**Introduction**

For this activity, you will be manipulating professional sumo wrestling data from over decades of tournaments. Using the “dplyr” package to transform and subset the data into more meaningful displays, you will learn how to make a multitude of insights more accessible and readily apparent. Implementing functions and expressions as efficiently as possible will provide the most impactful learning experience.

**Learning Goals**

By the end of the activity, you will gain an understanding of the dplyr verbs:

* filter
* slice
* mutate
* summarize
* group\_by
* arrange

**Data**

The data below shows the first six entries of the professional Japanese sumo wrestling data set that is utilized for this activity. Containing all records from 1957 to mid 2023, the results of individual wrestlers from the elite Makuuchi Banzuke division are displayed for each Basho (tournament). Several other important variables, explained in the data description, also accompany each wrestlers’ appearances in the data set. The full data (sumo\_since\_1957.csv) is available on the GitHub repository associated with this module.

**A picture containing text, font, screenshot, white

Description automatically generated**

**Methods**

For this activity, students will use the “dplyr” package in r studio to execute data transformations. Primarily using the filter, mutate, slice, summarize, group\_by, and arrange functions, students will likely make use of a the dplyr cheat sheet to determine which other expressions and functions will be needed.

**Excercises**

1. a. Subset the data to contain wrestlers from Mongolia only.

sumo%>%

filter(Shusshin == "Mongolia")

b. Using one line of code, subset the full data to contain wrestlers from Mongolia or Tokyo only.

sumo%>%

filter(Shusshin == "Mongolia" | Shusshin == "Tokyo")

1. c. Using one line of code, subset the full data to contain wrestlers from Mongolia who train at Kataonami.

sumo%>%

filter(Shusshin == "Mongolia" & Heya == "Kataonami")

1. a. Extract rows 25 through 85 from the full data set.

sumo%>%

slice(25:85)

b. Identify the rows with the single highest and lowest recorded weights of wrestlers. Show only the top result for both lines.

sumo%>%

slice\_max(weight\_kg, with\_ties = FALSE)

sumo%>%

slice\_min(weight\_kg, with\_ties = FALSE)

1. Create a new variable that displays the wrestlers’ win percentage for each appearance in a Basho.

sumo%>%

mutate(win\_percentage = (wins/(wins+losses+ties))\*100)

1. Calculate how many times each wrestler appears in the data set.

sumo%>%

group\_by(Rikishi)%>%

summarise(total\_appearances = n())

1. Calculate the mean weight for each wrestler

sumo%>%

group\_by(Rikishi)%>%

summarise("mean weight" = mean(weight\_kg))

1. Calculate the total losses for each wrestler and order the resulting data from least to most.

sumo%>%

group\_by(Rikishi)%>%

summarise(total\_losses = sum(losses))%>%

arrange(total\_losses)

1. Calculate the total wins for each wrestler and order the resulting data from most to least.

sumo%>%

group\_by(Rikishi)%>%

summarise(total\_wins = sum(wins))%>%

arrange(desc(total\_wins))

1. Calculate the overall win percentage of each wrestler and arrange from highest to lowest.

sumo%>%

group\_by(Rikishi)%>%

summarise(total\_win\_percetage = mean(wins/(wins+losses+ties))\*100)%>%

arrange(desc(total\_win\_percentage))

OR

sumo%>%

mutate(win\_percentage = (wins/(wins+losses+ties))\*100)%>%

group\_by(Rikishi)%>%

summarise(total\_win\_percentage = mean(win\_percentage))%>%

arrange(desc(total\_win\_percentage))