EDUC 641 Lab: Applied Statistics in Education and Human Services I

Lab 6: 11/1 and 11/2

## Focus on preparing the dataset using ‘ dplyr()‘ verbs: select(), filter(), mutate() and creating standardized scores using mutate()

**Goals:**

1. **Use select() to limit the dataframe to the main variables you want to look at**
2. **Use mutate() to create a *new* standardized variable/ column or change existing columns**
3. **Use filter() with the pipe operate (%>%)**

***What’s NEW?!***

**Preparing dataset for analysis:**

**mutate()** allows you to add new columns (i.e., variables) or update existing ones.

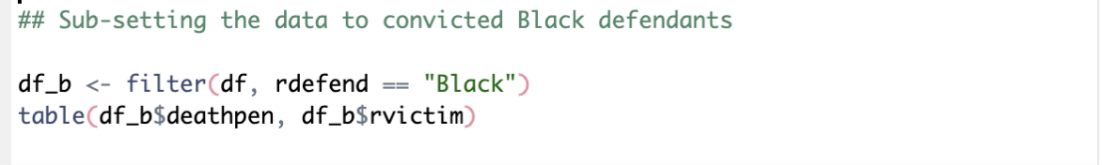
**select()** Used to choose and reorder specific variables of interest from a data frame.

**filter()** subsets rows of a data frame based on specified conditions (as seen weeks 4&5)

A close-up of text

Description automatically generated

\*\* Note that filter() can be used with %>% (pipe operator). Previously, we used filter like this -



**Difference between + and %>%**

\*\* Both %>% (pipe operator) and + (plus operator) are used to write multiple lines of code together. They allow you to do a sequence of things. However, they work with different packages.

**Use %>% when using dplyr() functions and + when using ggplot2() functions.**

**Using mutate() to make standardized z\_scores for a continuous variable:**

**Example:**

A screen shot of a computer code

Description automatically generated**\*\*^^ Notice** the **scale()** function in the second example

Also, remember that **sd()** is the standard deviation function, so we’re subtracting the mean and dividing by the standard deviation to standardize the *life\_expectancy* variable.

**Worksheet Don’t forget to load packages: library(**tidyverse**) and read the life\_expectancy dataset.**

1. **Use select() to isolate the variables of interest: country, year, BMI**
   1. your\_data <- your\_data %>%

**select(**variable\_1, variable\_2, variable\_3**)**

1. **Try using the mutate() function to store the standardized scores of BMI as a new variable in the same dataset.** 
   1. your\_data <- your\_data %>%

**mutate(**zscore\_ variable = **scale(**your\_continuous\_variable**))**

1. **Use filter() to look at rows where the zscore\_variable is 2 standard deviations above the mean (zscore\_variable > 2). How many such rows are there?**
   1. your\_data <- your\_data **%>% # (first pipe to the filter() function)**

**mutate(**zscore\_ variable = **scale(**your\_continuous\_variable**)) %>% #(second pipe)**

**filter(**zscore\_variable> 2**)**

1. **Let’s try one more example.** This time select country, year, and a continuous variable of your choice. Calculate the standardized score for the continuous variable and filter to rows where z-scores is 2 SD below the mean (z\_score < -2). **Challenge: do everything using pipe in one go.**