

# Project Info - Assignment 3

Code ▾

- Nadejda Boev (20056079)
- Due Date - 2022/01/26
- Github user - 16nbb1
- Github link - [https://github.com/16nbb1/Biol432\\_A3](https://github.com/16nbb1/Biol432_A3) ([https://github.com/16nbb1/Biol432\\_A3](https://github.com/16nbb1/Biol432_A3))

## Data Input and libraries

Loading in libraries I'll Need

Hide

```
library(ggplot2)
library(tidyverse)
library(dplyr)
library(stringr)
```

Loading in dataset with a relative path

Hide

```
(df = read.csv("../InputData/FallopiaData.csv"))
```

PotNum	Scenario	Nutrients	Taxon	Symphytum	Silene	Urtica	Geranium	Ge...	All_Natives
<int>	<chr>	<chr>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	low	low	japon	9.81	36.36	16.08	4.68	0.12	67.05
2	low	low	japon	8.64	29.65	5.59	5.75	0.55	50.18
3	low	low	japon	2.65	36.03	17.09	5.13	0.09	60.99
5	low	low	japon	1.44	21.43	12.39	5.37	0.31	40.94
6	low	low	japon	9.15	23.90	5.19	0.00	0.17	38.41
7	low	low	japon	6.31	24.40	7.00	9.05	0.97	47.73
8	low	low	japon	19.53	29.93	0.00	3.51	0.40	53.37
9	low	low	japon	6.08	18.25	8.44	9.64	0.01	42.42
10	low	low	japon	5.69	35.23	4.84	7.30	0.47	53.53
11	low	low	japon	15.58	20.81	2.81	6.36	0.33	45.89

1-10 of 123 rows | 1-10 of 13 columns

Previous 1 2 3 4 5 6 ... 13 Next

## Data Wrangling

1. Removing rows with 'Total' biomass < 60
  - Filtering for rows that have a biomass of 60 or less
2. Make a new column TotalG, which converts the 'Total' column from mg to grams AND replace Total with TotalG, and add it to the dataset.
  - Removing Total using NULL
3. Reorder the columns so that they are in the order: 'TotalG', 'Taxon', 'Scenario', 'Nutrients' AND Remove the other columns

Hide

```
df_filt = df %>%
  filter(Total <= 60) %>%
  mutate(TotalG = Total / 1000, Total = NULL) %>%
  select(TotalG, Taxon, Scenario, Nutrients)
```

## Function: basics

Input:

- A vector of data to process (e.g. column from a data.frame object)
- A string that defines what calculation to perform.
  - Options: "Average", "Sum", "Observations"
  - All other inputs receive the following message: "ERROR input"

Output:

- Based on the above command, perform the command with the column provided
- Count the total number of observations in the 'Taxon' column ONLY IF a Taxon column exists
- Calculate the average TotalG for each of the two Nutrient concentrations ONLY IF a TotalG column exists
- Return an error messages if input string does not perfectly match or if columns required do not exist in the dataframe provided

Hide

```
basics <- function (in_comm, in_vec, dataframe) {

  # IF statement based on command, includes the possibility for NA in columns
  if(in_comm == 'Average') {
    print(paste0('Average: ', mean(in_vec, na.rm=T)))
  } else if (in_comm == 'Sum') {
    print(paste0('Sum: ', sum(in_vec, na.rm=T)))
  } else if (in_comm == 'Observations') {
    print(paste0('Number of observations: ',length(in_vec)))
  } else {
    print('INPUT ERROR')
  }

  # IF statement for counting number of Taxon observations, if this column exists
  if(any(str_detect(names(dataframe), "Taxon")) == TRUE) {
    print(paste0('Number of observations in Taxon column: ',length(dataframe$Taxon)))
  } else {
    print('ERROR : No taxon column')
  }

  # IF statement for for groupby average by nutrient, the column TotalG must exist
  if(any(str_detect(names(dataframe), "TotalG")) == TRUE) {
    groupby_nut = dataframe %>%
      group_by(Nutrients) %>%
      summarize(Mean = mean(TotalG))

    # Printing out each average (low and high nutrients)
    print(paste0('Average biomass for high nutrients: ', signif(groupby_nut[groupby_nut$Nutrients == 'high',]$Mean, digits = 3), 'g'))
    print(paste0('Average biomass for low nutrients: ', signif(groupby_nut[groupby_nut$Nutrients == 'low',]$Mean, digits = 3), 'g'))

  } else {
    print('ERROR : No TotalG column')
  }

}
```

## Testing inputs

Successful cases

Hide

```
basics('Average', df_filt$TotalG, df_filt)
```

```
[1] "Average: 0.0505489743589744"
[1] "Number of observations in Taxon column: 78"
[1] "Average biomass for high nutrients: 0.0513g"
[1] "Average biomass for low nutrients: 0.0487g"
```

Hide

```
basics('Sum', df_filt$TotalG, df_filt)
```

```
[1] "Sum: 3.94282"
[1] "Number of observations in Taxon column: 78"
[1] "Average biomass for high nutrients: 0.0513g"
[1] "Average biomass for low nutrients: 0.0487g"
```

Hide

```
basics('Observations', df_filt$TotalG, df_filt)
```

```
[1] "Number of observations: 78"  
[1] "Number of observations in Taxon column: 78"  
[1] "Average biomass for high nutrients: 0.0513g"  
[1] "Average biomass for low nutrients: 0.0487g"
```

Other cases

Hide

```
# Spelling error  
basics('Observaions', df_filt$TotalG, df_filt)
```

```
[1] "INPUT ERROR"  
[1] "Number of observations in Taxon column: 78"  
[1] "Average biomass for high nutrients: 0.0513g"  
[1] "Average biomass for low nutrients: 0.0487g"
```

Hide

```
# Throwing error when wrong dataframe is used (df does not have a TotalG column)  
basics('Observations', df_filt$TotalG, df)
```

```
[1] "Number of observations: 78"  
[1] "Number of observations in Taxon column: 123"  
[1] "ERROR : No TotalG column"
```

## Saving the “clean” dataset

Hide

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