22.10.22_PU5058_REPORT.Rmd

2022-10-22

Aim

The aim is to highlight to local government, schools and parents the reduction in mental well being scores for girls in S4, specifically in rural areas.

Load Packages

```
# install of the following packages, if they are not already installed, this is
# done in the console as you only need to do it once.  The # in front of the
# code would need removed first, as I have left it as a comment as I have
# already installed all of the packages. Information about each of the packages
# is commented in the load libraries section below.

# install.packages('tidyverse') #remember the inverted commas around the name
# of the package

# install.packages('here')

# install.packages('sf')

# install.packages('mapview')

# install.packages('cowplot')

# install.packages('leafpop')
```

Libraries

```
# load various packages needed to create the visualisations


# here package is used to create file paths from a top level directory(folder)
library(here)

# tidyverse package has lots of functions useful for Data Science
```

```
library(tidyverse)

# sf (simple features) package is used to work with shapefiles, that includes
# spatial vector data, that are required for the creation of maps
library(sf)

# flexdashboard package is used to display multiple visualisations. I only have
# one map to display so this is not required.  In the written discussion I
# mention that a table of additional information to accompany the map would be
# beneficial and then this package would be required. library(flexdashboard)

# mapview package has multiple functions that are used to create interactive
# visualisations of spatial data
library(mapview)

# gt package has functions to create tables, such as add headers, change column
# width. This could be used as an improvement as discussed in the written
# report. library(gt)

# cowplot is used in addition to the graphing package ggplot (part of the
# tidyverse package) that offers functions to create publication quality
# visualisations. Examples are choose themes, annotate graphs and align plots.
library(cowplot)

# plotly is used to create interactive graphs to be displayed on the internet
library(plotly)

# this is not a package to be installed but this sets the theme of the map,
# including the overall font size and the font size of labels and the title,
# line width.
theme_set(theme_cowplot())

# Provides functions to edit the pop up table from the interactive map.
library(leafpop)
```

There are multiple sources of information about different packages, one source can be found here (1). Default settings for cowplot can be found here (2).

Read in the data

```
# Information about where the data came from can be found in the accompanying
# report.

# to read in the data we will use the readr package which was loaded within the
# tidyverse package We will assign the data files to objects by giving them
# names

# These files include multiple variables including the mean wellbeing score
# (measure) and the year and gender of the group (indicator)
```

```r
S2_Boys <- read_csv(here("Input/22.10.22_01_PU5058_REPORT_S2_BOYS.csv"))
S2_Girls <- read_csv(here("Input/22.10.22_02_PU5058_REPORT_S2_GIRLS.csv"))
S4_Girls <- read_csv(here("Input/22.10.22_03_PU5058_REPORT_S4_GIRLS.csv"))
S4_Boys <- read_csv(here("Input/22.10.22_04_PU5058_REPORT_S4_BOYS.csv"))

# These files are required for creating the map. They include data zone
# information and shape files.

# read in the data for the council areas providing all multiple zone areas for
# each council in Scotland
simd_indicators <- read_csv(here("Input/SIMD2020v2_indicators.csv"))

# read in the geospatial data (shape files) used to create the map. Note:
# quiet=T results in no messages being displayed during upload
datazone_sf <- st_read(here("Input/sc_dz_11.shp"), quiet = T)
```

Prepare the data

Bar Chart

The first preparation of the data is for the bar chart.

```r
# check the variables are the same in each file
head(S2_Boys)
```

```
## # A tibble: 6 x 12
##   indicator area_~1 area_~2 area_~3  year period numer~4 measure lower~5 upper~6
##   <chr>     <chr>   <chr>   <chr>   <dbl> <chr>    <dbl>   <dbl>   <dbl>   <dbl>
## 1 S2 boys ~ Scotla~ S00000~ Scotla~  2012 2010--   14946    51.1    50.9    51.2
## 2 S2 boys ~ Aberde~ S12000~ Counci~  2012 2010--     520    51.1    50.4    51.8
## 3 S2 boys ~ Aberde~ S12000~ Counci~  2012 2010--     756    50.8    50.2    51.4
## 4 S2 boys ~ Angus   S12000~ Counci~  2012 2010--     330    49.4    48.5    50.3
## 5 S2 boys ~ Argyll~ S12000~ Counci~  2012 2010--     263    50.3    49.2    51.3
## 6 S2 boys ~ City o~ S12000~ Counci~  2012 2010--    1152    50.7    50.2    51.2
## # ... with 2 more variables: definition <chr>, data_source <chr>, and
## #   abbreviated variable names 1: area_name, 2: area_code, 3: area_type,
## #   4: numerator, 5: lower_confidence_interval, 6: upper_confidence_interval
```

```r
head(S2_Girls)
```

```
## # A tibble: 6 x 12
##   indicator area_~1 area_~2 area_~3  year period numer~4 measure lower~5 upper~6
##   <chr>     <chr>   <chr>   <chr>   <dbl> <chr>    <dbl>   <dbl>   <dbl>   <dbl>
## 1 S2 girls~ Scotla~ S00000~ Scotla~  2012 2010--   15081    49.3    49.2    49.5
## 2 S2 girls~ Aberde~ S12000~ Counci~  2012 2010--     539    48.4    47.7    49.2
## 3 S2 girls~ Aberde~ S12000~ Counci~  2012 2010--     736    49.2    48.6    50.0
## 4 S2 girls~ Angus   S12000~ Counci~  2012 2010--     332    48.7    47.7    49.8
## 5 S2 girls~ Argyll~ S12000~ Counci~  2012 2010--     227    49.6    48.4    50.7
## 6 S2 girls~ City o~ S12000~ Counci~  2012 2010--    1245    49.5    49.0    50.0
## # ... with 2 more variables: definition <chr>, data_source <chr>, and
## #   abbreviated variable names 1: area_name, 2: area_code, 3: area_type,
## #   4: numerator, 5: lower_confidence_interval, 6: upper_confidence_interval
```

```
head(S4_Boys)
```

```
## # A tibble: 6 x 12
##   indicator area_~1 area_~2 area_~3  year period numer~4 measure lower~5 upper~6
##   <chr>     <chr>   <chr>   <chr>   <dbl> <chr>    <dbl>   <dbl>   <dbl>   <dbl>
## 1 S4 boys ~ Scotla~ S00000~ Scotla~  2012 2010--   16409    50.5    50.4    50.6
## 2 S4 boys ~ Aberde~ S12000~ Counci~  2012 2010--     603    49.2    48.4    49.9
## 3 S4 boys ~ Aberde~ S12000~ Counci~  2012 2010--     828    50.2    49.6    50.9
## 4 S4 boys ~ Angus   S12000~ Counci~  2012 2010--     361    50.0    49.2    50.9
## 5 S4 boys ~ Argyll~ S12000~ Counci~  2012 2010--     266    50.2    49.0    51.3
## 6 S4 boys ~ City o~ S12000~ Counci~  2012 2010--    1296    50.8    50.3    51.3
## # ... with 2 more variables: definition <chr>, data_source <chr>, and
## #   abbreviated variable names 1: area_name, 2: area_code, 3: area_type,
## #   4: numerator, 5: lower_confidence_interval, 6: upper_confidence_interval
```

```
head(S4_Girls)
```

```
## # A tibble: 6 x 12
##   indicator area_~1 area_~2 area_~3  year period numer~4 measure lower~5 upper~6
##   <chr>     <chr>   <chr>   <chr>   <dbl> <chr>    <dbl>   <dbl>   <dbl>   <dbl>
## 1 S4 girls~ Scotla~ S00000~ Scotla~  2012 2010--   16328    46.8    46.7    47.0
## 2 S4 girls~ Aberde~ S12000~ Counci~  2012 2010--     629    47.4    46.7    48.2
## 3 S4 girls~ Aberde~ S12000~ Counci~  2012 2010--     809    46.8    46.1    47.5
## 4 S4 girls~ Angus   S12000~ Counci~  2012 2010--     380    45.9    45.0    46.8
## 5 S4 girls~ Argyll~ S12000~ Counci~  2012 2010--     302    45.9    44.8    47.0
## 6 S4 girls~ City o~ S12000~ Counci~  2012 2010--    1320    46.4    45.9    47.0
## # ... with 2 more variables: definition <chr>, data_source <chr>, and
## #   abbreviated variable names 1: area_name, 2: area_code, 3: area_type,
## #   4: numerator, 5: lower_confidence_interval, 6: upper_confidence_interval
```

```
# join the files for S2, S4 boys and girls into one dataset
combined_data <- bind_rows(S2_Boys, S4_Boys, S2_Girls, S4_Girls, .id = NULL)

# it can be seen in the environment panel that the combined_data dataset has
# 100 observations, which is what is expected as the 4 individual datasets had
# 25 observations each.
```

The variables that we are interested in are indicator (giving year and gender of the pupils), area_name (council area) and measure(mean wellbeing score). Further information about how the mean wellbeing score is calculated can be found in the accompanying report.

We need to know which council areas are in the S4_Girls dataset. This information will be useful when preparing the data for the map.

```
S4_Girls$area_name %>%
    table()
```

```
## .
##        Aberdeen City       Aberdeenshire               Angus       Argyll & Bute
##                    1                   1                   1                   1
##    City of Edinburgh    Clackmannanshire Dumfries & Galloway         Dundee City
##                    1                   1                   1                   1
##        East Ayrshire East Dunbartonshire        East Lothian   East Renfrewshire
##                    1                   1                   1                   1
##              Falkirk                Fife        Glasgow City            Highland
##                    1                   1                   1                   1
##           Inverclyde          Midlothian               Moray  Na h-Eileanan Siar
##                    1                   1                   1                   1
##       North Ayrshire   North Lanarkshire      Orkney Islands     Perth & Kinross
##                    1                   1                   1                   1
##             Scotland
##                    1
```

There are 24 councils with mean wellbeing score data.

The combined data has a variable called indicator which includes the gender and the school year. This is not tidy data so the indicator column is split into School Year and Gender. The variable Gender_School_Year which includes both variables is also created but to only be used as a label on the bar chart x-axis.

```
# create a new dataset to store the new variables
combined_data_substr <- combined_data %>%
    # filter for only Scotland, this will result in 4 observations to be
    # plotted on the bar chart
filter(area_type == "Scotland") %>%
    # Create a new variable and select only the 1st to 7th character
mutate(Gender_School_Year = substr(indicator, 1, 7)) %>%
    # Create a new variable and select only the 1st to 2nd character
mutate(School_Year = substr(indicator, 1, 2)) %>%
    # Create a new variable and select only the 4th to 7th character
mutate(Gender = substr(indicator, 4, 7))

# check the new columns have been created
head(combined_data_substr)
```

```
## # A tibble: 4 x 15
##   indicator area_~1 area_~2 area_~3  year period numer~4 measure lower~5 upper~6
##   <chr>     <chr>   <chr>   <chr>   <dbl> <chr>    <dbl>   <dbl>   <dbl>   <dbl>
## 1 S2 boys ~ Scotla~ S00000~ Scotla~  2012 2010-~   14946    51.1    50.9    51.2
## 2 S4 boys ~ Scotla~ S00000~ Scotla~  2012 2010-~   16409    50.5    50.4    50.6
## 3 S2 girls~ Scotla~ S00000~ Scotla~  2012 2010-~   15081    49.3    49.2    49.5
## 4 S4 girls~ Scotla~ S00000~ Scotla~  2012 2010-~   16328    46.8    46.7    47.0
## # ... with 5 more variables: definition <chr>, data_source <chr>,
## #   Gender_School_Year <chr>, School_Year <chr>, Gender <chr>, and abbreviated
## #   variable names 1: area_name, 2: area_code, 3: area_type, 4: numerator,
## #   5: lower_confidence_interval, 6: upper_confidence_interval
```

The data is now only 4 observations and it can be seen that there are no missing values or unusual entries. No further investigation is required. The required variables will be selected when the bar chart is created.

Map

The preparation for the map can now be completed.

The map will require more than 4 observations for the mean wellbeing score. It will require one for each of the council areas. We will use the original S4_Girls data set for the map.

```
# Only the variables measure (mean Well being Score) and area_name (council
# area) are needed for the map Create a new dataset to save the changes
S4_Girls_col_rename <- S4_Girls %>%
    # select the variables needed to create the graph
select(measure, area_name) %>%
    # rename the column measure to be more meaningful when plotted in the map
rename(Average_Wellbeing_Score = measure)
```

The data zones to create the map will be sourced from a dataset that includes information from the Scottish Index of Multiple Deprivation report. Further information can be found in the accompanying report.

```
# Investigate the variables in the simd indicators dataset
head(simd_indicators)
```

```
## # A tibble: 6 x 37
##   Data_Z~1 Inter~2 Counc~3 Total~4 Worki~5 Incom~6 Incom~7 Emplo~8 Emplo~9 CIF
##   <chr>    <chr>   <chr>     <dbl>   <dbl> <chr>     <dbl> <chr>     <dbl> <chr>
## 1 S010065~ Culter  Aberde~     894     580 8%           71 8%           49 65
## 2 S010065~ Culter  Aberde~     793     470 5%           43 5%           25 45
## 3 S010065~ Culter  Aberde~     624     461 6%           40 4%           19 45
## 4 S010065~ Culter  Aberde~     537     307 10%          52 8%           26 80
## 5 S010065~ Culter  Aberde~     663     415 10%          68 8%           32 95
## 6 S010065~ Culter  Aberde~     759     453 4%           30 4%           17 50
## # ... with 27 more variables: ALCOHOL <dbl>, DRUG <dbl>, SMR <dbl>,
## #   DEPRESS <chr>, LBWT <chr>, EMERG <dbl>, Attendance <chr>, Attainment <chr>,
## #   no_qualifications <dbl>, not_participating <chr>, University <chr>,
## #   drive_petrol <dbl>, drive_GP <dbl>, drive_post <dbl>, drive_primary <dbl>,
## #   drive_retail <dbl>, drive_secondary <dbl>, PT_GP <dbl>, PT_post <dbl>,
## #   PT_retail <dbl>, Broadband <chr>, crime_count <chr>, crime_rate <chr>,
## #   overcrowded_count <dbl>, nocentralheat_count <dbl>, ...
```

The variables of interest will be Data_Zone and Council_area. Scotland is split into 6,976 geographic data zones.

```
# check for missing values in the simd_indicators data set
simd_indicators %>%
    summarise_all(~sum(is.na(.)))
```

```
## # A tibble: 1 x 37
##   Data_Z~1 Inter~2 Counc~3 Total~4 Worki~5 Incom~6 Incom~7 Emplo~8 Emplo~9   CIF
##      <int>   <int>   <int>   <int>   <int>   <int>   <int>   <int>   <int> <int>
## 1        0       0       0       0       0       0       0       0       0     0
## # ... with 27 more variables: ALCOHOL <int>, DRUG <int>, SMR <int>,
## #   DEPRESS <int>, LBWT <int>, EMERG <int>, Attendance <int>, Attainment <int>,
## #   no_qualifications <int>, not_participating <int>, University <int>,
## #   drive_petrol <int>, drive_GP <int>, drive_post <int>, drive_primary <int>,
## #   drive_retail <int>, drive_secondary <int>, PT_GP <int>, PT_post <int>,
## #   PT_retail <int>, Broadband <int>, crime_count <int>, crime_rate <int>,
## #   overcrowded_count <int>, nocentralheat_count <int>, ...
```

```
# check for unusual characters in the simd_indicators dataset(*Acknowledgments)
simd_indicators %>%
    select(Data_Zone, Council_area) %>%
    filter_all(any_vars(str_detect(., pattern = "%")))
```

```
## # A tibble: 0 x 2
## # ... with 2 variables: Data_Zone <chr>, Council_area <chr>
```

```
# check for unusual characters in the simd_indicators
# dataset*(Acknowledgements)
simd_indicators %>%
    select(Data_Zone, Council_area) %>%
    filter_all(any_vars(str_detect(., pattern = "\\*")))
```

```
## # A tibble: 0 x 2
## # ... with 2 variables: Data_Zone <chr>, Council_area <chr>
```

```
# Check which council areas are listed within the simd_indicators dataset
simd_indicators$Council_area %>%
    table()
```

```
## .
##         Aberdeen City         Aberdeenshire                 Angus
##                   283                   340                   155
##       Argyll and Bute     City of Edinburgh      Clackmannanshire
##                   125                   597                    72
## Dumfries and Galloway           Dundee City         East Ayrshire
##                   201                   188                   163
##   East Dunbartonshire          East Lothian     East Renfrewshire
##                   130                   132                   122
##               Falkirk                  Fife          Glasgow City
##                   214                   494                   746
##              Highland            Inverclyde            Midlothian
##                   312                   114                   115
##                 Moray  Na h-Eileanan an Iar        North Ayrshire
##                   126                    36                   186
##     North Lanarkshire        Orkney Islands     Perth and Kinross
##                   447                    29                   186
##          Renfrewshire      Scottish Borders      Shetland Islands
##                   225                   143                    30
##        South Ayrshire     South Lanarkshire              Stirling
##                   153                   431                   121
##   West Dunbartonshire          West Lothian
##                   121                   239
```

This information was checked because some of the council areas were not included in the research for the mean wellbeing scores.The number of times each council appears represents how many data zones make up the council area.

```
# create a new dataset to include only the variables Data_Zone and
# Council-area.
simd_selected_col <- simd_indicators %>%
    # select the variables needed to be able to join to the S4_Girls_col_rename
    # dataset
select(Data_Zone, Council_area) %>%
    # remove the council areas that have no mean wellbeing score
filter(Council_area != "Scottish Borders" & Council_area != "West Dunbartonshire" &
    Council_area != "Shetland Islands" & Council_area != "West Lothian" & Council_area !=
    "South Ayrshire" & Council_area != "South Lanarkshire" & Council_area != "Renfrewshire" &
    Council_area != "Stirling")
```

There are no missing values, unusual characters or council areas with no mean wellbeing score in the data set.

The council areas in both data sets simd_selected_col (area_name) and in the S4_Girls_col_rename (Council_area) can now be used to join the datasets. This will create one dataset with the mean wellbeing scores and data zone information. This dataset will then be combined to the datazone shape files needed to create the map.

```
# updated version use this joining councils
S4_councils_DZ <- left_join(simd_selected_col, S4_Girls_col_rename, by = c(Council_area = "area_name"))
```

```
head(S4_councils_DZ)
```

```
## # A tibble: 6 x 3
##   Data_Zone Council_area  Average_Wellbeing_Score
##   <chr>     <chr>                           <dbl>
## 1 S01006506 Aberdeen City                    47.4
## 2 S01006507 Aberdeen City                    47.4
## 3 S01006508 Aberdeen City                    47.4
## 4 S01006509 Aberdeen City                    47.4
## 5 S01006510 Aberdeen City                    47.4
## 6 S01006511 Aberdeen City                    47.4
```

The datazone shapefile dataset will now be checked.

```
# Investigate the variables in the datazone_sf dataset
head(datazone_sf)
```

```
## Simple feature collection with 6 features and 9 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -2.317044 ymin: 57.07619 xmax: -2.251077 ymax: 57.10491
## Geodetic CRS:  WGS 84
##    DataZone        Name TotPop2011 ResPop2011 HHCnt2011  StdAreaHa StdAreaKm2
## 1 S01006506 Culter - 01        872        852       424 438.880218   4.388801
## 2 S01006507 Culter - 02        836        836       364  22.349739   0.223498
## 3 S01006508 Culter - 03        643        643       340  27.019476   0.270194
## 4 S01006509 Culter - 04        580        580       274   9.625426   0.096254
## 5 S01006510 Culter - 05        644        577       256  18.007657   0.180076
## 6 S01006511 Culter - 06        751        749       315  40.048802   0.400487
##   Shape_Leng Shape_Area                       geometry
## 1  11801.872 4388802.12 MULTIPOLYGON (((-2.27748 57...
## 2   2900.406  221746.84 MULTIPOLYGON (((-2.273543 5...
## 3   3468.762  270194.75 MULTIPOLYGON (((-2.274429 5...
## 4   1647.461   96254.26 MULTIPOLYGON (((-2.266113 5...
## 5   3026.111  180076.58 MULTIPOLYGON (((-2.260134 5...
## 6   4300.089  400488.04 MULTIPOLYGON (((-2.253576 5...
```

```
# datazone_sf%>% summarise_all(~sum(is.na(.))) I tried to check for missing
# values in the datazone_sf but I received a message 'no loop for break/next,
# jumping to top level. This would need further investigation. I assume it is
# to do with being a shapefile.
```

The shape files now need to be joined using the data zones.

```
# Add shape files by joining the data by the data zones.

S4_councils_DZ_sf <- left_join(datazone_sf, S4_councils_DZ, by = c(DataZone = "Data_Zone"))
```

```
# Investigate the variables in the S4_councils_DZ_sf dataset
head(S4_councils_DZ_sf)
```

```
## Simple feature collection with 6 features and 11 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -2.317044 ymin: 57.07619 xmax: -2.251077 ymax: 57.10491
## Geodetic CRS:  WGS 84
##    DataZone        Name TotPop2011 ResPop2011 HHCnt2011  StdAreaHa StdAreaKm2
## 1 S01006506 Culter - 01        872        852       424 438.880218   4.388801
## 2 S01006507 Culter - 02        836        836       364  22.349739   0.223498
## 3 S01006508 Culter - 03        643        643       340  27.019476   0.270194
## 4 S01006509 Culter - 04        580        580       274   9.625426   0.096254
## 5 S01006510 Culter - 05        644        577       256  18.007657   0.180076
## 6 S01006511 Culter - 06        751        749       315  40.048802   0.400487
##   Shape_Leng Shape_Area  Council_area Average_Wellbeing_Score
## 1  11801.872 4388802.12 Aberdeen City                   47.43
## 2   2900.406  221746.84 Aberdeen City                   47.43
## 3   3468.762  270194.75 Aberdeen City                   47.43
## 4   1647.461   96254.26 Aberdeen City                   47.43
## 5   3026.111  180076.58 Aberdeen City                   47.43
## 6   4300.089  400488.04 Aberdeen City                   47.43
##                         geometry
## 1 MULTIPOLYGON (((-2.27748 57...
## 2 MULTIPOLYGON (((-2.273543 5...
## 3 MULTIPOLYGON (((-2.274429 5...
## 4 MULTIPOLYGON (((-2.266113 5...
## 5 MULTIPOLYGON (((-2.260134 5...
## 6 MULTIPOLYGON (((-2.253576 5...
```

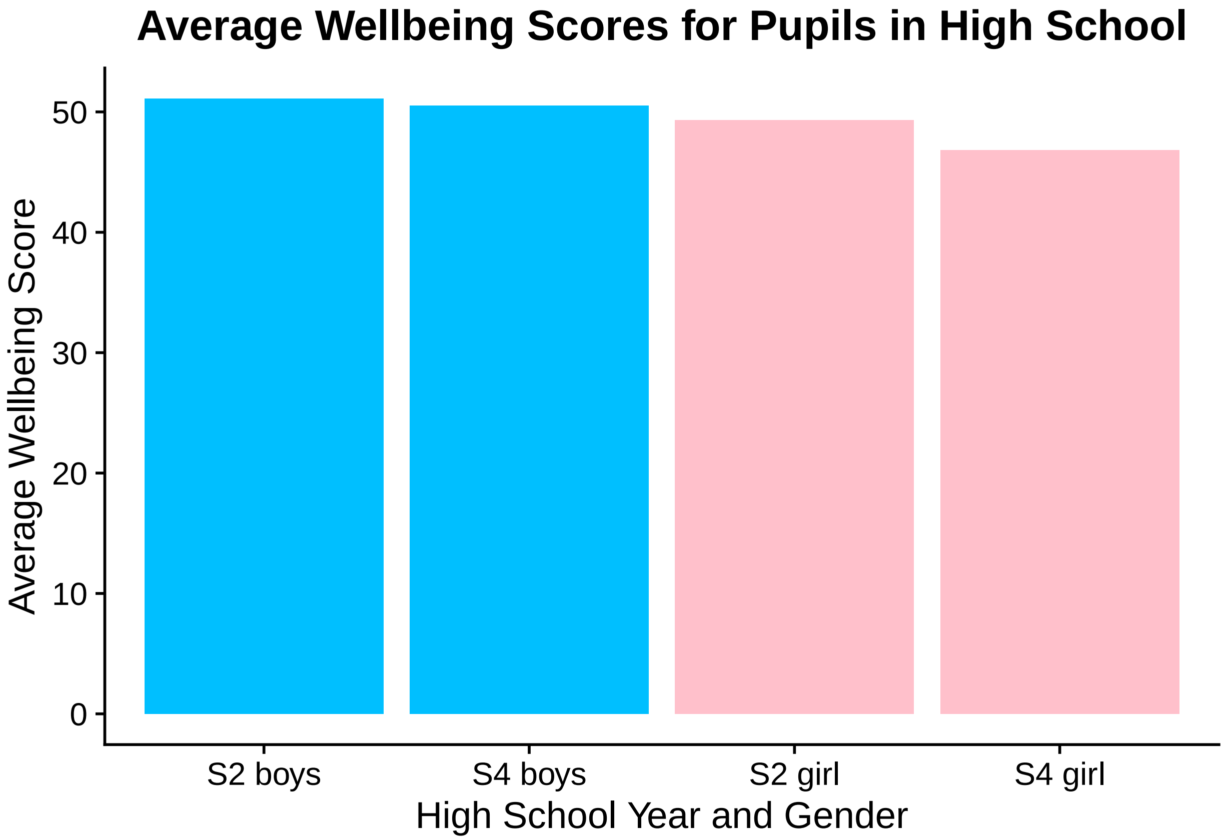
The dataset S4_councils_DZ_sf now contains the shape files for each datazone, council area and the mean wellbeing score that will be used to create the map.

Create Visualisation

Column Chart

The column chart can now be created.

```
# Select the dataset that we want to use to create the graph Select the
# variables (columns) that we will use to create the graph Reorder the columns
# so Gender_School_Year variable is put in order by descending measure variable
# on the x-axis. The y-axis is the variable measure and we are going to choose
# the colour of the column depending on the gender. Define which colours you
# want the columns to be, boys will be deep sky blue and girls pink. A column
# chart is chosen instead of a bar because we already have the total measure
# values. Remove the legend, we have the gender on the x-axis for people who
# are colour blind. Centre the title of the graph Add an x-axis label Add a
# y-axis label Add a title
combined_data_substr %>%
    select(Gender_School_Year, Gender, measure) %>%
    ggplot(aes(x = reorder(Gender_School_Year, -measure), y = measure, fill = Gender)) +
    geom_col() + scale_fill_manual(values = c("deepskyblue", "pink")) + theme(legend.position = "none",
    plot.title = element_text(hjust = 0.5)) + xlab("High School Year and Gender") +
    ylab("Average Wellbeing Score") + ggtitle("Average Wellbeing Scores for Pupils in High School")
```



The colours were chosen from a large selection available in R, colours in R (3).

Map

The map can now be created.

```
# Define the colour palette to use when creating the map. The argument
# direction = -1 has not been used because the low numbers are to have dark
# colours rather than vice versa. The number of colours required is 7.  pal =
# viridisLite::rocket(n = 7)

# Create a new object to store the selected variables for the map
# S4_councils_DZ_sf %>%

# Select the relevant variables to create the map
# select(DataZone,Council_area,Average_Wellbeing_Score) %>%

# mapview is a package that can quickly create interactive maps mapview(

# Select the type of maps to use map.types = 'OpenStreetMap',

# Define which variable will be used to determine the colour shades of the
# polygons zcol = 'Average_Wellbeing_Score',

# The council areas will be visible when the mouse hovers over an area on the
# map label = S4_councils_DZ_sf$Council_area,

# Select the variables that you want to be visible in the pop up table in the
# interactive map popup=popupTable(S4_councils_DZ_sf, zcol=c('Council_area',
# 'Average_Wellbeing_Score')),

# Define the name of the layer that we want to show on the map layer.name =
# 'Average_Wellbeing_Score',

# Set the opacity of the colour fills to 0.8, 1 is fully opaque (you can't see
# through). The boundaries of the council areas have to be visible.
# alpha.regions = 0.8,

# This defines that pal (defined above) will be the colour pallette used to
# fill the polygons in the map. Break points have been defined so only 7
# colours are required.  col.regions = pal,at=c(45,45.5,46,46.5,47,47.5,48) )
```

Further information about the virisisLite (4). package can found using the link.

References

https://rdocumentation.org/ (1)

https://rdrr.io/cran/cowplot/man/theme_cowplot.html (2)

https://r-graph-gallery.com/42-colors-names.html (3)

https://cran.r-project.org/web/packages/viridisLite/viridisLite.pdf (4)

Acknowledgements

https://jessbutler.github.io/simd/ *(You can see the code and data used under "Sources".)