ - percentage` <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
## $ `25+ µg/m³ - percentage` <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
## Rows: 10,935  
## Columns: 7  
## $ ...1 <dbl> 0, 1, 2, 3, 4, 5, 6, 7, 8, …  
## $ Day <date> 2018-12-31, 2019-12-31, 20…  
## $ `NUTS ID` <chr> "DE254", "DE254", "DE254", …  
## $ `Country code` <chr> "DE", "DE", "DE", "DE", "DE…  
## $ `Name (latin characters)` <chr> "Nürnberg, Kreisfreie Stadt…  
## $ `Daily PM 2.5 average (µg/m³)` <dbl> 11.77, 10.52, 10.07, 10.23,…  
## $ `EU Air Quality Guidelines classification` <chr> "Fair (10-20)", "Fair (10-2…

From a clinical perspective, **overweight and obesity** data is of interest as it is a risk factor for many health-related problems. The data taken is available on the OECD website.

## Processing overweight / obesity data  
{  
 # Source: https://data.oecd.org/healthrisk/overweight-or-obese-population.htm  
 # Reading data  
 obesity\_data\_raw <- read\_csv(here("raw\_data/", "DP\_LIVE\_02112023125750641.csv"))  
   
 # Eyeballing data  
 obesity\_data\_raw %>% glimpse() # Country names are in a 3-letter country-code format  
 obesity\_data\_raw %>% skimr::skim()  
 obesity\_data\_raw %>% tabyl(LOCATION, TIME, show\_na = FALSE) %>% # Shows 2019 the highest number of observations  
 adorn\_totals("row") %>%  
 adorn\_percentages("row") %>%  
 adorn\_pct\_formatting(digits = 1) %>%  
 adorn\_ns %>%  
 adorn\_title %>%  
 head(5)  
   
 # Preparing the obesity data  
 obesity\_data\_tidy <- obesity\_data\_raw %>%  
 filter(TIME == YEAR) %>%  
 select(LOCATION, Value) %>%  
 rename(Country = LOCATION) %>%  
 group\_by(Country) %>%  
 # Explain why I used the mean and not dropped the `self\_measured` column  
 summarise(Value = mean(Value, na.rm = TRUE)) %>%  
 mutate(Country = Country %>%  
 factor())  
}

## Rows: 80  
## Columns: 8  
## $ LOCATION <chr> "AUT", "BEL", "CAN", "CAN", "CAN", "CAN", "CAN", "CZE", "…  
## $ INDICATOR <chr> "OVEROBESE", "OVEROBESE", "OVEROBESE", "OVEROBESE", "OVER…  
## $ SUBJECT <chr> "SELFREPORTED", "SELFREPORTED", "MEASURED", "SELFREPORTED…  
## $ MEASURE <chr> "PC\_POP15", "PC\_POP15", "PC\_POP15", "PC\_POP15", "PC\_POP15…  
## $ FREQUENCY <chr> "A", "A", "A", "A", "A", "A", "A", "A", "A", "A", "A", "A…  
## $ TIME <dbl> 2019, 2018, 2019, 2018, 2019, 2020, 2021, 2019, 2018, 201…  
## $ Value <dbl> 51.1, 49.3, 59.8, 53.8, 54.2, 54.4, 55.5, 58.4, 55.0, 57.…  
## $ `Flag Codes` <chr> NA, NA, NA, NA, NA, NA, NA, NA, "D", NA, "D", NA, NA, NA,…

Smoking is a major risk factor for many diseases, as an extensive body of academic literature and research is showing. Smoking is predominantly associated with the etiology of lung cancer and a range of pulmonary diseases. Therefore, data on the percentage of smokers in the population are also included in the analysis.

## Processing smoke data  
{  
 # Source: https://www.statista.com/statistics/433390/individuals-who-currently-smoke-cigarettes-in-european-countries/  
 # Reading data   
 smoking\_data\_raw <- read\_csv(here("raw\_data/", "statistic\_id433390\_current-smokers-in-europe-2020-by-country.csv"))  
   
 # Eyeballing data  
 smoking\_data\_raw %>% glimpse()  
   
 # Preparing the obesity data  
 smoking\_data\_tidy <- smoking\_data\_raw %>%  
 mutate(Country = Country %>%  
 factor())  
}

## Rows: 28  
## Columns: 2  
## $ Country <chr> "Greece", "Bulgaria", "Croatia", "Latvia", "Cz…  
## $ `Percentage of smokers` <dbl> 42, 38, 36, 32, 30, 30, 28, 28, 28, 28, 27, 26…

On the **CIA Factbook**, the index of the *median age*, that summarizes the age distribution of a population, is distributed. Currently, the median age per country ranges from a low of about 15 in Niger or Uganda to 45 or more in several European countries or Japan. The Median age variable can be seen as a potential confounder on the correlation between respiratory diseases (dependent) and other variables like air pollution (independent).

## Processing median age data  
{  
 # Source: https://www.cia.gov/the-world-factbook/field/median-age/country-comparison/  
 # Reading data  
 age\_data\_raw <- read\_csv(here("raw\_data/", "cia\_factbook\_median\_age.csv"))  
   
 # Eyeballing data  
 age\_data\_raw %>% glimpse()  
   
 # Preparing the obesity data  
 age\_data\_tidy <- age\_data\_raw %>%  
 select(name, value) %>%  
 rename(Country = name, `Median age` = value) %>%  
 mutate(Country = Country %>%  
 factor())  
   
 # Remove unused data / values from Global Environment  
 rm(list = c("air\_pollution\_data\_raw", "obesity\_data\_raw", "oecd\_data\_raw", "air\_pollution\_cities\_data\_raw", "air\_pollution\_cities\_data\_tidy", "smoking\_data\_raw", "age\_data\_raw"))  
}

## Rows: 227  
## Columns: 6  
## $ name <chr> "Monaco", "Japan", "Saint Pierre and Miquelon", "G…  
## $ slug <chr> "monaco", "japan", "saint-pierre-and-miquelon", "g…  
## $ value <dbl> 55.4, 48.6, 48.5, 47.8, 46.5, 46.2, 45.6, 45.6, 45…  
## $ date\_of\_information <dbl> 2020, 2020, 2020, 2020, 2020, 2020, 2020, 2020, 20…  
## $ ranking <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,…  
## $ region <chr> "Europe", "East and Southeast Asia", "North Americ…

**Note:** *The different datasets used incompatible encodings for the country variable, English language, local language or a code, which lead to the need to normalize them to a standardized convention. In this case the English naming for country and continent is chosen.*

As Large Language Models (LLM) are an effective way to automate the processing of large amounts of text, the OpenAI API is used to convert the country variables in a consistent matching format. Secondly, the API is used to associate each state with its corresponding continental landmass, as this data is missing and needed for grouping by continents.

## Checking common countries in the datasets  
## ChatGPT or loading existing data  
if (nchar(OPENAI\_API\_KEY) == 51) {  
 message("API key is valid, requesting data from OpenAI")  
  
 ## Filter unique countries and continents  
 # Filter dataset `oecd\_data\_tidy`  
 countries\_oecd <- oecd\_data\_tidy %>%  
 select(Country) %>%  
 distinct() %>%  
 arrange(Country)  
   
 # Filter dataset `air\_pollution\_data\_tidy`  
 countries\_airpol <- air\_pollution\_data\_tidy %>%  
 select(Country) %>%  
 distinct() %>%  
 arrange(Country)  
   
 # Filter dataset `obesity\_data\_tidy`  
 countries\_obesity <- obesity\_data\_tidy %>%  
 select(Country) %>%  
 distinct() %>%  
 arrange(Country)  
   
 # Filter dataset `smoking\_data\_tidy`  
 countries\_smoking <- smoking\_data\_tidy %>%  
 select(Country) %>%  
 distinct() %>%  
 arrange(Country)  
   
 # Filter dataset `age\_data\_tidy`  
 countries\_age <- age\_data\_tidy %>%  
 select(Country) %>%  
 distinct() %>%  
 arrange(Country)  
   
 ## Convert country tibbles to chr  
 # Create empty vector  
 countries <- character()  
 # OECD data,  
 countries <- append(countries, list(levels(countries\_oecd$Country)))  
 # air pollution data  
 countries <- append(countries, list(levels(countries\_airpol$Country)))  
 # smoking data,  
 countries <- append(countries, list(levels(countries\_smoking$Country)))  
 # age data,  
 countries <- append(countries, list(levels(countries\_age$Country)))  
 # OECD obesity data  
 countries <- append(countries, list(levels(countries\_obesity$Country)))  
   
 ## Request AI completion to standardize the country names  
 # Loop through OpenAI ChatGPT completion  
 countries\_cleaned <- vector()  
 check <- character()  
 country <- character()  
 continent\_list <- ""  
 for (i in 1:5) {  
 # Remove unneeded character from country lists  
 countries[i] <- countries[i] %>% gsub("\n", "", .) %>% gsub('\\"', "", .) %>% gsub("c(", "", ., fixed = TRUE) %>% gsub(")", "", ., fixed = TRUE)  
   
 # Request AI completion, as long as output has same length as input  
 while (length(strsplit(as.character(countries[i]), ", ")[[1]]) != length(strsplit(as.character(countries\_cleaned[i]), ", ")[[1]])) {  
 print("Input")  
 print(paste(countries[i], collapse = ", "))  
 print(paste0("Countries: ", length(strsplit(as.character(countries[i]), ", ")[[1]])))  
 request <- create\_chat\_completion(model = "gpt-3.5-turbo", openai\_api\_key = OPENAI\_API\_KEY, messages = list(  
 list("role" = "system", "content" = "You are a translator of country names or descriptions which comes often in national language, several anmes or abbrevations, into a single standard english country name. Use `Türkiye` instead of `Turkey`."),  
 list("role" = "user", "content" = "I will give you a list of countries and I need you to reply just the english country names, comma seperated, without any further content! Please give me an sample output of '  
 Türkiye, Danmark, DEU, Schweiz/Suisse/Svizzera, Shqipëria'."),  
 list("role" = "assistant", "content" = "Türkiye, Denmark, Germany, Switzerland, Albania"),  
 list("role" = "user", "content" = "Exactly, like that! Now I give you the list. `Türkiye` is the new name for formerly `Turkey`"),  
 list("role" = "user", "content" = paste(countries[i], collapse = ", "))))  
 countries\_cleaned[i] <- request$choices$message.content  
 print("Output")  
 print(countries\_cleaned[i])  
 print(paste0("Countries: ", length(strsplit(as.character(countries\_cleaned[i]), ", ")[[1]])))  
 print("------------------")  
 }  
   
 # From chr to vector  
 check <- append(check, list(strsplit(as.character(countries[i]), split = ", ")[[1]]))  
 country <- append(country, list(strsplit(countries\_cleaned[i], split = ", ")[[1]]))  
   
 # Continent list  
 if (i == 5) {  
 # Loop as long number of countries are not equal to number continents  
 while (length(unlist(country[i])) != length(continent\_list)) {  
 print("Input")  
 print(paste(countries\_cleaned[i], collapse = ", "))  
 print(paste0("Country list length: ", length(unlist(country[i]))))  
 request <- create\_chat\_completion(model = "gpt-3.5-turbo", openai\_api\_key = OPENAI\_API\_KEY, messages = list(  
 list("role" = "system", "content" = "You convert country names to the english continent names they are on."),  
 list("role" = "user", "content" = "I will give you a list of countries and I need you to print out the english continent names, comma seperate all countries, without any further content! Please give me an sample output of 'Germany, Mexico, Japan, Chile'."),  
 list("role" = "assistant", "content" = "Europe, North Amercia, Asia, South America"),  
 list("role" = "user", "content" = "Exactly, like that! Now I give you the list."),  
 list("role" = "user", "content" = paste(countries\_cleaned[i], collapse = ", "))))  
 continent\_list <- request$choices$message.content  
 continent\_list <- strsplit(continent\_list, split = ", ")[[1]]  
 print("Output")  
 print(paste(continent\_list, collapse = ", "))  
 print(paste0("Continent list length: ", length(continent\_list)))  
 }  
   
 # Find common countries in the lists  
 country\_list <- intersect(intersect(intersect(intersect(unlist(country[1]), unlist(country[2])), unlist(country[3])), unlist(country[4])), unlist(country[5]))  
   
 # Save lists  
 saveRDS(country\_list, here("processed\_data/", "country\_list.rds"))  
 saveRDS(continent\_list, here("processed\_data/", "continent\_list.rds"))  
 saveRDS(check, here("processed\_data/", "check.rds"))  
 saveRDS(country, here("processed\_data/", "country.rds"))  
 }  
 }  
} else {  
 message("API key is invalid, loading data from disk ...")  
  
 # Load lists, if API key is missing  
 country\_list <- readRDS(here("processed\_data/", "country\_list.rds"))  
 continent\_list <- readRDS(here("processed\_data/", "continent\_list.rds"))  
 check <- readRDS(here("processed\_data/", "check.rds"))  
 country <- readRDS(here("processed\_data/", "country.rds"))  
}  
  
## Show common countries  
{  
 print("############################")  
 print(paste0("### Common countries: ", length(country\_list), " ###"))  
 print("############################")  
  
 # Remove unused data / values from Global Environment  
 rm(list = c("i", "request", "countries" ,"countries\_cleaned", "countries\_airpol", "countries\_obesity", "countries\_oecd", "countries\_smoking", "countries\_age"))  
}

## [1] "############################"  
## [1] "### Common countries: 13 ###"  
## [1] "############################"

**Note:** *If no OpenAI API Key is provided, the code falls back to already stored data in the repository.*

To prepare the datasets for plotting, it is needed to drop all observations in the datasets which are not in the common country list. This is done in the following code chunk.

## Cleaning country descriptions  
{  
 ## Create dataset `oecd\_data\_plot`  
 # Standardize country names ...  
 oecd\_plot <- oecd\_data\_tidy  
 for (i in 1:length(unlist(country[1]))) {  
 oecd\_plot <- oecd\_plot %>%  
 mutate(Country = if\_else(Country == unlist(check[1])[i], unlist(country[1])[i], Country))  
 }  
 # ... and filter for common countries in datasets  
 oecd\_plot <- oecd\_plot %>%  
 filter(Country %in% country\_list)  
   
 ## Create dataset `air\_pollution\_plot`  
 # Standardize country names ...  
 air\_pollution\_plot <- air\_pollution\_data\_tidy  
 for (i in 1:length(unlist(country[2]))) {  
 air\_pollution\_plot <- air\_pollution\_plot %>%  
 mutate(Country = if\_else(Country == unlist(check[2])[i], unlist(country[2])[i], Country))  
 }  
 # ... and filter for common countries in datasets  
 air\_pollution\_plot <- air\_pollution\_plot %>%  
 filter(Country %in% country\_list)  
   
 ## Create dataset `smoking\_data\_plot`  
 # Standardize country names ...  
 smoking\_plot <- smoking\_data\_tidy  
 for (i in 1:length(unlist(country[3]))) {  
 smoking\_plot <- smoking\_plot %>%  
 mutate(Country = if\_else(Country == unlist(check[3])[i], unlist(country[3])[i], Country))  
 }  
 # ... and filter for common countries in datasets  
 smoking\_plot <- smoking\_plot %>%  
 filter(Country %in% country\_list)  
   
 ## Create dataset `age\_data\_plot`  
 # Standardize country names ...  
 age\_plot <- age\_data\_tidy  
 for (i in 1:length(unlist(country[4]))) {  
 smoking\_plot <- smoking\_plot %>%  
 mutate(Country = if\_else(Country == unlist(check[4])[i], unlist(country[4])[i], Country))  
 }  
 # ... and filter for common countries in datasets  
 age\_plot <- age\_plot %>%  
 filter(Country %in% country\_list)  
   
 ## Create dataset `obesity\_plot`  
 # Standardize country names  
 obesity\_plot <- obesity\_data\_tidy  
 for (i in 1:length(unlist(country[5]))) {  
 obesity\_plot <- obesity\_plot %>%  
 mutate(Country = if\_else(Country == unlist(check[5])[i], unlist(country[5])[i], Country))  
 }  
 # ... and filter for common countries in datasets  
 obesity\_plot <- obesity\_plot %>%  
 filter(Country %in% country\_list)  
   
 # Remove unused data / values from Global Environment  
 rm(list = c("air\_pollution\_data\_tidy", "obesity\_data\_tidy", "oecd\_data\_tidy", "smoking\_data\_tidy", "i", "check", "country", "country\_list", "continent\_list"))  
}

To create the final dataset for plotting, the OECD data about resp. diseases is enriched with other datasets (air\_pollution, obesity, smoking and age).

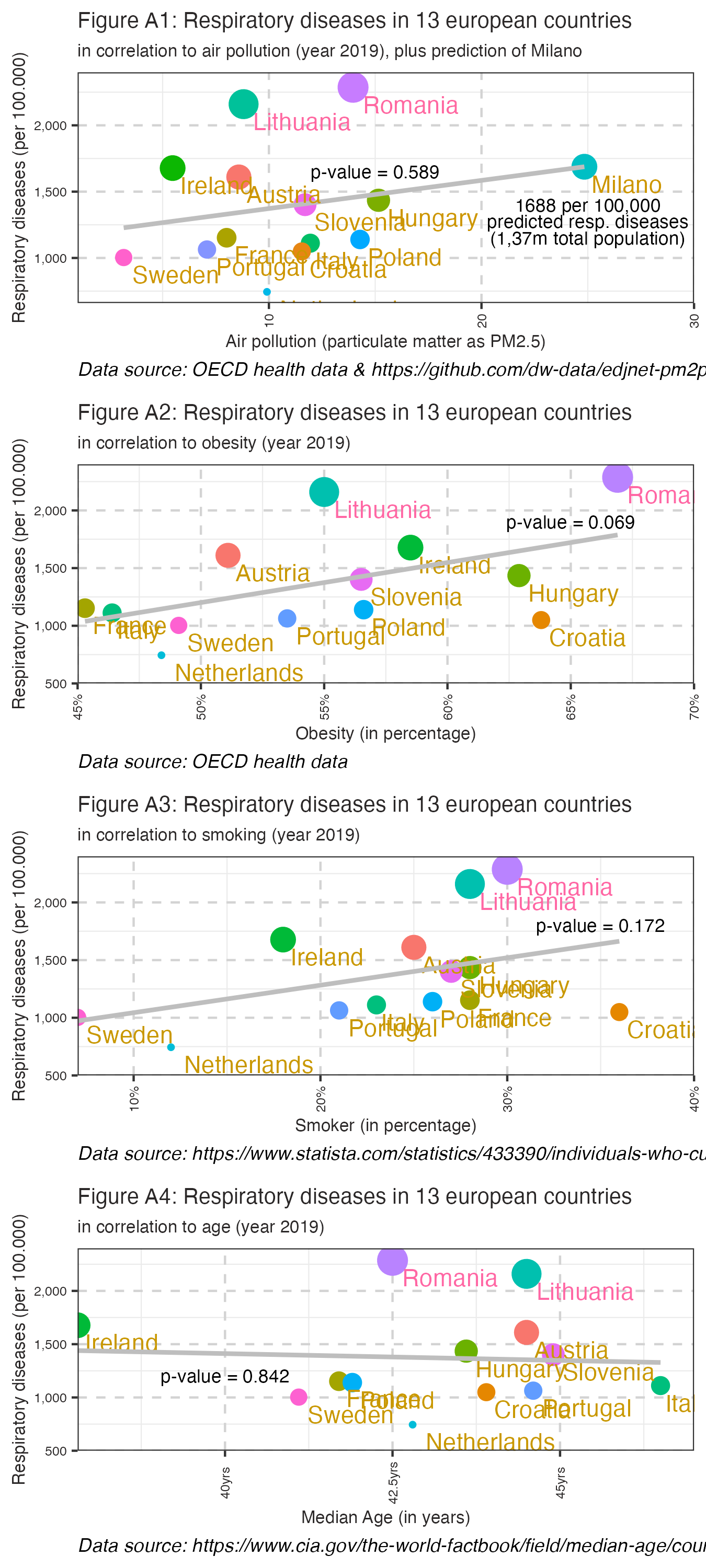
## Preparation of the plotting datasets  
{  
 ## Combining the datasets  
 # Joining OECD data and air pollution data  
 data <- left\_join(oecd\_plot, air\_pollution\_plot, by = "Country")  
   
 # Joining obesity data  
 data <- left\_join(data, obesity\_plot, by = "Country")  
   
 # Joining smoking data  
 data <- left\_join(data, smoking\_plot, by = "Country")  
   
 # Joining age data  
 data <- left\_join(data, age\_plot, by = "Country")  
   
 ## Prepare the dataset for plotting  
 # Calculate the diseases per 100.000  
 data\_plot <- data %>%  
 mutate("Diseases of the respiratory system" = `Diseases of the respiratory system` / Pop \* 100000) %>%  
 rename("Resp. diseases per 100.000" = `Diseases of the respiratory system`) %>%  
 rename("Obesity" = Value) %>%  
 rename("Smoker" = `Percentage of smokers`)  
   
 # Remove unused data / values from Global Environment  
 rm(list = c("data", "oecd\_plot", "air\_pollution\_plot", "obesity\_plot", "smoking\_plot", "age\_plot"))  
}

**Note:** *The data is in a tidy format, meaning each variable is in it’s own column and each observation in it’s own row, and ready for plotting.*

# 3 Data visualisation

After finding the data for median age, smoking, obesity / overweight and air pollution, it is now set into relation to the respiratory diseases variable. The following visualizations is proving if there are **correlations** between any of them.

## Visualisation  
{  
 ## Figure A1: Point figure with a linear regression line  
 # Calculate dependence of respiratory diseases in correlation to air pollution  
 model = lm(`Resp. diseases per 100.000` ~ PM2.5, data = data\_plot)  
 p = round(summary(model)$coefficients[,4][2], digits = 3)  
   
 # Predict resp. diseases per 100,000 for selected city  
 city\_value <- data.frame(PM2.5 = city\_pm)  
 city\_resp\_diseases <- as.numeric(predict(model, newdata = city\_value))  
 data\_plot\_with\_city <- data\_plot %>%  
 add\_row(Country = CITY, `Resp. diseases per 100.000` = city\_resp\_diseases, PM2.5 = city\_pm, `Median age` = NA)  
 # Show correlation between respiratory diseases and air pollution, plus selected city  
 p1a <- data\_plot\_with\_city %>%  
 ggplot(aes(x = PM2.5, y = `Resp. diseases per 100.000`, color = Country)) +  
 geom\_point(aes(color = Country, size = `Resp. diseases per 100.000`)) +  
 geom\_smooth(method = "lm", se = FALSE, color = "gray") +  
 geom\_text(aes(label = Country, color = `Resp. diseases per 100.000` > 2000), hjust = -0.1, vjust = 1.5) +  
 labs(title = paste0("Figure A1: Respiratory diseases in ", length(data\_plot$Country)," european countries"),  
 subtitle = paste0("in correlation to air pollution (year ", YEAR, "), plus prediction of ", CITY),  
 x = "Air pollution (particulate matter as PM2.5)", y = "Respiratory diseases (per 100.000)",   
 caption = "Data source: OECD health data & https://github.com/dw-data/edjnet-pm2p5") +  
 scale\_y\_continuous(labels = scales::label\_comma()) +  
 expand\_limits(x = c(1, 30), y = c(660, 2400)) +  
 coord\_cartesian(expand = FALSE) +  
 annotate("text", x = 15, y = 1650, label = paste0("p-value = ", p), size = 3) +  
 annotate("text", x = 25, y = 1400, label = paste0(round(city\_resp\_diseases, digits = 0), " per 100,000"), size = 3) +  
 annotate("text", x = 25, y = 1275, label = "predicted resp. diseases", size = 3) +  
 annotate("text", x = 25, y = 1150, label = "(1,37m total population)", size = 3) +  
 theme\_bw() +  
 theme(legend.position = "none",   
 plot.caption = element\_text(face = "italic", hjust = 0),  
 panel.grid.major = element\_line(colour = "lightgrey", linetype = "dashed"),  
 plot.title = element\_text(size = 10, colour = "#2b2828", hjust = 0),  
 plot.subtitle = element\_text(size = 8, colour = "#2b2828", hjust = 0),  
 axis.text.y = element\_text(size = 6, colour = "#2b2828"),  
 axis.text.x = element\_text(size = 6, colour = "#2b2828", angle = 90, vjust = 0.5),  
 axis.title = element\_text(size = 8, hjust = 0.5, colour = "#2b2828"))  
  
 ## Figure A2: Point figure with a linear regression line  
 # Calculate dependence of respiratory diseases in correlation to obesity  
 model = lm(`Resp. diseases per 100.000` ~ Obesity, data = data\_plot)  
 p = round(summary(model)$coefficients[,4][2], digits = 3)  
 # Correlation between respiratory diseases and overweight & obesity  
 p1b <- data\_plot %>%  
 ggplot(aes(x = Obesity, y = `Resp. diseases per 100.000`, color = Country)) +  
 geom\_point(aes(color = Country, size = `Resp. diseases per 100.000`)) +  
 geom\_text(aes(label = Country, color = `Resp. diseases per 100.000` > 2000), hjust = -0.1, vjust = 1.5) +  
 geom\_smooth(method = "lm", se = FALSE, color = "gray") +  
 annotate("text", x = 65, y = 1900, label = paste0("p-value = ", p), size = 3) +  
 labs(title = paste0("Figure A2: Respiratory diseases in ", length(data\_plot$Country), " european countries"),  
 subtitle = paste0("in correlation to obesity (year ", YEAR, ")"),  
 x = "Obesity (in percentage)", y = "Respiratory diseases (per 100.000)",   
 caption = "Data source: OECD health data") +  
 scale\_x\_continuous(labels = function(x) paste0(x, '%')) +  
 scale\_y\_continuous(labels = scales::label\_comma()) +  
 expand\_limits(x = c(45, 70), y = c(500, 2400)) +  
 coord\_cartesian(expand = FALSE) +  
 theme\_bw() +  
 theme(legend.position = "none",   
 plot.caption = element\_text(face = "italic",hjust = 0),  
 panel.grid.major = element\_line(colour = "lightgrey", linetype = "dashed"),  
 plot.title = element\_text(size = 10, colour = "#2b2828", hjust = 0),  
 plot.subtitle = element\_text(size = 8, colour = "#2b2828", hjust = 0),  
 axis.text.y = element\_text(size = 6, colour = "#2b2828"),  
 axis.text.x = element\_text(size = 6, colour = "#2b2828", angle = 90, vjust = 0.5),  
 axis.title = element\_text(size = 8, colour = "#2b2828"))  
  
 ## Figure A3: Point figure with a linear regression line  
 # Calculate dependence of respiratory diseases in correlation to smoking  
 model = lm(`Resp. diseases per 100.000` ~ Smoker, data = data\_plot)  
 p = round(summary(model)$coefficients[,4][2], digits = 3)  
 # Correlation between respiratory diseases and smoking  
 p1c <- data\_plot %>%  
 ggplot(aes(x = Smoker, y = `Resp. diseases per 100.000`, color = Country)) +  
 geom\_point(aes(color = Country, size = `Resp. diseases per 100.000`)) +  
 geom\_text(aes(label = Country, color = `Resp. diseases per 100.000` > 2000), hjust = -0.1, vjust = 1.5) +  
 geom\_smooth(method = "lm", se = FALSE, color = "gray") +  
 annotate("text", x = 35, y = 1800, label = paste0("p-value = ", p), size = 3) +  
 labs(title = paste0("Figure A3: Respiratory diseases in ", length(data\_plot$Country), " european countries"),  
 subtitle = paste0("in correlation to smoking (year ", YEAR, ")"),  
 x = "Smoker (in percentage)", y = "Respiratory diseases (per 100.000)",   
 caption = "Data source: https://www.statista.com/statistics/433390/individuals-who-currently-smoke-cigarettes-in-european-countries/") +  
 scale\_x\_continuous(labels = function(x) paste0(x, '%')) +  
 scale\_y\_continuous(labels = scales::label\_comma()) +  
 expand\_limits(x = c(15, 40), y = c(500, 2400)) +  
 coord\_cartesian(expand = FALSE) +  
 theme\_bw() +  
 theme(legend.position = "none",   
 plot.caption = element\_text(face = "italic",hjust = 0),  
 panel.grid.major = element\_line(colour = "lightgrey", linetype = "dashed"),  
 plot.title = element\_text(size = 10, colour = "#2b2828", hjust = 0),  
 plot.subtitle = element\_text(size = 8, colour = "#2b2828", hjust = 0),  
 axis.text.y = element\_text(size = 6, colour = "#2b2828"),  
 axis.text.x = element\_text(size = 6, colour = "#2b2828", angle = 90, vjust = 0.5),  
 axis.title = element\_text(size = 8, colour = "#2b2828"))  
   
 ## Figure A4: Point figure with a linear regression line  
 # Calculate dependence of respiratory diseases in correlation to smoking  
 model = lm(`Resp. diseases per 100.000` ~ `Median age`, data = data\_plot)  
 p = round(summary(model)$coefficients[,4][2], digits = 3)  
 # Correlation between respiratory diseases and age  
 p1d <- data\_plot %>%  
 ggplot(aes(x = `Median age`, y = `Resp. diseases per 100.000`, color = Country)) +  
 geom\_point(aes(color = Country, size = `Resp. diseases per 100.000`)) +  
 geom\_text(aes(label = Country, color = `Resp. diseases per 100.000` > 2000), hjust = -0.1, vjust = 1.5) +  
 geom\_smooth(method = "lm", se = FALSE, color = "gray") +  
 annotate("text", x = 40, y = 1200, label = paste0("p-value = ", p), size = 3) +  
 labs(title = paste0("Figure A4: Respiratory diseases in ", length(data\_plot$Country), " european countries"),  
 subtitle = paste0("in correlation to age (year ", YEAR, ")"),  
 x = "Median Age (in years)", y = "Respiratory diseases (per 100.000)",   
 caption = "Data source: https://www.cia.gov/the-world-factbook/field/median-age/country-comparison/") +  
 scale\_x\_continuous(labels = function(x) paste0(x, 'yrs')) +  
 scale\_y\_continuous(labels = scales::label\_comma()) +  
 expand\_limits(x = c(38.5, 47), y = c(500, 2400)) +  
 coord\_cartesian(expand = FALSE) +  
 theme\_bw() +  
 theme(legend.position = "none",   
 plot.caption = element\_text(face = "italic",hjust = 0),  
 panel.grid.major = element\_line(colour = "lightgrey", linetype = "dashed"),  
 plot.title = element\_text(size = 10, colour = "#2b2828", hjust = 0),  
 plot.subtitle = element\_text(size = 8, colour = "#2b2828", hjust = 0),  
 axis.text.y = element\_text(size = 6, colour = "#2b2828"),  
 axis.text.x = element\_text(size = 6, colour = "#2b2828", angle = 90, vjust = 0.5),  
 axis.title = element\_text(size = 8, colour = "#2b2828"))  
  
 # Combine the plots  
 combined\_plot <- arrangeGrob(p1a, p1b, p1c, p1d, nrow = 4)  
   
 # Save the combined plot  
 ggsave(combined\_plot, file = here("figures/", "Plot\_A\_Combined.png"), width = 4.5, height = 10)  
   
 # Remove unused data / values from Global Environment  
 rm(list = c("combined\_plot", "p", "model", "city\_pm", "city\_value", "city\_resp\_diseases", "data\_plot\_with\_city", "data\_plot", "p1a", "p1b", "p1c", "p1d"))  
}  
  
# Show figure  
knitr::include\_graphics(here("figures", "Plot\_A\_Combined.png"))



**Note:** *The strongest correlation is been found in the independent variable Obesity, towards Resp. diseases per 100.000. This could underscore the significance of obesity in relation to resp. diseases. Milano has a predicted rate of 1688 cases per 100,000, which would be around 23,000 cases for the cities population.*

As the data found is restricted to **13 european countries**, the need for a global comparison is not achievable. Therefore, the following code will request global data on obesity from the ourworldindata repository.

In this case, the data does not consist of the percentage of overweight and obesity per country, instead it holds the **deaths related to obesity**.

## More data from countries on other continents needed!  
{  
 # Source: https://ourworldindata.org/obesity  
 obesity\_world\_data\_raw <- read\_csv(here("raw\_data/", "share-of-deaths-obesity.csv"))  
 obesity\_world\_data\_tidy <- obesity\_world\_data\_raw %>%  
 filter(Year == YEAR) %>%  
 # Drop NA rows  
 drop\_na(Code) %>%  
 # Drop `World`  
 filter(str\_length(Code) <= 3) %>%  
 rename("Country" = Code, "Obesity" = starts\_with("Share")) %>%  
 select(Country, Obesity)  
   
 # ChatGPT or loading existing data  
 if (nchar(OPENAI\_API\_KEY) == 51) {  
 message("API key is valid, requesting data from OpenAI")  
   
 # OpenAI ChatGPT API usage to map continents to countries  
 convert\_codes <- function(mode, data) {  
 output <- vector()  
 cat("Converting codes: ")  
 for (country in data) {  
 if (mode == "Country") {  
 request <- create\_chat\_completion(model = "gpt-3.5-turbo", openai\_api\_key = OPENAI\_API\_KEY, messages = list(  
 list("role" = "system", "content" = "You convert a country code to the english country name. Use `Unknown` for unknown country codes."),  
 list("role" = "user", "content" = "Please convert 'DEU'."),  
 list("role" = "assistant", "content" = "Germany"),  
 list("role" = "user", "content" = "Exactly, like that! Now I give you another country code."),  
 list("role" = "user", "content" = country)))  
 output <- append(output, request$choices$message.content)  
 }  
 if (mode == "Continent") {  
 request <- create\_chat\_completion(model = "gpt-3.5-turbo", openai\_api\_key = OPENAI\_API\_KEY, messages = list(  
 list("role" = "system", "content" = "You convert a country code to the english continent name it is located on. Use `Unknown` for unknown country codes or one of these 6 different continent names `Africa`, `Asia`, `Europe`, `North America`, `South America`, `Oceania`.)"),  
 list("role" = "user", "content" = "Please convert 'DEU'."),  
 list("role" = "assistant", "content" = "Europe"),  
 list("role" = "user", "content" = "Exactly, like that! Now I give you another country code."),  
 list("role" = "user", "content" = country)))  
 output <- append(output, request$choices$message.content)  
 }  
 cat("#")  
 }  
 return(output)  
 }  
   
 # Convert country codes to continents  
 continents\_world <- convert\_codes("Continent", data = obesity\_world\_data\_tidy$Country)  
 saveRDS(continents\_world, here("processed\_data/", "continents.rds"))  
   
 # Convert country codes to country names  
 countries\_world <- convert\_codes("Country", data = obesity\_world\_data\_tidy$Country)  
 saveRDS(countries\_world, here("processed\_data/", "countries.rds"))  
 } else {  
 message("API key is invalid, loading existing data")  
   
 # Load saved data (ChatGPT requests are charged!)  
 continents\_world <- readRDS(here("processed\_data/", "continents.rds"))  
 countries\_world <- readRDS(here("processed\_data/", "countries.rds"))  
 }  
   
 # Creating dataset for plotting  
 obesity\_world\_plot <- obesity\_world\_data\_tidy %>%  
 mutate(Country = countries\_world %>%  
 factor()) %>%  
 mutate(Continent = continents\_world %>%  
 factor()) %>%  
 filter(Country != "Unknown") %>%  
 filter(Continent != "Unknown")  
   
 # Remove unused data / values from Global Environment  
 rm(list = c("obesity\_world\_data\_raw", "obesity\_world\_data\_tidy", "continents\_world", "countries\_world", "OPENAI\_API\_KEY", "CITY", "convert\_codes"))  
   
 ## Checking new data  
 # Show levels (Continents)  
 obesity\_world\_plot %>%  
 pull(Continent) %>%  
 levels()  
   
 # Show count of countries per continent  
 obesity\_world\_plot %>%  
 select(Continent) %>%  
 summary()  
   
 # Count African countries  
 afr\_countries <- obesity\_world\_plot %>%  
 filter(Continent == "Africa") %>%  
 select(Continent) %>%  
 count()  
   
 # Count European countries  
 eur\_countries <- obesity\_world\_plot %>%  
 filter(Continent == "Europe") %>%  
 select(Continent) %>%  
 count()  
}

To display more than 200 different countries in a meaningful way, the data is presented as a table, ordered by continents and highlighting the percentage of obesity in brackets.

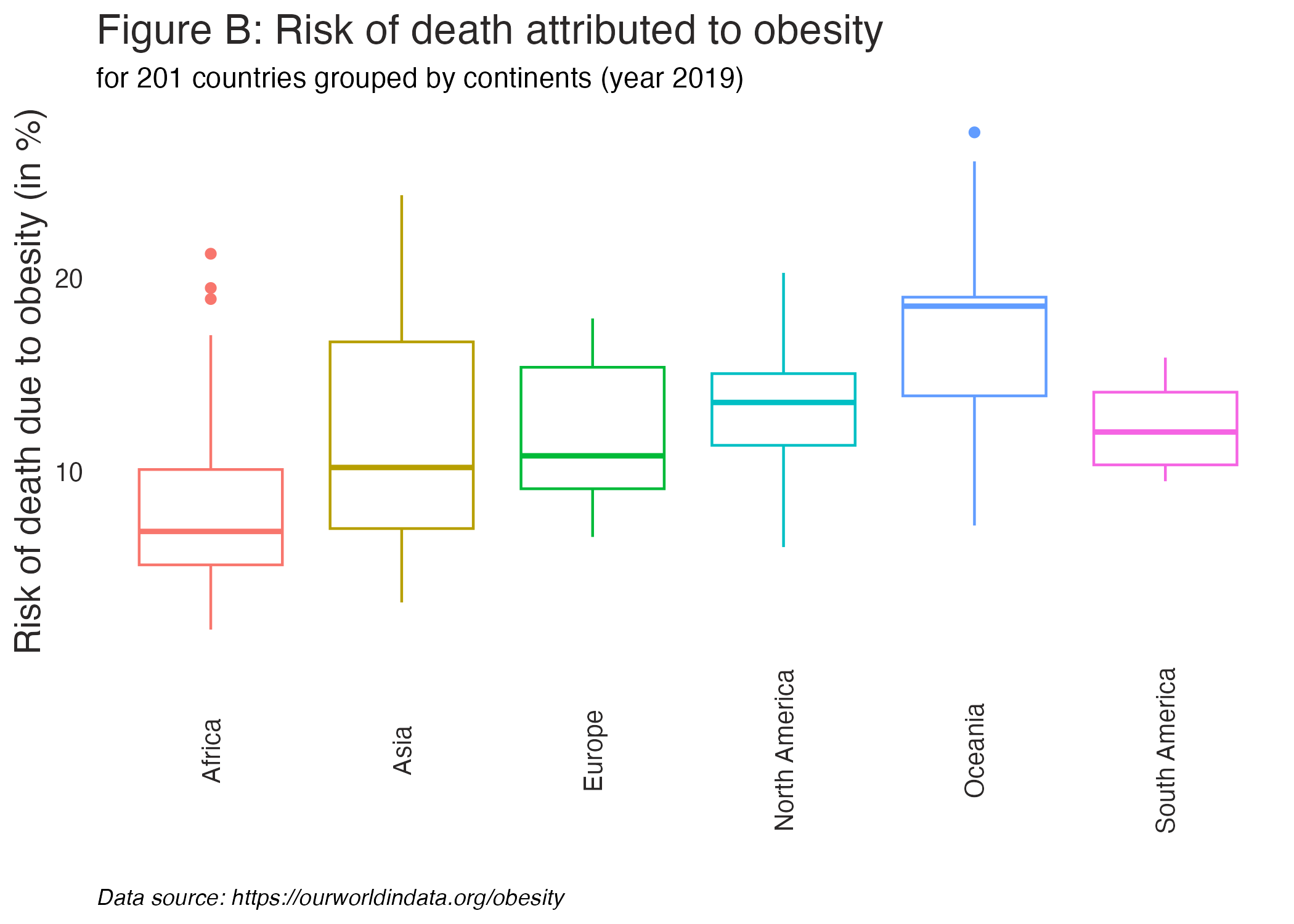
## Table A1: Table with 6 columns (grouped by continents)  
{  
 # Show overview over countries with deaths related to obesity  
 t1 <- obesity\_world\_plot %>%  
 mutate(Country = if\_else(Obesity < 7, md(paste0(Country, ' (<span style="color:green">', round(Obesity, digits = 1), '%</span>)')),   
 if\_else(Obesity < 15, md(paste0(Country, ' (<span style="color:#ff8c00">', round(Obesity, digits = 1), '%</span>)')),   
 md(paste0("\*\*", Country, ' (<span style="color:red">', round(Obesity, digits = 1), '%</span>)\*\*'))))) %>%  
 arrange(Obesity) %>%  
 select(Country, Continent) %>%  
 pivot\_wider(names\_from = Continent, values\_from = Country) %>%  
 gt() %>%  
 tab\_header(title = "Table A1: Percentage of deaths attributed to obesity",  
 subtitle = paste0("for ", length(obesity\_world\_plot$Country), " countries, grouped by continents (year ", YEAR, ")")) %>%  
 tab\_source\_note(source\_note = md("\*Data source: https://ourworldindata.org/obesity\*")) %>%  
 tab\_footnote(footnote = paste0("EUROPE includes ", eur\_countries, " countries (avg. age 42.5yrs), AFRICA includes ", afr\_countries, " countries (avg. age 19.7yrs)"),  
 locations = cells\_column\_labels(columns = c(Europe, Africa))) %>%  
 cols\_move(columns = Africa, after = Asia) %>%   
 cols\_move(columns = `North America`, after = Europe) %>%  
 cols\_move(columns = Oceania, after = `South America`) %>%  
 cols\_label(Africa = md("\*\*AFRICA\*\*"), Asia = md("\*\*ASIA\*\*"), `North America` = md("\*\*NORTH AMERICA\*\*"),  
 Europe = md("\*\*EUROPE\*\*"), `South America` = md("\*\*SOUTH AMERICA\*\*"), Oceania = md("\*\*OCEANIA\*\*")) %>%  
 cols\_align(align = "left", columns = everything()) %>%  
 tab\_spanner(label = md("\*Focus continents\*"), columns = c(Africa, Europe))  
   
 # Save the table  
 gtsave(t1, filename = "Table\_A1\_Deaths\_Obesity.pdf", path = here("tables/"))  
}  
  
# Remove unused data / values from Global Environment  
rm(list = c("t1", "eur\_countries", "afr\_countries"))  
  
# Show PDF  
knitr::include\_graphics(here("tables", "Table\_A1\_Deaths\_Obesity.pdf"))



**Note:** *At the bottom of the table the countries with the highest rates of deaths related to obesity can be found.*

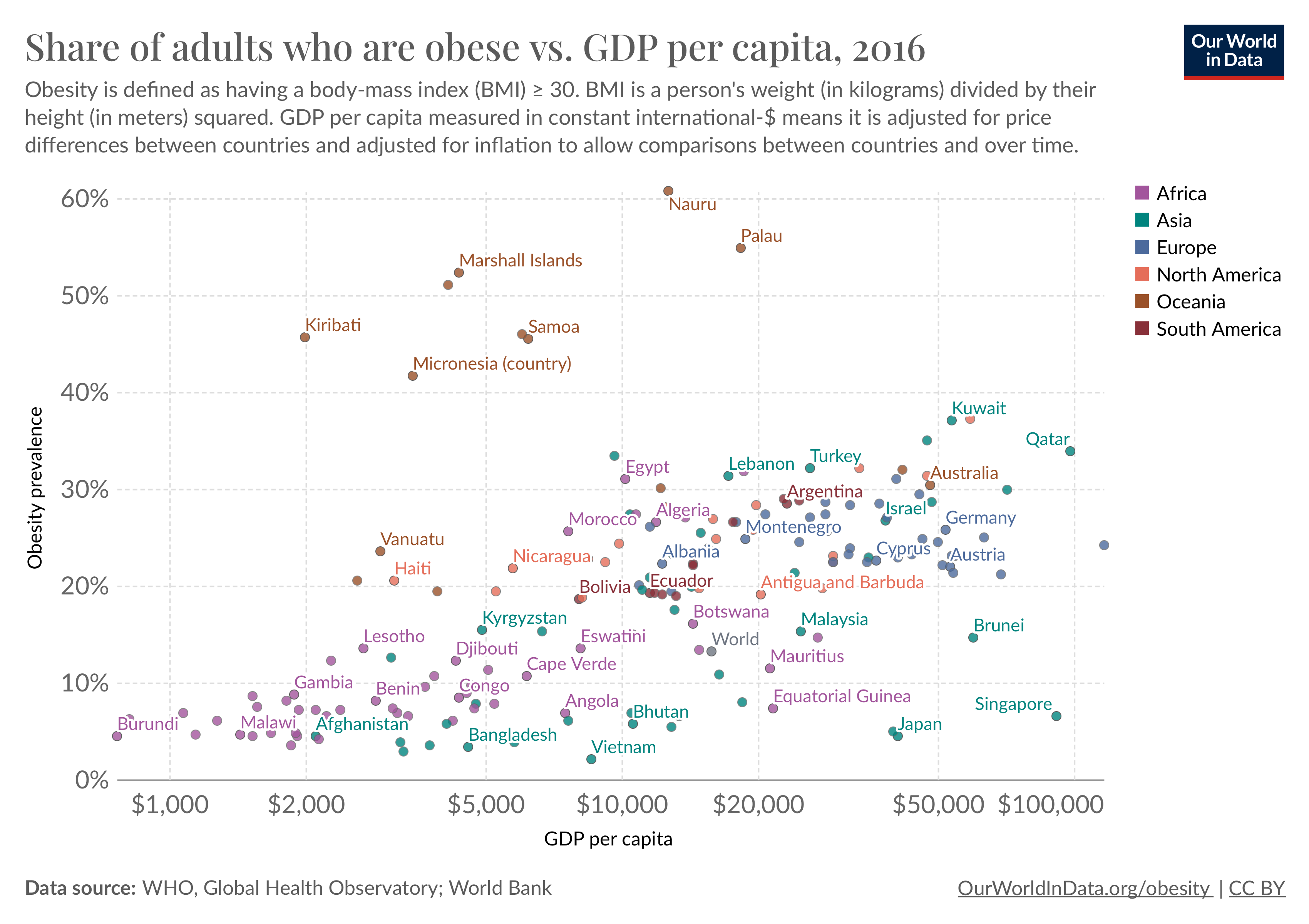
To summarize this observation per continent, a boxplot is used. The boxplot shows the median of the deaths attributed to obesity per continent.

## Figure B: Boxplot, grouped by continents  
{  
 # Rename `obesity` to `Risk of death`  
 obesity\_world\_plot <- obesity\_world\_plot %>%  
 rename("Risk of death" = "Obesity")  
   
 # Show deaths attributed to obesity worldwide  
 p2 <- obesity\_world\_plot %>%  
 ggplot(aes(x = Continent, y = `Risk of death`)) +  
 geom\_boxplot(aes(color = Continent)) +  
 labs(title = "Figure B: Risk of death attributed to obesity",  
 subtitle = paste0("for ", length(obesity\_world\_plot$Country), " countries grouped by continents (year ", YEAR, ")"),  
 x = "", y = "Risk of death due to obesity (in %)",   
 caption = "Data source: https://ourworldindata.org/obesity") +  
 theme\_bw() +  
 theme(legend.position = "none",   
 plot.caption = element\_text(face = "italic", hjust = 0),  
 plot.background = element\_blank(), panel.grid.major = element\_blank(),  
 panel.grid.minor = element\_blank(), panel.border = element\_blank(),  
 plot.title = element\_text(size = 16, colour = "#2b2828", hjust = 0),  
 axis.ticks.x = element\_blank(), axis.ticks.y = element\_blank(),  
 axis.text.x = element\_text(size = 10, colour = "#2b2828", angle = 90, vjust = 0.5),  
 axis.text.y = element\_text(size = 10, colour = "#2b2828"),  
 axis.title = element\_text(size = 14, hjust = 0.5, colour = "#2b2828"))  
   
 # Save the table  
 ggsave(p2, file = here("figures/", "Plot\_B\_Deaths\_Obesity.png"), width = 7, height = 5)  
}  
  
# Show figure  
knitr::include\_graphics(here("figures", "Plot\_B\_Deaths\_Obesity.png"))



# Remove unused data / values from Global Environment  
rm("p2")

**Note:** *Africa has the lowest median, while Oceania has the highest. Europa also shows many countries with a high rate of health issues related to obesity.*



This figure from **Our World in Data** shows the relationship between obesity and GDP. It indicates that *the higher the GDP*, *the higher the obesity rate*.

As second table is created for grouping countries not by continent but by risk factor as also the median age in three groups.

## Table A2: Table grouped by risks and age groups  
{  
 # Add avg. age per country  
 obesity\_world\_plot2 <- left\_join(obesity\_world\_plot, age\_data\_tidy, by = "Country")  
   
 # Convert variable `Risk of death` values from numeric to categorical  
 obesity\_world\_plot2 <- obesity\_world\_plot2 %>%  
 mutate(`Risk of death` = cut(`Risk of death`, breaks = c(0, 7, 15, 100), labels = c("Low", "Medium", "High"), include.lowest = TRUE)) %>%  
 mutate(`Median age` = cut(`Median age`, breaks = c(0, 25, 40, 100), labels = c("Young", "Middle", "Old"), include.lowest = TRUE)) %>%  
 rename("Age" = "Median age") %>%  
 drop\_na(Age)  
   
 # Calculate highest risk group  
 len\_list <- obesity\_world\_plot2 %>%  
 filter(`Risk of death` == "High" & Age == "Old") %>%  
 count()  
   
 # Show overview of risk of death related to obesity and age group  
 t2 <- obesity\_world\_plot2 %>%  
 arrange(`Risk of death`) %>%  
 mutate(Country = if\_else(`Risk of death` == "Low", md(paste0('<span style="color:green">', Country, ' </span>')),   
 if\_else(`Risk of death` == "Medium", md(paste0('<span style="color:#ff8c00">', Country, '</span>')),   
 if\_else(Age == "Old", md(paste0('\*\*<span style="color:darkred">', Country, '</span>\*\*')),  
 md(paste0('\*\*<span style="color:red">', Country, '</span>\*\*'))))),  
 Age = toupper(Age)) %>%  
 select(Country, `Risk of death`, Age) %>%  
 pivot\_wider(names\_from = `Risk of death`, values\_from = Country) %>%  
 gt() %>%  
 tab\_header(title = "Table A2: Risk of death due to obesity",  
 subtitle = paste0("for ", length(obesity\_world\_plot2$Country), " countries, grouped by risk and age (year ", YEAR, ")")) %>%  
 tab\_source\_note(source\_note = md("\*Data sources: https://ourworldindata.org/obesity\*, https://en.wikipedia.org/wiki/List\_of\_countries\_by\_median\_age")) %>%  
 tab\_footnote(footnote = paste0("Legend: `YOUNG` ≙ under 25yrs avg. age; `MIDDLE` ≙ 25 - 40yrs avg. age; `OLD` ≙ over 40yrs avg. age."),  
 locations = cells\_column\_labels(columns = Age)) %>%   
 tab\_footnote(footnote = paste0("Subgroup of high risk & and high avg. age includes ", len\_list, " countries."),  
 locations = cells\_column\_labels(columns = High)) %>%  
 cols\_move(columns = Low, after = Age) %>%  
 cols\_label(Age = md("\*\*AGE\*\*"), Low = md("\*LOW RISK\*"), Medium = md("\*MEDIUM RISK\*"), High = md("\*HIGH RISK\*")) %>%  
 cols\_align(align = "left", columns = everything()) %>%  
 tab\_spanner(label = md("\*Focus countries\*"), columns = High)  
  
 # Save the table  
 gtsave(t2, filename = "Table\_A2\_Deaths\_Risk.pdf", path = here("tables/"))  
}  
  
# Remove unused data / values from Global Environment  
rm(list = c("t2", "obesity\_world\_plot", "obesity\_world\_plot2", "age\_data\_tidy", "YEAR", "len\_list"))  
  
# Show PDF  
knitr::include\_graphics(here("tables", "Table\_A2\_Deaths\_Risk.pdf"))



**Note:** *Here we see that the countries with the highest risk of death due to obesity and a high median age. Many of them in Europe.*

# 4 Discussion and analysis

The data analysis shows a strong correlation (p = 0.07) between the appearance of overweight / obesity and the risk of developing a respiratory disease. In clinical experience of Physiotherapists, obesity is known as a critical factor in health outcomes of patients. The abdominal fat, is suspected to negatively affect the mobility and function of the lungs in the upper torso Rauch, E. (2015). European countries have a high prevalence of obesity, consequently reducing obesity would serve as an initial step towards the diminution of prevalence rates of resp. diseases.

The data also shows moderate correlation between smoking and resp. diseases (p = 0.17). This is not surprising, as smoking is known to be a risk factor for a long time and there are many initiatives to reduce smoking and to prevent younger people from starting to consume tobacco.

On the other hand, air pollution and the median age of a country both do not seem to impact the risk of developing a resp. disease. Particular the air pollution was under suspicion to be a high risk factor, but the data does not show a significant correlation.

# 5 Conclusion and Recommendations

As the data shows, obesity leads to a high risk to develop a resp. disease. Countries with a higher median age (confounder / mediator) are often found in Europe. Physiotherapy is an important pillar in the health system not only to treat resp. diseases directly (awareness of breathing movements, postures that make breathing easier, stretching positions or mucolytic exercises), but also to prevent or cure obesity through physical exercise and diet education. Rauch, E. (2015) posits that accumulations of visceral adipose tissue influence the respiratory mechanics.

Considering that the factors air pollution, smoking and median age seemingly exert an insubstantial impact on the risk of developing a resp. disease, and given that these can not be directly influenced by physiotherapeutic measures, it is reasonable to omit them from this conclusion.

Why do factors like *air pollution* and *smoking* not seem to have a strong impact, as obesity have? Why does obesity have such a strong impact on the risk of developing a respiratory disease. Can the research of F. X. Mayr, Rauch, E. (2015) referred to his foundational work, provide valid explanations for this phenomenon? Will there be a rise in obesity and also in the prevalence of resp. diseases in Counties which are still developing, like on the African continent?

# 6 Reflective practice

**Description:** My experience of working on the project was showing different phases of the working process. In the beginning I was very motivated to do the coding. I used a R script file to try out different data wrangling, analysis and visualisation methods for the presentation and report and I was very curious about the data and how I can enrich the OECD data with data from other sources. This changed over time with more data and aspects needed to be covered. The participation in the discussion boards gave me further insights, but due to lack of time I was not able to participate in all of the discussions.

**Feelings:** At some point with rising complexity, finding the red thread, I was a bit overwhelmed. I found it hard to express my thoughts in a structured way, taking also into account that English is not my native language.

**Evaluation:** I really enjoyed to work with R code, RStudio and plotting libraries such as *ggplot2*. Also to use ChatGPT for the first time in R for normalizing purposes was a rewarding experience. The finalization of the project was a bit draining, as it already reached a higher complexity.

**Analysis:** My time management was good, as after taking my long leave in my expiring job, I kept the working hours I had before to finish the assessment in good time. With my previous coding experience I found my self in a good learning position to reach a higher level of understanding in R and data science at all.

**Conclusion:** Keeping a journal of my steps and thoughts would have helped me to reach a better understanding of the data and the research question.

**Action plan:** I need to keep the overview of my research question and the data which is used. Also I need to be more structured in my approach and follow a red thread.

# References

Daniels, J. P., 2023. [Colombia introduces junk food tax](https://doi.org/10.1016/S0140-6736(23)02628-4). THE LANCET, (Accessed: 24.11.2023).

Hawkings, A., 2023. [China supplies data to WHO about clusters of respiratory illness](https://www.theguardian.com/world/2023/nov/23/china-respiratory-illness-spike-children-who). The Guardian, (Accessed: 27.11.2023).

Jura, M. & Kozak, L. P., 2016. [Obesity and related consequences to ageing](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5005878/). National Library of Medicine, (Accessed: 24.11.2023).

Rauch, E., 2015. [Lehrbuch der diagnostik und therapie nach f. X. Mayr, pp. 73](https://www.narayana-verlag.de/homoeopathie/pdf/Lehrbuch-der-Diagnostik-und-Therapie-nach-F-X-Mayr-Erich-Rauch.19559_4Heilprinzipien_Nach_Mayr.pdf). Karl F. Haug Verlag, Stuttgart.

World Health Organization, 2023. [Obesity](https://www.who.int/health-topics/obesity). National Library of Medicine, (Accessed: 27.11.2023).

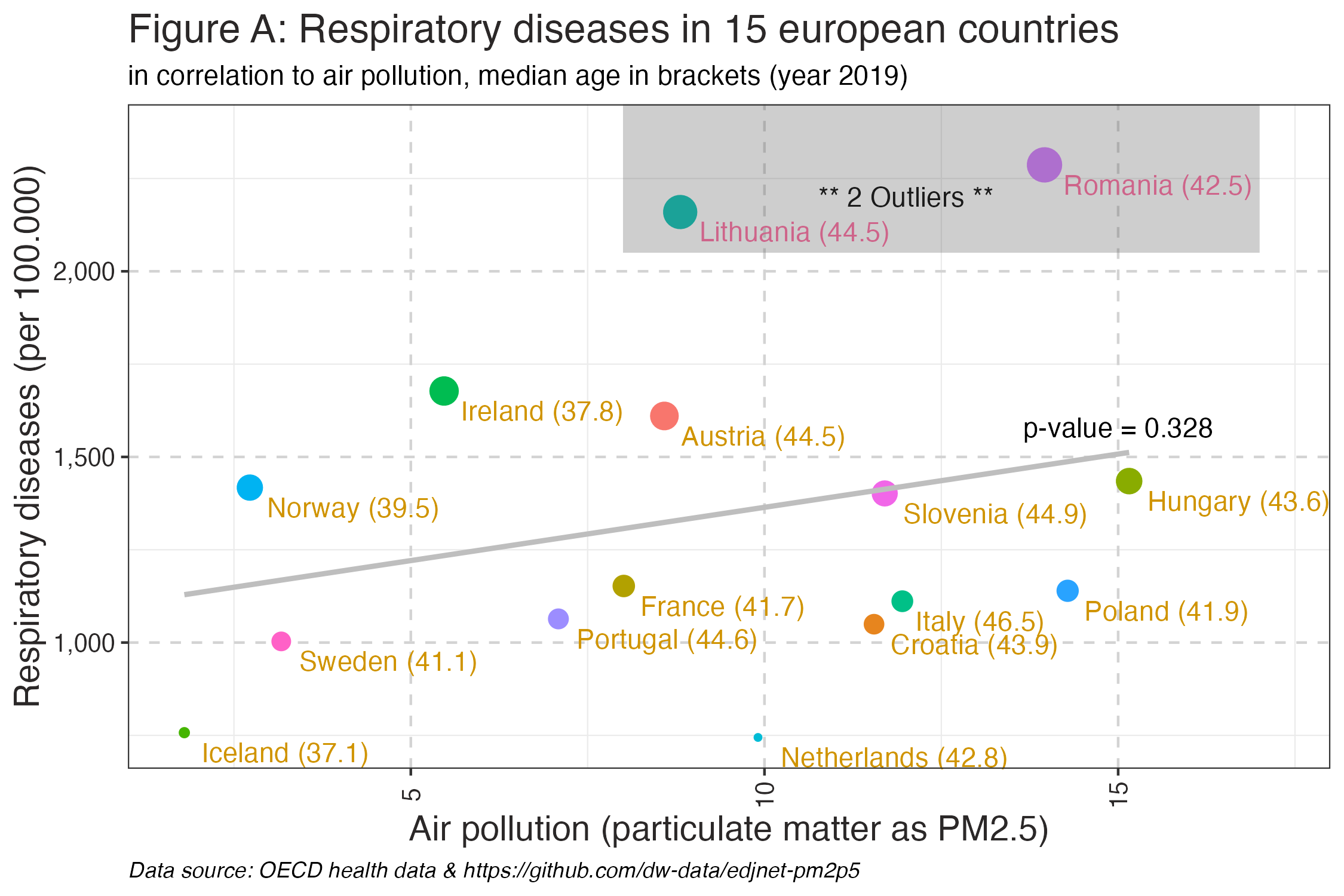
**Soft copies:**

[Daniels, J. P., 2023](../meta_data/PIIS0140673623026284.pdf)

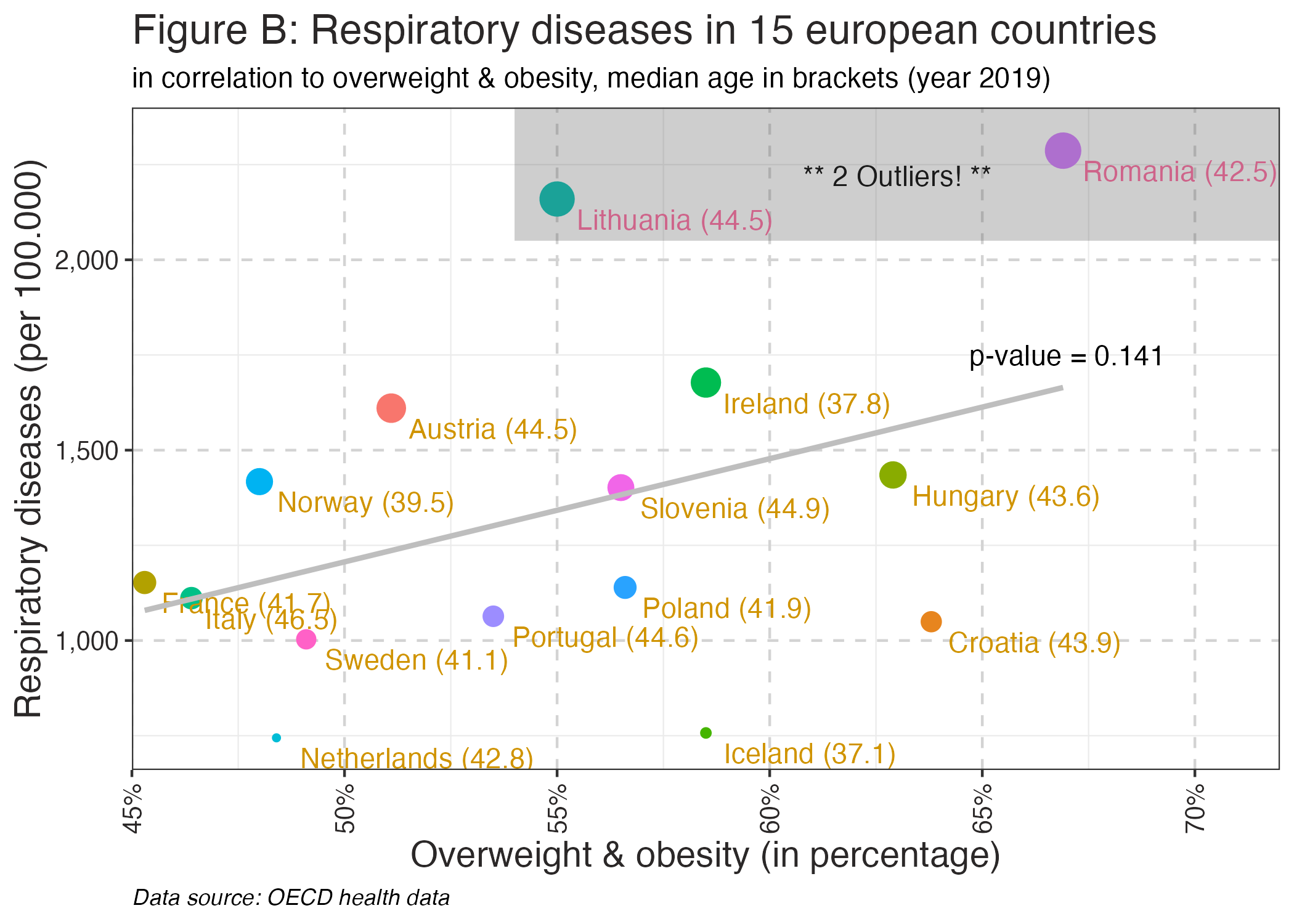
[Jura, M. & Kozak, L. P., 2016](../meta_data/11357_2016_Article_9884.pdf)

[WHO, 2023](../meta_data/11357_2016_Article_9884.pdf)

**Figures from the management presentation:**



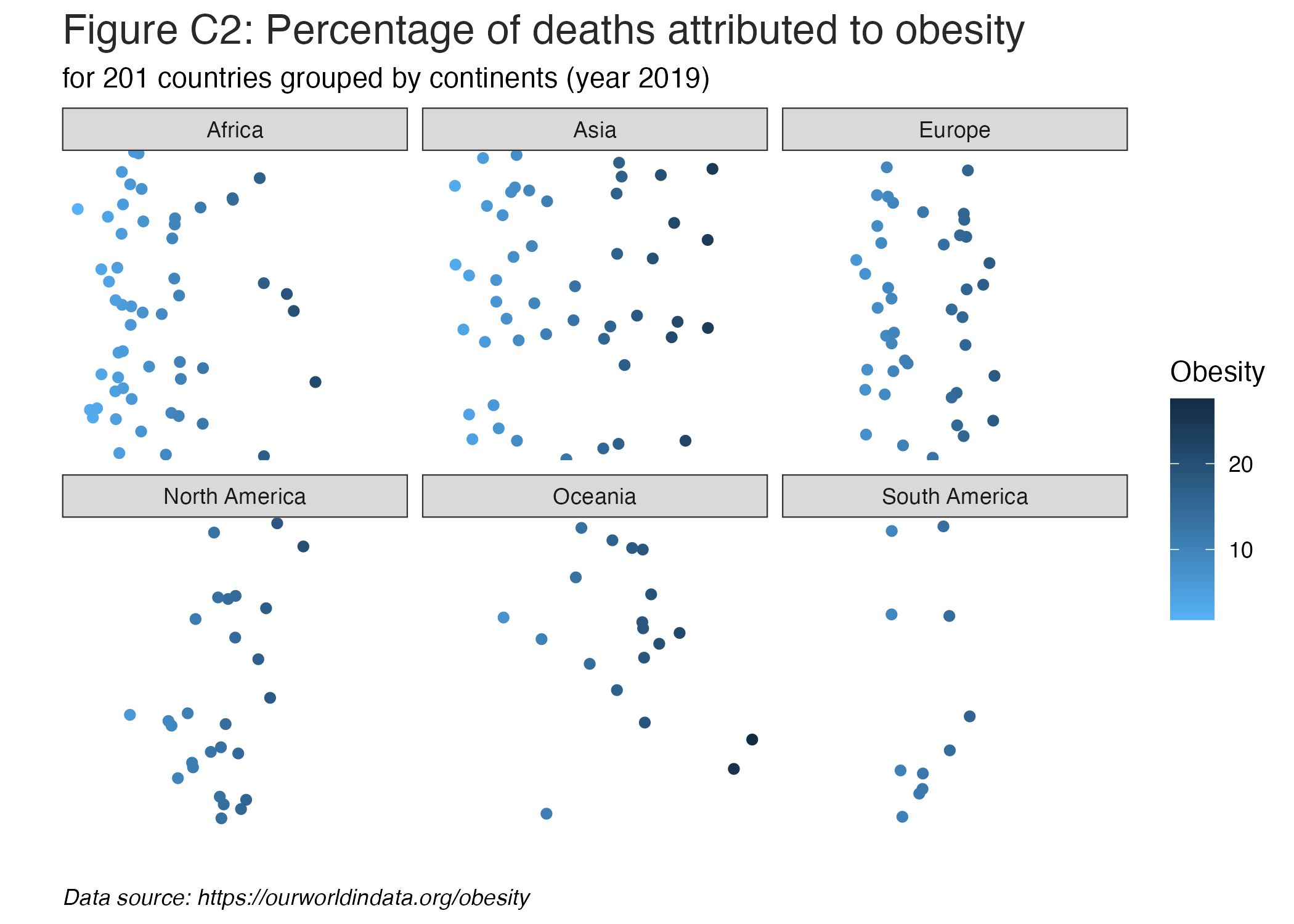
PM2.5



Obesity



Obesity continental



Deaths continental

**Datathon Assessment Guidelines**

[Datathon Assessment Guidelines](../meta_data/Datathon%20Assessment%20Guidelines%20HEIN11037%20AY2324.pdf)

**Session Info**

Debug variable and output of used libraries.

# Knitted with RStudio (2023.09.1+494 "Desert Sunflower" for macOS)  
# For word counting the code and it's output can be neglected,   
# setting the `debug\_var` variable to `TRUE`.  
sessionInfo()

## R version 4.1.2 (2021-11-01)  
## Platform: x86\_64-apple-darwin17.0 (64-bit)  
## Running under: macOS Big Sur 10.16  
##   
## Matrix products: default  
## BLAS: /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRblas.0.dylib  
## LAPACK: /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRlapack.dylib  
##   
## locale:  
## [1] en\_US.UTF-8/en\_US.UTF-8/en\_US.UTF-8/C/en\_US.UTF-8/en\_US.UTF-8  
##   
## attached base packages:  
## [1] stats graphics grDevices utils datasets methods base   
##   
## other attached packages:  
## [1] here\_1.0.1 janitor\_2.2.0 webshot2\_0.1.1 gt\_0.10.0   
## [5] gridExtra\_2.3 openai\_0.4.1 lubridate\_1.9.3 forcats\_1.0.0   
## [9] stringr\_1.5.0 dplyr\_1.1.3 purrr\_1.0.2 readr\_2.1.4   
## [13] tidyr\_1.3.0 tibble\_3.2.1 ggplot2\_3.4.3 tidyverse\_2.0.0  
##   
## loaded via a namespace (and not attached):  
## [1] Rcpp\_1.0.8 lattice\_0.20-45 png\_0.1-7 ps\_1.6.0   
## [5] rprojroot\_2.0.2 digest\_0.6.29 utf8\_1.2.2 R6\_2.5.1   
## [9] repr\_1.1.6 evaluate\_0.22 pillar\_1.9.0 rlang\_1.1.1   
## [13] rstudioapi\_0.15.0 Matrix\_1.3-4 rmarkdown\_2.25 textshaping\_0.3.6  
## [17] labeling\_0.4.2 splines\_4.1.2 bit\_4.0.4 munsell\_0.5.0   
## [21] compiler\_4.1.2 xfun\_0.40 systemfonts\_1.0.4 pkgconfig\_2.0.3   
## [25] base64enc\_0.1-3 mgcv\_1.8-38 htmltools\_0.5.7 websocket\_1.4.1   
## [29] tidyselect\_1.2.0 fansi\_1.0.2 crayon\_1.4.2 tzdb\_0.2.0   
## [33] withr\_2.5.0 later\_1.3.0 commonmark\_1.9.0 grid\_4.1.2   
## [37] nlme\_3.1-153 jsonlite\_1.8.7 gtable\_0.3.0 lifecycle\_1.0.3   
## [41] pacman\_0.5.1 magrittr\_2.0.3 scales\_1.2.1 cli\_3.6.1   
## [45] stringi\_1.7.6 vroom\_1.6.3 farver\_2.1.0 fs\_1.6.3   
## [49] promises\_1.2.0.1 snakecase\_0.11.1 skimr\_2.1.5 xml2\_1.3.3   
## [53] ragg\_1.2.5 generics\_0.1.2 vctrs\_0.6.3 tools\_4.1.2   
## [57] bit64\_4.0.5 glue\_1.6.2 markdown\_1.11 hms\_1.1.3   
## [61] processx\_3.8.2 parallel\_4.1.2 fastmap\_1.1.0 yaml\_2.2.2   
## [65] timechange\_0.2.0 chromote\_0.1.2 colorspace\_2.0-2 knitr\_1.44   
## [69] sass\_0.4.7