Population prescribed drugs for anxiety or depression or psychosis

Fionnuala Cousins 52319277

14 November, 2023, 20:31

```
# tidyverse includes dplyr and ggplot2 so I don't need to load them separately
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## 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/Fionnuala/OneDrive - University of Aberdeen/PU5063 Intro to HDS/Assessment
```

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
library(plotly)
```

```
##
## Attaching package: 'plotly'
##
## The following object is masked from 'package:ggplot2':
##
##     last_plot
##
## The following object is masked from 'package:stats':
##
##     filter
##
## The following object is masked from 'package:graphics':
##
##     layout
```

Question

What are the regional trends for the number of people prescribed drugs for anxiety, depression and psychosis in Scotland over the last ten years? What might these mean for employers' allocation of support resources? The next sections follow the Health Data Science Workflow to address these questions.

Data Acquisition

The data was downloaded from https://scotland.shinyapps.io/ScotPHO_profiles_tool/ on 05/11/23 for the item "population prescribed drugs for anxiety/depression/psychosis" for all available years and all health boards. The downloaded file was called timetrend_data.csv

```
#reading in the data and checking its columns:
adp_data <- read_csv(here("Inputs/timetrend_data.csv"))
```

```
## Rows: 180 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (7): indicator, area_name, area_code, area_type, period, definition, dat...
## dbl (5): year, numerator, measure, lower_confidence_interval, upper_confiden...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
glimpse(adp_data)
```

```
## Rows: 180
## Columns: 12
## $ indicator                <chr> "Population prescribed drugs for anxiety/dep~
## $ area_name                <chr> "Scotland", "NHS Ayrshire & Arran", "NHS Bor~
## $ area_code                <chr> "S00000001", "S08000015", "S08000016", "S080~
## $ area_type                <chr> "Scotland", "Health board", "Health board", ~
## $ year                     <dbl> 2010, 2010, 2010, 2010, 2010, 2010, 2010, 20~
## $ period                   <chr> "2010/11 financial year", "2010/11 financial~
## $ numerator                <dbl> 787040, 60822, 17226, 22280, 55334, 43976, 7~
## $ measure                  <dbl> 14.96, 16.31, 15.15, 14.75, 15.26, 14.86, 12~
## $ lower_confidence_interval <dbl> 14.93, 16.20, 14.94, 14.57, 15.14, 14.73, 12~
## $ upper_confidence_interval <dbl> 14.99, 16.43, 15.36, 14.92, 15.38, 14.98, 12~
## $ definition               <chr> "Percentage", "Percentage", "Percentage", "P~
## $ data_source              <chr> "Public Health Scotland (Prescribing Informa~
```

#Prepare/Clean Data

```
# There are no missing values
# This chunk is for selecting and renaming columns, removing the rows for the
# whole of Scotland and removing the NHS prefix.
# The mutate line was suggested by chatgpt when I gave it the preceding lines
# in this chunk and asked it what to add to strip out the "NHS " prefix.

clean_data <- adp_data %>%
```

```
  select('area_name','year','numerator', 'measure') %>%
  rename(number = 'numerator', NHS = 'area_name', percentage = 'measure') %>%
  filter(NHS !='Scotland') %>%
  mutate(NHS = sub("^NHS ","", NHS))
head(clean_data)
```

```
## # A tibble: 6 x 4
##   NHS                  year number percentage
##   <chr>               <dbl>  <dbl>      <dbl>
## 1 Ayrshire & Arran     2010  60822       16.3
## 2 Borders              2010  17226       15.2
## 3 Dumfries & Galloway  2010  22280       14.8
## 4 Fife                 2010  55334       15.3
## 5 Forth Valley         2010  43976       14.9
## 6 Grampian             2010  70337       12.4
```

```
# The below code plots the data as-is. It's here so anyone can plot it and see
# for themselves that 14 is too many categories, but it's commented out so it
# doesn't get confused with the intended plot.
# options(scipen = 999)
# clean_data %>%
#   ggplot() +

#   geom_area(aes(x = year, y = number, fill = NHS))
```

#Analyse

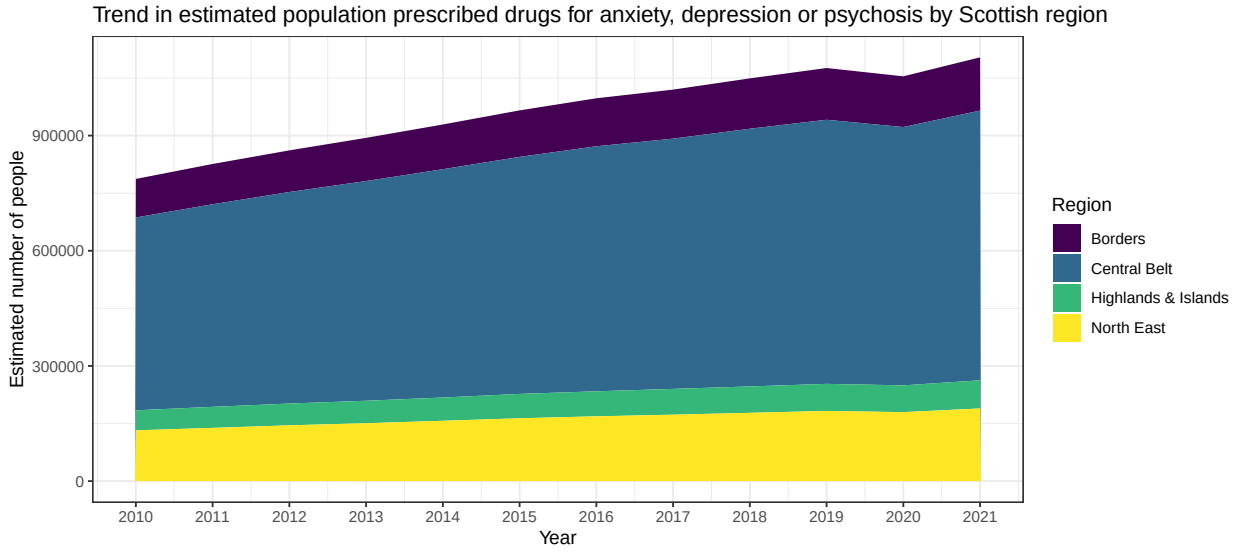
14 Health Boards are too many to plot in the same visualisation; the audience would be overwhelmed. So, I will create a new column, classifying neighbouring NHS boards into Central Belt, Borders, Highlands and Islands and North East. Then, I have to sum the old NHS Board percentages for each year into a single value for the Region for that year.

```
#I wanted to use functions from the course but when I looked up how to recategorise a categorical varia
group_data <- clean_data %>%
  mutate(Region = case_when(
    NHS %in% c("Ayrshire & Arran" , "Borders" , "Dumfries & Galloway")
    ~ "Borders",
    NHS %in% c("Fife" , "Forth Valley" , "Greater Glasgow & Clyde" ,
               "Lanarkshire" , "Lothian")
    ~ "Central Belt",
    NHS %in% c("Grampian" , "Tayside")
    ~ "North East",
    NHS %in% c("Highland" , "Western Isles" , "Orkney" , "Shetland")
    ~ "Highlands & Islands"))
```

```
#Now to re-calculate the regional percentages
summed_data <- group_data %>%
  group_by(Region, year) %>%
  summarise(total_people = sum(number))
```

```
## `summarise()` has grouped output by 'Region'. You can override using the
## `.groups` argument.
```

```r
options(scipen = 999)
summed_data %>%
  ggplot(aes(x = year, y = total_people, fill = Region)) +
  geom_area()+
  xlab("Year")+
  ylab("Estimated number of people")+
  ggtitle("Trend in estimated population prescribed drugs for anxiety, depression or psychosis by Scotti
  scale_fill_viridis(discrete=TRUE)+
  scale_x_continuous(breaks = unique(summed_data$year))+
  theme_bw()
```




```r
# If switching the output to html, this line shows values when hovered over.
# It's commented out here because the output is PDF.
# ggplotly(tooltip = c("x", "y"))
```