

# *Manipulation of data-frame data with dutility functions*

*Klaus Holst & Thomas Scheike*

*March 3, 2020*

---

## *Simple data manipulation for data-frames*

- Renaming variables, Deleting variables
- Looking at the data
- Making new variables for the analysis
- Making factors (groupings)
- Working with factors
- Making a factor from existing numeric variable and vice versa

Here are some key data-manipulation steps on a data-frame which is how we typically organize our data in R. After having read the data into R it will typically be a data-frame, if not we can force it to be a data-frame. The basic idea of the utility functions is to get a simple and easy to type way of making simple data-manipulation on a data-frame much like what is possible in SAS or STATA.

The functions, `say`, `dcut`, `dfactor` and so on are all functions that basically does what the base R `cut`, `factor` do, but are easier to use in the context of data-frames and have additional functionality.

---

```
1 library(mets)
2 data(melanoma)
```

---

```
1 is.data.frame(melanoma)
```

---

```
[1] TRUE
```

Here we work on the melanoma data that is already read into R and is a data-frame.

## *dUtility functions*

The structure for all functions is

- `dfunction(dataframe, y~x | ifcond, ...)`

to use the function on `y` in a dataframe grouped by `x` if condition `ifcond` is valid. The basic functions are

Data processing

- `dsort`

- dreshape
- dcut
- drm, drename, ddrop, dkeep, dsubset
- drelevel
- dlag
- dfactor, dnumeric

#### Data aggregation

- dby, dby2
- dscalar, deval, daggregate
- dmean, dsd, dsum, dquantile, dcor
- dtable, dcount

#### Data summaries

- dhead, dtail,
- dsummary,
- dprint, dlist, dlevels, dunique

A generic function daggregate, daggr, can be called with a function as the argument

- daggregate(dataframe, y~x | ifcond, fun=function, ...)

without the grouping variable (x)

- daggregate(dataframe, ~y | ifcond, fun=function, ...)

A useful feature is that y and x as well as the subset condition can be specified using regular-expressions or by wildcards (default). Here to illustrate this, we compute the means of certain variables.

First just overall

---

```
1 dmean(melanoma, ~thick+I(log(thick)))
```

---

```
      thick I(log(thick))
291.985366      5.223341
```

now only when days>500

---

```
1 dmean(melanoma, ~thick+I(log(thick)) | I(days>500))
```

---

```
      thick I(log(thick))
271.582011      5.168691
```

and now after sex but only when days>500

---

```
1 dmean(melanoma, thick+I(log(thick)) ~sex | I(days>500))
```

---

```
sex    thick I(log(thick))
1  0 242.9580    5.060086
2  1 320.2429    5.353321
```

and finally after quartiles of days (via the dcut function)

```
1 dmean(melanoma,thick+I(log(thick))-I(dcut(days)))
```

```
      I(dcut(days))    thick I(log(thick))
1      [10,1.52e+03] 482.1731    5.799525
2      (1.52e+03,2e+03] 208.5490    4.987652
3      (2e+03,3.04e+03] 223.2941    4.974759
4      (3.04e+03,5.56e+03] 250.1961    5.120129
```

or summary of all variables starting with "s" and that contains "a"

```
1 dmean(melanoma, "s"+"*a*"~sex|I(days>500))
```

```
sex    status    days
1  0 1.831933 2399.143
2  1 1.714286 2169.800
```

*Renaming, deleting, keeping, dropping variables*

```
1 melanoma=drename(melanoma,tykkelse~thick)
2 names(melanoma)
```

```
[1] "no"      "status"  "days"   "ulc"     "tykkelse" "sex"
```

Deleting variables

```
1 data(melanoma)
2 melanoma=drm(melanoma,~thick+sex)
3 names(melanoma)
```

```
[1] "no"      "status"  "days"   "ulc"
```

or sas style

```
1 data(melanoma)
2 melanoma=ddrop(melanoma,~thick+sex)
3 names(melanoma)
```

```
[1] "no"      "status"  "days"   "ulc"
```

alternatively we can also keep certain variables

```
1 data(melanoma)
2 melanoma=dkeep(melanoma,~thick+sex+status+days)
3 names(melanoma)
```

```
[1] "thick"  "sex"    "status"  "days"
```

This can also be done with direct assignment

```
1 data(melanoma)
2 ddrop(melanoma) <- ~thick+sex
3 names(melanoma)
```

```
[1] "no"      "status"  "days"   "ulc"
```

*Looking at the data*


---

```

1 data(melanoma)
2 dstr(melanoma)

```

---

```

'data.frame':      205 obs. of  6 variables:
 $ no      : int  789 13 97 16 21 469 685 7 932 944 ...
 $ status: int   3 3 2 3 1 1 1 1 3 1 ...
 $ days  : int  10 30 35 99 185 204 210 232 232 279 ...
 $ ulc   : int   1 0 0 0 1 1 1 1 1 1 ...
 $ thick : int  676 65 134 290 1208 484 516 1288 322 741 ...
 $ sex   : int   1 1 1 0 1 1 1 1 0 0 ...

```

The data can in Rstudio be seen as a data-table but to list certain parts of the data in output window

---

```

1 dlist(melanoma)

```

---

```

      no status days ulc thick sex
1  789 3      10 1    676 1
2   13 3      30 0     65 1
3   97 2      35 0    134 1
4   16 3      99 0    290 0
5   21 1     185 1   1208 1
---
201 317 2     4492 1    706 1
202 798 2     4668 0    612 0
203 806 2     4688 0     48 0
204 606 2     4926 0    226 0
205 328 2     5565 0    290 0

```

---

```

1 dlist(melanoma, ~.|sex==1)

```

---

```

      no status days ulc thick
1  789 3      10 1    676
2   13 3      30 0     65
3   97 2      35 0    134
5   21 1     185 1   1208
6  469 1     204 1    484
---
191 445 2     3909 1    806
195 415 2     4119 0     65
197 175 2     4207 0     65
198 493 2     4310 0    210
201 317 2     4492 1    706

```

---

```

1 dlist(melanoma, ~ulc+days+thick+sex|sex==1)

```

---

```

      ulc days thick sex
1     1  10  676 1
2     0  30   65 1
3     0  35  134 1
5     1 185 1208 1
6     1 204  484 1
---
191 1 3909 806 1
195 0 4119  65 1
197 0 4207  65 1
198 0 4310 210 1
201 1 4492 706 1

```

Getting summaries

---

```
1 dsummary(melanoma)
```

---

no	status	days	ulc	thick
Min. : 2.0	Min. :1.00	Min. : 10	Min. :0.000	Min. : 10
1st Qu.:222.0	1st Qu.:1.00	1st Qu.:1525	1st Qu.:0.000	1st Qu.: 97
Median :469.0	Median :2.00	Median :2005	Median :0.000	Median : 194
Mean :463.9	Mean :1.79	Mean :2153	Mean :0.439	Mean : 292
3rd Qu.:731.0	3rd Qu.:2.00	3rd Qu.:3042	3rd Qu.:1.000	3rd Qu.: 356
Max. :992.0	Max. :3.00	Max. :5565	Max. :1.000	Max. :1742

sex

Min. :0.0000
1st Qu.:0.0000
Median :0.0000
Mean :0.3854
3rd Qu.:1.0000
Max. :1.0000

or for specific variables

---

```
1 dsummary(melanoma,~thick+status+sex)
```

---

thick	status	sex
Min. : 10	Min. :1.00	Min. :0.0000
1st Qu.: 97	1st Qu.:1.00	1st Qu.:0.0000
Median : 194	Median :2.00	Median :0.0000
Mean : 292	Mean :1.79	Mean :0.3854
3rd Qu.: 356	3rd Qu.:2.00	3rd Qu.:1.0000
Max. :1742	Max. :3.00	Max. :1.0000

Summaries in different groups (sex)

---

```
1 dsummary(melanoma,thick+days+status~sex)
```

---

sex: 0

thick	days	status
Min. : 10.0	Min. : 99	Min. :1.000
1st Qu.: 97.0	1st Qu.:1636	1st Qu.:2.000
Median : 162.0	Median :2059	Median :2.000
Mean : 248.6	Mean :2283	Mean :1.833
3rd Qu.: 306.0	3rd Qu.:3131	3rd Qu.:2.000
Max. :1742.0	Max. :5565	Max. :3.000

sex: 1

thick	days	status
Min. : 16.0	Min. : 10	Min. :1.000
1st Qu.: 105.0	1st Qu.:1052	1st Qu.:1.000
Median : 258.0	Median :1860	Median :2.000
Mean : 361.1	Mean :1946	Mean :1.722
3rd Qu.: 484.0	3rd Qu.:2784	3rd Qu.:2.000
Max. :1466.0	Max. :4492	Max. :3.000

and only among those with thin-tumours or only females (sex==1)

---

```
1 dsummary(melanoma,thick+days+status~sex|thick<97)
```

---

sex: 0

thick	days	status
Min. :10.00	Min. : 355	Min. :1.000
1st Qu.:32.00	1st Qu.:1762	1st Qu.:2.000
Median :64.00	Median :2227	Median :2.000
Mean :51.48	Mean :2425	Mean :2.034
3rd Qu.:65.00	3rd Qu.:3185	3rd Qu.:2.000
Max. :81.00	Max. :4688	Max. :3.000

```
-----
sex: 1
  thick      days      status
Min.   :16.00  Min.   : 30  Min.   :1.000
1st Qu.:30.00  1st Qu.:1820  1st Qu.:2.000
Median :65.00  Median :2886  Median :2.000
Mean   :55.75  Mean   :2632  Mean   :1.875
3rd Qu.:81.00  3rd Qu.:3328  3rd Qu.:2.000
Max.   :81.00  Max.   :4207  Max.   :3.000
```

```
1 dsummary(melanoma,thick+status~+1|sex==1)
```

```
-----
  thick      status
Min.   : 16.0  Min.   :1.000
1st Qu.:105.0  1st Qu.:1.000
Median :258.0  Median :2.000
Mean   :361.1  Mean   :1.722
3rd Qu.:484.0  3rd Qu.:2.000
Max.   :1466.0  Max.   :3.000
```

OR

```
1 dsummary(melanoma,~thick+status|sex==1)
```

```
-----
  thick      status
Min.   : 16.0  Min.   :1.000
1st Qu.:105.0  1st Qu.:1.000
Median :258.0  Median :2.000
Mean   :361.1  Mean   :1.722
3rd Qu.:484.0  3rd Qu.:2.000
Max.   :1466.0  Max.   :3.000
```

To make more complex conditions need to use the I()

```
1 dsummary(melanoma,thick+days+status~sex|I(thick<97 & sex==1)
  )
```

```
sex: 1
  thick      days      status
Min.   :16.00  Min.   : 30  Min.   :1.000
1st Qu.:30.00  1st Qu.:1820  1st Qu.:2.000
Median :65.00  Median :2886  Median :2.000
Mean   :55.75  Mean   :2632  Mean   :1.875
3rd Qu.:81.00  3rd Qu.:3328  3rd Qu.:2.000
Max.   :81.00  Max.   :4207  Max.   :3.000
```

Tables between variables

```
1 dtable(melanoma,~status+sex)
```

```
-----
      sex  0  1
status
1       28 29
2       91 43
3        7  7
```

All bivariate tables

```
1 dtable(melanoma,~status+sex+ulc,level=2)
```

```

      status
sex  1  2  3
  0 28 91  7
  1 29 43  7

```

```

      status
ulc  1  2  3
  0 16 92  7
  1 41 42  7

```

```

      sex
ulc  0  1
  0 79 36
  1 47 43

```

### All univariate tables

---

```
1 dtable(melanoma,~status+sex+ulc,level=1)
```

---

```

status
  1  2  3
57 134 14

```

```

sex
  0  1
126 79

```

```

ulc
  0  1
115 90

```

### and with new variables

---

```
1 dtable(melanoma,~status+sex+ulc+dcut(days)+I(days>300),level
      =1)
```

---

```

status
  1  2  3
57 134 14

```

```

sex
  0  1
126 79

```

```

ulc
  0  1
115 90

```

```

dcut(days)
  [10,1.52e+03]  (1.52e+03,2e+03]  (2e+03,3.04e+03]  (3.04e+03,5.56e+03]
                52                51                51                51

```

```

I(days > 300)
FALSE TRUE
  11  194

```

### *Sorting the data*

To sort the data

---

```

1 data(melanoma)
2 mel= dsort(melanoma,~days)
3 dsort(melanoma) <- ~days
4 head(mel)

```

---

```

      no status days ulc thick sex
1 789      3   10   1   676   1
2  13      3   30   0    65   1
3  97      2   35   0   134   1
4  16      3   99   0   290   0
5  21      1  185   1  1208   1
6 469      1  204   1   484   1

```

and to sort after multiple variables increasing and decreasing

---

```

1 dsort(melanoma) <- ~days~status
2 head(melanoma)

```

---

```

      no status days ulc thick sex
1 789      3   10   1   676   1
2  13      3   30   0    65   1
3  97      2   35   0   134   1
4  16      3   99   0   290   0
5  21      1  185   1  1208   1
6 469      1  204   1   484   1

```

### *Making new variables for the analysis*

To define a bunch of new covariates within a data-frame

---

```

1 data(melanoma)
2 melanoma= transform(melanoma, thick2=thick^2, lthick=log(
      thick) )
3 dhead(melanoma)

```

---

```

      no status days ulc thick sex  thick2  lthick
1 789      3   10   1   676   1  456976  6.516193
2  13      3   30   0    65   1   4225  4.174387
3  97      2   35   0   134   1  17956  4.897840
4  16      3   99   0   290   0  84100  5.669881
5  21      1  185   1  1208   1 1459264  7.096721
6 469      1  204   1   484   1  234256  6.182085

```

When the above definitions are done using a condition this can be achieved using the `dtransform` function that extends `transform` with a possible condition

---

```

1 melanoma=dtransform(melanoma,ll=thick*1.05^ulc,sex==1)
2 melanoma=dtransform(melanoma,ll=thick,sex!=1)
3 dmean(melanoma,ll~sex+ulc)

```

---

```

      sex ulc      ll
1  0  0  173.7342
2  1  0  197.3611
3  0  1  374.5532
4  1  1  523.1198

```

## Making factors (groupings)

On the melanoma data the variable `thick` gives the thickness of the melanom tumour. For some analyses we would like to make a factor depending on the thickness. This can be done in several different ways

---

```
1 melanoma=dcut(melanoma,~thick,breaks=c(0,200,500,800,2000))
```

---

New variable is named `thickcat.o` by default.

To see levels of factors in data-frame

---

```
1 dlevels(melanoma)
```

---

```
thickcat.o #levels=:4
[1] "[0,200]"      "(200,500]"    "(500,800]"    "(800,2e+03]"
-----
```

### Checking group sizes

---

```
1 dtable(melanoma,~thickcat.o)
```

---

```
thickcat.o
  [0,200]  (200,500]  (500,800] (800,2e+03]
      109         64         20         12
```

### With adding to the data-frame directly

---

```
1 dcut(melanoma,breaks=c(0,200,500,800,2000)) <- gr.thick1~
  thick
2 dlevels(melanoma)
```

---

```
thickcat.o #levels=:4
[1] "[0,200]"      "(200,500]"    "(500,800]"    "(800,2e+03]"
-----
gr.thick1 #levels=:4
[1] "[0,200]"      "(200,500]"    "(500,800]"    "(800,2e+03]"
-----
```

new variable is named `thickcat.o` (after first cut-point), or to get quartiles with default names `thick.cat.4`

---

```
1 dcut(melanoma) <- ~ thick # new variable is thickcat.4
2 dlevels(melanoma)
```

---

```
thickcat.o #levels=:4
[1] "[0,200]"      "(200,500]"    "(500,800]"    "(800,2e+03]"
-----
gr.thick1 #levels=:4
[1] "[0,200]"      "(200,500]"    "(500,800]"    "(800,2e+03]"
-----
thickcat.4 #levels=:4
[1] "[10,97]"      "(97,194]"     "(194,356]"     "(356,1.74e+03]"
-----
```

or median groups, here starting again with the original data,

---

```
1 data(melanoma)
2 dcut(melanoma,breaks=2) <- ~ thick # new variable is thick.2
3 dlevels(melanoma)
```

---

```
thickcat.2 #levels=:2
[1] "[10,194]" "(194,1.74e+03]"
-----
```

to control new names

---

```
1 data(melanoma)
2 mela= dcut(melanoma,thickcat4+dayscat4~thick+days,breaks=4)
3 dlevels(mela)
```

---

```
thickcat4 #levels=:4
[1] "[10,97]" "(97,194]" "(194,356]" "(356,1.74e+03]"
-----
```

```
dayscat4 #levels=:4
[1] "[10,1.52e+03]" "(1.52e+03,2e+03]" "(2e+03,3.04e+03]"
[4] "[3.04e+03,5.56e+03]"
-----
```

OR

---

```
1 data(melanoma)
2 dcut(melanoma,breaks=4) <- thickcat4+dayscat4~thick+days
3 dlevels(melanoma)
```

---

```
thickcat4 #levels=:4
[1] "[10,97]" "(97,194]" "(194,356]" "(356,1.74e+03]"
-----
```

```
dayscat4 #levels=:4
[1] "[10,1.52e+03]" "(1.52e+03,2e+03]" "(2e+03,3.04e+03]"
[4] "[3.04e+03,5.56e+03]"
-----
```

This can also be typed out more specifically

---

```
1 melanoma$gthick = cut(melanoma$thick,breaks=c
  (0,200,500,800,2000))
2 melanoma$gthick = cut(melanoma$thick,breaks=quantile(
  melanoma$thick),include.lowest=TRUE)
```

---

## *Working with factors*

To see levels of covariates in data-frame

---

```
1 data(melanoma)
2 dcut(melanoma,breaks=4) <- thickcat4~thick
3 dlevels(melanoma)
```

---

```
thickcat4 #levels=:4
[1] "[10,97]" "(97,194]" "(194,356]" "(356,1.74e+03]"
-----
```

To relevel the factor

---

```
1 dtable(melanoma,~thickcat4)
2 melanoma = drelevel(melanoma,~thickcat4,ref="(194,356]")
3 dlevels(melanoma)
```

---

```

thickcat4
  [10,97]      (97,194]      (194,356] (356,1.74e+03]
      56          53          45          51
thickcat4 #levels=:4
[1] "[10,97]"      "(97,194]"      "(194,356]"      "(356,1.74e+03]"
-----
thickcat4.(194,356] #levels=:4
[1] "(194,356]"      "[10,97]"      "(97,194]"      "(356,1.74e+03]"
-----

```

or to take the third level in the list of levels, same as above,

```

1 melanoma = drelevel(melanoma,~thickcat4,ref=2)
2 dlevels(melanoma)
-----

```

```

thickcat4 #levels=:4
[1] "[10,97]"      "(97,194]"      "(194,356]"      "(356,1.74e+03]"
-----
thickcat4.(194,356] #levels=:4
[1] "(194,356]"      "[10,97]"      "(97,194]"      "(356,1.74e+03]"
-----
thickcat4.2 #levels=:4
[1] "(97,194]"      "[10,97]"      "(194,356]"      "(356,1.74e+03]"
-----

```

To combine levels of a factor (first combining first 3 groups into one)

```

1 melanoma = drelevel(melanoma,~thickcat4,newlevels=1:3)
2 dlevels(melanoma)
-----

```

```

thickcat4 #levels=:4
[1] "[10,97]"      "(97,194]"      "(194,356]"      "(356,1.74e+03]"
-----
thickcat4.(194,356] #levels=:4
[1] "(194,356]"      "[10,97]"      "(97,194]"      "(356,1.74e+03]"
-----
thickcat4.2 #levels=:4
[1] "(97,194]"      "[10,97]"      "(194,356]"      "(356,1.74e+03]"
-----
thickcat4.1:3 #levels=:2
[1] "[10,97]-(194,356]" "(356,1.74e+03]"
-----

```

or to combine groups 1 and 2 into one group and 3 and 4 into another

```

1 dkeep(melanoma) <- ~thick+thickcat4
2 melanoma = drelevel(melanoma,gthick2~thickcat4,newlevels=
  list(1:2,3:4))
3 dlevels(melanoma)
-----

```

```

thickcat4 #levels=:4
[1] "[10,97]"      "(97,194]"      "(194,356]"      "(356,1.74e+03]"
-----
gthick2 #levels=:2
[1] "[10,97]-(97,194]"      "(194,356]-(356,1.74e+03]"
-----

```

Changing order of factor levels

---

```

1 dfactor(melanoma,levels=c(3,1,2,4)) <- thickcat4.2~thickcat4
2 dlevel(melanoma,~ "thickcat4*")
3 dtable(melanoma,~thickcat4+thickcat4.2)

```

---

```

thickcat4 #levels=:4
[1] "[10,97]"      "(97,194)"      "(194,356)"      "(356,1.74e+03)"
-----
thickcat4.2 #levels=:4
[1] "(194,356)"      "[10,97]"      "(97,194)"      "(356,1.74e+03)"
-----

                thickcat4.2 (194,356) [10,97] (97,194) (356,1.74e+03)
thickcat4
[10,97]                0      56      0      0
(97,194)                0      0      53      0
(194,356)               45      0      0      0
(356,1.74e+03)         0      0      0      51

```

Combine levels but now control factor-level names

---

```

1 melanoma=drelevel(melanoma,gthick3~thickcat4,newlevels=list(
   group1.2=1:2,group3.4=3:4))
2 dlevels(melanoma)

```

---

```

thickcat4 #levels=:4
[1] "[10,97]"      "(97,194)"      "(194,356)"      "(356,1.74e+03)"
-----
gthick2 #levels=:2
[1] "[10,97]-(97,194)"      "(194,356)-(356,1.74e+03)"
-----
thickcat4.2 #levels=:4
[1] "(194,356)"      "[10,97]"      "(97,194)"      "(356,1.74e+03)"
-----
gthick3 #levels=:2
[1] "group1.2" "group3.4"
-----

```

### *Making a factor from existing numeric variable and vice versa*

A numeric variable "status" with values 1,2,3 into a factor by

---

```

1 data(melanoma)
2 melanoma = dfactor(melanoma,~status, labels=c("malignant-
   melanoma","censoring","dead-other"))
3 melanoma = dfactor(melanoma,sex1~sex,labels=c("females","
   males"))
4 dtable(melanoma,~sex1+status.f)

```

---

```

                status.f malignant-melanoma censoring dead-other
sex1
females                28      91      7
males                  29      43      7

```

A gender factor with values "M", "F" can be converted into numerics by

---

```

1 melanoma = dnumeric(melanoma,~sex1)
2 dstr(melanoma,"sex*")
3 dtable(melanoma,~'sex*',level=2)

```

---

```
'data.frame':      205 obs. of  3 variables:
 $ sex   : int  1 1 1 0 1 1 1 1 0 0 ...
 $ sexl  : Factor w/ 2 levels "females","males": 2 2 2 1 2 2 2 2 1 1 ...
 $ sexl.n: num  2 2 2 1 2 2 2 2 1 1 ...
```

```
      sex
sexl   0  1
females 126 0
males   0 79
```

```
      sex
sexl.n 0  1
  1 126  0
  2  0 79
```

```
      sexl
sexl.n females males
  1      126     0
  2       0    79
```