

# Assignment3

Zhijun Li

26/01/2022

## Project Info

Github Username: zazauwu

Student Name: Zhijun Li

Student ID: 20136835

Github Link: <https://github.com/zazauwu/assignment3.git>

## Load the data and required package

```
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

Data <- read.csv("./InputData/FallopiaData.csv")
```

## Check the data

```
str(Data)

## 'data.frame':   123 obs. of  13 variables:
## $ PotNum      : int  1 2 3 5 6 7 8 9 10 11 ...
## $ Scenario    : chr  "low" "low" "low" "low" ...
## $ Nutrients   : chr  "low" "low" "low" "low" ...
## $ Taxon       : chr  "japon" "japon" "japon" "japon" ...
## $ Symphytum   : num  9.81 8.64 2.65 1.44 9.15 ...
## $ Silene      : num  36.4 29.6 36 21.4 23.9 ...
## $ Urtica      : num  16.08 5.59 17.09 12.39 5.19 ...
## $ Geranium    : num  4.68 5.75 5.13 5.37 0 9.05 3.51 9.64 7.3 6.36 ...
## $ Geum        : num  0.12 0.55 0.09 0.31 0.17 0.97 0.4 0.01 0.47 0.33 ...
## $ All_Natives : num  67 50.2 61 40.9 38.4 ...
## $ Fallopia    : num  0.01 0.04 0.09 0.77 3.4 0.54 2.05 0.26 0 0 ...
## $ Total       : num  67.1 50.2 61.1 41.7 41.8 ...
## $ Pct_Fallopia: num  0.01 0.08 0.15 1.85 8.13 1.12 3.7 0.61 0 0 ...
```

```
if (any(is.na(Data) == T)){
  print("There is missing data")
}else{
  print("There is no missing data")}
```

```
## [1] "There is no missing data"
```

## Data Modification

```
Data <- Data %>%
  #remove rows with 'Total' biomass < 60
  filter(Total >= 60) %>%
  #select the columns in the order: 'Total', 'Taxon', 'Senario', 'Nutrients'
  select(Total, Taxon, Scenario, Nutrients) %>%
  #convert the 'Total' column from mg to grams and replace it with 'TotalG'
  transmute(TotalG = Total/1000, Taxon, Scenario, Nutrients)

head(Data)
```

```
##   TotalG Taxon Scenario Nutrients
## 1 0.06706 japon      low      low
## 2 0.06108 japon      low      low
## 3 0.06082 japon     high     high
## 4 0.06674 japon     high     high
## 5 0.06318 japon     high     high
## 6 0.06131 japon     high     high
```

## Custom Function

```
#write a function that will take two inputs
#'x' is any vector of Data to process
myfun <- function(x, calculation, verbose = F){
  #calculate the average value for the column named in 'x'
  if (calculation == "Average"){
    return(mean(x))
    #calculate the sum of values for the column named in 'x'
  } else if (calculation == "Sum"){
    return(sum(x))
    #count the number of observed values for the column named in 'x'
  } else if (calculation == "Observations"){
    return(length(x))
    #otherwise output an error to the user
  } else {
    message("Error! Try 'Average', 'Sum', 'Observations' instead.")
  }
}
```

## Count the total number of observations in the 'Taxon' column

```
myfun(Data$Taxon, "Observations") #should work
```

```
## [1] 45
```

```
myfun(Data$Taxon, "Count") #should output error message
```

```
## Error! Try 'Average', 'Sum', 'Observations' instead.
```

Calculate the average TotalG for each of the two Nutrient concentrations

```
Data %>%  
  group_by(Nutrients) %>%  
  summarise(Average_TotalG = myfun(TotalG, "Average"))
```

```
## # A tibble: 2 x 2  
##   Nutrients Average_TotalG  
##   <chr>          <dbl>  
## 1 high           0.0665  
## 2 low            0.0641
```

Save the New Data

```
write.csv(Data, "./Output/WrangledData.csv")
```