Apply the Same Function over Columns and Row Groups

Fan Wang

2023-03-09

Contents

1 Generate Replace Variables

Go to the **RMD**, **R**, **PDF**, or **HTML** version of this file. Go back to fan's REconTools research support package, R4Econ examples page, PkgTestR packaging guide, or Stat4Econ course page.

1.1 Sum Across Columns

We compute sum over several variables in the mtcars dataset. We will sum over several variables with shared prefix, after adding these prefix first. We introduce an NA value to make sure that we can sum ignoring NA

We sum using three different methods below: (1) purrr:reduce(), (2) base::rowSums(), (3) Manual sum. Note that the rowSums option is able to sum ignoring NA.

```
# we introduce NA value to first row
mtcars[1,1] <- NA
# Rename variables, and sum across
mtcars_rowsum <- mtcars %>%
    rename(stats_mpg = mpg, stats_cyl = cyl, stats_hp = hp) %>%
    mutate(
        cs_reduce = purrr::reduce(
            dplyr::pick(contains("stats")),
            `+`
        ),
        cs_rowsum = base::rowSums(
            dplyr::pick(contains("stats")),
            na.rm = TRUE
        ),
        cs_manual = stats_mpg + stats_cyl + stats_hp
    ) %>%
    select(matches("stats|cs"), gear)
# Display
st_caption <- "sum across columns"
kable(mtcars_rowsum %>% slice_head(n = 5),
    caption = st_caption
) %>% kable_styling_fc_wide()
```

sum across columns

	stats_mpg	stats_cyl	stats_hp	cs_reduce	cs_rowsum	cs_manual	gear
Mazda RX4	NA	6	110	NA	116.0	NA	4
Mazda RX4 Wag	21.0	6	110	137.0	137.0	137.0	4
Datsun 710	22.8	4	93	119.8	119.8	119.8	4
Hornet 4 Drive	21.4	6	110	137.4	137.4	137.4	3
Hornet Sportabout	18.7	8	175	201.7	201.7	201.7	3

See this discussion for column sum peed comparisons.

1.2 Sum Across Rows within Group

Following from the prior section, we now sum across rows within group.

```
# we introduce NA value to first row
# mtcars[1,1] <- NA
# Rename variables, and sum across
mtcars_grpsum <- mtcars_rowsum %>%
    arrange(gear) %>% group_by(gear) %>%
    # srs = sum row sum
    mutate_at(vars(matches("stats|cs")),
        .funs = list(gs = ~sum(., na.rm=TRUE))
    ) %>%
    select(gear, matches("gs")) %>%
    slice_head(n=1)
# Display
st_caption <- "gs = group sum, cs = col sum over the columns with stats as prefix, sum across rows afte
kable(mtcars_grpsum ,
    caption = st_caption
) %>% kable_styling_fc_wide()
```

\begin{table}[!h]

\caption{gs = group sum, cs = col sum over the columns with stats as prefix, sum across rows after col sum; gear = 4 difference for cs_rowsum_gs because it allowed for summing ignoring NA for values across columns}

gear	stats_mpg_gs	stats_cyl_gs	stats_hp_gs	cs_reduce_gs	cs_rowsum_gs	cs_manual_gs
3	241.6	112	2642	2995.6	2995.6	2995.6
4	273.4	56	1074	1287.4	1403.4	1287.4
5	106.9	30	978	1114.9	1114.9	1114.9

\end{table}

Now, we sum across rows within group. ## Replace NA for Multiple Variables

Replace some variables NA by some values, and other variables' NAs by other values.

```
# Define
it_N <- 3
it_M <- 5
svr_id <- "date"

# NA dataframe, note need to define as NA_real_
# if define as NA, will not be able to replace with 99 column
# would be logical rather than double.
df_NA <- as_tibble(matrix(NA_real_, nrow = it_N, ncol = it_M)) %>%
```

```
    rowid_to_column(var = svr_id) %>%
    rename_at(
        vars(starts_with("V")),
        funs(str_replace(., "V", "var"))
    )
kable(df_NA) %>%
    kable_styling_fc()
```

date	var1	var2	var3	var4	var5
1	NA	NA	NA	NA	NA
2	NA	NA	NA	NA	NA
3	NA	NA	NA	NA	NA

```
# Replace NA
df_NA_replace <- df_NA %>%
    mutate_at(vars(one_of(c("var1", "var2"))), list(~ replace_na(., 0))) %>%
    mutate_at(vars(one_of(c("var3", "var5"))), list(~ replace_na(., 99)))

kable(df_NA_replace) %>%
    kable_styling_fc()
```

date	var1	var2	var3	var4	var5
1	0	0	99	NA	99
2	0	0	99	NA	99
3	0	0	99	NA	99

1.3 Cumulative Sum Multiple Variables

Each row is a different date, each column is the profit a firms earns on a date, we want to compute cumulatively how much a person is earning. Also renames variable names below jointly.

```
# Define
it_N <- 3
it_M <- 5
svr_id <- "date"

# random dataframe, daily profit of firms
# dp_fx: daily profit firm ID something
set.seed(123)
df_daily_profit <- as_tibble(matrix(rnorm(it_N * it_M), nrow = it_N, ncol = it_M)) %>%
    rowid_to_column(var = svr_id) %>%
    rename_at(
        vars(starts_with("V")),
        funs(str_replace(., "V", "dp_f"))
    )
kable(df_daily_profit) %>%
    kable_styling_fc()
```

date	dp_f1	dp_f2	dp_f3	dp_f4	dp_f5
1	-0.5604756	0.0705084	0.4609162	-0.4456620	0.4007715
2	-0.2301775	0.1292877	-1.2650612	1.2240818	0.1106827
3	1.5587083	1.7150650	-0.6868529	0.3598138	-0.5558411

```r
# cumulative sum with suffix
df_cumu_profit_suffix <- df_daily_profit %>%
    mutate_at(vars(contains("dp_f")), .funs = list(cumu = ~ cumsum(.)))
kable(df_cumu_profit_suffix) %>%
    kable_styling_fc_wide()
```

date	dp_f1	dp_f2	dp_f3	dp_f4	dp_f5	dp_f1_cumu	dp_f2_cumu	dp_f3_cumu	dp_f4_cumu	dp_f5_cumu
1	-0.5604756	0.0705084	0.4609162	-0.4456620	0.4007715	-0.5604756	0.0705084	0.4609162	-0.4456620	0.4007715
2	-0.2301775	0.1292877	-1.2650612	1.2240818	0.1106827	-0.7906531	0.1997961	-0.8041450	0.7784198	0.5114542
3	1.5587083	1.7150650	-0.6868529	0.3598138	-0.5558411	0.7680552	1.9148611	-1.4909979	1.1382337	-0.0443870

```r
# cumulative sum variables naming to prefix
df_cumu_profit <- df_cumu_profit_suffix %>%
    rename_at(vars(contains("_cumu")), list(~ paste("cp_f", gsub("_cumu", "", .), sep = ""))) %>%
    rename_at(vars(contains("cp_f")), list(~ gsub("dp_f", "", .)))
kable(df_cumu_profit) %>%
    kable_styling_fc_wide()
```

date	dp_f1	dp_f2	dp_f3	dp_f4	dp_f5	cp_f1	cp_f2	cp_f3	cp_f4	cp_f5
1	-0.5604756	0.0705084	0.4609162	-0.4456620	0.4007715	-0.5604756	0.0705084	0.4609162	-0.4456620	0.4007715
2	-0.2301775	0.1292877	-1.2650612	1.2240818	0.1106827	-0.7906531	0.1997961	-0.8041450	0.7784198	0.5114542
3	1.5587083	1.7150650	-0.6868529	0.3598138	-0.5558411	0.7680552	1.9148611	-1.4909979	1.1382337	-0.0443870