

# Mesh join

Fan Wang

2023-12-24

## Contents

1	Mesh Join	1
1.1	Expand Multiple Rows with the Same Expansion	1

## 1 Mesh Join

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 Expand Multiple Rows with the Same Expansion

File A is at ID x Week x DayOfWeek level, file B is at ID x DayOfWeek x Product. Product is the product ID bought, could also store other info on product as additional variables. We want to create a file that is at ID x Week x DayOfWeek x Product level.

The idea is that products bought made on Monday by household 1, for example are always the same, and file A records a “shopping-record”, which week, which day each household went shopping.

We do not store in file A what was bought because a particular household always buys the same thing on the same day of the week. We store data in A and B separately to save space since all products by the same household on the same day of week is always identical.

However, we need to join the two files together compute how many units of each product all households bought throughout some timeframe.

Step 1, construct File A, by fully messing ID, Week and Day of Week. In the simulated file below, household 1 shopped 3 times, twice on 3rd day of week, once on 2nd day of week, across two weeks. Household 2 shopped once, on the 3rd day of week.

```
# Mesh
ar_st_varnames <- c('hh','week','dayofweek')
ar_it_ids <- c(1,2)
ar_it_weeks <- c(1,2)
ar_it_daysofweek <- c(1,2,3)
df_idwkday_mesh <- tidyr::expand_grid(
  ar_it_ids, ar_it_weeks, ar_it_daysofweek) %>%
  rename_all(~c(ar_st_varnames))

# Randomly drop a subset of rows
# Different subset of ID and Week for each DayOfWeek.
it_M <- 4
set.seed(456)
df_idwkday_mesh <- df_idwkday_mesh[sample(dim(df_idwkday_mesh)[1], it_M, replace=FALSE),] %>%
```

File A (ID x Week x DayOfWeek)

hh	week	dayofweek
1	1	3
1	2	2
1	2	3
2	2	3

File B (ID x DayOfWeek x Product)

hh	dayofweek	product
1	1	12
1	2	14
1	3	10
1	3	13
1	3	14
2	1	12
2	1	13
2	2	11

```

arrange(!!!syms(ar_st_varnames))

# Display
st_caption <- "File A (ID x Week x DayOfWeek)"
kable(df_idwkday_mesh, caption=st_caption) %>% kable_styling_fc()

```

Step 2, construct File B. We have shopping list for the 1st household on shopping from 1st, 2nd, and 3rd days of a week. We have a shopping list for 2nd household only for shopping on the 1st and 2nd day.

```

# Generate day of week specific product file
ar_st_varnames <- c('hh', 'dayofweek', 'product')
ar_it_product <- c(10,11,12,13,14)
df_dayproduct_mesh <- tidyr::expand_grid(
  ar_it_ids, ar_it_daysofweek, ar_it_product) %>%
  rename_all(~c(ar_st_varnames))

# Make each day product list not identical
it_M <- 8
set.seed(123)
df_dayproduct_mesh <- df_dayproduct_mesh[sample(dim(df_dayproduct_mesh)[1], it_M, replace=FALSE),] %>%
  arrange(!!!syms(ar_st_varnames))

# Display
st_caption <- "File B (ID x DayOfWeek x Product)"
kable(df_dayproduct_mesh, caption=st_caption) %>% kable_styling_fc()

```

Step 3. we combine files A and B together via `dplyr::left_join`.

Given the sample files we have constructed we have:

- multiple items in shopping list for household 1 on day 3
- no shopping list for household 2 on day 3
- shopping list available on days that do not appear on shopping days tracking list

When we `left_join`, we do not include in combined file shopping list from days for households not in the

File C, left-join (ID x Week x DayOfweek x Product)

hh	week	dayofweek	product
1	1	3	10
1	1	3	13
1	1	3	14
1	2	2	14
1	2	3	10
1	2	3	13
1	2	3	14
2	2	3	NA

tracking list. Note that from the output below, we achieved several things:

- the day 3 shopping list for household 1 is merged in twice, to household's trips on day 3 in both week 1 and 2, rows expanded because 3 items bought on each day
- the day 2 shopping list for household 1 is merged in once, there are no row-expansion, since there was one item bought on this shopping list
- the day 3 shopping list for household 2 is not merged in, since the shopping list does not exist, but the row remains.

```
# left join
df_left_join <- df_idwday_mesh %>%
  left_join(df_dayproduct_mesh,
    by= c('hh'='hh', 'dayofweek'='dayofweek'))
# Display left-join
st_caption <- "File C, left-join (ID x Week x DayOfweek x Product)"
kable(df_left_join, caption=st_caption) %>% kable_styling_fc()
```

Step 4, now, we also try `dplyr::full_join`. Note that the full-join result is not what we want, it added shopping list by household to the file, but these shopping lists were un-realized, since the households did not shop in any week on those days. So our desired result is achieved by `dplyr::left_join`.

```
# full join
df_full_join <- df_idwday_mesh %>%
  full_join(df_dayproduct_mesh,
    by= c('hh'='hh', 'dayofweek'='dayofweek'))
# Display full-join
st_caption <- "File C, full-join (ID x Week x DayOfweek x Product)"
kable(df_full_join, caption=st_caption) %>% kable_styling_fc()
```

File C, full-join (ID x Week x DayOfweek x Product)

hh	week	dayofweek	product
1	1	3	10
1	1	3	13
1	1	3	14
1	2	2	14
1	2	3	10
1	2	3	13
1	2	3	14
2	2	3	NA
1	NA	1	12
2	NA	1	12
2	NA	1	13
2	NA	2	11