[r-bloggers.com](https://www.r-bloggers.com/padr-is-updated/)

**padr is updated**

3-4 minutes

[This article was first published on [**That’s so Random**](https://edwinth.github.io/blog/padr-v-0-5-0/), and kindly contributed to [R-bloggers](https://www.r-bloggers.com/)]. (You can report issue about the content on this page [here](https://www.r-bloggers.com/contact-us/))

Want to share your content on R-bloggers? [click here](https://www.r-bloggers.com/add-your-blog/) if you have a blog, or [here](https://r-posts.com/) if you don't.

Yesterday v.0.5.0 of the padr package hit CRAN. You will find the main changes in the thicken function, that has gained two new arguments. First of all, by an idea of Adam Stone, you are now enabled to drop the original datetime variable from the data frame by using drop = TRUE. This argument defaults to FALSE to ensure backwards compatibility. Without setting drop to TRUE the datetime variable will be returned alongside the added, thickened variable:

library(padr)

thicken(coffee, interval = "hour")

## time\_stamp amount time\_stamp\_hour

## 1 2016-07-07 09:11:21 3.14 2016-07-07 09:00:00

## 2 2016-07-07 09:46:48 2.98 2016-07-07 09:00:00

## 3 2016-07-09 13:25:17 4.11 2016-07-09 13:00:00

## 4 2016-07-10 10:45:11 3.14 2016-07-10 10:00:00

Now, with drop = TRUE the original datetime variable will no longer be returned:

thicken(coffee, interval = "hour", drop = TRUE)

## amount time\_stamp\_hour

## 1 3.14 2016-07-07 09:00:00

## 2 2.98 2016-07-07 09:00:00

## 3 4.11 2016-07-09 13:00:00

## 4 3.14 2016-07-10 10:00:00

Secondly, thicken has gained the ties\_to\_earlier argument. By default when the rounding argument in thicken is set to “up” and the original observation is equal to a value in the higher interval variable, the observation is mapped to the next value in the new variable. (For example 2019-04-14 13:00:00 would be mapped to 2019-04-14 14:00:00 when rounding is “up” and interval is “hour”.) This can be undesired. When this argument is set to TRUE tied observations are mapped to their own value (thus to one value earlier in the new variable). For completeness this argument also works when rounding is “down”. Then, when original and new value are equal, the original value is mapped to the previous value of the higher level interval variable. (For example 2019-04-14 13:00:00 will be mapped to 2019-04-14 12:00:00 when the interval is hour). Feature request by github user stribstrib.

Finally, along some minor bug fixes, there is a major bug fix that was reported by github user levi-nagy. thicken preserves missing values; missing values in the original datetime column are also found in the newly added variable. The missing values were placed on the wrong position however. They were placed on the original position + the number of NAs already seen in the original datetime variable, instead of on the NA position where they are supposed to be. Only the first missing value was on the correct position, all the others had an unwanted offset. This is now fixed, all the missing values are in the correct place in the thickened variable.