Exercise III: Apply Functions

Wen-wen Tung

5/25/2019

R has a family of apply functions: `apply` , `lapply` , `sapply` , `tapply` , `mapply` , and related `by` and `split` . These functions perform vectorized operations to process data in batches, so that the analysts do not need to use loops. You are strongly encouraged to use them whenever possible.

---

1. Defining groups via a factor

1.1 `split`

R uses a factor to define a group. For example, each element of vector x below can be associated with a group in G. The vector x can then be split according to the group.

```
x <- c(10, 15, 31, 25, 44)
G <- factor(c("A","C","C","B","C"))
Gx <- split(x,G)
Gx
```

```
## $A
## [1] 10
##
## $B
## [1] 25
##
## $C
## [1] 15 31 44
```

We can try to split the Atlantic Hurricane Data according to Storm status.

```
load("/Users/wen-wen/Something/Class/TaiwanNTU2019/NTU_Exercises/ALHURDAT.RDATA")
AL[which(AL$Status=="ET"), "Status"] <- "EX"
GAL <- split(AL,AL$Status)
summary(GAL)
```

```
##    Length Class      Mode
## DB 22     data.frame list
## EX 22     data.frame list
## HU 22     data.frame list
## LO 22     data.frame list
## SD 22     data.frame list
## SS 22     data.frame list
## TD 22     data.frame list
## TS 22     data.frame list
## WV 22     data.frame list
```

2. The `tapply` function

The `tapply` function is useful when we need to break up a vector into groups defined by some classifying factor, compute a function on the subsets, and return the results in a convenient table form (the 't' stands for 'table').

2.1 Example of getting monthly mean of daily measurements of airquality data

Take the `airquality` data frame included in the base **R** datasets for example. We used this dataset at the end of Exercise I. It has daily measurements of ozone, solar radiation, wind speed, temperature in New York from May to September in 1973.

Here we perform some initial exploration of the data.

```
str(airquality)
```

```
## 'data.frame':    153 obs. of  6 variables:
##  $ Ozone  : int  41 36 12 18 NA 28 23 19 8 NA ...
##  $ Solar.R: int  190 118 149 313 NA NA 299 99 19 194 ...
##  $ Wind   : num  7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
##  $ Temp   : int  67 72 74 62 56 66 65 59 61 69 ...
##  $ Month  : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ Day    : int  1 2 3 4 5 6 7 8 9 10 ...
```

```
summary(airquality)
```

```
##      Ozone           Solar.R           Wind             Temp
##  Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :56.00
##  1st Qu.: 18.00   1st Qu.:115.8   1st Qu.: 7.400   1st Qu.:72.00
##  Median : 31.50   Median :205.0   Median : 9.700   Median :79.00
##  Mean   : 42.13   Mean   :185.9   Mean   : 9.958   Mean   :77.88
##  3rd Qu.: 63.25   3rd Qu.:258.8   3rd Qu.:11.500   3rd Qu.:85.00
##  Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00
##  NA's   :37       NA's   :7
##      Month            Day
##  Min.   :5.000   Min.   : 1.0
##  1st Qu.:6.000   1st Qu.: 8.0
##  Median :7.000   Median :16.0
##  Mean   :6.993   Mean   :15.8
##  3rd Qu.:8.000   3rd Qu.:23.0
##  Max.   :9.000   Max.   :31.0
##
```

```
head(airquality)
```

```
##   Ozone Solar.R Wind Temp Month Day
## 1    41     190  7.4   67     5   1
## 2    36     118  8.0   72     5   2
## 3    12     149 12.6   74     5   3
## 4    18     313 11.5   62     5   4
## 5    NA      NA 14.3   56     5   5
## 6    28      NA 14.9   66     5   6
```

```
tail(airquality)
```

```
##     Ozone Solar.R Wind Temp Month Day
## 148    14      20 16.6   63     9  25
## 149    30     193  6.9   70     9  26
## 150    NA     145 13.2   77     9  27
## 151    14     191 14.3   75     9  28
## 152    18     131  8.0   76     9  29
## 153    20     223 11.5   68     9  30
```

We observe that:

- The data can be broken up into groups by month
- We intend to subset the data by month
- Within each subset, we can calculate mean values of ozone, solar radiation, wind speed, and temperature

This is a situation that `tapply` is most useful. For example, to calculate monthly mean temperature, we invoke the `tapply` function with three arguments: the response variable (`airquality$Temp`), the categorical explanatory variable (`airquality$Month`), and the name of the function that we want to apply (`mean`):

```
MTemp <- tapply(airquality$Temp, airquality$Month, mean)
MTemp
```

```
##        5        6        7        8        9
## 65.54839 79.10000 83.90323 83.96774 76.90000
```

Question 1: Similar to the example above, use `tapply` to calculate the monthly mean ozone concentration. Note that there are missing values in ozone measurements.

```
#### Answer
```

```
#### Answer
Mozone <- tapply(airquality$Ozone, airquality$Month, mean, na.rm=TRUE)
Mozone
```

```
##        5        6        7        8        9
## 23.61538 29.44444 59.11538 59.96154 31.44828
```

2.2 Example of getting mean life expentancy in Gapminder data grouped by year and continent

The `tapply` function is very flexible. It can produce multi-dimensional tables simply by replacing the one categorical variable by a list of categorical variables.

Attach the Gapminder data; install the package if you have not:

```
library(gapminder)
str(gapminder)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    1704 obs. of  6 variables:
##  $ country  : Factor w/ 142 levels "Afghanistan",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ continent: Factor w/ 5 levels "Africa","Americas",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ year     : int  1952 1957 1962 1967 1972 1977 1982 1987 1992 1997 ...
##  $ lifeExp  : num  28.8 30.3 32 34 36.1 ...
##  $ pop      : int  8425333 9240934 10267083 11537966 13079460 14880372 12881816 13867957 1
6317921 22227415 ...
##  $ gdpPercap: num  779 821 853 836 740 ...
```

```
summary(gapminder)
```

```
##         country        continent        year         lifeExp
##  Afghanistan:  12   Africa  :624   Min.   :1952   Min.   :23.60
##  Albania    :  12   Americas:300   1st Qu.:1966   1st Qu.:48.20
##  Algeria    :  12   Asia    :396   Median :1980   Median :60.71
##  Angola     :  12   Europe  :360   Mean   :1980   Mean   :59.47
##  Argentina  :  12   Oceania : 24   3rd Qu.:1993   3rd Qu.:70.85
##  Australia  :  12                  Max.   :2007   Max.   :82.60
##  (Other)    :1632
##       pop              gdpPercap
##  Min.   :6.001e+04   Min.   :   241.2
##  1st Qu.:2.794e+06   1st Qu.:  1202.1
##  Median :7.024e+06   Median :  3531.8
##  Mean   :2.960e+07   Mean   :  7215.3
##  3rd Qu.:1.959e+07   3rd Qu.:  9325.5
##  Max.   :1.319e+09   Max.   :113523.1
##
```

Next, calculate mean of life expectancy by continent and by year. This results in a table.

```
Mlife <- tapply(gapminder$lifeExp, list(gapminder$continent, gapminder$year), mean)
Mlife
```

```
##              1952     1957     1962     1967     1972     1977     1982
## Africa   39.13550 41.26635 43.31944 45.33454 47.45094 49.58042 51.59287
## Americas 53.27984 55.96028 58.39876 60.41092 62.39492 64.39156 66.22884
## Asia     46.31439 49.31854 51.56322 54.66364 57.31927 59.61056 62.61794
## Europe   64.40850 66.70307 68.53923 69.73760 70.77503 71.93777 72.80640
## Oceania  69.25500 70.29500 71.08500 71.31000 71.91000 72.85500 74.29000
##              1987     1992     1997     2002     2007
## Africa   53.34479 53.62958 53.59827 53.32523 54.80604
## Americas 68.09072 69.56836 71.15048 72.42204 73.60812
## Asia     64.85118 66.53721 68.02052 69.23388 70.72848
## Europe   73.64217 74.44010 75.50517 76.70060 77.64860
## Oceania  75.32000 76.94500 78.19000 79.74000 80.71950
```

1.3 Practice with HURDAT2

Question 2: What is the maximum sustained 10-m wind for each Atlantic storm in 2005?

```
#### Answer
```

```
#### Answer
AL05 <- subset(AL, Year==2005)
tapply(AL05$Wind,list(AL05$Name, AL05$Year),max, na.rm=TRUE)
```

```
##           2005
## ALPHA       45
## ARLENE      60
## BETA       100
## BRET        35
## CINDY       65
## DELTA       60
## DENNIS     130
## EMILY      140
## EPSILON     75
## FRANKLIN    60
## GAMMA       45
## GERT        40
## HARVEY      55
## IRENE       90
## JOSE        50
## KATRINA    150
## LEE         35
## MARIA      100
## NATE        80
## NINETEEN    30
## OPHELIA     75
## PHILIPPE    70
## RITA       155
## STAN        70
## TAMMY       45
## TEN         30
## TWENTY-TWO  40
## UNNAMED     45
## VINCE       65
## WILMA      160
## ZETA        50
```

2. `lapply` and `sapply` functions

To apply a function to each list element, we use `lapply` or `sapply` functions, depending on the desired outcome. `lapply` always returns the results in list; `l` means “list”. `sapply` returns the results in a vector or matrix if all returned values have the same length; `s` means “simplify”.

Get the maximum sustained 10-m wind for Katrina (2005).

```
Kt <- subset(AL, Year==2005 & Name== "KATRINA")
lapply(Kt,summary)
```

```
## $Key
##    Length     Class      Mode
##        34 character character
##
## $Name
##    Length     Class      Mode
##        34 character character
##
## $DateTime
##    Length     Class      Mode
##        34 character character
##
## $Record
##    Length     Class      Mode
##        34 character character
##
## $Status
##    Length     Class      Mode
##        34 character character
##
## $Lat
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.10   24.82   26.00   27.81   29.45   40.10
##
## $Lon
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -89.60  -88.45  -84.35  -83.70  -79.72  -75.10
##
## $Wind
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   25.00   40.00   70.00   74.12  100.00  150.00
##
## $Pressure

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   902.0   941.2   978.5   965.7   994.0  1008.0
##
## $NE34
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00   60.00   75.00   86.61  135.00  200.00       3
##
## $SE34
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00   60.00   75.00   83.55   95.00  200.00       3
##
## $SW34
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   50.00   58.23   95.00  150.00       3
##
## $NW34
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   50.00   62.42  130.00  180.00       3
##
## $NE50
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   50.00   45.65   72.50  120.00       3
##
## $SE50
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   60.00   46.29   75.00  120.00       3
##
## $SW50
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   20.00   29.03   62.50   75.00       3
##
## $NW50
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   20.00   34.52   72.50  120.00       3
##
## $NE64
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   26.94   47.50   90.00       3
##
## $SE64
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   25.81   40.00   90.00       3
##
## $SW64
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   18.23   32.50   60.00       3
##
## $NW64
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   21.61   32.50   90.00       3
##
## $Year
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    2005    2005    2005    2005    2005    2005
```

```
sapply(Kt,summary)
```

```
## $Key
##    Length     Class      Mode
##        34 character character
##
## $Name
##    Length     Class      Mode
##        34 character character
##
## $DateTime
##    Length     Class      Mode
##        34 character character
##
## $Record
##    Length     Class      Mode
##        34 character character
##
## $Status
##    Length     Class      Mode
##        34 character character
##
## $Lat
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.10   24.82   26.00   27.81   29.45   40.10
##
## $Lon
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -89.60  -88.45  -84.35  -83.70  -79.72  -75.10
##
## $Wind
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   25.00   40.00   70.00   74.12  100.00  150.00
##
## $Pressure

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   902.0   941.2   978.5   965.7   994.0  1008.0
##
## $NE34
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00   60.00   75.00   86.61  135.00  200.00       3
##
## $SE34
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00   60.00   75.00   83.55   95.00  200.00       3
##
## $SW34
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   50.00   58.23   95.00  150.00       3
##
## $NW34
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   50.00   62.42  130.00  180.00       3
##
## $NE50
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   50.00   45.65   72.50  120.00       3
##
## $SE50
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   60.00   46.29   75.00  120.00       3
##
## $SW50
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   20.00   29.03   62.50   75.00       3
##
## $NW50
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   20.00   34.52   72.50  120.00       3
##
## $NE64
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   26.94   47.50   90.00       3
##
## $SE64
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   25.81   40.00   90.00       3
##
## $SW64
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   18.23   32.50   60.00       3
##
## $NW64
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   21.61   32.50   90.00       3
##
## $Year
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    2005    2005    2005    2005    2005    2005
```

3. `apply` function operates on a matrix

3.1 Apply a function to every row

Such as calculating mean 34-kt wind maximum extent (in nautical miles) for Katrina

```
SubKt <- Kt[,10:13]
# Take mean
apply(SubKt,1,mean)
```

```
##  44066  44067  44068  44069  44070  44071  44072  44073  44074  44075
##   0.00   0.00   0.00  30.00  30.00  30.00  30.00  50.00  62.50     NA
##  44076  44077  44078  44079  44080  44081  44082  44083  44084  44085
##  57.50  55.00  55.00  60.00  78.75 110.00 110.00 112.50 120.00 146.25
##  44086  44087  44088  44089  44090  44091  44092  44093  44094  44095
## 156.25 171.25 182.50 175.00     NA 162.50     NA 120.00  76.25  72.50
##  44096  44097  44098  44099
##   0.00   0.00   0.00   0.00
```

```
apply(SubKt,1,mean, na.rm=TRUE)
```

```
##  44066  44067  44068  44069  44070  44071  44072  44073  44074  44075
##   0.00   0.00   0.00  30.00  30.00  30.00  30.00  50.00  62.50    NaN
##  44076  44077  44078  44079  44080  44081  44082  44083  44084  44085
##  57.50  55.00  55.00  60.00  78.75 110.00 110.00 112.50 120.00 146.25
##  44086  44087  44088  44089  44090  44091  44092  44093  44094  44095
## 156.25 171.25 182.50 175.00    NaN 162.50    NaN 120.00  76.25  72.50
##  44096  44097  44098  44099
##   0.00   0.00   0.00   0.00
```

3.2 Apply a function to every column

Such as calculating the range of longitude, latitude, maximum sustained 10-m wind, and minimum sea-level pressure for Katrina

```
SubKt <- Kt[,6:9]
# Take range
apply(SubKt,2,range)
```

```
##       Lat   Lon Wind Pressure
## [1,] 23.1 -89.6   25      902
## [2,] 40.1 -75.1  150     1008
```