data.table

Tracey S. Frescino

Aug. 18, 2016

I am making this up

Help Links

http://user2014.stat.ucla.edu/files/tutorial_Matt.pdf http://datatable.r-forge.r-project.org/datatable-intro.pdf

Initialize R Environment

```
# Set working directory.
setwd("C:/_tsf/_GitHub/help/datatable")

# Set options bias for scientific notation
options(scipen=6)

# Load libraries
require(data.table)
```

```
## Loading required package: data.table
```

```
require(microbenchmark)
```

```
## Loading required package: microbenchmark
```

```
require(dplyr)
```

```
## Loading required package: dplyr

## ------------------------------------------------------------------------

## data.table + dplyr code now lives in dtplyr.
## Please library(dtplyr)!

## ------------------------------------------------------------------------

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:data.table':
##
##     between, last
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# Read in Wyoming tree dataset to use as a bit data set to test with.
system.time(tree.df <- read.csv("data/WYtree.csv", header=TRUE))
```

```
##    user  system elapsed
##    0.82    0.00    0.83
```

```
class(tree.df)
```

```
## [1] "data.frame"
```

```
# Converting a data frame to a data.table
# setDT converts lists and data.frames to data.tables by reference (no copy is made)
microbenchmark(
    tree.dt <- data.table(tree.df),
    tree.dt <- setDT(tree.df),
    tree.dt <- setDT(tree.df, key="PLT_CN")
)
```

```
## Unit: microseconds
##                                       expr      min        lq       mean
##             tree.dt <- data.table(tree.df) 3293.203 3447.9695 9341.63109
##                  tree.dt <- setDT(tree.df)   46.389   52.9575   91.73974
##  tree.dt <- setDT(tree.df, key = "PLT_CN")  321.439  352.0230  437.53417
##     median       uq       max neval
##  4434.4515 4929.746 55389.181   100
##    64.6580   75.947  2329.709   100
##   387.5335  418.322  3732.871   100
```

```
## fread vs read.csv
## read.csv() reads rows into memory as character and then tries to convert them to integer and factors
## freed() reads everything as character

# Create a data.table from tree.df
system.time(tree.dt <- fread("data/WYtree.csv"))
```

```
##    user  system elapsed
##    0.28    0.00    0.28
```

```
class(tree.dt)
```

```
## [1] "data.table" "data.frame"
```

```
head(tree.dt)
```

```
##             PLT_CN CONDID SUBP TREE STATUSCD SPCD SPGRPCD  DIA HT ACTUALHT
## 1: 282479222489998      1    1    1        1   19      12  9.9 56       56
## 2: 282479222489998      1    1    2        2  108      21 11.2 62       62
## 3: 282479222489998      1    1    3        2  101      24 14.0 68       68
## 4: 282479222489998      1    1    4        2  101      24 12.0 69       69
## 5: 282479222489998      1    1    5        2  101      24 12.9 63       63
## 6: 282479222489998      1    1    6        2  101      24  6.2 31       31
##    HTCD TREECLCD CR CCLCD AGENTCD CULL DECAYCD STOCKING WDLDSTEM UNCRCD
## 1:    1        2 55     3      NA   10      NA    1.556       NA     60
## 2:    1        3 NA    NA      NA   35       3    0.000       NA     NA
## 3:    1        3 NA    NA      NA   25       2    0.000       NA     NA
## 4:    1        3 NA    NA      NA   30       2    0.000       NA     NA
## 5:    1        3 NA    NA      NA   30       3    0.000       NA     NA
## 6:    1        3 NA    NA      NA   35       3    0.000       NA     NA
##    BHAGE TOTAGE MIST_CL_CD STANDING_DEAD_CD PREV_STATUS_CD PREV_WDLDSTEM
## 1:     0      0          0                NA             NA            NA
## 2:     0      0          0                 1             NA            NA
## 3:     0      0          0                 1             NA            NA
## 4:     0      0          0                 1             NA            NA
## 5:     0      0          0                 1             NA            NA
## 6:     0      0          0                 1             NA            NA
##    RECONCILECD PREVDIA  VOLCFNET  VOLCFGRS FGROWCFAL FMORTCFAL FREMVCFAL
## 1:          NA      NA 10.713023 11.903359  0.239587         0         0
## 2:          NA      NA 13.248148 20.381766  0.000000         0         0
## 3:          NA      NA 24.678143 32.904190  0.000000         0         0
## 4:          NA      NA 17.676287 25.251838  0.000000         0         0
## 5:          NA      NA 18.541419 26.487742  0.000000         0         0
## 6:          NA      NA  1.487479  2.288429  0.000000         0         0
##    TPA_UNADJ TPAGROW_UNADJ TPAMORT_UNADJ TPAREMV_UNADJ  CARBON_BG
## 1:  6.018046      6.018046             0             0  34.573198
## 2:  6.018046      6.018046             0             0  48.078692
## 3:  6.018046      6.018046             0             0 105.259606
## 4:  6.018046      6.018046             0             0  76.234960
## 5:  6.018046      6.018046             0             0  79.534437
## 6:  6.018046      6.018046             0             0   6.898005
##    CARBON_AG        BA DRYBIO_AG TREEAGE
## 1: 150.60414 0.5345465 301.20828       0
## 2: 180.99964 0.6841498 361.99929       0
## 3: 395.67490 1.0689840 791.34979       0
## 4: 285.05869 0.7853760 570.11737       0
## 5: 268.75655 0.9076001 537.51310       0
## 6:  22.27666 0.2096518  44.55331       0
```

```
# Create new data table to use as a small dataset to test with
## strlut
ESTUNIT <- c(1,1,3,3,5,7,7,9,9,11,11,13,13,15,17,19,19,21,23,23,25)
STRATA <-  c(1,2,1,2,2,1,2,1,2,1,2,1,2,2,2,1,2,2,1,2,2)
ACRES <- c(472611,2285002,245547,1776182,3072988,779515,4317444,215408,
           2514245,540035,1297089,935801,4994287,1428579,1283969,380750,
           2291052,1720074,762318,1854636,3440445)
NBRPLOTS <- round(runif(length(ESTUNIT), 1, 500))
```

```
strlut <- data.table(ESTUNIT, STRATA, ACRES)
strlut2 <- data.table(ESTUNIT, STRATA, NBRPLOTS)

unitvar <- "ESTUNIT"
strvar <- "STRATA"
acrevar <- "ACRES"


# Read in species look up table
ref_spcd <- fread("data/ref_spcd.csv")
```

Meaning of data.table

DT[i, j, by]

Take data.table **DT**, subset rows using **i**, then calculate **j** grouped by **by**

Relationship of commands to SQL

data.table SQL i where j select := update by group by i order by (in compound syntax) i having (in compound syntax) nomatch=NA outer join nomatch=0 inner join

DT[where, select|update, group by][having][order by][ ]...[ ]

Data Exploration 1 - subset columns

```
# Subset using numbers to identify columns
strlut[,1:2, with=FALSE]
```

```
##     ESTUNIT STRATA
##  1:       1      1
##  2:       1      2
##  3:       3      1
##  4:       3      2
##  5:       5      2
##  6:       7      1
##  7:       7      2
##  8:       9      1
##  9:       9      2
## 10:      11      1
## 11:      11      2
## 12:      13      1
## 13:      13      2
## 14:      15      2
## 15:      17      2
## 16:      19      1
## 17:      19      2
## 18:      21      2
## 19:      23      1
## 20:      23      2
## 21:      25      2
##     ESTUNIT STRATA
```

```
#Bad:
#strlut[,1:2]

# Subset column as vector
strlut[,ESTUNIT]
```

```
##  [1]  1  1  3  3  5  7  7  9  9 11 11 13 13 15 17 19 19 21 23 23 25
```

```
# Subset column as data.table with 1 column
strlut[,list(ESTUNIT)]
```

```
##     ESTUNIT
##  1:       1
##  2:       1
##  3:       3
##  4:       3
##  5:       5
##  6:       7
##  7:       7
##  8:       9
##  9:       9
## 10:      11
## 11:      11
## 12:      13
## 13:      13
## 14:      15
## 15:      17
## 16:      19
## 17:      19
## 18:      21
## 19:      23
## 20:      23
## 21:      25
##     ESTUNIT
```

```
# Subset column as vector, passing variable
strlut[[unitvar]]
```

```
##  [1]  1  1  3  3  5  7  7  9  9 11 11 13 13 15 17 19 19 21 23 23 25
```

```
strlut[,"ESTUNIT", with=FALSE]
```

```
##     ESTUNIT
##  1:       1
##  2:       1
##  3:       3
##  4:       3
##  5:       5
##  6:       7
##  7:       7
##  8:       9
```

```
##  9:       9
## 10:      11
## 11:      11
## 12:      13
## 13:      13
## 14:      15
## 15:      17
## 16:      19
## 17:      19
## 18:      21
## 19:      23
## 20:      23
## 21:      25
##     ESTUNIT
```

```
# Subset 1 column as data.table, passing variable
strlut[,unitvar, with=FALSE]
```

```
##     ESTUNIT
##  1:       1
##  2:       1
##  3:       3
##  4:       3
##  5:       5
##  6:       7
##  7:       7
##  8:       9
##  9:       9
## 10:      11
## 11:      11
## 12:      13
## 13:      13
## 14:      15
## 15:      17
## 16:      19
## 17:      19
## 18:      21
## 19:      23
## 20:      23
## 21:      25
##     ESTUNIT
```

```
# Subset more than 1 column as data.table, passing variable
strlut[,c(unitvar, strvar), with=FALSE]
```

```
##     ESTUNIT STRATA
##  1:       1      1
##  2:       1      2
##  3:       3      1
##  4:       3      2
##  5:       5      2
##  6:       7      1
##  7:       7      2
```

```
##  8:       9      1
##  9:       9      2
## 10:      11      1
## 11:      11      2
## 12:      13      1
## 13:      13      2
## 14:      15      2
## 15:      17      2
## 16:      19      1
## 17:      19      2
## 18:      21      2
## 19:      23      1
## 20:      23      2
## 21:      25      2
##     ESTUNIT STRATA
```

```
# Subset rows 2 and 4 and columns ESTUNIT and STRATA
strlut[c(2,4), list(ESTUNIT, STRATA)]
```

```
##    ESTUNIT STRATA
## 1:       1      2
## 2:       3      2
```

```
# Subset rows 2 and 4 and columns ESTUNIT and STRATA, passing variables
STRATA <- "STRATA"
#strlut[c(2,4), list(get(eval(unitvar)), get(STRATA))]

strlut[c(2,4), list(get(eval(unitvar)), get(eval(STRATA)))]
```

```
##    V1 V2
## 1:  1  2
## 2:  3  2
```

```
# Subset a column and add another column that is product of 2 other columns
strlut[, list(ESTUNIT, newcol = ESTUNIT + STRATA)]
```

```
##     ESTUNIT newcol
##  1:       1      2
##  2:       1      3
##  3:       3      4
##  4:       3      5
##  5:       5      7
##  6:       7      8
##  7:       7      9
##  8:       9     10
##  9:       9     11
## 10:      11     12
## 11:      11     13
## 12:      13     14
## 13:      13     15
## 14:      15     17
## 15:      17     19
```

```
## 16:      19     20
## 17:      19     21
## 18:      21     23
## 19:      23     24
## 20:      23     25
## 21:      25     27
##     ESTUNIT newcol
```

```
# Subset a column and add another column that is product of 2 other columns, passing vars
strlut[, .(get(eval(unitvar)), newcol = get(unitvar) + get(strvar))]
```

```
##     V1 newcol
##  1:  1      2
##  2:  1      3
##  3:  3      4
##  4:  3      5
##  5:  5      7
##  6:  7      8
##  7:  7      9
##  8:  9     10
##  9:  9     11
## 10: 11     12
## 11: 11     13
## 12: 13     14
## 13: 13     15
## 14: 15     17
## 15: 17     19
## 16: 19     20
## 17: 19     21
## 18: 21     23
## 19: 23     24
## 20: 23     25
## 21: 25     27
##     V1 newcol
```

```
## Get a count by ESTUNIT and STRATA
strlut[, table(ESTUNIT, STRATA)]
```

```
##        STRATA
## ESTUNIT 1 2
##       1  1 1
##       3  1 1
##       5  0 1
##       7  1 1
##       9  1 1
##       11 1 1
##       13 1 1
##       15 0 1
##       17 0 1
##       19 1 1
##       21 0 1
##       23 1 1
##       25 0 1
```

```
#makes a table

## or passing variables
strlut[, table(get(unitvar), get(strvar))]
```

```
##
##      1 2
##   1  1 1
##   3  1 1
##   5  0 1
##   7  1 1
##   9  1 1
##   11 1 1
##   13 1 1
##   15 0 1
##   17 0 1
##   19 1 1
##   21 0 1
##   23 1 1
##   25 0 1
```

```
## Add a column names STRWT that is proportion of ACRES by ESTUNIT
microbenchmark( strlut[,STRWT:=ACRES/sum(ACRES), by=ESTUNIT],
strlut.dplyr <-
  strlut %>%
  group_by(ESTUNIT) %>%
  mutate(STRWT = ACRES/sum(ACRES)))
```

```
## Unit: microseconds
##                                                                                 expr
##                                 strlut[, `:=`(STRWT, ACRES/sum(ACRES)), by = ESTUNIT]
##  strlut.dplyr <- strlut %>% group_by(ESTUNIT) %>% mutate(STRWT = ACRES/sum(ACRES))
##       min        lq      mean    median        uq       max neval
##   592.793  633.8455  723.8194  700.5555  776.0915  2163.038   100
##  1484.446 1604.3180 2061.7618 1631.8230 1706.9480 38113.212   100
```

```
strlut.dplyr <-
  strlut %>%
  group_by(ESTUNIT) %>%
  mutate(STRWT = ACRES/sum(ACRES))

## Passing variables
strlut[, STRWT:=get(eval(acrevar))/sum(get(eval(acrevar))), by=get(eval(unitvar))]

## Changing names of columns
setnames(strlut2, "NBRPLOTS", "n.strata")

## Subset unique values
subset(unique(strlut), select = unitvar)
```

```
##     ESTUNIT
##  1:       1
```

```
##  2:       1
##  3:       3
##  4:       3
##  5:       5
##  6:       7
##  7:       7
##  8:       9
##  9:       9
## 10:      11
## 11:      11
## 12:      13
## 13:      13
## 14:      15
## 15:      17
## 16:      19
## 17:      19
## 18:      21
## 19:      23
## 20:      23
## 21:      25
##     ESTUNIT
```

```
## Get number of unique values
strlut[, uniqueN(get(unitvar))]
```

```
## [1] 13
```

```
## Set column order
#setcolorder(strlut, c("ESTUNIT", "STRATA", "STRWT", "ACRES"))
```

###Data Exploration - Sum data by group

```
# Compare tapply and data.table method for speed
# Note: On small dataset, tapply is faster, but on larger dataset, the other is faster

tapply(strlut$ACRES, strlut$STRATA, sum)  ## data.frame method
```

```
##       1       2
##  4331985 32275992
```

```
strlut[, sum(ACRES), by=STRATA]           ## data.table method
```

```
##    STRATA       V1
## 1:      1  4331985
## 2:      2 32275992
```

```
# Using small dataset
microbenchmark(
  tapply(strlut$ACRES, strlut$STRATA, sum),
  strlut[, sum(ACRES), by=STRATA]
)
```

```
## Unit: microseconds
##                                       expr     min      lq     mean
##  tapply(strlut$ACRES, strlut$STRATA, sum) 164.620 177.962 195.3389
##         strlut[, sum(ACRES), by = STRATA] 628.509 647.803 689.9638
##    median       uq      max neval
##  190.2775 203.8240  410.933   100
##  655.1925 666.0715 2202.858   100
```

```
## Using larger dataset
microbenchmark(
  tapply(tree.df$BA, tree.df$PLT_CN, sum),
  tree.dt[, sum(BA), by=PLT_CN]
)
```

```
## Unit: milliseconds
##                                     expr       min        lq      mean
##  tapply(tree.df$BA, tree.df$PLT_CN, sum) 55.118237 56.495126 57.845203
##          tree.dt[, sum(BA), by = PLT_CN]  2.041523  2.187258  2.335366
##    median       uq       max neval
##  57.39622 58.62163 74.718167   100
##   2.27470  2.40894  4.136824   100
```

```
# Add a new column to data table by group (sum of basal area by species)
tree.dt[, sumba:=sum(BA, na.rm=TRUE), by=SPCD]
head(tree.dt)
```

```
##             PLT_CN CONDID SUBP TREE STATUSCD SPCD SPGRPCD  DIA HT ACTUALHT
## 1: 282479222489998      1    1    1        1   19      12  9.9 56       56
## 2: 282479222489998      1    1    2        2  108      21 11.2 62       62
## 3: 282479222489998      1    1    3        2  101      24 14.0 68       68
## 4: 282479222489998      1    1    4        2  101      24 12.0 69       69
## 5: 282479222489998      1    1    5        2  101      24 12.9 63       63
## 6: 282479222489998      1    1    6        2  101      24  6.2 31       31
##    HTCD TREECLCD CR CCLCD AGENTCD CULL DECAYCD STOCKING WDLDSTEM UNCRCD
## 1:    1        2 55     3      NA   10      NA    1.556       NA     60
## 2:    1        3 NA    NA      NA   35       3    0.000       NA     NA
## 3:    1        3 NA    NA      NA   25       2    0.000       NA     NA
## 4:    1        3 NA    NA      NA   30       2    0.000       NA     NA
## 5:    1        3 NA    NA      NA   30       3    0.000       NA     NA
## 6:    1        3 NA    NA      NA   35       3    0.000       NA     NA
##    BHAGE TOTAGE MIST_CL_CD STANDING_DEAD_CD PREV_STATUS_CD PREV_WDLDSTEM
## 1:     0      0          0               NA             NA            NA
## 2:     0      0          0                1             NA            NA
## 3:     0      0          0                1             NA            NA
## 4:     0      0          0                1             NA            NA
## 5:     0      0          0                1             NA            NA
## 6:     0      0          0                1             NA            NA
##    RECONCILECD PREVDIA  VOLCFNET  VOLCFGRS FGROWCFAL FMORTCFAL FREMVCFAL
## 1:          NA      NA 10.713023 11.903359  0.239587         0         0
## 2:          NA      NA 13.248148 20.381766  0.000000         0         0
## 3:          NA      NA 24.678143 32.904190  0.000000         0         0
## 4:          NA      NA 17.676287 25.251838  0.000000         0         0
## 5:          NA      NA 18.541419 26.487742  0.000000         0         0
```

```
## 6:          NA      NA  1.487479  2.288429  0.000000         0         0
##    TPA_UNADJ TPAGROW_UNADJ TPAMORT_UNADJ TPAREMV_UNADJ  CARBON_BG
## 1:  6.018046      6.018046             0             0  34.573198
## 2:  6.018046      6.018046             0             0  48.078692
## 3:  6.018046      6.018046             0             0 105.259606
## 4:  6.018046      6.018046             0             0  76.234960
## 5:  6.018046      6.018046             0             0  79.534437
## 6:  6.018046      6.018046             0             0   6.898005
##    CARBON_AG        BA DRYBIO_AG TREEAGE     sumba
## 1: 150.60414 0.5345465 301.20828       0 1803.2537
## 2: 180.99964 0.6841498 361.99929       0 3608.9223
## 3: 395.67490 1.0689840 791.34979       0  855.6096
## 4: 285.05869 0.7853760 570.11737       0  855.6096
## 5: 268.75655 0.9076001 537.51310       0  855.6096
## 6:  22.27666 0.2096518  44.55331       0  855.6096
```

```
# Remove a column from data table
tree.dt[,sumba:=NULL]
head(tree.dt)
```

```
##             PLT_CN CONDID SUBP TREE STATUSCD SPCD SPGRPCD  DIA HT ACTUALHT
## 1: 282479222489998      1    1    1        1   19      12  9.9 56       56
## 2: 282479222489998      1    1    2        2  108      21 11.2 62       62
## 3: 282479222489998      1    1    3        2  101      24 14.0 68       68
## 4: 282479222489998      1    1    4        2  101      24 12.0 69       69
## 5: 282479222489998      1    1    5        2  101      24 12.9 63       63
## 6: 282479222489998      1    1    6        2  101      24  6.2 31       31
##    HTCD TREECLCD CR CCLCD AGENTCD CULL DECAYCD STOCKING WDLDSTEM UNCRCD
## 1:    1        2 55     3      NA   10      NA    1.556       NA     60
## 2:    1        3 NA    NA      NA   35       3    0.000       NA     NA
## 3:    1        3 NA    NA      NA   25       2    0.000       NA     NA
## 4:    1        3 NA    NA      NA   30       2    0.000       NA     NA
## 5:    1        3 NA    NA      NA   30       3    0.000       NA     NA
## 6:    1        3 NA    NA      NA   35       3    0.000       NA     NA
##    BHAGE TOTAGE MIST_CL_CD STANDING_DEAD_CD PREV_STATUS_CD PREV_WDLDSTEM
## 1:     0      0          0               NA             NA            NA
## 2:     0      0          0                1             NA            NA
## 3:     0      0          0                1             NA            NA
## 4:     0      0          0                1             NA            NA
## 5:     0      0          0                1             NA            NA
## 6:     0      0          0                1             NA            NA
##    RECONCILECD PREVDIA  VOLCFNET  VOLCFGRS FGROWCFAL FMORTCFAL FREMVCFAL
## 1:          NA      NA 10.713023 11.903359  0.239587         0         0
## 2:          NA      NA 13.248148 20.381766  0.000000         0         0
## 3:          NA      NA 24.678143 32.904190  0.000000         0         0
## 4:          NA      NA 17.676287 25.251838  0.000000         0         0
## 5:          NA      NA 18.541419 26.487742  0.000000         0         0
## 6:          NA      NA  1.487479  2.288429  0.000000         0         0
##    TPA_UNADJ TPAGROW_UNADJ TPAMORT_UNADJ TPAREMV_UNADJ  CARBON_BG
## 1:  6.018046      6.018046             0             0  34.573198
## 2:  6.018046      6.018046             0             0  48.078692
## 3:  6.018046      6.018046             0             0 105.259606
## 4:  6.018046      6.018046             0             0  76.234960
## 5:  6.018046      6.018046             0             0  79.534437
```

```
## 6:  6.018046      6.018046             0             0   6.898005
##    CARBON_AG        BA DRYBIO_AG TREEAGE
## 1: 150.60414 0.5345465 301.20828       0
## 2: 180.99964 0.6841498 361.99929       0
## 3: 395.67490 1.0689840 791.34979       0
## 4: 285.05869 0.7853760 570.11737       0
## 5: 268.75655 0.9076001 537.51310       0
## 6:  22.27666 0.2096518  44.55331       0
```

```
# To pass a new column variable name in
newcol <- "sumba"
tree.dt[, (newcol):=sum(BA, na.rm=TRUE), by=SPCD]
head(tree.dt)
```

```
##             PLT_CN CONDID SUBP TREE STATUSCD SPCD SPGRPCD  DIA HT ACTUALHT
## 1: 282479222489998      1    1    1        1   19      12  9.9 56       56
## 2: 282479222489998      1    1    2        2  108      21 11.2 62       62
## 3: 282479222489998      1    1    3        2  101      24 14.0 68       68
## 4: 282479222489998      1    1    4        2  101      24 12.0 69       69
## 5: 282479222489998      1    1    5        2  101      24 12.9 63       63
## 6: 282479222489998      1    1    6        2  101      24  6.2 31       31
##    HTCD TREECLCD CR CCLCD AGENTCD CULL DECAYCD STOCKING WDLDSTEM UNCRCD
## 1:    1        2 55     3      NA   10      NA    1.556       NA     60
## 2:    1        3 NA    NA      NA   35       3    0.000       NA     NA
## 3:    1        3 NA    NA      NA   25       2    0.000       NA     NA
## 4:    1        3 NA    NA      NA   30       2    0.000       NA     NA
## 5:    1        3 NA    NA      NA   30       3    0.000       NA     NA
## 6:    1        3 NA    NA      NA   35       3    0.000       NA     NA
##    BHAGE TOTAGE MIST_CL_CD STANDING_DEAD_CD PREV_STATUS_CD PREV_WDLDSTEM
## 1:     0      0          0               NA             NA            NA
## 2:     0      0          0                1             NA            NA
## 3:     0      0          0                1             NA            NA
## 4:     0      0          0                1             NA            NA
## 5:     0      0          0                1             NA            NA
## 6:     0      0          0                1             NA            NA
##    RECONCILECD PREVDIA  VOLCFNET  VOLCFGRS FGROWCFAL FMORTCFAL FREMVCFAL
## 1:          NA      NA 10.713023 11.903359  0.239587         0         0
## 2:          NA      NA 13.248148 20.381766  0.000000         0         0
## 3:          NA      NA 24.678143 32.904190  0.000000         0         0
## 4:          NA      NA 17.676287 25.251838  0.000000         0         0
## 5:          NA      NA 18.541419 26.487742  0.000000         0         0
## 6:          NA      NA  1.487479  2.288429  0.000000         0         0
##    TPA_UNADJ TPAGROW_UNADJ TPAMORT_UNADJ TPAREMV_UNADJ  CARBON_BG
## 1:  6.018046      6.018046             0             0  34.573198
## 2:  6.018046      6.018046             0             0  48.078692
## 3:  6.018046      6.018046             0             0 105.259606
## 4:  6.018046      6.018046             0             0  76.234960
## 5:  6.018046      6.018046             0             0  79.534437
## 6:  6.018046      6.018046             0             0   6.898005
##    CARBON_AG        BA DRYBIO_AG TREEAGE     sumba
## 1: 150.60414 0.5345465 301.20828       0 1803.2537
## 2: 180.99964 0.6841498 361.99929       0 3608.9223
## 3: 395.67490 1.0689840 791.34979       0  855.6096
## 4: 285.05869 0.7853760 570.11737       0  855.6096
```

```
## 5: 268.75655 0.9076001 537.51310       0  855.6096
## 6:  22.27666 0.2096518  44.55331       0  855.6096
```

```
# Remove a column using a passed in variable
tree.dt[, (newcol):=NULL]
head(tree.dt)
```

```
##             PLT_CN CONDID SUBP TREE STATUSCD SPCD SPGRPCD  DIA HT ACTUALHT
## 1: 282479222489998      1    1    1        1   19      12  9.9 56       56
## 2: 282479222489998      1    1    2        2  108      21 11.2 62       62
## 3: 282479222489998      1    1    3        2  101      24 14.0 68       68
## 4: 282479222489998      1    1    4        2  101      24 12.0 69       69
## 5: 282479222489998      1    1    5        2  101      24 12.9 63       63
## 6: 282479222489998      1    1    6        2  101      24  6.2 31       31
##    HTCD TREECLCD CR CCLCD AGENTCD CULL DECAYCD STOCKING WDLDSTEM UNCRCD
## 1:    1        2 55     3      NA   10      NA    1.556       NA     60
## 2:    1        3 NA    NA      NA   35       3    0.000       NA     NA
## 3:    1        3 NA    NA      NA   25       2    0.000       NA     NA
## 4:    1        3 NA    NA      NA   30       2    0.000       NA     NA
## 5:    1        3 NA    NA      NA   30       3    0.000       NA     NA
## 6:    1        3 NA    NA      NA   35       3    0.000       NA     NA
##    BHAGE TOTAGE MIST_CL_CD STANDING_DEAD_CD PREV_STATUS_CD PREV_WDLDSTEM
## 1:     0      0          0                NA             NA            NA
## 2:     0      0          0                 1             NA            NA
## 3:     0      0          0                 1             NA            NA
## 4:     0      0          0                 1             NA            NA
## 5:     0      0          0                 1             NA            NA
## 6:     0      0          0                 1             NA            NA
##    RECONCILECD PREVDIA  VOLCFNET  VOLCFGRS FGROWCFAL FMORTCFAL FREMVCFAL
## 1:          NA      NA 10.713023 11.903359  0.239587         0         0
## 2:          NA      NA 13.248148 20.381766  0.000000         0         0
## 3:          NA      NA 24.678143 32.904190  0.000000         0         0
## 4:          NA      NA 17.676287 25.251838  0.000000         0         0
## 5:          NA      NA 18.541419 26.487742  0.000000         0         0
## 6:          NA      NA  1.487479  2.288429  0.000000         0         0
##    TPA_UNADJ TPAGROW_UNADJ TPAMORT_UNADJ TPAREMV_UNADJ  CARBON_BG
## 1:  6.018046      6.018046             0             0  34.573198
## 2:  6.018046      6.018046             0             0  48.078692
## 3:  6.018046      6.018046             0             0 105.259606
## 4:  6.018046      6.018046             0             0  76.234960
## 5:  6.018046      6.018046             0             0  79.534437
## 6:  6.018046      6.018046             0             0   6.898005
##    CARBON_AG        BA DRYBIO_AG TREEAGE
## 1: 150.60414 0.5345465 301.20828       0
## 2: 180.99964 0.6841498 361.99929       0
## 3: 395.67490 1.0689840 791.34979       0
## 4: 285.05869 0.7853760 570.11737       0
## 5: 268.75655 0.9076001 537.51310       0
## 6:  22.27666 0.2096518  44.55331       0
```

```
# Create a new data table with sum of basal area by species
sumba <- tree.dt[, sum(BA, na.rm=TRUE), by=SPCD]
sumba
```

```
##     SPCD          V1
##  1:   19 1803.253675
##  2:  108 3608.922272
##  3:  101  855.609612
##  4:   93 2207.414655
##  5:  113  299.443744
##  6:   66  301.927768
##  7:  202 1111.021250
##  8:  122  821.667952
##  9:  746  304.035466
## 10:  745   46.674787
## 11:  749   27.500759
## 12:   65  675.239451
## 13:  544    5.136795
## 14:  313    4.887984
## 15:  106    1.019080
## 16:  475    2.948432
## 17:  375    1.422785
## 18:  823   41.700357
## 19:   96   11.979220
## 20:  814    2.313914
```

```
# Create a new data table (with new name) with sum of basal area by species
spba <- tree.dt[, .(sumba=sum(BA, na.rm=TRUE)), by=SPCD]
spba
```

```
##     SPCD       sumba
##  1:   19 1803.253675
##  2:  108 3608.922272
##  3:  101  855.609612
##  4:   93 2207.414655
##  5:  113  299.443744
##  6:   66  301.927768
##  7:  202 1111.021250
##  8:  122  821.667952
##  9:  746  304.035466
## 10:  745   46.674787
## 11:  749   27.500759
## 12:   65  675.239451
## 13:  544    5.136795
## 14:  313    4.887984
## 15:  106    1.019080
## 16:  475    2.948432
## 17:  375    1.422785
## 18:  823   41.700357
## 19:   96   11.979220
## 20:  814    2.313914
```

```
# Create a new data table (with new name) with sum of basal area by species - passing variable
ba <- "BA"
spba <- tree.dt[, list(sumba=sum(get(ba), na.rm=TRUE)), by=SPCD]
spba
```

```
##     SPCD       sumba
##  1:   19 1803.253675
##  2:  108 3608.922272
##  3:  101  855.609612
##  4:   93 2207.414655
##  5:  113  299.443744
##  6:   66  301.927768
##  7:  202 1111.021250
##  8:  122  821.667952
##  9:  746  304.035466
## 10:  745   46.674787
## 11:  749   27.500759
## 12:   65  675.239451
## 13:  544    5.136795
## 14:  313    4.887984
## 15:  106    1.019080
## 16:  475    2.948432
## 17:  375    1.422785
## 18:  823   41.700357
## 19:   96   11.979220
## 20:  814    2.313914
```

```
# Sum basal area by 1 group (species=202)
tree.dt[SPCD==202, sum(BA, na.rm=TRUE)]
```

```
## [1] 1111.021
```

```
# Get a count of species that fall within height ranges (to nearest 10 ft) for
# height classes greater than 10.
#.N =freq
tree.dt[HT > 10, list(Count = .N), by = list(HTCL = 10 * round(HT / 10))]
```

```
##     HTCL Count
##  1:   60  3424
##  2:   70  1596
##  3:   30  3311
##  4:   50  3852
##  5:   40  5404
##  6:   80   970
##  7:   90   303
##  8:   20  2524
##  9:   10  1069
## 10:  100   114
## 11:  120    22
## 12:  110    40
## 13:  140     2
## 14:  130     1
```

```
## Calculate sum of BA by SPCD and HTCL (from above)
tree.dt[HT > 10, .(sum(BA)), by = list(SPCD, HTCL = 10 * round(HT / 10))]
```

```
##      SPCD HTCL          V1
```

```
##   1:   19   60 416.7985523
##   2:  108   60 964.2566738
##   3:  101   70  70.1852353
##   4:  101   60 145.1319217
##   5:  101   30 169.8424141
##  ---
## 121:  823   50   1.9735299
## 122:  375   40   0.4599358
## 123:   93  130   8.4665714
## 124:  814   20   2.1828544
## 125:  814   10   0.1021534
```

```
## Get sum of BA and number of trees by STATUSCD and by PLT_CN
tree.dt[PLT_CN==282479222489998, {
    sumba = sum(BA, na.rm=TRUE)
    n = .N
    .SD[, .(n, .N, sumba_in_STATUSCD = sum(BA, na.rm=TRUE), sumba_in_PLT=sumba), by=STATUSCD] },
    by=PLT_CN]
```

```
##             PLT_CN STATUSCD  n  N sumba_in_STATUSCD sumba_in_PLT
## 1: 282479222489998        1 74 21          7.907973     50.47513
## 2: 282479222489998        2 74 53         42.567161     50.47513
```

```
## Perform more than 1 operation on a column within a data.table
tree.dt[, {tmp1=BA*TPA_UNADJ; tmp2=mean(tmp1, na.rm=TRUE); tmp3=round(tmp2, 2)}, by=SPCD]
```

```
##     SPCD    V1
##  1:   19  2.77
##  2:  108  2.76
##  3:  101  3.24
##  4:   93  4.56
##  5:  113  3.07
##  6:   66  3.53
##  7:  202  4.37
##  8:  122  3.78
##  9:  746  2.29
## 10:  745 14.78
## 11:  749  6.22
## 12:   65  6.15
## 13:  544  2.81
## 14:  313  2.94
## 15:  106  3.07
## 16:  475  3.04
## 17:  375  1.02
## 18:  823  2.02
## 19:   96  5.91
## 20:  814  1.64
```

```
## Keep more than one variable
tree.dt[, {tmp1=BA*TPA_UNADJ; tmp2=mean(tmp1, na.rm=TRUE); tmp3=round(tmp2, 2);   list(tmp2=tmp2, tmp3=
```

```
##     SPCD      tmp2  tmp3
```

```
##  1:   19  2.765436  2.77
##  2:  108  2.755351  2.76
##  3:  101  3.235487  3.24
##  4:   93  4.563856  4.56
##  5:  113  3.074976  3.07
##  6:   66  3.526703  3.53
##  7:  202  4.370663  4.37
##  8:  122  3.775341  3.78
##  9:  746  2.288436  2.29
## 10:  745 14.783738 14.78
## 11:  749  6.216706  6.22
## 12:   65  6.145466  6.15
## 13:  544  2.810316  2.81
## 14:  313  2.941611  2.94
## 15:  106  3.066435  3.07
## 16:  475  3.037039  3.04
## 17:  375  1.021695  1.02
## 18:  823  2.023044  2.02
## 19:   96  5.905230  5.91
## 20:  814  1.640103  1.64
```

```
## Multiply multiple columns by a constant
t1 <- copy(tree.dt)
t2 <- copy(tree.dt)
t3 <- copy(tree.dt)
t4 <- copy(tree.dt)
vars2convert <- c("CARBON_BG", "CARBON_AG")
microbenchmark(
    for(j in vars2convert){ set(t1, i=NULL, j=j, value=t1[[j]] * 1000) },
    t2[, (vars2convert) := lapply(.SD, function(x) x * 1000 ), .SDcols=vars2convert],
    t3[, (vars2convert) := lapply(.SD, "*", 1000 ), .SDcols=vars2convert],
    t4[, (vars2convert) := get(eval(vars2convert)) * 1000]
)
```

```
## Unit: microseconds
##                                                                                 expr
##            for (j in vars2convert) {     set(t1, i = NULL, j = j, value = t1[[j]] * 1000) }
##  t2[, `:=`((vars2convert), lapply(.SD, function(x) x * 1000)),      .SDcols = vars2convert]
##                 t3[, `:=`((vars2convert), lapply(.SD, "*", 1000)), .SDcols = vars2convert]
##                                   t4[, `:=`((vars2convert), get(eval(vars2convert)) * 1000)]
##       min        lq      mean    median        uq      max neval
##   123.567  159.2830  286.2447  169.9560  182.6825 1631.823   100
##   779.991  845.0590  987.8626  908.6895  994.0780 2445.887   100
##   676.129  731.5495  841.2531  776.7070  855.1165 2133.480   100
##  1637.981 1767.2945 1929.5534 1813.4785 1925.1400 3903.237   100
```

```
t1[1:2, c("PLT_CN", "TREE", vars2convert), with=FALSE]
```

```
##             PLT_CN TREE     CARBON_BG     CARBON_AG
## 1: 282479222489998    1 3.457320e+301 1.506041e+302
## 2: 282479222489998    2 4.807869e+301 1.809996e+302
```

```
t2[1:2, c("PLT_CN", "TREE", vars2convert), with=FALSE]
```

```
##             PLT_CN TREE     CARBON_BG     CARBON_AG
## 1: 282479222489998    1 3.457320e+301 1.506041e+302
## 2: 282479222489998    2 4.807869e+301 1.809996e+302
```

```
t3[1:2, c("PLT_CN", "TREE", vars2convert), with=FALSE]
```

```
##             PLT_CN TREE     CARBON_BG     CARBON_AG
## 1: 282479222489998    1 3.457320e+301 1.506041e+302
## 2: 282479222489998    2 4.807869e+301 1.809996e+302
```

```
t4[1:2, c("PLT_CN", "TREE", vars2convert), with=FALSE]
```

```
##             PLT_CN TREE     CARBON_BG     CARBON_AG
## 1: 282479222489998    1 3.457320e+301 3.457320e+301
## 2: 282479222489998    2 4.807869e+301 4.807869e+301
```

```
    ## CHANGE NA VALUES TO 0
#    for(col in tsumvarnmlst2) set(sumdat, which(is.na(sumdat[[col]])), col, 0)
```

Using Keys

```
# Set key for tree.dt as SPCD
setkey(tree.dt, SPCD)

# or if passing variable
var <- "SPCD"
setkeyv(tree.dt, var)


# Get sum basal area for spcd = 202
tree.dt[SPCD==202, sum(BA, na.rm=TRUE)]
```

```
## [1] 1111.021
```

```
# Get sum basal area for all species
tree.dt[, sum(BA, na.rm=TRUE), by=SPCD]
```

```
##     SPCD          V1
##  1:   19 1803.253675
##  2:   65  675.239451
##  3:   66  301.927768
##  4:   93 2207.414655
##  5:   96   11.979220
##  6:  101  855.609612
##  7:  106    1.019080
##  8:  108 3608.922272
```

```
##  9:  113  299.443744
## 10:  122  821.667952
## 11:  202 1111.021250
## 12:  313    4.887984
## 13:  375    1.422785
## 14:  475    2.948432
## 15:  544    5.136795
## 16:  745   46.674787
## 17:  746  304.035466
## 18:  749   27.500759
## 19:  814    2.313914
## 20:  823   41.700357
```

```
# Get sum of basal area, average height, maximum height, maximum diameter by species
# by= and keyby=  both retain row order within groups (by-order of first appearance)

key(tree.dt)
```

```
## [1] "SPCD"
```

```
spsum <- tree.dt[,list(sumba=sum(BA, na.rm=TRUE),
              avght=mean(HT, na.rm=TRUE),
              maxht=max(HT, na.rm=TRUE),
              maxdia=max(DIA, na.rm=TRUE)), by=key(tree.dt)]
spsum
```

```
##     SPCD       sumba     avght maxht maxdia
##  1:   19 1803.253675 42.376534   116   31.7
##  2:   65  675.239451  9.572280    28   40.5
##  3:   66  301.927768 12.023339    33   26.2
##  4:   93 2207.414655 50.546166   140   43.0
##  5:   96   11.979220 41.785714    98   32.5
##  6:  101  855.609612 35.762073    94   36.7
##  7:  106    1.019080 12.500000    15   11.8
##  8:  108 3608.922272 46.367831   111   26.7
##  9:  113  299.443744 26.921348    85   31.5
## 10:  122  821.667952 39.079399    89   28.2
## 11:  202 1111.021250 47.957009   124   34.1
## 12:  313    4.887984 33.700000    39   14.3
## 13:  375    1.422785 28.100000    42    7.2
## 14:  475    2.948432  6.866667    23   11.5
## 15:  544    5.136795 38.000000    58   12.8
## 16:  745   46.674787 54.157895    89   64.6
## 17:  746  304.035466 38.332645    82   19.4
## 18:  749   27.500759 37.333333    63   27.6
## 19:  814    2.313914 16.857143    22    7.5
## 20:  823   41.700357 24.500000    51   16.4
```

```
spsum2 <- tree.dt[,list(sumba=sum(BA, na.rm=TRUE),
              avght=mean(HT, na.rm=TRUE),
              maxht=max(HT, na.rm=TRUE),
              maxdia=max(DIA, na.rm=TRUE)), keyby=key(tree.dt)]
spsum2
```

```
##     SPCD       sumba     avght maxht maxdia
##  1:   19 1803.253675 42.376534   116   31.7
##  2:   65  675.239451  9.572280    28   40.5
##  3:   66  301.927768 12.023339    33   26.2
##  4:   93 2207.414655 50.546166   140   43.0
##  5:   96   11.979220 41.785714    98   32.5
##  6:  101  855.609612 35.762073    94   36.7
##  7:  106    1.019080 12.500000    15   11.8
##  8:  108 3608.922272 46.367831   111   26.7
##  9:  113  299.443744 26.921348    85   31.5
## 10:  122  821.667952 39.079399    89   28.2
## 11:  202 1111.021250 47.957009   124   34.1
## 12:  313    4.887984 33.700000    39   14.3
## 13:  375    1.422785 28.100000    42    7.2
## 14:  475    2.948432  6.866667    23   11.5
## 15:  544    5.136795 38.000000    58   12.8
## 16:  745   46.674787 54.157895    89   64.6
## 17:  746  304.035466 38.332645    82   19.4
## 18:  749   27.500759 37.333333    63   27.6
## 19:  814    2.313914 16.857143    22    7.5
## 20:  823   41.700357 24.500000    51   16.4
```

```
# For just one species
# Note: Because key is numeric, must include list or J in front of category
sp202 <- tree.dt[list(202),list(sumba=sum(BA, na.rm=TRUE),
              avght=mean(HT, na.rm=TRUE),
              maxht=max(HT, na.rm=TRUE),
              maxdia=max(DIA, na.rm=TRUE)), keyby=key(tree.dt)]
sp202
```

```
##    SPCD    sumba    avght maxht maxdia
## 1:  202 1111.021 47.95701   124   34.1
```

```
# Without specifying key
sp202 <- tree.dt[list(202),list(sumba=sum(BA, na.rm=TRUE),
              avght=mean(HT, na.rm=TRUE),
              maxht=max(HT, na.rm=TRUE),
              maxdia=max(DIA, na.rm=TRUE))]
sp202
```

```
##       sumba    avght maxht maxdia
## 1: 1111.021 47.95701   124   34.1
```

```
# For two species
microbenchmark(
sp202_746 <- tree.dt[list(c(202,746)),list(sumba=sum(BA, na.rm=TRUE),
              avght=mean(HT, na.rm=TRUE),
              maxht=max(HT, na.rm=TRUE),
              maxdia=max(DIA, na.rm=TRUE)), by=.EACHI],
sp202_746 <- tree.dt[list(c(202,746)),list(sumba=sum(BA, na.rm=TRUE),
              avght=mean(HT, na.rm=TRUE),
              maxht=max(HT, na.rm=TRUE),
              maxdia=max(DIA, na.rm=TRUE)), by=key(tree.dt)])
```

```
## Unit: milliseconds
##
##        sp202_746 <- tree.dt[list(c(202, 746)), list(sumba = sum(BA,      na.rm = TRUE), avght = mean
##  sp202_746 <- tree.dt[list(c(202, 746)), list(sumba = sum(BA,      na.rm = TRUE), avght = mean(HT, na
##       min       lq     mean   median       uq      max neval
##  1.626076 1.679032 1.867655 1.760112 1.829900 3.901185   100
##  2.016892 2.053223 2.193453 2.123832 2.181511 4.009152   100
```

```
sp202_746
```

```
##    SPCD     sumba    avght maxht maxdia
## 1:  202 1111.0213 47.95701   124   34.1
## 2:  746  304.0355 38.33264    82   19.4
```

```
# All species
spall <- tree.dt[,list(sumba=sum(BA, na.rm=TRUE),
              avght=mean(HT, na.rm=TRUE),
              maxht=max(HT, na.rm=TRUE),
              maxdia=max(DIA, na.rm=TRUE)), by=key(tree.dt)]
spall
```

```
##     SPCD       sumba     avght maxht maxdia
##  1:   19 1803.253675 42.376534   116   31.7
##  2:   65  675.239451  9.572280    28   40.5
##  3:   66  301.927768 12.023339    33   26.2
##  4:   93 2207.414655 50.546166   140   43.0
##  5:   96   11.979220 41.785714    98   32.5
##  6:  101  855.609612 35.762073    94   36.7
##  7:  106    1.019080 12.500000    15   11.8
##  8:  108 3608.922272 46.367831   111   26.7
##  9:  113  299.443744 26.921348    85   31.5
## 10:  122  821.667952 39.079399    89   28.2
## 11:  202 1111.021250 47.957009   124   34.1
## 12:  313    4.887984 33.700000    39   14.3
## 13:  375    1.422785 28.100000    42    7.2
## 14:  475    2.948432  6.866667    23   11.5
## 15:  544    5.136795 38.000000    58   12.8
## 16:  745   46.674787 54.157895    89   64.6
## 17:  746  304.035466 38.332645    82   19.4
## 18:  749   27.500759 37.333333    63   27.6
## 19:  814    2.313914 16.857143    22    7.5
## 20:  823   41.700357 24.500000    51   16.4
```

Joining Tables

Join type DT syntax Merge INNER X[Y, nomatch=0] merge(X,Y,all=FALSE) LEFT OUTER Y[X] merge(X,Y,all.x=TRUE) RIGHT OUTER X[Y] merge(X,Y,all.y=TRUE) FULL OUTER - merge(X,Y,all=TRUE)

```
## Testing different ways of merging
setkey(tree.dt, SPCD)
ref_spcd <- setDT(ref_spcd, key="VALUE")
```

```
microbenchmark(
  a1 <- merge(tree.df, ref_spcd, by.x="SPCD", by.y="VALUE"), # using merge with data frame
  a2 <- merge(tree.dt, ref_spcd, by.x="SPCD", by.y="VALUE"), # using merge with data table
  a3 <- tree.dt[ref_spcd, on=c(SPCD="VALUE"), nomatch=0],    # using on with data table
  a4 <- tree.dt[ref_spcd, nomatch=0L],                       # using on with data table and keys
  setDT(tree.dt)[ref_spcd, on=c(SPCD="VALUE")],
  a5 <- inner_join(x=tree.df, y=ref_spcd, by = c("SPCD"="VALUE"))

)
```

```
## Unit: milliseconds
##                                                                 expr
##        a1 <- merge(tree.df, ref_spcd, by.x = "SPCD", by.y = "VALUE")
##        a2 <- merge(tree.dt, ref_spcd, by.x = "SPCD", by.y = "VALUE")
##         a3 <- tree.dt[ref_spcd, on = c(SPCD = "VALUE"), nomatch = 0]
##                              a4 <- tree.dt[ref_spcd, nomatch = 0L]
##                    setDT(tree.dt)[ref_spcd, on = c(SPCD = "VALUE")]
##  a5 <- inner_join(x = tree.df, y = ref_spcd, by = c(SPCD = "VALUE"))
##       min       lq      mean   median       uq      max neval
##  7.508846 7.832953 11.467019 8.023640 9.717041 54.81979   100
##  5.590890 5.924850 11.403787 6.157616 7.824742 57.70411   100
##  6.141811 6.492806 11.918326 6.690882 8.558960 55.46964   100
##  6.063811 6.361849 13.248460 6.613294 8.416508 51.60335   100
##  6.386891 6.665225 10.194444 6.876028 8.121960 53.43551   100
##  3.536231 3.709267  9.611955 3.815592 5.247079 48.25843   100
```

```
## Testing group with merge as separate commands vs in same command
key(a4)
```

```
## [1] "SPCD"
```

```
a4 <- tree.dt[ref_spcd, nomatch=0L]
a4[, sum(BA, na.rm=TRUE), by=key(a4)]
```

```
##     SPCD          V1
##  1:   19 1803.253675
##  2:   65  675.239451
##  3:   66  301.927768
##  4:   93 2207.414655
##  5:   96   11.979220
##  6:  101  855.609612
##  7:  106    1.019080
##  8:  108 3608.922272
##  9:  113  299.443744
## 10:  122  821.667952
## 11:  202 1111.021250
## 12:  313    4.887984
## 13:  375    1.422785
## 14:  475    2.948432
## 15:  544    5.136795
## 16:  745   46.674787
```

```
## 17:  746  304.035466
## 18:  749   27.500759
## 19:  814    2.313914
## 20:  823   41.700357
```

```
tree.dt[ref_spcd, nomatch=0, sum(BA, na.rm=TRUE), by=key(tree.dt)]
```

```
##     SPCD          V1
##  1:   19 1803.253675
##  2:   65  675.239451
##  3:   66  301.927768
##  4:   93 2207.414655
##  5:   96   11.979220
##  6:  101  855.609612
##  7:  106    1.019080
##  8:  108 3608.922272
##  9:  113  299.443744
## 10:  122  821.667952
## 11:  202 1111.021250
## 12:  313    4.887984
## 13:  375    1.422785
## 14:  475    2.948432
## 15:  544    5.136795
## 16:  745   46.674787
## 17:  746  304.035466
## 18:  749   27.500759
## 19:  814    2.313914
## 20:  823   41.700357
```

```
#or
merge.dt <- merge(tree.dt[,c("PLT_CN", "SPCD", "BA"), with=FALSE], ref_spcd, by.x="SPCD", by.y="VALUE")
head(merge.dt)
```

```
##    SPCD          PLT_CN         BA       MEANING
## 1:   19 282479222489998 0.53454654 subalpine fir
## 2:   19 282479222489998 0.34038414 subalpine fir
## 3:   19 282479222489998 0.34905600 subalpine fir
## 4:   19 282479222489998 0.67198734 subalpine fir
## 5:   19 282479222489998 0.07068384 subalpine fir
## 6:   19 282479222489998 0.54540000 subalpine fir
```

```
cols <- c("PLT_CN", "SPCD", "BA")
merge.dt <- merge(tree.dt[,cols, with=FALSE], ref_spcd, by.x="SPCD", by.y="VALUE")
head(merge.dt)
```

```
##    SPCD          PLT_CN         BA       MEANING
## 1:   19 282479222489998 0.53454654 subalpine fir
## 2:   19 282479222489998 0.34038414 subalpine fir
## 3:   19 282479222489998 0.34905600 subalpine fir
## 4:   19 282479222489998 0.67198734 subalpine fir
## 5:   19 282479222489998 0.07068384 subalpine fir
## 6:   19 282479222489998 0.54540000 subalpine fir
```

```
## Test difference between data.frame and data.table
# microbenchmark(
#   merge.df <- merge(tree.df[,c("PLT_CN", "SPCD", "BA")], ref_spcd, by.x="SPCD", by.y="VALUE"),
#   merge.dt <- merge(tree.dt[,c("PLT_CN", "SPCD", "BA"), with=FALSE], ref_spcd, by.x="SPCD", by.y="VAL
# )

microbenchmark(
  a4 <- tree.dt[ref_spcd, nomatch=0L],
  a4[, sum(BA, na.rm=TRUE), by=key(a4)],
  tree.dt[ref_spcd, nomatch=0, sum(BA, na.rm=TRUE), by=key(a4)]
)
```

```
## Unit: milliseconds
##                                                                    expr
##                                a4 <- tree.dt[ref_spcd, nomatch = 0L]
##                             a4[, sum(BA, na.rm = TRUE), by = key(a4)]
##  tree.dt[ref_spcd, nomatch = 0, sum(BA, na.rm = TRUE), by = key(a4)]
##       min       lq     mean   median       uq       max neval
##  5.996485 6.117795 7.618932 6.458937 7.498788 48.214910   100
##  1.050525 1.132423 1.237119 1.179634 1.292937  2.479549   100
##  2.367477 2.451634 2.679568 2.579306 2.728941  5.172570   100
```

```
## Merging a subset of one data table to another data table
a4 <- tree.dt[ref_spcd, nomatch=0L]


## Add a new column to first data.table in merge using columns from second data.table
setkey(strlut, ESTUNIT, STRATA)
setkey(strlut2, ESTUNIT, STRATA)
strlut[strlut2, newcol:=ACRES*NBRPLOTS]
```

Symbols (.N, .SD, .I, .BY, .GRP)

```
# Frequency table (Number of records by SPCD)
tree.dt[, .N, by=SPCD]
```

```
##     SPCD    N
##  1:   19 4824
##  2:   65  671
##  3:   66  579
##  4:   93 3150
##  5:   96   14
##  6:  101 1721
##  7:  106    2
##  8:  108 8966
##  9:  113  682
## 10:  122 1486
## 11:  202 1659
## 12:  313   13
## 13:  375   11
## 14:  475   15
```

```
## 15:  544   11
## 16:  745   19
## 17:  746 1080
## 18:  749   29
## 19:  814   14
## 20:  823  151
```

```
# Frequency table (Number of records by SPCD) - with named column
tree.dt[, .(NBR=.N), by=SPCD]
```

```
##     SPCD  NBR
##  1:   19 4824
##  2:   65  671
##  3:   66  579
##  4:   93 3150
##  5:   96   14
##  6:  101 1721
##  7:  106    2
##  8:  108 8966
##  9:  113  682
## 10:  122 1486
## 11:  202 1659
## 12:  313   13
## 13:  375   11
## 14:  475   15
## 15:  544   11
## 16:  745   19
## 17:  746 1080
## 18:  749   29
## 19:  814   14
## 20:  823  151
```

```
# Frequency table (Number of records by SPCD)
tree.dt[, as.data.table(table(SPCD))]
```

```
##     SPCD    N
##  1:   19 4824
##  2:   65  671
##  3:   66  579
##  4:   93 3150
##  5:   96   14
##  6:  101 1721
##  7:  106    2
##  8:  108 8966
##  9:  113  682
## 10:  122 1486
## 11:  202 1659
## 12:  313   13
## 13:  375   11
## 14:  475   15
## 15:  544   11
## 16:  745   19
## 17:  746 1080
```

```
## 18:  749   29
## 19:  814   14
## 20:  823  151
```

```
# Frequency table by 2 columns (Number of records by SPCD)
tree.dt[,.N, by=list(SPCD, STATUSCD)]
```

```
##     SPCD STATUSCD    N
##  1:   19        1 3601
##  2:   19        2 1223
##  3:   65        1  571
##  4:   65        2  100
##  5:   66        1  505
##  6:   66        2   74
##  7:   93        1 2156
##  8:   93        2  994
##  9:   96        1   14
## 10:  101        2  841
## 11:  101        1  880
## 12:  106        1    2
## 13:  108        2 2730
## 14:  108        1 6236
## 15:  113        2  338
## 16:  113        1  344
## 17:  122        1 1255
## 18:  122        2  231
## 19:  202        1 1113
## 20:  202        2  546
## 21:  313        1    9
## 22:  313        2    4
## 23:  375        1    7
## 24:  375        2    4
## 25:  475        1   14
## 26:  475        2    1
## 27:  544        1   11
## 28:  745        1   18
## 29:  745        2    1
## 30:  746        2  368
## 31:  746        1  712
## 32:  749        1   26
## 33:  749        2    3
## 34:  814        1   13
## 35:  814        2    1
## 36:  823        1  132
## 37:  823        2   19
##     SPCD STATUSCD    N
```

```
tree.dt[, as.data.table(table(SPCD, STATUSCD))]
```

```
##     SPCD STATUSCD    N
##  1:   19        1 3601
##  2:   65        1  571
##  3:   66        1  505
```

```
##  4:   93        1 2156
##  5:   96        1   14
##  6:  101        1  880
##  7:  106        1    2
##  8:  108        1 6236
##  9:  113        1  344
## 10:  122        1 1255
## 11:  202        1 1113
## 12:  313        1    9
## 13:  375        1    7
## 14:  475        1   14
## 15:  544        1   11
## 16:  745        1   18
## 17:  746        1  712
## 18:  749        1   26
## 19:  814        1   13
## 20:  823        1  132
## 21:   19        2 1223
## 22:   65        2  100
## 23:   66        2   74
## 24:   93        2  994
## 25:   96        2    0
## 26:  101        2  841
## 27:  106        2    0
## 28:  108        2 2730
## 29:  113        2  338
## 30:  122        2  231
## 31:  202        2  546
## 32:  313        2    4
## 33:  375        2    4
## 34:  475        2    1
## 35:  544        2    0
## 36:  745        2    1
## 37:  746        2  368
## 38:  749        2    3
## 39:  814        2    1
## 40:  823        2   19
##     SPCD STATUSCD    N
```

```
## Passing variables
var1 <- "SPCD"
var2 <- "STATUSCD"
vars <- c(var1, var2)

tree.dt[,.N, by=c(var1, var2)]
```

```
##     SPCD STATUSCD    N
##  1:   19        1 3601
##  2:   19        2 1223
##  3:   65        1  571
##  4:   65        2  100
##  5:   66        1  505
##  6:   66        2   74
##  7:   93        1 2156
```

```
##  8:   93        2  994
##  9:   96        1   14
## 10:  101        2  841
## 11:  101        1  880
## 12:  106        1    2
## 13:  108        2 2730
## 14:  108        1 6236
## 15:  113        2  338
## 16:  113        1  344
## 17:  122        1 1255
## 18:  122        2  231
## 19:  202        1 1113
## 20:  202        2  546
## 21:  313        1    9
## 22:  313        2    4
## 23:  375        1    7
## 24:  375        2    4
## 25:  475        1   14
## 26:  475        2    1
## 27:  544        1   11
## 28:  745        1   18
## 29:  745        2    1
## 30:  746        2  368
## 31:  746        1  712
## 32:  749        1   26
## 33:  749        2    3
## 34:  814        1   13
## 35:  814        2    1
## 36:  823        1  132
## 37:  823        2   19
##     SPCD STATUSCD    N
```

```
tree.dt[,.N, by=vars]
```

```
##     SPCD STATUSCD    N
##  1:   19        1 3601
##  2:   19        2 1223
##  3:   65        1  571
##  4:   65        2  100
##  5:   66        1  505
##  6:   66        2   74
##  7:   93        1 2156
##  8:   93        2  994
##  9:   96        1   14
## 10:  101        2  841
## 11:  101        1  880
## 12:  106        1    2
## 13:  108        2 2730
## 14:  108        1 6236
## 15:  113        2  338
## 16:  113        1  344
## 17:  122        1 1255
## 18:  122        2  231
## 19:  202        1 1113
```

```
## 20:  202        2  546
## 21:  313        1    9
## 22:  313        2    4
## 23:  375        1    7
## 24:  375        2    4
## 25:  475        1   14
## 26:  475        2    1
## 27:  544        1   11
## 28:  745        1   18
## 29:  745        2    1
## 30:  746        2  368
## 31:  746        1  712
## 32:  749        1   26
## 33:  749        2    3
## 34:  814        1   13
## 35:  814        2    1
## 36:  823        1  132
## 37:  823        2   19
##     SPCD STATUSCD    N
```

```
tree.dt[,.(Freq=.N), by=vars]
```

```
##     SPCD STATUSCD Freq
##  1:   19        1 3601
##  2:   19        2 1223
##  3:   65        1  571
##  4:   65        2  100
##  5:   66        1  505
##  6:   66        2   74
##  7:   93        1 2156
##  8:   93        2  994
##  9:   96        1   14
## 10:  101        2  841
## 11:  101        1  880
## 12:  106        1    2
## 13:  108        2 2730
## 14:  108        1 6236
## 15:  113        2  338
## 16:  113        1  344
## 17:  122        1 1255
## 18:  122        2  231
## 19:  202        1 1113
## 20:  202        2  546
## 21:  313        1    9
## 22:  313        2    4
## 23:  375        1    7
## 24:  375        2    4
## 25:  475        1   14
## 26:  475        2    1
## 27:  544        1   11
## 28:  745        1   18
## 29:  745        2    1
## 30:  746        2  368
## 31:  746        1  712
```

```
## 32:  749        1   26
## 33:  749        2    3
## 34:  814        1   13
## 35:  814        2    1
## 36:  823        1  132
## 37:  823        2   19
##     SPCD STATUSCD Freq
```

```
tree.dt[SPCD %in% c(202, 746),.(Freq=.N), by=vars]
```

```
##    SPCD STATUSCD Freq
## 1:  202        1 1113
## 2:  202        2  546
## 3:  746        2  368
## 4:  746        1  712
```

```
tree.dt[, as.data.table(table(get(var1), get(var2)))]
```

```
##       V1 V2    N
##  1:  19  1 3601
##  2:  65  1  571
##  3:  66  1  505
##  4:  93  1 2156
##  5:  96  1   14
##  6: 101  1  880
##  7: 106  1    2
##  8: 108  1 6236
##  9: 113  1  344
## 10: 122  1 1255
## 11: 202  1 1113
## 12: 313  1    9
## 13: 375  1    7
## 14: 475  1   14
## 15: 544  1   11
## 16: 745  1   18
## 17: 746  1  712
## 18: 749  1   26
## 19: 814  1   13
## 20: 823  1  132
## 21:  19  2 1223
## 22:  65  2  100
## 23:  66  2   74
## 24:  93  2  994
## 25:  96  2    0
## 26: 101  2  841
## 27: 106  2    0
## 28: 108  2 2730
## 29: 113  2  338
## 30: 122  2  231
## 31: 202  2  546
## 32: 313  2    4
## 33: 375  2    4
## 34: 475  2    1
```

```
## 35: 544  2    0
## 36: 745  2    1
## 37: 746  2  368
## 38: 749  2    3
## 39: 814  2    1
## 40: 823  2   19
##      V1 V2    N
```

```
## Get PLT_CN values where there are more than 50 live trees
tuniqueid <- "PLT_CN"
tree.dt[STATUSCD == 1, (.N > 100), by=c(tuniqueid, "STATUSCD")][V1==TRUE][[tuniqueid]]
```

```
## [1] "40405497010690" "40406999010690"
```

```
## Check results
tplt <- tree.dt[PLT_CN == 40405497010690]
dim(tplt)
```

```
## [1] 129  42
```

```
tplt2 <- tree.dt[PLT_CN == 40406999010690]
dim(tplt2)
```

```
## [1] 107  42
```

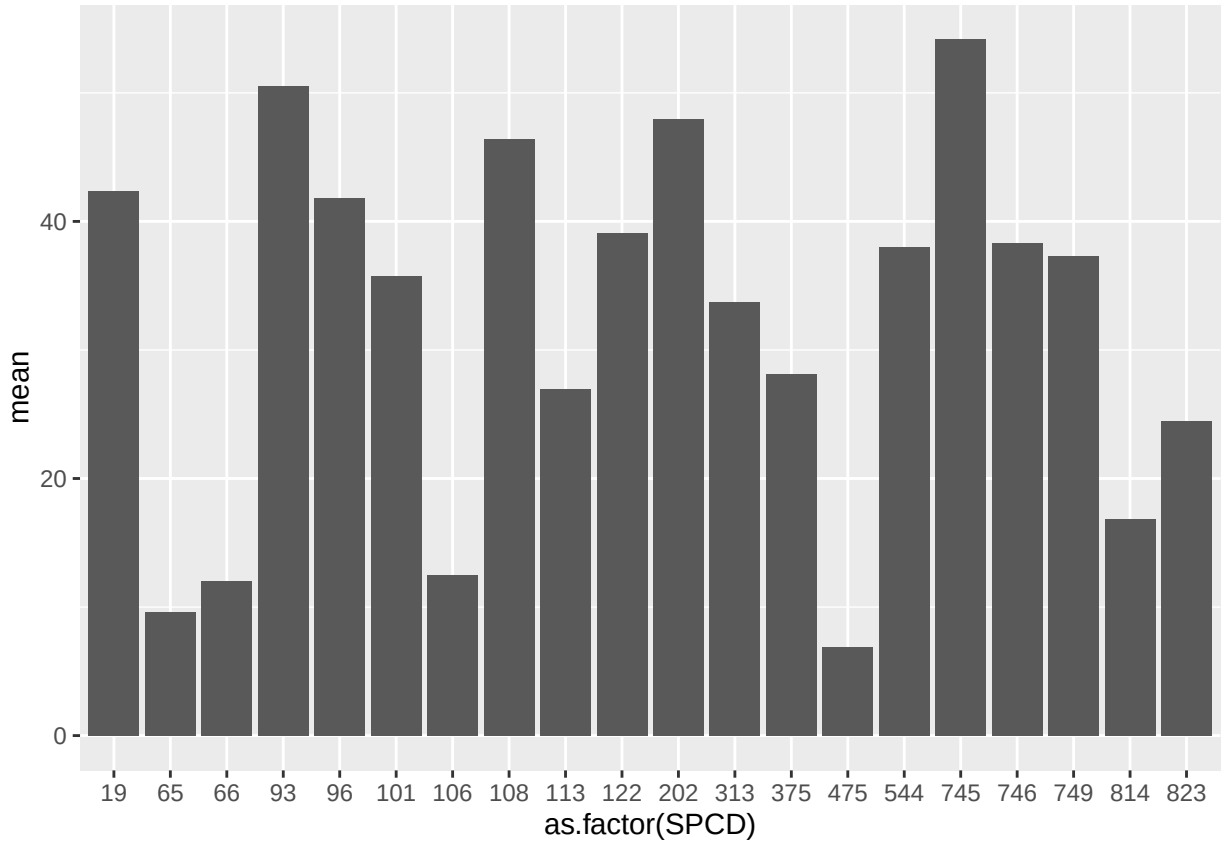
```
## Changing values of columns (ex. NA values to 0 values)
#na.to.0 <- function(x){x[is.na(x)] <- 0; x}
#sumtreef.prop[, (tdomscols) := lapply(.SD, na.to.0), .SDcols=tdomscols]

## this is faster
#for(col in tdomscols) set(sumtreef.prop2, which(is.na(sumtreef.prop2[[col]])), col, 0)
#DANGER ZONE
```

```
mean.ht.dt <- tree.dt[,list(mean= mean(HT, na.rm=TRUE)), by="SPCD"]

mean.ht.dt2 <- tree.dt[,mean:= mean(HT, na.rm=TRUE), by="SPCD"]

library(ggplot2)
ggplot(mean.ht.dt, aes(y=mean, x=as.factor(SPCD))) + geom_bar(stat="identity")
```


mean
40
20
0
19
65
66
93
96
101
106
108
113
122
202
313
375
475
544
745
746
749
814
823
as.factor(SPCD)