MM916 Week 2 Question Sheet

This question sheet covers material from Week 2 of MM916. There may be some things in this sheet that have not specifically been covered in lectures. Just remember that you are always allowed to make use of the help files in R and please feel free to ask any of the tutors for help during the live session.

When starting this lab please make sure to open a new R Script so that you don't lose any work.

Question 1

The data set `Population.txt` contains the population in each data zone in Scotland between 2001 and 2013. A data zone is a small region defined for data analysis purposes.

a) Download the data `Population.txt` and read this into R.

```
# assuming that you have set your working directory
# You need to specify header=TRUE
population <- read.table("Population.txt", header=TRUE)
```

b) Describe why this data set cannot be described as tidy

```
# Time should be a variable, rather than column headings -
# this data would be better suited to a wide format`
```

c) Using appropriate functions from the `tidyr` package reshape this data set into a tidy format

```
# We've used -> to assign the data to the object population_tidy
# This is called right assign and makes the code flow

population %>%
  pivot_longer(Y2001:Y2013, names_to="Year", values_to="Population", names_prefix="Y") ->
  population_tidy
```

d) Calculate the average population per year. Has this changed over time?

```
# Use the tidy data!

population_tidy %>%
  group_by(Year) %>%
  summarise(mean(Population))
```

```
## # A tibble: 13 x 2
##    Year  `mean(Population)`
##    <chr>              <dbl>
##  1 2001                779.
##  2 2002                779.
##  3 2003                779.
##  4 2004                782.
##  5 2005                786.
##  6 2006                789.
##  7 2007                795.
```

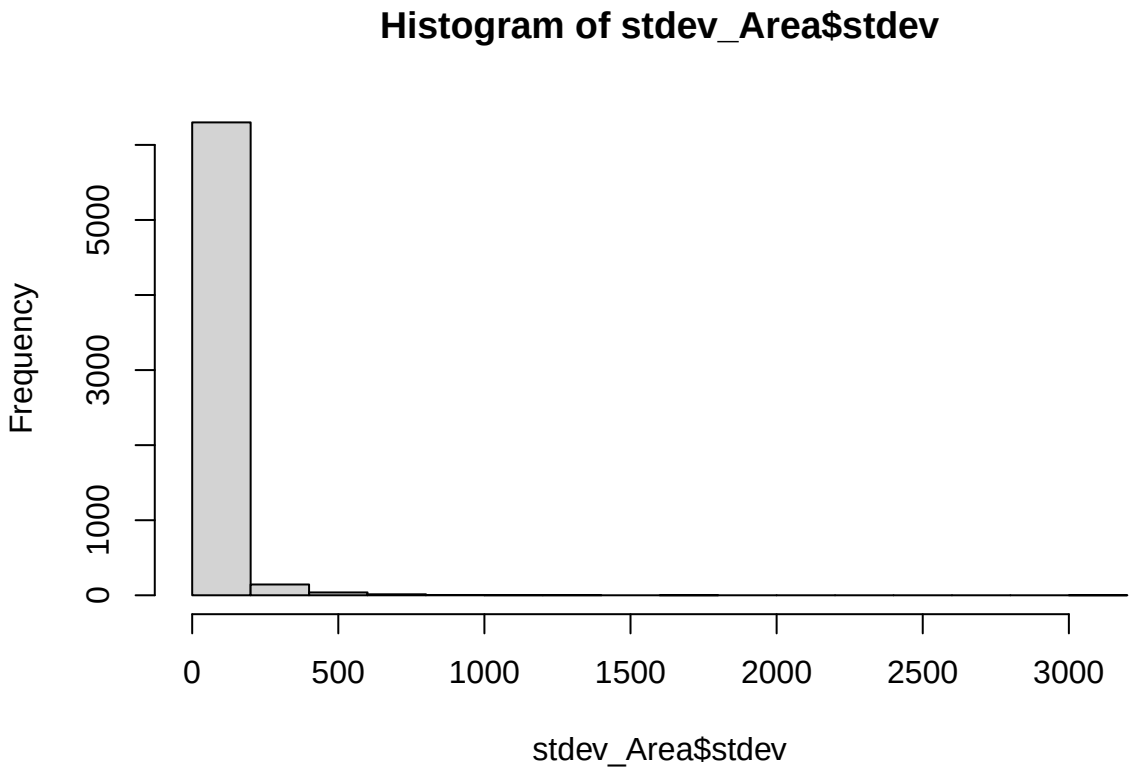
```
##  8 2008                800.
##  9 2009                804.
## 10 2010                809.
## 11 2011                815.
## 12 2012                817.
## 13 2013                819.
```

```
# It does look like there has been a consistent increase in population sizes.
# Greater increases in later years.`
```

e) Calculate the standard deviation of the populations for each region. Plot these in a histogram using `hist()`.

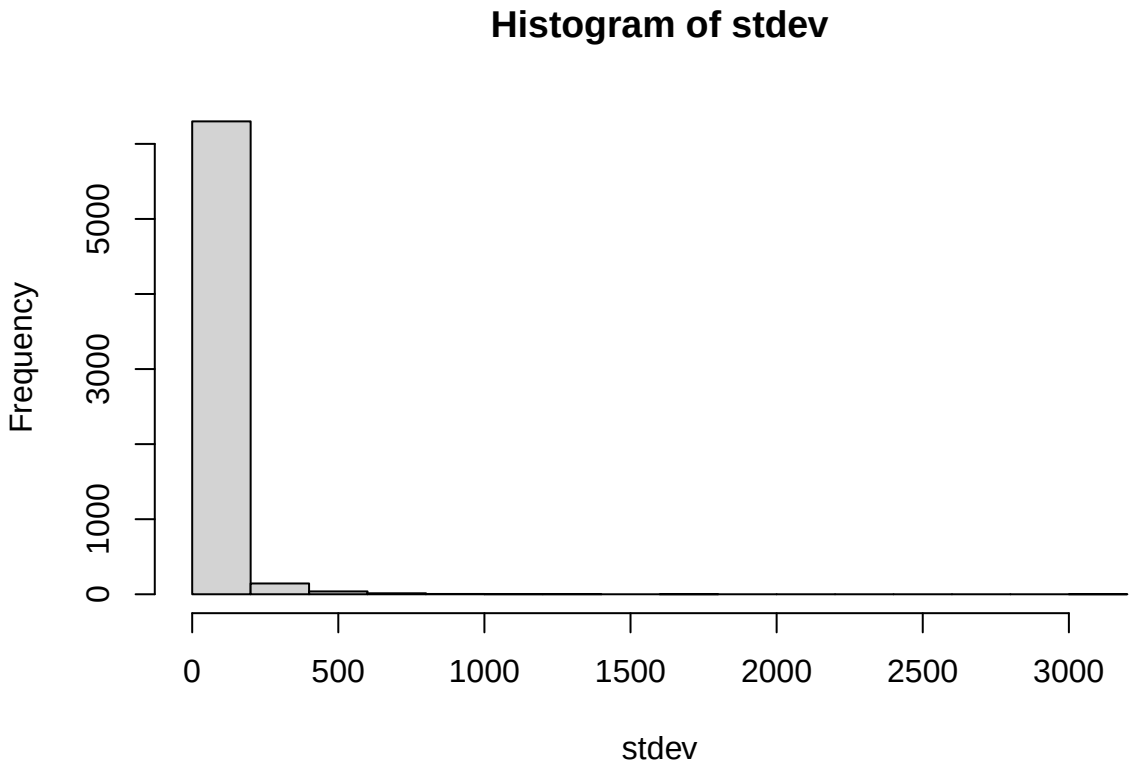
```
# Option 1: Manipulate data and save it in an object then use hist
population_tidy %>%
  group_by(Area) %>%
  summarise(stdev=sd(Population)) ->
  stdev_Area

hist(stdev_Area$stdev)
```




```
# Option 2: Use with()

population_tidy %>%
  group_by(Area) %>%
  summarise(stdev=sd(Population)) %>%
  with(., hist(stdev))
```




```
# When using with() the . specifies the data that is coming through the pipe.
# It says that you want to apply the specific function in the second argument
# using the data specified in the first argument (in this case our summarised
# data frame). This is a useful way to avoid storing objects that will only be used once!`


# Option 3: Use ggplot() - Next week!
```

Question 2

The data contained in `suicides.csv` contains the number of suicides and the population for males vs females at different age groups.

a) Download the file `suicides.csv` and read this into R. (Hint: this is a case where your choice of `read` function makes a big difference to how easy the data is to manipulate.)

```
# assuming that you have set your working directory
# Since the names in the csv file contain special characters
# Use read_csv() from readr to maintain column names
suicides <- readr::read_csv("Suicides.csv")

# Using read_csv from readr will make manipulation much easier further down the line!
# Remember that specifying something that looks like package_name::function() just says
# that I want to call function from the package called package_name.`
```

b) Describe why this data set cannot be described as tidy.

```
# These data are untidy in pretty much every way they can be. Population and Suicide
# should be variables (you would usually use population to standardise!) Gender and
# Age should be in their own cell and these should each be a variable.`
```

c) Using appropriate functions from the `tidyr` package reshape this data set into a tidy format.

```
# The manipulations here need to be done in a sensible order - to us it makes most
# sense to make the data long first, and then force the type column into separate
# columns for population and suicide.`

# Plan of action:
# pivot_longer to move observations into one column
# I can separate gender and age in pivot_longer
# pivot_wider to get separate columns for population and suicide

suicides %>%
  # Perform separate at the same time by specifying multiple names in names_to
  # and a separator in names_sep
  pivot_longer(4:ncol(suicides), names_to=c("Gender","Age"), values_to="Obs", names_sep=":") %>%
  pivot_wider(names_from=Type, values_from=Obs) ->
  suicides_tidy
```

d) Calculate the number of suicides per year

```
suicides_tidy %>%
  group_by(year) %>%
  summarise(suicides=sum(suicides_no))
```

```
## # A tibble: 31 x 2
##     year suicides
##    <dbl>    <dbl>
##  1  1985     5105
##  2  1986     4839
##  3  1987     4594
##  4  1988     4971
##  5  1989     4361
##  6  1990     4643
##  7  1991     4547
##  8  1992     4628
##  9  1993     4462
## 10  1994     4380
## # i 21 more rows
```

e) Add a new column to the data which contains the rate of suicide per 100,000

```
suicides_tidy %>%
  mutate(suicide_rate=suicides_no*100000/population) ->
  # I need to update suicides_tidy to use this column
  suicides_tidy
```

f) Find the average rate of suicide by gender regardless of age. Are there any gender differences?

```
suicides_tidy %>%
  group_by(Gender) %>%
  summarise(mean(suicide_rate))
```

```
## # A tibble: 2 x 2
##   Gender `mean(suicide_rate)`
##   <chr>                <dbl>
## 1 female                3.40
## 2 male                 11.6
```

g) Find the average rate of suicide for each gender/age combination. Are the gender differences consistent across these groups?

```
suicides_tidy %>%
  group_by(Gender, Age) %>%
  summarise(rate=mean(suicide_rate)) %>%
  # I'm going to pivot wider to make comparison easier
  pivot_wider(names_from=Gender, values_from=rate)
```

```
## # A tibble: 6 x 3
##   Age         female   male
##   <chr>        <dbl>  <dbl>
## 1 15-24 years 2.29    9.58
## 2 25-34 years 3.75   15.9
## 3 35-54 years 4.85   16.9
## 4 5-14 years  0.0829  0.129
## 5 55-74 years 4.71   12.3
## 6 75+ years   4.73   14.8
```

```
# Difference between males and females is lower in the 5-14 age group,
# but is generally consistent across the other age groups with suicide
# rates being ~4 times higher for males.`
```