Subsetting Summary

May 4, 2018

1 .Subsetting Vector
Load the data for example.

```
nba <- read.csv("nba2017-players.csv", header = TRUE)
player = as.vector(nba$player)
head(player)

## [1] "Al Horford"        "Amir Johnson"      "Avery Bradley"
## [4] "Demetrius Jackson" "Gerald Green"      "Isaiah Thomas"
```

player is a vector of player names.
(1)Numeric index

```
player[1]

## [1] "Al Horford"

player[c(1,2,3)]

## [1] "Al Horford"    "Amir Johnson"  "Avery Bradley"
```

(2)Logical index
when the length of index vector is same as length of vector a

```
a = c(1,2,3,4,5,6,7)
index = c(TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, TRUE)
a[index]

## [1] 1 2 3 7
```

Return numbers when index is TRUE
when the length of index vector is not same as length of vector a, it will be recycled to be the same length.

```
a = c(1,2,3,4,5,6,7)
## shorter length
index = c(TRUE, FALSE)
a[index]

## [1] 1 3 5 7
```

(3)Character index

```
names(a) = c("one","two","three","four","five","six","seven") ### assign a name for each number
print(a) # the vector looks like this now

##   one   two three  four  five   six seven
##     1     2     3     4     5     6     7
```

```
a[c("one", "four")] # using character index to extract elements
```

```
##  one four
##    1    4
```

Factor is pretty much similar with vectors in terms of sub setting
2. Subsetting a Data Frame: [row index, column index]
example with nba data

```
head(nba)
```

```
##              player team position height weight age experience
## 1        Al Horford  BOS        C     82    245  30          9
## 2      Amir Johnson  BOS       PF     81    240  29         11
## 3     Avery Bradley  BOS       SG     74    180  26          6
## 4 Demetrius Jackson  BOS       PG     73    201  22          0
## 5      Gerald Green  BOS       SF     79    205  31          9
## 6     Isaiah Thomas  BOS       PG     69    185  27          5
##                           college   salary games minutes points points3
## 1         University of Florida 26540100    68    2193    952      86
## 2                               12000000    80    1608    520      27
## 3 University of Texas at Austin  8269663    55    1835    894     108
## 4      University of Notre Dame  1450000     5      17     10       1
## 5                                1410598    47     538    262      39
## 6      University of Washington  6587132    76    2569   2199     245
##   points2 points1
## 1     293     108
## 2     186      67
## 3     251      68
## 4       2       3
## 5      56      33
## 6     437     590
```

(1)Numeric index

```
nba[c(1,2,3), ] ## first three rows
```

```
##          player team position height weight age experience
## 1    Al Horford  BOS        C     82    245  30          9
## 2  Amir Johnson  BOS       PF     81    240  29         11
## 3 Avery Bradley  BOS       SG     74    180  26          6
##                          college   salary games minutes points points3
## 1         University of Florida 26540100    68    2193    952      86
## 2                               12000000    80    1608    520      27
## 3 University of Texas at Austin  8269663    55    1835    894     108
##   points2 points1
## 1     293     108
## 2     186      67
## 3     251      68
```

```
head(nba[,c(1,2,3)]) ## first three columns
```

```
##              player team position
## 1        Al Horford  BOS        C
## 2      Amir Johnson  BOS       PF
## 3     Avery Bradley  BOS       SG
## 4 Demetrius Jackson  BOS       PG
## 5      Gerald Green  BOS       SF
## 6     Isaiah Thomas  BOS       PG
```

```
nba[c(1,2,3),c(1,2,3)]
```

```
##          player team position
## 1    Al Horford  BOS        C
## 2  Amir Johnson  BOS       PF
## 3 Avery Bradley  BOS       SG
```

```
# extracting observations with position is center. This is an example of numeric
#subseting since which function returns
# numeric index which meet the criteria.
head(nba[which(nba$position == "C"), ])
```

```
##              player team position height weight age experience
## 1        Al Horford  BOS        C     82    245  30          9
## 12     Kelly Olynyk  BOS        C     84    238  25          3
## 15     Tyler Zeller  BOS        C     84    253  27          4
## 16    Channing Frye  CLE        C     83    255  33         10
## 20      Edy Tavares  CLE        C     87    260  24          1
## 30 Tristan Thompson  CLE        C     81    238  25          5
##                           college   salary games minutes points points3
## 1          University of Florida 26540100    68    2193    952      86
## 12            Gonzaga University  3094014    75    1538    678      68
## 15  University of North Carolina  8000000    51     525    178       0
## 16         University of Arizona  7806971    74    1398    676     137
## 20                                   5145     1      24      6       0
## 30 University of Texas at Austin 15330435    78    2336    630       0
##    points2 points1
## 1      293     108
## 12     192      90
## 15      78      22
## 16     101      63
## 20       3       0
## 30     262     106
```

(2)Logical index

```
## extracting observations with position is center. This is an example of
##logical subsetting wince nba£position == "C"
## returns a vector of TRUE / FALSE
head(nba[nba$position == "C",])
```

```
##              player team position height weight age experience
## 1        Al Horford  BOS        C     82    245  30          9
## 12     Kelly Olynyk  BOS        C     84    238  25          3
## 15     Tyler Zeller  BOS        C     84    253  27          4
## 16    Channing Frye  CLE        C     83    255  33         10
## 20      Edy Tavares  CLE        C     87    260  24          1
## 30 Tristan Thompson  CLE        C     81    238  25          5
##                           college   salary games minutes points points3
## 1          University of Florida 26540100    68    2193    952      86
## 12            Gonzaga University  3094014    75    1538    678      68
## 15  University of North Carolina  8000000    51     525    178       0
## 16         University of Arizona  7806971    74    1398    676     137
## 20                                   5145     1      24      6       0
## 30 University of Texas at Austin 15330435    78    2336    630       0
##    points2 points1
## 1      293     108
```

```
## 12     192      90
## 15      78      22
## 16     101      63
## 20       3       0
## 30     262     106
```

(3)Character index

```
head(nba[ ,c("player","salary")])
```

```
##              player   salary
## 1        Al Horford 26540100
## 2      Amir Johnson 12000000
## 3     Avery Bradley  8269663
## 4 Demetrius Jackson  1450000
## 5      Gerald Green  1410598
## 6     Isaiah Thomas  6587132
```

Using dollar sign to extract columns as vectors or factors.. The reason why team of nba is factor is when I read the data, I didn't specify the type of columns, so R automatically stored team column as factor.

```
head(nba$team)
```

```
## [1] BOS BOS BOS BOS BOS BOS
## 30 Levels: ATL BOS BRK CHI CHO CLE DAL DEN DET GSW HOU IND LAC LAL ... WAS
```

```
head(nba$age)
```

```
## [1] 30 29 26 22 31 27
```

3. Subsetting a Tibble: Modern type of Data Frame
convert data frame into tibble

```
library(dplyr)
```

```
##
## Attaching package:  'dplyr'
## The following objects are masked from 'package:stats':
##
##    filter, lag
## The following objects are masked from 'package:base':
##
##    intersect, setdiff, setequal, union
```

```
iris_tb = tbl_df(iris) # convert data frame into tibble
iris_tb
```

```
## # A tibble: 150 x 5
##    Sepal.Length Sepal.Width Petal.Length Petal.Width Species
##           <dbl>       <dbl>        <dbl>       <dbl> <fct>
##  1          5.1         3.5          1.4         0.2 setosa
##  2          4.9         3            1.4         0.2 setosa
##  3          4.7         3.2          1.3         0.2 setosa
##  4          4.6         3.1          1.5         0.2 setosa
##  5          5           3.6          1.4         0.2 setosa
##  6          5.4         3.9          1.7         0.4 setosa
##  7          4.6         3.4          1.4         0.3 setosa
##  8          5           3.4          1.5         0.2 setosa
```

```
##  9          4.4         2.9          1.4         0.2 setosa
## 10          4.9         3.1          1.5         0.1 setosa
## # ... with 140 more rows
```

```
iris_tb[[1]] ## return the first column as a vector
```

```
##   [1] 5.1 4.9 4.7 4.6 5.0 5.4 4.6 5.0 4.4 4.9 5.4 4.8 4.8 4.3 5.8 5.7 5.4
##  [18] 5.1 5.7 5.1 5.4 5.1 4.6 5.1 4.8 5.0 5.0 5.2 5.2 4.7 4.8 5.4 5.2 5.5
##  [35] 4.9 5.0 5.5 4.9 4.4 5.1 5.0 4.5 4.4 5.0 5.1 4.8 5.1 4.6 5.3 5.0 7.0
##  [52] 6.4 6.9 5.5 6.5 5.7 6.3 4.9 6.6 5.2 5.0 5.9 6.0 6.1 5.6 6.7 5.6 5.8
##  [69] 6.2 5.6 5.9 6.1 6.3 6.1 6.4 6.6 6.8 6.7 6.0 5.7 5.5 5.5 5.8 6.0 5.4
##  [86] 6.0 6.7 6.3 5.6 5.5 5.5 6.1 5.8 5.0 5.6 5.7 5.7 6.2 5.1 5.7 6.3 5.8
## [103] 7.1 6.3 6.5 7.6 4.9 7.3 6.7 7.2 6.5 6.4 6.8 5.7 5.8 6.4 6.5 7.7 7.7
## [120] 6.0 6.9 5.6 7.7 6.3 6.7 7.2 6.2 6.1 6.4 7.2 7.4 7.9 6.4 6.3 6.1 7.7
## [137] 6.3 6.4 6.0 6.9 6.7 6.9 5.8 6.8 6.7 6.7 6.3 6.5 6.2 5.9
```

```
iris_tb[1]## returns the first column as tibble
```

```
## # A tibble: 150 x 1
##    Sepal.Length
##           <dbl>
##  1          5.1
##  2          4.9
##  3          4.7
##  4          4.6
##  5          5
##  6          5.4
##  7          4.6
##  8          5
##  9          4.4
## 10          4.9
## # ... with 140 more rows
```

```
iris_tb$Sepal.Length  ## returns Spepal.length as vector
```

```
##   [1] 5.1 4.9 4.7 4.6 5.0 5.4 4.6 5.0 4.4 4.9 5.4 4.8 4.8 4.3 5.8 5.7 5.4
##  [18] 5.1 5.7 5.1 5.4 5.1 4.6 5.1 4.8 5.0 5.0 5.2 5.2 4.7 4.8 5.4 5.2 5.5
##  [35] 4.9 5.0 5.5 4.9 4.4 5.1 5.0 4.5 4.4 5.0 5.1 4.8 5.1 4.6 5.3 5.0 7.0
##  [52] 6.4 6.9 5.5 6.5 5.7 6.3 4.9 6.6 5.2 5.0 5.9 6.0 6.1 5.6 6.7 5.6 5.8
##  [69] 6.2 5.6 5.9 6.1 6.3 6.1 6.4 6.6 6.8 6.7 6.0 5.7 5.5 5.5 5.8 6.0 5.4
##  [86] 6.0 6.7 6.3 5.6 5.5 5.5 6.1 5.8 5.0 5.6 5.7 5.7 6.2 5.1 5.7 6.3 5.8
## [103] 7.1 6.3 6.5 7.6 4.9 7.3 6.7 7.2 6.5 6.4 6.8 5.7 5.8 6.4 6.5 7.7 7.7
## [120] 6.0 6.9 5.6 7.7 6.3 6.7 7.2 6.2 6.1 6.4 7.2 7.4 7.9 6.4 6.3 6.1 7.7
## [137] 6.3 6.4 6.0 6.9 6.7 6.9 5.8 6.8 6.7 6.7 6.3 6.5 6.2 5.9
```

double brackets subsetting iris return vectors. single brackets subsetting iris return tibbles.