Assessing whether there was an association between anxiety levels and threat estimation during the Covid-19 pandemic across four European countries

2024-11-26

Contents


```
knitr::opts_chunk$set(echo = TRUE)

library(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   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- 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 errors
```

```
library(here)
```

```
## here() starts at C:/Users/kiren/OneDrive/Documents/psy6422_assessment/covid
```

```
library(dplyr)
library(ggplot2)
library(extrafont)
```

```
## Registering fonts with R
```

Project Description

This project is an assessed piece of work as part of the MSc Psychological Research Methods with Data Science course at the University of Sheffield. The assessment requires students to produce a visualisation of an open source dataset using R. In line with the movement towards increasing the reproducibility of scientific findings, the processes employed for the of data visualisation, along with the associated code for the exploratory data analysis are reported.

Background

The entire world was impacted by the effects of the Covid-19 (SARS-CoV-2) virus as it resulted in a pandemic. Following the rapid spread of the disease, many countries took action by implementing lockdown measures and introducing social distancing and hygiene protocols (Miyah et al., 2022).

The data collected during this period is plentiful and has provided new insights into human behaviour when faced with the uncertainty of a novel and harmful disease. Using an online survey, Abadi et al., (2023) obtained data from 2031 participants from the UK, Spain, Germany and Netherlands. The authors report that they chose to include these specific countries on the basis that each differed significantly on the number of Covid-19 related deaths at the time of the survey completion, where Spain had the most number of cases, followed by the UK,the Netherlands and Germany. Moreover, the implementation of lockdown by these countries' respective governments and the socio-political attitudes within their differed. Participants who engaged with the survey completed measures on several variables of interest, such as socio-political attitudes, conspiracy mentality and perceptions of threat, to name a few.

Data Origins

The original article was retrieved from the Journal of Open Psychology Data. Click here to view the full article.

The data and codebook was retrieved from the authors'*figshare* repository which is available here.

The authors of the paper have indicated that their dataset is published under the CC BY-SA license. This allows the re-use of the data, provided that the original authors are acknowledged.

The URL is this visualisation project is here

The repository for these pages is: https://github.com/IJames8/covid.git

Import the data

```
raw_data <- read.csv("data/raw_data.csv")

head(raw_data)
```

```
##   RespondentID Country A3.1 A3.2 A3.3 A3.4 A3.5 A3.6 A3.7 A3.8 A3.9 A3.10 A5.1
## 1            1       1    2    3   52   52    4    1    5   12    1     4    7
## 2            2       1    2    5   52   52    3    1    8    2    5     2   10
## 3            3       1    1    4   94   94    4    1    5    2    1     4   10
## 4            4       1    1    2  221  221    1    1   10    3    4     5    8
## 5            5       1    1    3   52   52    3    1    8    2    0     6    7
## 6            6       1    1    1   52   52    1    4    9    3    4     2    8
##   A5.2 A5.3. A5.4 A5.5 A6.1 A7.1 A7.2 A9.1 A9.2 A9.3. A9.4 A9.5 A9.6 A9.7 A9.8.
## 1    3     1    3    8    7    2    2    2    1     1    3    3    2    5     1
```

```
## 2   10     1    8    1    5    2    2    4    3     4    6    1    1    1     6
## 3   10     1   10    7    6    2    2    5    4     3    7    4    7    7     7
## 4    8     9    7    7    8    4    4    7    6     5    6    7    7    6     6
## 5    7     2    6    3    7    1    1    1    1     5    3    5    1    1     6
## 6   10     9    9    9    4    3    4    6    6     7    7    6    5    7     5
##   A11.1 A11.2 A11.3 A11.4 A12.1 A12.2 A12.3 A12.4 A13.1 A13.2 A13.3 A13.4 A14.1
## 1     2     2     2     2     1     1     1     2     2     2     2     2     2
## 2     1     1     1     1     1     1     1     1     1     1     1     1     1
## 3     1     1     1     1     1     1     1     1     1     1     1     2     1
## 4     2     1     1     2     2     1     1     2     1     1     2     2     2
## 5     1     2     1     1     1     1     1     1     1     2     1     2     2
## 6     1     2     1     1     1     2     1     2     2     1     1     2     1
##   A14.2 A14.3 A14.4 A16.1 A16.2 A16.3. A16.4. A16.5 A16.6 A17.1 A17.2 A17.3
## 1     1     2     2     5     2      9      9     7     7     5     2     2
## 2     1     2     1     4     8      2      2    10     9     9     9     8
## 3     1     1     2     7     7      7      3    10    10     8    10     6
## 4     2     2     2     7     8      2      2     8     8     8     8     9
## 5     2     2     2     6     2      4      4     7     5     4     3     8
## 6     2     1     2     9    10      4      7     5     7     9     6     9
##   A17.4 A17.5 A17.6 A17.7 A17.8 A17.9 A17.10. A17.11 A17.12 A18.1 A20.1 A20.2
## 1     5     3     2     6    10     8       9      7      7     5     7     7
## 2     9     6     5     1     1     1       2      9      8     5     7     7
## 3     8    10    10    10     7    10       4     10      9     5     7     5
## 4     8     7     9     7     8     7       2      8      8     5     7     7
## 5     6     4     3     7     2     2       3      5      3     5     6     6
## 6    10     8    10     8    10     9       4      6      9     5     6     7
##   A20.3 A21.1 A21.2. A22.1 A22.2 A23.1 A23.2 A23.3 A24.1 A24.2 A24.3 A24.4
## 1     5     4      2     3     7     6     6     6     5     5     5     4
## 2     7     7      7     7     7     7     7     7     7     7     7     7
## 3     7     6      3     5     6     7     7     5     7     7     7     4
## 4     6     7      6     6     7     7     6     7     6     7     6     7
## 5     5     5      5     6     5     5     5     6     7     7     5     5
## 6     7     6      7     7     7     7     7     5     7     7     5     6
##   A24.5 A24.6 A25.1 A25.2 A27.1 A27.2. A27.3 A27.4 A27.5 A27.6. A27.7. A27.8
## 1     4     1     1     1     4      5     2     4     2      5      5     7
## 2     7     7     8     8     2      6     2     2     1      7      1     7
## 3     2     7    10     6     7      7     7     6     5      5      7     7
## 4     6     5     7     8     7      6     7     7     6      6      6     5
## 5     6     5     3     2     6      6     7     6     6      6      6     4
## 6     7     5    10     8     5      7     7     7     5      7      6     6
##   A27.9 A27.10 A27.11 A29.1 A29.2 A29.3 A29.4 A29.5 A29.6 A29.7 A29.8 A29.9
## 1     7      5      5     7     7     7     7     7     7     7     7     7
## 2     7      2      6     7     7     7     7     7     7     7     7     7
## 3     5      7      5     7     7     7     7     7     7     7     7     7
## 4     7      7      6     5     7     6     6     7     7     6     7     6
## 5     7      6      5     5     5     6     6     6     6     6     6     7
## 6     6      6      7     7     6     7     5     7     5     6     6     7
##   A31.1 A31.2 A31.3 A31.4 A31.5 A31.6 A31.7 A31.8 A31.9 A32.1 A32.2 A34.1 A34.2
## 1     7     7     7     1     1     7     4     1     1     1     1     1     2
## 2     7     7     7     7     1     7     7     1     1     1    10     6     6
## 3     7     7     7     6     7     7     7     7     7    10     5     2     6
## 4     6     7     7     6     5     7     6     6     4     8     9     7     7
## 5     7     6     7     1     7     6     7     7     6     3     8     2     3
## 6     7     7     5     4     4     7     6     5     6     9    10     7     5
```

```
##   A34.3 A34.4 A36.1 A36.2 A36.3 A36.4. A36.5 A36.6 A38.1. A38.2. A38.3 A38.4
## 1     2     4     5     7     7      1     6     6      4      1     7     1
## 2     4     6     1     1     1      5     4     1      1      7     7     1
## 3     4     1     3     2     6      6     1     1      5      2     6     2
## 4     7     6     6     5     7      7     7     6      7      6     6     7
## 5     2     2     6     5     6      2     5     6      5      3     4     1
## 6     7     7     7     4     5      7     6     6      6      6     7     5
##   A38.5 A40.1 A40.2 A40.3 A40.4 A40.5 A40.6 A40.7 A40.8 A40.9 A40.10 A40.11
## 1     2     4     7     5     7     5     7     7     6     1      3      7
## 2     1     7     7     6     7     7     7     7     7     7      1      7
## 3     6     7     7     7     7     7     7     7     7     7      7      7
## 4     7     5     6     6     4     5     7     6     7     7      6      7
## 5     5     6     6     6     7     5     5     2     6     5      5      7
## 6     7     6     6     5     7     5     7     5     7     6      6      7
```

```
# check that the data has imported properly. The number of observations and variables appears to be cor
```

Research Question

In reviewing the reuse potential of the data as suggested by the Abadi et al., (2013), this data visualisation project aims to investigate whether anxiety around the coronavirus, as indicated by the scores on the anxiety measure, is associated with an overestimation of various threats, such as those related to coronavirus, climate, symbolic/material or safety, as indicated by the scores on the threat estimation measure for each participant.

Data Preparation

For this particular visualisation, the codebook indicates that the variables of interest are items 5.1 -5.5. These relate to anxiety about coronavirus. It is important to note that item 5.3. is reverse coded. Variables 16.1 - 16.3 (threat estimation around cornoavirus) and 17.1 - 17.12 (threat estimation around climate, symbolic, material and safety) are also needed as these relate to the scores around threat estimation. Within this category, items 16.3, 16.4 and 17.10 are reverse coded.

The remaining items will be removed from the dataframe, leaving only information about the relevant variables, along with annonymised participant ID and country.

Wrangle the data

```
# remove the unwanted variables.

df1 <- raw_data %>%
  select(-A3.1 : -A3.10, -A6.1 : -A14.4, -A18.1:-A40.11)
```

```
# Items 5.3, 16.3, 16.4 and 17.10, are reverse coded. This will need to be adjusted in the dataframe.
# Item 5.3. is scored on a scale from 1 - 10. High scores indicate, high levels of anxiety.
# Item 16.4 and 16.3 are scored on a scale from 1 - 10. High scores indicate, high levels of threat est
# Item 17.10. is scored on a scale from 1 - 10. High scores indicate, high levels of threat estimation.

#Create function to reverse score the items indicated in the codebook. Utilising the " N-plus-one-minus

reversed_scores <- function(df1, columns, max_score = 11) {
```

```
  df1 %>%
    mutate(across(all_of(columns), ~ max_score - ., .names = "{.col}_reversed"))
  }

# Apply the function to relevant items and assign this to an updated dataframe.
df2 <- reversed_scores(df1, c("A5.3.", "A16.3.", "A16.4.", "A17.10."))


# View the dataframe to check that new columns have been added to df2 for the reversed scored items. Ch

head(df2)
```

```
##   RespondentID Country A5.1 A5.2 A5.3. A5.4 A5.5 A16.1 A16.2 A16.3. A16.4.
## 1            1       1    7    3     1    3    8     5     2      9      9
## 2            2       1   10   10     1    8    1     4     8      2      2
## 3            3       1   10   10     1   10    7     7     7      7      3
## 4            4       1    8    8     9    7    7     7     8      2      2
## 5            5       1    7    7     2    6    3     6     2      4      4
## 6            6       1    8   10     9    9    9     9    10      4      7
##   A16.5 A16.6 A17.1 A17.2 A17.3 A17.4 A17.5 A17.6 A17.7 A17.8 A17.9 A17.10.
## 1     7     7     5     2     2     5     3     2     6    10     8       9
## 2    10     9     9     9     8     9     6     5     1     1     1       2
## 3    10    10     8    10     6     8    10    10    10     7    10       4
## 4     8     8     8     8     9     8     7     9     7     8     7       2
## 5     7     5     4     3     8     6     4     3     7     2     2       3
## 6     5     7     9     6     9    10     8    10     8    10     9       4
##   A17.11 A17.12 A5.3._reversed A16.3._reversed A16.4._reversed A17.10._reversed
## 1      7      7             10               2               2                2
## 2      9      8             10               9               9                9
## 3     10      9             10               4               8                7
## 4      8      8              2               9               9                9
## 5      5      3              9               7               7                8
## 6      6      9              2               7               4                7
```

```
# Remove old scores and replace with reversed scores.

df2$A5.3. <- df2$A5.3._reversed
df2$A16.3. <- df2$A16.3._reversed
df2$A16.4. <- df2$A16.4._reversed
df2$A17.10. <- df2$A17.10._reversed


# Delete the reversed score columns at the end so it is less confusing and assign this to new dataframe

df3 <- select(df2, -c(A5.3._reversed, A16.3._reversed, A16.4._reversed, A17.10._reversed))

# Check that this has transferred correctly.

head(df3)
```

```
##   RespondentID Country A5.1 A5.2 A5.3. A5.4 A5.5 A16.1 A16.2 A16.3. A16.4.
## 1            1       1    7    3    10    3    8     5     2      2      2
## 2            2       1   10   10    10    8    1     4     8      9      9
```

```
## 3            3       1   10   10    10   10    7     7     7      4      8
## 4            4       1    8    8     2    7    7     7     8      9      9
## 5            5       1    7    7     9    6    3     6     2      7      7
## 6            6       1    8   10     2    9    9     9    10      7      4
##   A16.5 A16.6 A17.1 A17.2 A17.3 A17.4 A17.5 A17.6 A17.7 A17.8 A17.9 A17.10.
## 1     7     7     5     2     2     5     3     2     6    10     8       2
## 2    10     9     9     9     8     9     6     5     1     1     1       9
## 3    10    10     8    10     6     8    10    10    10     7    10       7
## 4     8     8     8     8     9     8     7     9     7     8     7       9
## 5     7     5     4     3     8     6     4     3     7     2     2       8
## 6     5     7     9     6     9    10     8    10     8    10     9       7
##   A17.11 A17.12
## 1      7      7
## 2      9      8
## 3     10      9
## 4      8      8
## 5      5      3
## 6      6      9
```

```
# Calculate mean score for anxiety around Covid-19 and the mean for threat variables for each participa

anxiety_mean <- rowMeans(df3[, c("A5.1","A5.2", "A5.3.", "A5.4", "A5.5")]) #, na.rm = TRUE)

c19_threat_mean <- rowMeans(df3[, c("A16.1", "A16.2", "A16.3.", "A16.4.", "A16.5", "A16.6")]) #, na.rm

safety_threat_mean <-  rowMeans(df3[, c("A17.1", "A17.2", "A17.3", "A17.4", "A17.5", "A17.6", "A17.7",


# Add mean score for anxiety to dataframe

df3$anxiety_mean <- anxiety_mean
df3$c19_threat_mean <- c19_threat_mean
df3$safety_threat_mean <- safety_threat_mean


# Remove the individual questionnaire scores now that the mean has been calculated for the variables of

df4 <- df3[, -c(3:25)]

# Recode the country column

df5 <- df4 %>%
  mutate(Country = recode(Country,
                          '1' = "Germany",
                          '2' = "Spain",
                          '3' = "Netherlands",
                          '4' = "UK"))

# Check that the new dataframe has all the relevant information needed.


head(df5)
```

```
##   RespondentID Country anxiety_mean c19_threat_mean safety_threat_mean
```

```
## 1            1 Germany          6.2        4.166667           4.916667
## 2            2 Germany          7.8        8.166667           6.250000
## 3            3 Germany          9.4        7.666667           8.750000
## 4            4 Germany          6.4        8.166667           8.000000
## 5            5 Germany          6.4        5.666667           4.583333
## 6            6 Germany          7.6        7.000000           8.416667
```

```
#As df5 is in wide format, it will need to be converted to long format.

long_data <- df5%>%
  pivot_longer(cols = c("anxiety_mean", "c19_threat_mean", "safety_threat_mean"),
               names_to = "Variable",
               values_to = "Score")
head(long_data)
```

```
## # A tibble: 6 x 4
##   RespondentID Country Variable           Score
##          <int> <chr>   <chr>              <dbl>
## 1            1 Germany anxiety_mean        6.2
## 2            1 Germany c19_threat_mean     4.17
## 3            1 Germany safety_threat_mean  4.92
## 4            2 Germany anxiety_mean        7.8
## 5            2 Germany c19_threat_mean     8.17
## 6            2 Germany safety_threat_mean  6.25
```

```
# Aggregate variables to produce a summaried overall mean score for each variable according to the coun

data_summary <-  long_data %>%
  group_by(Country, Variable) %>%
  summarise( mean_score= mean(Score), .groups = "drop")


# Creating factors as the countries are categorical data and to prevent issues when creating the plot l

data_summary <- data_summary %>%
  mutate(Country = factor (Country, levels = c("Germany", "Netherlands", "Spain", "UK")))

data_summary
```

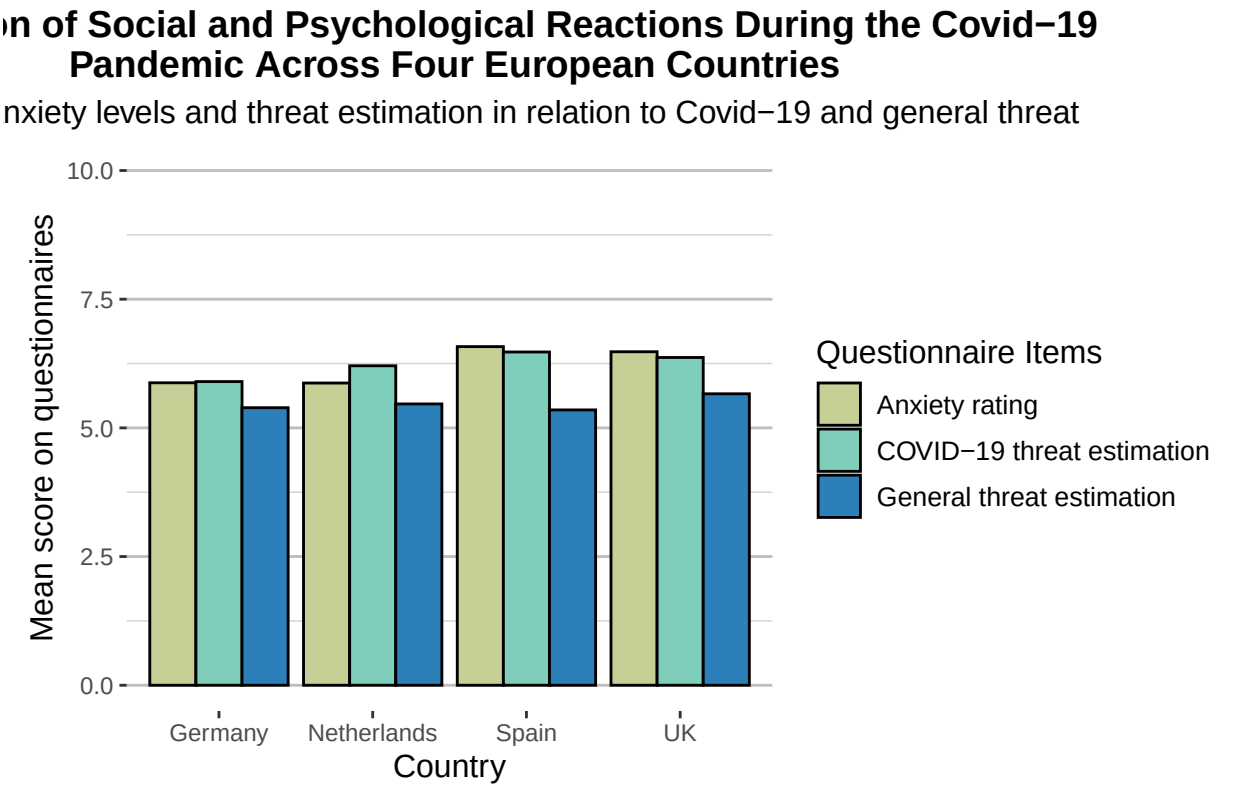
```
## # A tibble: 12 x 3
##    Country     Variable           mean_score
##    <fct>       <chr>                   <dbl>
##  1 Germany     anxiety_mean             5.88
##  2 Germany     c19_threat_mean          5.90
##  3 Germany     safety_threat_mean       5.39
##  4 Netherlands anxiety_mean             5.87
##  5 Netherlands c19_threat_mean          6.21
##  6 Netherlands safety_threat_mean       5.47
##  7 Spain       anxiety_mean             6.58
##  8 Spain       c19_threat_mean          6.47
##  9 Spain       safety_threat_mean       5.35
## 10 UK          anxiety_mean             6.48
## 11 UK          c19_threat_mean          6.37
## 12 UK          safety_threat_mean       5.66
```

Visualisation

The best way to visualise this data may be through a bar chart with the countries as the categorical variable and the questionnaire scores as continuous. With this, it is possible to check if the predictions made about the data,as stated previosly, are met.

```
# Create a bar chart of anxiety and threat estimation scores for each country.

p1 <- ggplot(data_summary, aes(x = Country, y = mean_score, fill = Variable)) +
  geom_bar(stat = "identity", position = "dodge", colour = "black") +

  # Set Y-axis scale from 0 -10.

  ylim(0, 10) +

  # Add labels

  labs(
    title = "Visualisation of Social and Psychological Reactions During the Covid-19\n Pandemic Across
    subtitle = "Comparing anxiety levels and threat estimation in relation to Covid-19 and general threa
    x = "Country",
    y = "Mean score on questionnaires",
    fill = "Questionnaire Items",
    caption = "Source: A Dataset of Social-Psychological and Emotional Reactions During the COVID-19 Pa
  ) +

  # Customise by adding complementary colours to aid better viewing experience.

  scale_fill_manual(
    values = c("anxiety_mean" = "#c6cf95", "c19_threat_mean" = "#7fcdbb", "safety_threat_mean" = "#2c7f
    labels = c("Anxiety rating", "COVID-19 threat estimation", "General threat estimation")
  ) +


  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face = "bold"),
    plot.subtitle = element_text(hjust = 0.5, size = 12),
    plot.caption = element_text(hjust = 0, face = "italic"),
    axis.title = element_text(size = 12),
    legend.title = element_text(size = 12),
    legend.text = element_text(size = 10),
    panel.background = element_rect(fill = "#ffffff", color = NA),

    # Remove the vertical grid lines
    panel.grid.major.x = element_blank(),
    panel.grid.minor.x = element_blank(),

    # Keep the horizontal grid lines
    panel.grid.major.y = element_line(color = "grey", linewidth = 0.5),

    # Adding a secondary line makes it easier to estimate the values on the y-axis.
    panel.grid.minor.y = element_line(color = "lightgrey", linewidth = 0.25),

     # Adjust margins
```

```
    plot.margin = margin(10, 10, 10, 10))
p1
```



Source: A Dataset of Social−Psychological and Emotional Reactions During the COVID−19 Pandemic Acr

```
# plot that with the weekly count cases of c-19 for each country.
# the mean anxiety and threat score for each country as a whole needs to be calculated.
```

```
#ggsave(here("visualisations/p1.png"))
```

As predicted, in looking at the average anxiety scores, Spain had marginally higher rating followed by the UK then Germany and Netherlands. In terms of average Covid-19 threat detection score, Spain had the highest scores, followed by the UK, Netherlands and Germany. In terms of general threat estimation, UK scored the highest and Spain the lowest.

Import data (part 2)

It will be helpful to see the relationship between these findings to that of the total number of Covid- 19 related deaths for each country. According to Abadi et al., (2023) the data collection for their study took place in April 2020. The European Centre for Disease Prevention and Control (ECDC) provides data on the weekly number of Covid-19 related cases and deaths worldwide.The data for April 2020 is also reported and is available to download from their official website: https://www.ecdc.europa.eu/en/publications-data/download-historical-data-20-june-2022-weekly-number-new-reported-covid-19-cases

```
ecdc_raw <- read.csv("https://opendata.ecdc.europa.eu/covid19/casedistribution/csv", na.strings = "", fi

head(ecdc_raw)
```

```
##      dateRep day month year cases deaths countriesAndTerritories geoId
## 1 14/12/2020  14    12 2020   746      6             Afghanistan    AF
## 2 13/12/2020  13    12 2020   298      9             Afghanistan    AF
## 3 12/12/2020  12    12 2020   113     11             Afghanistan    AF
## 4 11/12/2020  11    12 2020    63     10             Afghanistan    AF
## 5 10/12/2020  10    12 2020   202     16             Afghanistan    AF
## 6 09/12/2020   9    12 2020   135     13             Afghanistan    AF
##   countryterritoryCode popData2019 continentExp
## 1                  AFG    38041757         Asia
## 2                  AFG    38041757         Asia
## 3                  AFG    38041757         Asia
## 4                  AFG    38041757         Asia
## 5                  AFG    38041757         Asia
## 6                  AFG    38041757         Asia
##   Cumulative_number_for_14_days_of_COVID.19_cases_per_100000
## 1                                                  9.013779
## 2                                                  7.052776
## 3                                                  6.868768
## 4                                                  7.134266
## 5                                                  6.968658
## 6                                                  6.963401
```

Wrangle the dataset

This dataset has information on all the countries in the world. For the purpose of this visualisation, only the data for Germany, Netherlands, Spain and UK are needed.

```
# Remove all the other countries

ecdc2 <- ecdc_raw %>%
  filter(countriesAndTerritories %in% c("Germany", "Netherlands", "Spain", "United_Kingdom"))

# Remove data for all other months, except April.

ecdc2 <- ecdc2 %>%
  filter(month == 4)

# Remove the other wanted variables also.

ecdc3 <- ecdc2 %>%
  select(-day, -month, -year, -cases, -geoId, -countryterritoryCode, -popData2019, -continentExp)

# Calucate the total deaths for April for each country.

total_deaths <- ecdc3 %>%
  group_by(countriesAndTerritories) %>%
  summarise(total_deaths = sum(deaths, na.rm = TRUE))
```

```r
# Add total deaths column to dataframe.

ecdc3 <- ecdc3 %>%
  left_join(total_deaths, by = "countriesAndTerritories")


# Table to display the total deaths calculated for each country in April 2020 when the survey was condu

ecdc_summary <- ecdc3 %>%
  group_by(countriesAndTerritories, total_deaths) %>%
  summarise(count = n(), .groups = "drop")%>%
  select(-count)

ecdc_summary
```

```
## # A tibble: 4 x 2
##   countriesAndTerritories total_deaths
##   <chr>                          <int>
## 1 Germany                         5705
## 2 Netherlands                     3847
## 3 Spain                          17203
## 4 United_Kingdom                 23999
```

```r
# Ensure that the names match for the countires

ecdc_summary <- ecdc_summary %>%
  mutate(countriesAndTerritories = if_else(countriesAndTerritories == "United_Kingdom", "UK", countries

ecdc_summary
```

```
## # A tibble: 4 x 2
##   countriesAndTerritories total_deaths
##   <chr>                          <int>
## 1 Germany                         5705
## 2 Netherlands                     3847
## 3 Spain                          17203
## 4 UK                             23999
```

Final Visualisation

The final visualisation will comprise of the previous graph overlaid with the data around the of number of deaths for each of the four countries.

```r
  # Adding data around total number of deaths reported to the previous graph.

p2 <- ggplot(data_summary, aes(x = Country, y = mean_score, fill = Variable)) +
  # Bar chart layer
  geom_bar(stat = "identity", position = "dodge", colour = "black") +

  # Scatter plot for EDCD data and define the scale for the y-axis.
  geom_point(
```

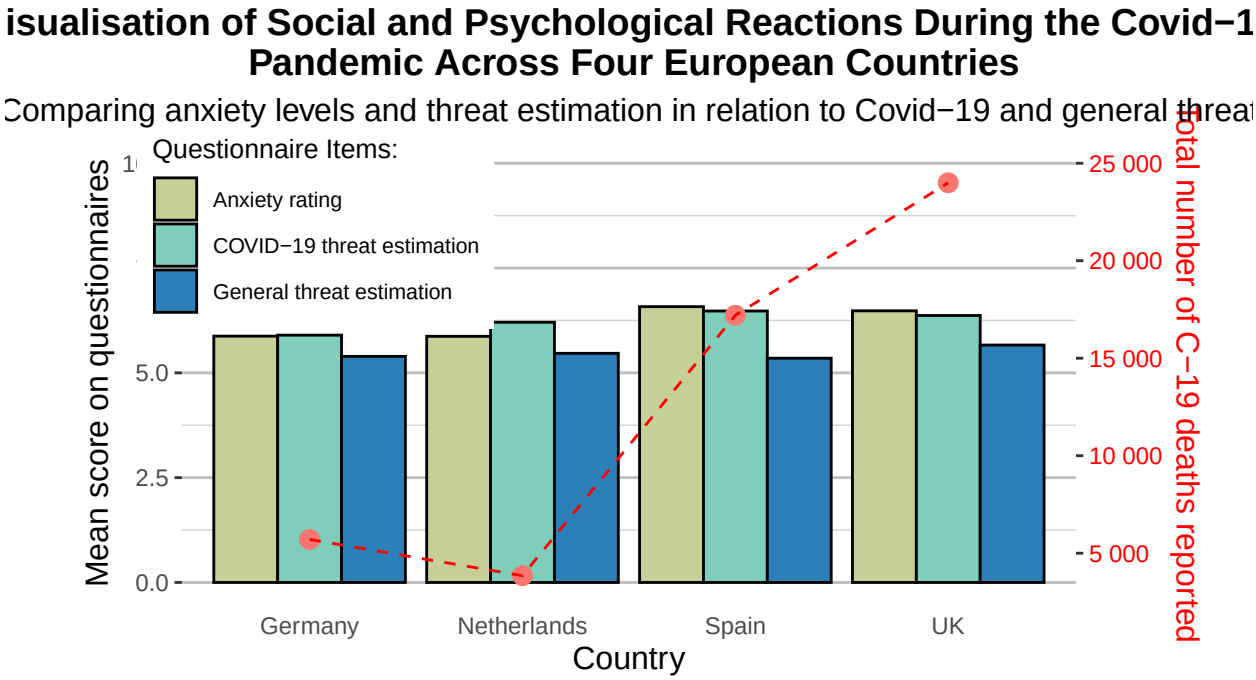
```
  data = ecdc_summary,
  aes(x = countriesAndTerritories, y = (total_deaths - 3500) * (10 / 21500), color = "Total Deaths", 
  size = 3,
  inherit.aes = FALSE,
  show.legend = FALSE) +

# Add dashed red line through the points
geom_line(
  data = ecdc_summary,
  aes(x = as.numeric(factor(countriesAndTerritories)), y = (total_deaths - 3500) * (10 / 21500)),
  color = "red",
  linetype = "dashed",

# Remove the legend as the axis for the secondary axis will be red
  inherit.aes = FALSE,
  show.legend = FALSE) +

# Specify the primary and secondary y-axis
  scale_y_continuous(
  name = "Mean score on questionnaires",
  limits = c(0, 10),

# Add second axis and specify the scale.
  sec.axis = sec_axis(~ . * 21500 / 10 + 3500,
    name = "Total number of C-19 deaths reported",
    labels = scales::label_number()))+

# Labels and colors
labs(
  title = "Visualisation of Social and Psychological Reactions During the Covid-19\n Pandemic Across 
  subtitle = "Comparing anxiety levels and threat estimation in relation to Covid-19 and general threa
  x = "Country",
  fill = "Questionnaire Items:  ",
  caption = "Source: A Dataset of Social-Psychological and Emotional Reactions During the COVID-19 Pa
) +
scale_fill_manual(
  values = c("anxiety_mean" = "#c6cf95", "c19_threat_mean" = "#7fcdbb", "safety_threat_mean" = "#2c7f
  labels = c("Anxiety rating", "COVID-19 threat estimation", "General threat estimation")
) +

# Customise the themes

theme(
  plot.title = element_text(hjust = 0.5, size = 14, face = "bold"),
  plot.subtitle = element_text(hjust = 0.5, size = 12),
  plot.caption = element_text(hjust = 0, face = "italic"),


# Make the secondary axis and text red so it corresponds with the dashed red line for the death numbe
  axis.title.y.right = element_text(size = 12, color = "red"),
  axis.text.y.right = element_text(color = "red"),
  axis.title = element_text(size = 12),
  axis.ticks.x = element_blank(),
```

```
  # Adjust the legend
    legend.title = element_text(size = 10),
    legend.text = element_text(size = 8),
    legend.position = c(0.15, 0.82),
    legend.key = element_rect(colour = NA, fill = NA),


 # Add white background
    panel.background = element_rect(fill = "#ffffff", colour = NA),
    panel.grid.major.x = element_blank(),
    panel.grid.minor.x = element_blank(),

  # Keep horizontal lines
    panel.grid.major.y = element_line(color = "grey", linewidth = 0.5),

  # Adding a secondary horizontal line makes it easier to estimate the values on the y-axis.
    panel.grid.minor.y = element_line(color = "lightgrey", linewidth = 0.25),

  #Adjust margins
    plot.margin = margin(30, 20, 20, 30))
```

```
## Warning in geom_point(data = ecdc_summary, aes(x = countriesAndTerritories, :
## Ignoring unknown aesthetics: text
```

```
## Warning: A numeric ‘legend.position‘ argument in ‘theme()‘ was deprecated in ggplot2
## 3.5.0.
## i Please use the ‘legend.position.inside‘ argument of ‘theme()‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
p2
```



Source: A Dataset of Social−Psychological and Emotional Reactions During the COVID−19 Pandemi
Data on total number of deaths as reported by the ECDC for the month of April 2020.

```
#ggsave(here("visualisations/p2.png"), width = 13, height = 8.5)
```

Summary

The results suggest that participants from Spain reported the highest levels of anxiety and the Netherlands reported the lowest levels according to the questionnaire results. Participants from Spain reported high levels of threat estimation in terms of Covid-19.

An obvious difference that stood out from this visualisation is the total number of Covid-19 related deaths for each of the four countries. Abadi et al., (2023) reported that at the time of their survey, Spain had the highest number of deaths but the ECDC data indicates that the UK had the highest number of Covid-19 related deaths for the month of April 2020.

The authors report that they obtained a representative sample from each of the countries (approximately 500 people from each country). From this data, there is no significant differences between the countries for these variables that can be observed. Future research may consider recruiting a larger sample which may produced much clearer distinction of the differences in scores. It may be that participant characteristics may have impacted the outcomes on the survey, e.g., voluntary sample that was willing to complete the surveys.

Follow up

This visualisation may be made better with the addition of an interactive element. I had attempted to make the graph interactive via ggplotly, however I found that this removed some of the helpful labels that I felt was important to the visualisation. It also seemed to remove the secondary axis. Although with the

interactive plot, when the cursor hovers over the point, the information is provided, it was felt that this may be confusing at first glance without the secondary axis and may lead to the graph being read wrong.

If additional data is acquired, for example, obtaining data from more countries in Europe and how the population levels of anxiety changed across may be good to visualise in terms of an interactive line or bar chart. It will also be good to potentially visualise the information of all the countries in Europe in a map.

References

Abadi, D., Arnaldo, I., & Fischer,A. (2023). A Dataset of SocialPsychological and EmotionalReactions During the COVID-19 Pandemic Across FourEuropean Countries. *Journal of Open Psychology Data*, 11: 11, pp. 1–11. DOI: https://doi.org/10.5334/jopd.86

Miyah, Y., Benjelloun, M., Lairini, S., & Lahrichi, A. (2022). COVID-19 impact on public health, environment, human psychology, global socioeconomy, and education. *The Scientific World Journal*, 2022(5578284) 1-8. https://doi.org/10.1155/2022/5578284.