Processing a WHO air quality data file

Load the Excel spreadsheet from the WHO Global Urban Ambient Air Pollution Database

```
# Load the packages for reading Excel files
# Use the Bristol R package mirror
bristol_cran <- 'https://www.stats.bris.ac.uk/R'
install.packages('rJava', repo=bristol_cran)
```

```
##
## The downloaded binary packages are in
##  /var/folders/2z/3z9jy4411qj2b_pmmt5fm64h0000gq/T//RtmpcUXYWz/downloaded_packages
```

```
install.packages("XLConnect", repo=bristol_cran)
```

```
##
## The downloaded binary packages are in
##  /var/folders/2z/3z9jy4411qj2b_pmmt5fm64h0000gq/T//RtmpcUXYWz/downloaded_packages
```

```
library(XLConnect)
```

```
## Loading required package: XLConnectJars
```

```
## XLConnect 0.2-15 by Mirai Solutions GmbH [aut],
##   Martin Studer [cre],
##   The Apache Software Foundation [ctb, cph] (Apache POI),
##   Graph Builder [ctb, cph] (Curvesapi Java library)
```

```
## http://www.mirai-solutions.com
## https://github.com/miraisolutions/xlconnect
```

Load the relevant worksheet from the Excel file:

```
workbook <- loadWorkbook('aap_air_quality_database_2018_v14.xlsx')
data <- XLConnect::readWorksheet(workbook, 'database', startRow=3)
head(data)
```

```
##          Region iso3 Country City.Town Year Annual.mean..ug.m3
## 1 Europe (LMIC)  ALB Albania     Korce 2015                 45
## 2 Europe (LMIC)  ALB Albania     Korce 2016                 40
## 3 Europe (LMIC)  ALB Albania    Tirana 2013                 32
## 4 Europe (LMIC)  ALB Albania     Vlore 2014                 15
## 5 Europe (LMIC)  ALB Albania     Vlore 2015                 19
## 6 Europe (LMIC)  ALB Albania     Vlore 2016                 23
##   Temporal.coverage note.on.converted.PM10 Annual.mean..ug.m3.1
## 1              >75%               Measured                   30
## 2              >75%               Measured                   29
## 3                NA               Measured                   16
## 4              >75%               Measured (10)-converted value
## 5              >75%               Measured (13)-converted value
## 6              >75%               Measured (15)-converted value
##   Temporal.coverage.1 note.on.converted.PM2.5
## 1                >75%               Measured
## 2                >75%               Measured
## 3                  NA               Measured
## 4                  NA              Converted
## 5                  NA              Converted
```

```
## 6                   NA               Converted
##   Number.and.type.of.monitoring.stations
## 1                   1 Suburban-Background
## 2                   1 Suburban-Background
## 3              1 station, traffic, urban
## 4                     1 Urban-Background
## 5                     1 Urban-Background
## 6                     1 Urban-Background
##                                       Reference.for.air.quality
## 1 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 2 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 3   European Environment Agency, Air quality e-reporting database
## 4 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 5 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6 The European Environmental Agency (EEA) [downloaded 2018/04/14]
##   Database.version..year. status
## 1                    2018   <NA>
## 2                    2018   <NA>
## 3                    2016   <NA>
## 4                    2018   <NA>
## 5                    2018   <NA>
## 6                    2018   <NA>
```

```
str(data)
```

```
## 'data.frame':    11962 obs. of  15 variables:
##  $ Region                                : chr  "Europe (LMIC)" "Europe (LMIC)" "Europe (LMIC)" "Eur
##  $ iso3                                  : chr  "ALB" "ALB" "ALB" "ALB" ...
##  $ Country                               : chr  "Albania" "Albania" "Albania" "Albania" ...
##  $ City.Town                             : chr  "Korce" "Korce" "Tirana" "Vlore" ...
##  $ Year                                  : num  2015 2016 2013 2014 2015 ...
##  $ Annual.mean..ug.m3                    : chr  "45" "40" "32" "15" ...
##  $ Temporal.coverage                     : chr  ">75%" ">75%" "NA" ">75%" ...
##  $ note.on.converted.PM10                : chr  "Measured" "Measured" "Measured" "Measured" ...
##  $ Annual.mean..ug.m3.1                  : chr  "30" "29" "16" "(10)-converted value" ...
##  $ Temporal.coverage.1                   : chr  ">75%" ">75%" "NA" "NA" ...
##  $ note.on.converted.PM2.5               : chr  "Measured" "Measured" "Measured" "Converted" ...
##  $ Number.and.type.of.monitoring.stations: chr  "1 Suburban-Background" "1 Suburban-Background" "1 s
##  $ Reference.for.air.quality             : chr  "The European Environmental Agency (EEA) [downloaded
##  $ Database.version..year.               : num  2018 2018 2016 2018 2018 ...
##  $ status                                : chr  NA NA NA NA ...
```

We are going to use some Tidyverse packages to organize the data.

```
install.packages("tidyverse", repo=bristol_cran)
```

```
##
## The downloaded binary packages are in
##  /var/folders/2z/3z9jy4411qj2b_pmmt5fm64h0000gq/T//RtmpcUXYWz/downloaded_packages
```

`dplyr` is part of the Tidyverse. It has some nice utilities for selecting and renaming columns in a data frame. You can so this without `dplyr`, but `dplyr` makes it much easier to express in a few lines of code.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Notice there are two sets of columns starting `Annual.mean`.

```
str(select(data, starts_with('Annual.mean')))
```

```
## 'data.frame':    11962 obs. of  2 variables:
##  $ Annual.mean..ug.m3  : chr  "45" "40" "32" "15" ...
##  $ Annual.mean..ug.m3.1: chr  "30" "29" "16" "(10)-converted value" ...
```

Likewise for `Temporal.coverage`:

```
str(select(data, starts_with('Temporal.coverage')))
```

```
## 'data.frame':    11962 obs. of  2 variables:
##  $ Temporal.coverage  : chr  ">75%" ">75%" "NA" ">75%" ...
##  $ Temporal.coverage.1: chr  ">75%" ">75%" "NA" "NA" ...
```

Looking at the column names, and at the original Excel file, the first in each pair refers to PM10, and the second to PM2.5 : https://en.wikipedia.org/wiki/Particulates

We rename the columns accordingly:

```
data <- rename(data,
               'pm10_yr_mean'='Annual.mean..ug.m3',
               'pm10_temp_cover'='Temporal.coverage',
               'pm25_yr_mean'='Annual.mean..ug.m3.1',
               'pm25_temp_cover'='Temporal.coverage.1'
)
head(data)
```

```
##          Region iso3 Country City.Town Year pm10_yr_mean pm10_temp_cover
## 1 Europe (LMIC)  ALB Albania     Korce 2015           45            >75%
## 2 Europe (LMIC)  ALB Albania     Korce 2016           40            >75%
## 3 Europe (LMIC)  ALB Albania    Tirana 2013           32              NA
## 4 Europe (LMIC)  ALB Albania     Vlore 2014           15            >75%
## 5 Europe (LMIC)  ALB Albania     Vlore 2015           19            >75%
## 6 Europe (LMIC)  ALB Albania     Vlore 2016           23            >75%
##   note.on.converted.PM10         pm25_yr_mean pm25_temp_cover
## 1               Measured                   30            >75%
## 2               Measured                   29            >75%
## 3               Measured                   16              NA
## 4               Measured (10)-converted value              NA
## 5               Measured (13)-converted value              NA
## 6               Measured (15)-converted value              NA
##   note.on.converted.PM2.5 Number.and.type.of.monitoring.stations
## 1                Measured                   1 Suburban-Background
## 2                Measured                   1 Suburban-Background
## 3                Measured               1 station, traffic, urban
## 4               Converted                      1 Urban-Background
## 5               Converted                      1 Urban-Background
## 6               Converted                      1 Urban-Background
##                                   Reference.for.air.quality
```

```
## 1 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 2 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 3   European Environment Agency, Air quality e-reporting database
## 4 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 5 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6 The European Environmental Agency (EEA) [downloaded 2018/04/14]
##   Database.version..year. status
## 1                    2018   <NA>
## 2                    2018   <NA>
## 3                    2016   <NA>
## 4                    2018   <NA>
## 5                    2018   <NA>
## 6                    2018   <NA>
```

Looking through the data (perhaps with the `View` function), it looks like the particulate concentration columns have form `(<number>)-converted value` if the value is converted.

```
head(data$pm25_yr_mean)
```

```
## [1] "30"                   "29"                   "16"
## [4] "(10)-converted value" "(13)-converted value" "(15)-converted value"
```

First we check that the `note.on.converted.PM25` column always says `Converted` when the PM2.5 concentration column has `-converted value`:

```
is_converted <- data$note.on.converted.PM2.5 == 'Converted'
head(is_converted)
```

```
## [1] FALSE FALSE FALSE  TRUE  TRUE  TRUE
```

Do all the `Converted` values have `-converted value` in their concentration column? Do `Measured` values never have `-converted value` in their concentration column?

Here we use the `stringr` package to make it easier to work with strings. `stringr` is one of the packages in the Tidyverse.

```
library(stringr)
```

For each value in the concentration column, we test whether it contains the string `-converted value`.

```
has_converted_value = str_detect(data$pm25_yr_mean, '-converted value')
head(has_converted_value)
```

```
## [1] FALSE FALSE FALSE  TRUE  TRUE  TRUE
```

We confirm that there is complete equivalence between the `Converted` values in the `note.on.converted.PM2.5` column, and the presence of `-converted value` in the concentration column.

```
all(is_converted == has_converted_value)
```

```
## [1] TRUE
```

Do the same for the PM10 columns:

```
is_conv_pm10 <- data$note.on.converted.PM10 == 'Converted'
all(is_conv_pm10 == str_detect(data$pm10_yr_mean, '-converted value'))
```

```
## [1] TRUE
```

The equivalence we found means that we can throw away the `converted value` part of the concentration values, because the information is already in the matching `note.on.converted.PM` column.

Now we need to convert the concentration column into numbers, by removing the `-converted value` part, and converting to numeric values. Because we will do this for two columns, we put the logic into a function, so we can repeat it, without retyping.

```
recode_col <- function(col) {
  # Recode column with "converted" values as numeric
  # Get number out of '(number)-converted value', if present
  conv_strs <- str_match(col, '\\((\\d+)\\)-converted value')[, 2]
  # If value did not match '(number)-converted value', we get NA
  not_converted = is.na(conv_strs)
  conv_values <- numeric(length(col))
  conv_values[not_converted] = as.numeric(col[not_converted])
  converted = !not_converted
  conv_values[converted] = as.numeric(conv_strs[converted])
  conv_values
}
```

Apply the function to the two concentration columns:

```
data$pm10_yr_mean = recode_col(data$pm10_yr_mean)
data$pm25_yr_mean = recode_col(data$pm25_yr_mean)
head(data)
```

```
##          Region iso3 Country City.Town Year pm10_yr_mean pm10_temp_cover
## 1 Europe (LMIC)  ALB Albania     Korce 2015           45            >75%
## 2 Europe (LMIC)  ALB Albania     Korce 2016           40            >75%
## 3 Europe (LMIC)  ALB Albania    Tirana 2013           32              NA
## 4 Europe (LMIC)  ALB Albania     Vlore 2014           15            >75%
## 5 Europe (LMIC)  ALB Albania     Vlore 2015           19            >75%
## 6 Europe (LMIC)  ALB Albania     Vlore 2016           23            >75%
##   note.on.converted.PM10 pm25_yr_mean pm25_temp_cover
## 1               Measured           30            >75%
## 2               Measured           29            >75%
## 3               Measured           16              NA
## 4               Measured           10              NA
## 5               Measured           13              NA
## 6               Measured           15              NA
##   note.on.converted.PM2.5 Number.and.type.of.monitoring.stations
## 1                Measured                   1 Suburban-Background
## 2                Measured                   1 Suburban-Background
## 3                Measured               1 station, traffic, urban
## 4               Converted                     1 Urban-Background
## 5               Converted                     1 Urban-Background
## 6               Converted                     1 Urban-Background
##                                      Reference.for.air.quality
## 1 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 2 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 3   European Environment Agency, Air quality e-reporting database
## 4 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 5 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6 The European Environmental Agency (EEA) [downloaded 2018/04/14]
##   Database.version..year. status
## 1                    2018   <NA>
## 2                    2018   <NA>
## 3                    2016   <NA>
## 4                    2018   <NA>
```

```
## 5                    2018   <NA>
## 6                    2018   <NA>
```

Actually, there are a few typos in the `notes.on.converted.PM` fields:

```
table(data$note.on.converted.PM2.5)
```

```
##
## converted Converted  measured  Measured
##         2      5875         1      6084
```

```
table(data$note.on.converted.PM10)
```

```
##
## Converted  measured  Measured
##      1887         3     10072
```

We need to set `measured` to be `Measured` and `converted` to be `Converted`. To do this, we use the `str_to_title` function from the `stringr` library that we have already loaded:

It works like this:

```
str_to_title('measured')
```

```
## [1] "Measured"
```

We apply it to the whole vector, for both columns:

```
data$note.on.converted.PM2.5 <- str_to_title(data$note.on.converted.PM2.5)
table(data$note.on.converted.PM2.5)
```

```
##
## Converted  Measured
##      5877      6085
```

```
data$note.on.converted.PM10 <- str_to_title(data$note.on.converted.PM10)
table(data$note.on.converted.PM10)
```

```
##
## Converted  Measured
##      1887     10075
```

Next we process the `Region` field. It consists of two pieces of information: the region name, and the income label, that can be `HIC` (High Income Country) or `LMIC` (Lower Middle Income Country):

```
region_values <- data$Region
head(region_values)
```

```
## [1] "Europe (LMIC)" "Europe (LMIC)" "Europe (LMIC)" "Europe (LMIC)"
## [5] "Europe (LMIC)" "Europe (LMIC)"
```

Time for some more fancy string manipulation:

```
# Split the string into the region name and the income catagory.
regions_types <- str_match(region_values, '(.*)\\s+\\((.*)\\)')
head(regions_types)
```

```
##      [,1]            [,2]     [,3]
## [1,] "Europe (LMIC)" "Europe" "LMIC"
## [2,] "Europe (LMIC)" "Europe" "LMIC"
## [3,] "Europe (LMIC)" "Europe" "LMIC"
## [4,] "Europe (LMIC)" "Europe" "LMIC"
## [5,] "Europe (LMIC)" "Europe" "LMIC"
```

```
## [6,] "Europe (LMIC)" "Europe" "LMIC"
```

Replace original `Region` column with region name, and add new `income_category` column:

```
data$Region <- regions_types[, 2]
data$income_category <- regions_types[, 3]
head(data)
```

```
##   Region iso3 Country City.Town Year pm10_yr_mean pm10_temp_cover
## 1 Europe  ALB Albania     Korce 2015           45            >75%
## 2 Europe  ALB Albania     Korce 2016           40            >75%
## 3 Europe  ALB Albania    Tirana 2013           32              NA
## 4 Europe  ALB Albania     Vlore 2014           15            >75%
## 5 Europe  ALB Albania     Vlore 2015           19            >75%
## 6 Europe  ALB Albania     Vlore 2016           23            >75%
##   note.on.converted.PM10 pm25_yr_mean pm25_temp_cover
## 1               Measured           30            >75%
## 2               Measured           29            >75%
## 3               Measured           16              NA
## 4               Measured           10              NA
## 5               Measured           13              NA
## 6               Measured           15              NA
##   note.on.converted.PM2.5 Number.and.type.of.monitoring.stations
## 1                Measured                  1 Suburban-Background
## 2                Measured                  1 Suburban-Background
## 3                Measured              1 station, traffic, urban
## 4               Converted                     1 Urban-Background
## 5               Converted                     1 Urban-Background
## 6               Converted                     1 Urban-Background
##                                     Reference.for.air.quality
## 1 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 2 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 3   European Environment Agency, Air quality e-reporting database
## 4 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 5 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6 The European Environmental Agency (EEA) [downloaded 2018/04/14]
##   Database.version..year. status income_category
## 1                    2018   <NA>            LMIC
## 2                    2018   <NA>            LMIC
## 3                    2016   <NA>            LMIC
## 4                    2018   <NA>            LMIC
## 5                    2018   <NA>            LMIC
## 6                    2018   <NA>            LMIC
```

Confirm the number of income categories:

```
table(data$income_category)
```

```
##
##  HIC LMIC
## 9702 2260
```

For convenience, we would like `income_category` to come after `Region` in the sequence of columns.

```
col_names <- names(data)
col_names
```

```
##  [1] "Region"
```

```
##  [2] "iso3"
##  [3] "Country"
##  [4] "City.Town"
##  [5] "Year"
##  [6] "pm10_yr_mean"
##  [7] "pm10_temp_cover"
##  [8] "note.on.converted.PM10"
##  [9] "pm25_yr_mean"
## [10] "pm25_temp_cover"
## [11] "note.on.converted.PM2.5"
## [12] "Number.and.type.of.monitoring.stations"
## [13] "Reference.for.air.quality"
## [14] "Database.version..year."
## [15] "status"
## [16] "income_category"
```

```
# Reorder, to put income category after Region
new_names = c(col_names[1], 'income_category', col_names[2:15])
new_names
```

```
##  [1] "Region"
##  [2] "income_category"
##  [3] "iso3"
##  [4] "Country"
##  [5] "City.Town"
##  [6] "Year"
##  [7] "pm10_yr_mean"
##  [8] "pm10_temp_cover"
##  [9] "note.on.converted.PM10"
## [10] "pm25_yr_mean"
## [11] "pm25_temp_cover"
## [12] "note.on.converted.PM2.5"
## [13] "Number.and.type.of.monitoring.stations"
## [14] "Reference.for.air.quality"
## [15] "Database.version..year."
## [16] "status"
```

Reorder data frame with new column order.

```
data <- select(data, new_names)
str(data)
```

```
## 'data.frame':    11962 obs. of  16 variables:
##  $ Region                                : chr  "Europe" "Europe" "Europe" "Europe" ...
##  $ income_category                       : chr  "LMIC" "LMIC" "LMIC" "LMIC" ...
##  $ iso3                                  : chr  "ALB" "ALB" "ALB" "ALB" ...
##  $ Country                               : chr  "Albania" "Albania" "Albania" "Albania" ...
##  $ City.Town                             : chr  "Korce" "Korce" "Tirana" "Vlore" ...
##  $ Year                                  : num  2015 2016 2013 2014 2015 ...
##  $ pm10_yr_mean                          : num  45 40 32 15 19 23 20 28 22 24 ...
##  $ pm10_temp_cover                       : chr  ">75%" ">75%" "NA" ">75%" ...
##  $ note.on.converted.PM10                : chr  "Measured" "Measured" "Measured" "Measured" ...
##  $ pm25_yr_mean                          : num  30 29 16 10 13 15 13 18 11 11 ...
##  $ pm25_temp_cover                       : chr  ">75%" ">75%" "NA" "NA" ...
##  $ note.on.converted.PM2.5               : chr  "Measured" "Measured" "Measured" "Converted" ...
##  $ Number.and.type.of.monitoring.stations: chr  "1 Suburban-Background" "1 Suburban-Background" "1 s
```

```
##  $ Reference.for.air.quality             : chr  "The European Environmental Agency (EEA) [downloaded
##  $ Database.version..year.               : num  2018 2018 2016 2018 2018 ...
##  $ status                                : chr  NA NA NA NA ...
```

We need a column for unique city / town name. For example, there is more than one `London`, meaning the `City.Town` values are not unique to a city or town:

```
town_is_london <- data$City.Town == 'London'
subset(data, town_is_london)
```

```
##        Region income_category iso3        Country City.Town Year
## 1483 Americas             HIC  CAN         Canada    London 2015
## 6811   Europe             HIC  GBR United Kingdom    London 2013
## 6812   Europe             HIC  GBR United Kingdom    London 2014
## 6813   Europe             HIC  GBR United Kingdom    London 2015
## 6814   Europe             HIC  GBR United Kingdom    London 2016
##      pm10_yr_mean pm10_temp_cover note.on.converted.PM10 pm25_yr_mean
## 1483           15              NA              Converted            8
## 6811           24            >75%               Measured           15
## 6812           23            >75%               Measured           14
## 6813           22            >75%               Measured           12
## 6814           23            >75%               Measured           12
##      pm25_temp_cover note.on.converted.PM2.5
## 1483            >75%               Measured
## 6811            >75%               Measured
## 6812            >75%               Measured
## 6813            >75%               Measured
## 6814            >75%               Measured
##                         Number.and.type.of.monitoring.stations
## 1483                                                1 Residential
## 6811                          5 Urban-Background, 5 Urban-Traffic
## 6812 1 Suburban-Background, 5 Urban-Background, 6 Urban-Traffic
## 6813 1 Suburban-Background, 7 Urban-Background, 6 Urban-Traffic
## 6814 1 Suburban-Background, 5 Urban-Background, 7 Urban-Traffic
##                                        Reference.for.air.quality
## 1483              Ontario Ministry of Environment and Climate Change
## 6811 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6812 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6813 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6814 The European Environmental Agency (EEA) [downloaded 2018/04/14]
##      Database.version..year. status
## 1483                    2018   <NA>
## 6811                    2018   <NA>
## 6812                    2018   <NA>
## 6813                    2018   <NA>
## 6814                    2018   <NA>
```

Put the country name into the `City.Town` name, to disambiguate.

```
data$City.Town <- paste(data$City.Town, ' (', data$Country, ')', sep='')
head(subset(data, town_is_london))
```

```
##        Region income_category iso3        Country               City.Town
## 1483 Americas             HIC  CAN         Canada         London (Canada)
## 6811   Europe             HIC  GBR United Kingdom London (United Kingdom)
## 6812   Europe             HIC  GBR United Kingdom London (United Kingdom)
```

```
## 6813   Europe             HIC  GBR United Kingdom London (United Kingdom)
## 6814   Europe             HIC  GBR United Kingdom London (United Kingdom)
##      Year pm10_yr_mean pm10_temp_cover note.on.converted.PM10 pm25_yr_mean
## 1483 2015           15              NA              Converted            8
## 6811 2013           24            >75%               Measured           15
## 6812 2014           23            >75%               Measured           14
## 6813 2015           22            >75%               Measured           12
## 6814 2016           23            >75%               Measured           12
##      pm25_temp_cover note.on.converted.PM2.5
## 1483            >75%               Measured
## 6811            >75%               Measured
## 6812            >75%               Measured
## 6813            >75%               Measured
## 6814            >75%               Measured
##                          Number.and.type.of.monitoring.stations
## 1483                                                1 Residential
## 6811                          5 Urban-Background, 5 Urban-Traffic
## 6812 1 Suburban-Background, 5 Urban-Background, 6 Urban-Traffic
## 6813 1 Suburban-Background, 7 Urban-Background, 6 Urban-Traffic
## 6814 1 Suburban-Background, 5 Urban-Background, 7 Urban-Traffic
##                                          Reference.for.air.quality
## 1483              Ontario Ministry of Environment and Climate Change
## 6811 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6812 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6813 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6814 The European Environmental Agency (EEA) [downloaded 2018/04/14]
##      Database.version..year. status
## 1483                    2018   <NA>
## 6811                    2018   <NA>
## 6812                    2018   <NA>
## 6813                    2018   <NA>
## 6814                    2018   <NA>
```

Write the converted data frame to a CSV file.

```
write.csv(data, 'aap_air_quality_database_2018_v14.csv', row.names=FALSE)
```

Check we can load the CSV file back into memory.

```
csv_data <- read.csv('aap_air_quality_database_2018_v14.csv')
head(csv_data)
```

```
##   Region income_category iso3 Country        City.Town Year pm10_yr_mean
## 1 Europe            LMIC  ALB Albania  Korce (Albania) 2015           45
## 2 Europe            LMIC  ALB Albania  Korce (Albania) 2016           40
## 3 Europe            LMIC  ALB Albania Tirana (Albania) 2013           32
## 4 Europe            LMIC  ALB Albania  Vlore (Albania) 2014           15
## 5 Europe            LMIC  ALB Albania  Vlore (Albania) 2015           19
## 6 Europe            LMIC  ALB Albania  Vlore (Albania) 2016           23
##   pm10_temp_cover note.on.converted.PM10 pm25_yr_mean pm25_temp_cover
## 1            >75%               Measured           30            >75%
## 2            >75%               Measured           29            >75%
## 3            <NA>               Measured           16            <NA>
## 4            >75%               Measured           10            <NA>
## 5            >75%               Measured           13            <NA>
## 6            >75%               Measured           15            <NA>
```

```
##   note.on.converted.PM2.5 Number.and.type.of.monitoring.stations
## 1                Measured                                1 Suburban-Background
## 2                Measured                                1 Suburban-Background
## 3                Measured                1 station, traffic, urban
## 4               Converted                                1 Urban-Background
## 5               Converted                                1 Urban-Background
## 6               Converted                                1 Urban-Background
##                                       Reference.for.air.quality
## 1 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 2 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 3   European Environment Agency, Air quality e-reporting database
## 4 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 5 The European Environmental Agency (EEA) [downloaded 2018/04/14]
## 6 The European Environmental Agency (EEA) [downloaded 2018/04/14]
##   Database.version..year. status
## 1                    2018   <NA>
## 2                    2018   <NA>
## 3                    2016   <NA>
## 4                    2018   <NA>
## 5                    2018   <NA>
## 6                    2018   <NA>
```