**dplyr : A Grammar of Data Manipulation**

**Basic functions of “dplyr”**

|  |  |
| --- | --- |
| **select()** | Selection of specific columns |
| **filter()** | Filtering data based on condition |
| **mutate()** | Creating new variables/columns |
| **arrange()** | Sorting dataset |
| **na.omit()** | Remove rows with missing value. |

* Loading “dplyr” package and importing data.

|  |
| --- |
| **library(dplyr)**  **classf <- read.csv('http://s.anilz.net/wb\_class') #World Bank country classification**  **energy <- read.csv('http://s.anilz.net/wb\_energy') #World Bank energy dataset**  **var\_def <- read.csv('http://s.anilz.net/wb\_var\_def') #Variables definition** |

* Examples of basic functions of “dplyr” package.
  1. Selecting columns [Syntax : **select(data, column1, column2, …..)**

|  |
| --- |
| **View(energy)**  **data\_select <- select(energy, country, ccode, year, tfec)**  **View(data\_select)** |

* 1. Filtering data [Syntax : **filter(data, condition1, condition2, ...) ]**

|  |
| --- |
| **data\_nepal <- filter(energy, country == "Nepal")**  **View(data\_nepal)** |

* 1. Create new columns [Syntax : **mutate(data, new\_col = expression, ...)]**

|  |
| --- |
| **data\_ren\_ele\_share <- mutate(energy, ren\_ele\_share = ren\_ele/tot\_ele\*100)**  **View(data\_ren\_ele\_share)** |

* 1. Sorting data [Syntax : **arrange(data, col1, col2, ….)]**

|  |
| --- |
| **data\_sort <- arrange(energy,year,desc(country))**  **View(data\_sort)** |

* 1. Remove rows with missing values [Syntax : **na.omit(data)]**

|  |
| --- |
| **data\_na\_omit <- na.omit(energy)**  **View(data\_na\_omit)** |

**Advance functions of “dplyr”**

* **Piping (%>%)** : Piping is used for chaining multiple operations together in a clean way.

*Example:* Suppose you are interested in renewable electricity output data in Nepal and India. Now, you want to perform the following operations with the help of piping (%>%).

* Select columns **year, country, ren\_ele, tot\_ele** from **energy** dataframe.
* Keep data of Nepal and India only.
* Sort the dataframe according to **country** and **year** columns.
* Create a new column **ren\_ele\_share** by calculating share of renewable electricity output in total output (i.e. **ren\_ele/tot\_ele\*100**).
* Save the new dataframe as **energy\_np\_in**

|  |
| --- |
| **energy\_np\_in <- energy %>%**  **select(year, country, ren\_ele, tot\_ele) %>%**  **filter(country == 'Nepal' | country =='India') %>%**  **arrange(country, year) %>%**  **mutate(ren\_ele\_share = ren\_ele/tot\_ele\*100)**  **View(energy\_np\_in)** |

* Summarizing by categories using **group\_by()** and **summarize()** functions.

*Example:* Suppose now you want to summarize the dataframe **energy\_np\_in** by calculating max, min, and average values of **ren\_ele\_share** in Nepal and India and save summarized dataframe as **energy\_summary**.

|  |
| --- |
| **energy\_summary <- energy\_np\_in %>%**  **na.omit() %>%**  **group\_by(country) %>%**  **summarize(max = max(ren\_ele\_share),**  **min = min(ren\_ele\_share),**  **mean = mean(ren\_ele\_share))**  **View(energy\_summary)** |

*Practice :* Let’s summarize the dataframe **energy** by calculating max, min, and average values of **ele\_total** [Access to electricity (% of total population)] for each country.

* **Merging dataframes** using join functions.

inner\_join() Return rows with matching keys in both data frames

left\_join() Return all rows from first data frame, matching rows from second

right\_join() Return all rows from second data frame, matching rows from first

full\_join() Return all rows from both data frames, matching by keys

semi\_join() Return rows from first data frame with matching keys in second

anti\_join() Return rows from first data frame without matching keys in second

|  |
| --- |
| **df1 <- data.frame(id = c(1, 2, 3), colA = c("A", "B", "C"))**  **df2 <- data.frame(id = c(1, 3, 5), colB = c("X", "Y", "Z"))**  **print(df1)**  **print(df2)**  **inner\_join(df1, df2, by = 'id')**  **left\_join(df1, df2, by = 'id')**  **right\_join(df1, df2, by = 'id')**  **full\_join(df1, df2, by = 'id')**  **semi\_join(df1, df2, by = 'id')**  **anti\_join(df1, df2, by = 'id')** |

*Practice :*

* Let’s left\_join dataframes **energy** and **classf** by common column **ccode**.
* Summarize by calculating average values of **ele\_total** [Access to electricity (% of total population)] for each **year** and **country** group (i.e., H, UM, LM, L).
* Save the summarized dataframe as **wb\_energy**.

|  |
| --- |
| **wb\_energy <- left\_join(energy, classf, by = 'ccode') %>%**  **na.omit() %>%**  **group\_by(wb\_class) %>%**  **summarize(average = mean(ele\_total))**  **View(wb\_energy)** |