Data Validation and Cleaning

Statistics 427: R Programming

Module 10

2020

Data validation

Data errors: occurs when data values are not appropriate for the `R` statements that are specified in a command. `R` will usually not produce any output when an error occurs and will attempt to relay what went wrong, all written in red in the console.

Syntax errors occur when program statements do not conform to the rules of the `R` language. Examples are misspelled words, unmatched quotes, missing commas, invalid operations, incorrect data type for specified command, occasionally missing data values, and many more (I see a new error almost every time I use `R`).

Validating data

Some `R` procedures can be used to detect invalid data; we can use them to attempt to ascertain whether or not we have valid data.

`summary()`, `table()`, `is.na()`, `complete.cases()`, `na.omit()`, and `contents()` (in `Hmisc` package) will be some of the functions used for validation.

Data within acceptable range

To show if numerical data are within an acceptable range of values, use `summary()`; it uses supplies summaries of vectors, data frames, or models. It calculates and displays mean and the 5 Number Summary (median, Q1, Q3, min and max). The min and max is where you can find if values are within the acceptable range

General form of summary()‘

```
summary(object, ...)
```

`object`: data frame or a vector
`...`: more options; `?summary` for options

Missing or invalid values of categorical values

To show if categorical variables have missing or invalid values, use `table()`. `table()` uses the cross-classifying factors to build a contingency table of the counts at each combination of factor levels.

General form of `table()`

```
table(object1,object2,useNA = c("no","ifany","always"),...)
```

`object1` and `object2`: vector names
`useNA`: whether or not to include NA values in the table
`...`: more options; `?table` for options

Missing values of all types

To see if values are NA, `is.na()` indicates which elements are missing.

General form of `is.na()`

`is.na(x)` where `x` is an object (usually a vector)

The generic function `is.na()` indicates which elements in the vector are missing.
The generic function `is.na<-` sets elements in the vector to NA (missing).

For cleaning, `is.na()` will work if you have values that should be considered missing.

`which()` with `is.na()` indicates which element are missing

Complete number of observations

The function `complete.cases()` returns a logical vector indicating which cases are complete. Use with a ! before the `complete.cases()` command will show the cases that are not complete.

General form of `complete.cases()`

`complete.cases(x)` where `x` is a data frame or vector

You can use Boolean logic operators to find the incomplete cases in the data set with use of indices.

Missing values

When dealing with missing data points, you can choose to leave the missing values in the dataset and tell R to ignore them in calculations.

```
x <- c(1,2,NA,3)
mean(x) # returns NA
```

```
[1] NA
```

```
mean(x, na.rm=TRUE) # returns 2
```

```
[1] 2
```

```
var(x,na.rm=T)
```

```
[1] 1
```

```
sd(x,na.rm=T)
```

```
[1] 1
```

```
summary(x,na.rm=T)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
    1.0     1.5     2.0     2.0     2.5     3.0       1
```

`complete.cases()`

The function `complete.cases()` returns a logical vector indicating which cases are complete. Use with a ! before the `complete.cases()` command will show the cases that are not complete. `mydata[!complete.cases(mydata),]`
The function `na.omit()` returns the object with list-wise deletion of missing values. `na.omit()` can be used to create a dataset without the missing values.

`for` loop to find missing values

The `for` loop will go through the dataset and look for missing values and display which values are missing from which variable. The loop will not tell you which observations are missing but will tell you how many from each variable are missing (*if* they are coded properly as NA to begin with)

```
for (Var in names(mydata)) {
missing <- sum(is.na(mydata[,Var]))
if (missing > 0) {
print(c(Var,missing))
}
}
```

The best validation function ever

However, with all of the previous methods, there is one that (in my opinion at this point in time) is the best for validation. It is in the `Hmisc` package, using the `contents()` command.

`contents()` creates an object containing the following attributes of the variables from a data frame: names, labels (if any), units (if any), number of factor levels (if any), factor levels, class, storage mode, and number of NAs. There are options for sorting the variables. `html(contents(dataframe))` creates HTML code for displaying the results. This code has hyperlinks so that if the user clicks on the number of levels the browser jumps to the correct part of a table of factor levels for all the factor variables.

General form of `contents()`

```
contents(object,sortlevels=FALSE,id=NULL,range=NULL,values=NULL,...)
```

`object`: a data frame; for html is an object created by contents
`sortlevels`: TRUE or T (default) will sort levels of all factor variables into alphabetic order. This is especially useful when two variables use the same levels but in different orders. They will still be recognized by the html method as having identical levels if sorted.
`id`: an optional subject ID variable name that if present will cause the number of unique IDs to be printed in the contents header
`range`: an optional variable name that if present in object will cause its range to be printed in the contents header
`values`: an optional variable name that if present in object will cause its unique values to be printed in the contents header

General form of `print()` for use with `contents()`

```
print(x,sort=c('none','names','labels','NAs'),prlevels=TRUE,maxlevels=Inf,number=FALSE,...)
```

`x`: an object created by `contents()` via assignment statement
`sort`: Print the variables in their original order in the data frame (default). Specify one of "names", "labels", or "NAs" to sort the variables by, respectively, alphabetically by names, alphabetically by labels, or by increasing order of number of missing values. For contents.list, sort may also be the value "vars" to cause sorting by the number of variables in the dataset
`prlevels`: set to FALSE to not print all levels of factor variables
`maxlevels`: maximum number of levels to print for a factor variable
`number`: TRUE to have the print and latex methods number the variables by their order in the data frame (default)

General form of `html()` for use with `contents()`

```
html(object,...)
```

`object`: a data frame
`...`: other options

Drawbacks of `contents()`

One issue with using `contents()` is that is only shows levels for categorical variables. To check invalid values of numerical variables, you would need to also run `summary()` on just the numerical variables.

There are many redundancies between all the methods; I personally like `contents()` and `summary()` to do validation.

Cleaning

Cleaning can be done programmatically with the assignment statement, within the spreadsheet, or with the `fix()` command. `fix()` opens a spreadsheet window that looks similar to the window from `View()`. The spreadsheet can be edited and such for use only in the current session of `R` (unless you either write over (NOT RECOMMENDED!). To save any changes made with `fix()`, you could use `write.csv()` or `write.table()` to save that dataset, otherwise, when your `R` session is finished, the changes are not saved.

General form of `fix()`

`fix(object)`

`object`: a data frame

What will happen is that an editor window (that looks a bit like a spreadsheet) will open and you can make changes in the object. It will not override the raw datafile that was used to read in the data but as long as the object is in the environment, it will be the changed version.

Programmatically cleaning data

To do this, we use variable creation methods (assignment statement), transformations, and conditional statements (logic) with the `if` family functions

Upper and lower case

Convert text to uppercase and lower case

`toupper(x)` or `tolower(x)` where `x` is a quantity or vector of characters

Dates in `R` I

Dates are tricky buggers in any program. `R` makes the process a little less crazy, even if it looks like there are so many ways to manipulate dates in `R`.

`as.Date(x,...)` where `x` is an object to be converted, object must be inside quotes

```
dt1 <- as.Date("2012-07-22")
dt1
```

```
[1] "2012-07-22"
```

Dates in `R` II

Non-standard formats must be specified with a `format` option, which is `'%m/%d/%Y'` if doing `mmddyyyy`. Other format options are available, just switch around some of the letters `m` and `d`. To see list of format symbols, type `?as.Date`

```r
dt2 <- as.Date("04/20/2011",format="%m/%d/%Y")
dt2
```

```
[1] "2011-04-20"
```

```r
dt3 <- as.Date("October 6, 2010",format="%B %d, %Y")
dt3
```

```
[1] "2010-10-06"
```

Dates in R III

```r
dt1 - dt2
```

```
Time difference of 459 days
```

```r
# Time difference of 459 days
difftime(dt1,dt2,units='weeks')
```

```
Time difference of 65.57143 weeks
```

```r
# Time difference of 65.57 weeks
dt2 + 10
```

```
[1] "2011-04-30"
```

```r
dt2 - 10
```

```
[1] "2011-04-10"
```

Dates in R IV

Create a vector of dates and find the intervals between them

```r
three.dates <- as.Date(c("2010-07-22","2011-04-20","2012-10-06"))
three.dates
```

```
[1] "2010-07-22" "2011-04-20" "2012-10-06"
```

```r
diff(three.dates)
```

```
Time differences in days
[1] 272 535
```

```r
## create a sequence of dates
six.weeks <- seq(dt1,length=6,by='week'); six.weeks
```

```
[1] "2012-07-22" "2012-07-29" "2012-08-05" "2012-08-12" "2012-08-19"
[6] "2012-08-26"
```

```r
six.weeks <- seq(dt1,length=6,by=14); six.weeks
```

```
[1] "2012-07-22" "2012-08-05" "2012-08-19" "2012-09-02" "2012-09-16"
[6] "2012-09-30"
```

```r
six.weeks <- seq(dt1,length=6,by='2 weeks'); six.weeks
```

```
[1] "2012-07-22" "2012-08-05" "2012-08-19" "2012-09-02" "2012-09-16"
[6] "2012-09-30"
```

Replacing values in a dataset I

In the package `DataCombine`, there are methods for replacing values. The first is `FindReplace`, a function to replace multiple patterns found in a character string column of a data frame.

```
ABData <- data.frame(a=c("London, UK","Oxford, UK","Berlin, DE","Hamburg, DE","Oslo, NO"),b=c(8,0.1,3,2
ABData
```

```
            a   b
1  London, UK 8.0
2  Oxford, UK 0.1
3  Berlin, DE 3.0
4 Hamburg, DE 2.0
5    Oslo, NO 1.0
```

Replacing values in a dataset II

Replace the UK and DE parts of the strings with England and Germany. So I create a data frame with two columns. The first records the pattern and the second records what I want to replace the pattern with some replacement vectors, then use `FindReplace` to make the replacements all at once. If you set `exact=T` then FindReplace will only replace exact pattern matches. Also, you can set `vector=T` to return only a vector of the column you replaced (the Var column), rather than the whole data frame.

```
Replaces <- data.frame(from=c("UK","DE"),to=c("England","Germany"))
library(DataCombine) # must install first
ABNewDF <- FindReplace(data=ABData,Var="a",replaceData=Replaces,from="from",to ="to",exact=F)
ABNewDF
```

```
                a   b
1  London, England 8.0
2  Oxford, England 0.1
3  Berlin, Germany 3.0
4 Hamburg, Germany 2.0
5         Oslo, NO 1.0
```

Replacing values in a dataset III

The use of `if` functions will also be a way to replace values.

```
if(hours > 100) {net.price <- net.price * 0.9
```

Using `if` with `else`:

```
if(public) {
tot.price <- net.price * 1.06
} else {
tot.price <- net.price * 1.12
}
round(tot.price)
}
```

Replacing values in a dataset IV

`ifelse` is similar to if but works on more than a length-one vector

`ifelse(test,yes,no)`

`test`: the condition
`yes`: what to do if the condition is met
`no`: what to do if the condition is not met

`ifelse(salary==2401,salary=24401,salary)`

Validation and Cleaning example I

```
nonsales=read.csv('https://webpages.uidaho.edu/~renaes/Data/nonsales.csv',header=T)
```

`attach()` or `with()` could be used here; these examples will be using `with()`

Validation and Cleaning example II

`is.na()` will be used to find NAs and using `which()` to find where they are (observation number(s))

```
# using head() beacuse the list of 5+ pages long
head(is.na(nonsales),n=10L)
```

```
      EmployeeID Firstname Lastname Gender Salary Jobtitle Country Birthdate
 [1,]      FALSE     FALSE    FALSE  FALSE  FALSE    FALSE   FALSE     FALSE
 [2,]      FALSE     FALSE    FALSE  FALSE  FALSE    FALSE   FALSE     FALSE
 [3,]      FALSE     FALSE    FALSE  FALSE  FALSE    FALSE   FALSE     FALSE
 [4,]      FALSE     FALSE    FALSE  FALSE   TRUE    FALSE   FALSE     FALSE
 [5,]      FALSE     FALSE    FALSE  FALSE  FALSE    FALSE   FALSE     FALSE
 [6,]      FALSE     FALSE    FALSE  FALSE  FALSE    FALSE   FALSE     FALSE
 [7,]      FALSE     FALSE    FALSE  FALSE  FALSE    FALSE   FALSE     FALSE
 [8,]      FALSE     FALSE    FALSE  FALSE  FALSE    FALSE   FALSE     FALSE
 [9,]      FALSE     FALSE    FALSE  FALSE  FALSE    FALSE   FALSE     FALSE
[10,]      FALSE     FALSE    FALSE  FALSE  FALSE    FALSE   FALSE     FALSE
      Hiredate
 [1,]    FALSE
 [2,]    FALSE
 [3,]    FALSE
 [4,]    FALSE
 [5,]    FALSE
 [6,]    FALSE
 [7,]    FALSE
 [8,]    FALSE
 [9,]    FALSE
[10,]    FALSE
```

Validation and Cleaning example III

```
with(nonsales,which(is.na(EmployeeID)))
```

```
[1] 14
```

```
with(nonsales,which(is.na(Jobtitle)))
```

```
integer(0)
```

```
with(nonsales,which(is.na(Gender))) # integer(0) means there are no NAs
```

```
integer(0)
```

```
with(nonsales,which(is.na(Salary)))
```

```
[1] 4
```

```
with(nonsales,which(is.na(Hiredate)))
```

```
integer(0)
```

```
with(nonsales,which(is.na(Birthdate)))
```

```
integer(0)
```

Validation and Cleaning example IV

`complete.cases()` to find all complete cases. `which()` can also be used with it so that you can find the observation(s). This is usually best to use with the Boolean logic operator `!` to find the *incomplete* cases.

```
# using head() beacuse the list is long
head(complete.cases(nonsales),n=10L)
```

```
 [1]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

```
head(which(complete.cases(nonsales)),n=10L)
```

```
 [1]  1  2  3  5  6  7  8  9 10 11
```

```
head(nonsales[complete.cases(nonsales),],n=10L)
```

```
   EmployeeID Firstname   Lastname Gender Salary                Jobtitle
1      120101   Patrick         Lu      M 163040                Director
2      120104    Kareen Billington      F  46230  Administration Manager
3      120105       Liz      Povey      F  27110             Secretary I
5      120107    Sherie     Sheedy      F  30475    Office Assistant III
6      120108    Gladys     Gromek      F  27660  Warehouse Assistant II
7      120108  Gabriele      Baker      F  26495   Warehouse Assistant I
8      120110    Dennis   Entwisle      M  28615 Warehouse Assistant III
9      120111    Ubaldo   Spillane      M  26895       Security Guard II
10     120112     Ellis  Glattback      F  26550
11     120113       Riu     Horsey      F  26870       Security Guard II
   Country Birthdate  Hiredate
1       AU 18-Aug-76  1-Jul-03
2       au 11-May-54  1-Jan-81
3       AU 21-Dec-74  1-May-99
5       AU  1-Feb-78 21-Jan-53
6       AU 23-Feb-84  1-Aug-06
7       AU 15-Dec-86  1-Oct-06
8       AU 20-Nov-49  1-Nov-79
9       AU 23-Jul-49 99NOV1978
10      AU 17-Feb-69  1-Jul-90
11      AU 10-May-44  1-Jan-74
```

Validation and Cleaning example V

Here find the *incomplete* cases and use `summary()` to see if values are missing or outside feasible ranges; `summary()` does not need `with()` if I am summarizing the entire dataset

```
!complete.cases(nonsales)
```

```
  [1] FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [13] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [49] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [61] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [73] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [85] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [97] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[109] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[121] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[133] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[145] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[157] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[169] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[193] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[205] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[217] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[229] FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
which(!complete.cases(nonsales)) # shows which cases are incomplete
```

```
[1]  4 14
```

```
nonsales[!complete.cases(nonsales),] # shows all incomplete case data
```

```
   EmployeeID Firstname Lastname Gender Salary             Jobtitle Country
4      120106      John  Hornsey      M     NA  Office Assistant II      AU
14         NA    Austen  Ralston      M  29250 Service Assistant II      AU
   Birthdate Hiredate
4  23-Dec-44 1-Jan-74
14 13-Jun-59 1-Feb-80
```

```
summary(nonsales)
```

```
   EmployeeID      Firstname           Lastname           Gender
 Min.   :120101   Length:235         Length:235         Length:235
 1st Qu.:120668   Class :character   Class :character   Class :character
 Median :120740   Mode  :character   Mode  :character   Mode  :character
 Mean   :120688
 3rd Qu.:120799
 Max.   :121148
 NA's   :1
     Salary         Jobtitle           Country           Birthdate
 Min.   :  2401   Length:235         Length:235         Length:235
 1st Qu.: 27804   Class :character   Class :character   Class :character
 Median : 34020   Mode  :character   Mode  :character   Mode  :character
```

```
 Mean   : 43955
 3rd Qu.: 47252
 Max.   :433800
 NA's   :1
   Hiredate
 Length:235
 Class :character
 Mode  :character
```

Validation and Cleaning example VI

The `for` loop from earlier will look for missing data (but not invalid data points) and it can be used with any dataset

```
for (Var in names(nonsales)) {
  missing <- sum(is.na(nonsales[,Var]))
  if (missing > 0) {
    print(c(Var,missing))
  }
}
```

```
[1] "EmployeeID" "1"
[1] "Salary" "1"
```

Validation and Cleaning example VI

Use `table()` to find invalid or missing values of categorical variables

```
with(nonsales,table(Gender,useNA="ifany"))
```

```
Gender
      F   G   M
  1 110   1 123
```

```
with(nonsales,table(Jobtitle,useNA="ifany"))
```

```
Jobtitle
                                  Account Manager              Accountant I
                         1                        1                         5
            Accountant II            Accountant III    Administration Manager
                         5                        2                         2
 Applications Developer I Applications Developer II Applications Developer IV
                         4                        3                         3
         Auditing Manager                Auditor I                Auditor II
                         1                        2                         2
              Auditor III       BI Administrator IV           BI Architect II
                         1                        1                         1
         BI Specialist II   Building Admin. Manager       Business Analyst II
                         1                        1                         2
     Business Analyst III          Cabinet Maker II         Cabinet Maker III
```

1	1	1
Chief Executive Officer	Chief Financial Officer	Chief Marketing Officer
1	1	1
Clerk I	Concession Assistant I	Concession Assistant II
2	1	1
Concession Assistant III	Concession Consultant I	Concession Consultant II
2	2	2
Concession Consultant III	Concession Director	Concession Manager
1	1	1
Corp. Comm. Manager	Corp. Comm. Specialist I	Corp. Comm. Specialist II
1	1	1
Director	Electrician II	Electrician III
3	2	1
Electrician IV	ETL Specialist I	ETL Specialist II
1	1	1
Events Manager	Finance Manager	Financial Analyst II
1	1	1
Financial Analyst III	Financial Controller I	Financial Controller II
1	1	1
Financial Controller III	HR Analyst II	HR Generalist I
3	2	1
HR Generalist II	HR Generalist III	HR Manager
1	1	1
HR Specialist I	HR Specialist II	IS Administrator I
2	1	1
IS Administrator II	IS Administrator III	IS Architect III
1	1	1
IS Director	Logistics Coordinator I	Logistics Coordinator II
1	1	2
Logistics Manager	Marketing Assistant I	Marketing Assistant II
2	1	5
Marketing Assistant III	Marketing Assistant IV	Marketing Manager
1	1	3
Office Administrator I	Office Administrator II	Office Assistant I
1	1	2
Office Assistant II	Office Assistant III	Office Assistant IV
2	3	1
Pricing Manager	Pricing Specialist	Project Manager
1	1	1
Purchasing Agent I	Purchasing Agent II	Purchasing Agent III
6	5	4
Purchasing Manager	Recruiter I	Recruiter II
3	1	1
Recruiter III	Recruitment Manager	Secretary I
1	1	1
Secretary II	Secretary III	Secretary IV
2	2	1
Security Guard I	Security Guard II	Security Manager
2	4	2
Senior Logistics Manager	Senior Marketing Manager	Senior Project Manager
2	1	1
Senior Strategist	Service Administrator I	Service Administrator III
1	2	1
Service Assistant I	Service Assistant II	Services Assistant IV

```
                     3                          2                          1
      Services Manager  Shipping Administrator I          Shipping Agent I
                     1                          1                          3
      Shipping Agent II        Shipping Agent III          Shipping Manager
                     3                          3                          5
         Strategist II      Systems Architect II      Systems Architect IV
                     1                          1                          1
      Technical Manager              Technician I             Technician II
                     1                          1                          1
                Trainee                  Trainer I                Trainer II
                    18                          2                          1
           Trainer III          Training Manager            Vice President
                     1                          1                          1
  Warehouse Assistant I    Warehouse Assistant II   Warehouse Assistant III
                    10                          4                          2
 Warehouse Assistant IV         Warehouse Manager
                     1                          1
```

```
with(nonsales,table(Gender,Jobtitle,useNA="ifany")) # do more at one time
```

```
      Jobtitle
Gender    Account Manager Accountant I Accountant II Accountant III
        0               0            0             0              0
     F  1               0            2             1              1
     G  0               0            0             0              0
     M  0               1            3             4              1
      Jobtitle
Gender Administration Manager Applications Developer I
                           0                        0
     F                     1                        2
     G                     0                        0
     M                     1                        2
      Jobtitle
Gender Applications Developer II Applications Developer IV Auditing Manager
                             0                         0                0
     F                       0                         2                1
     G                       0                         0                0
     M                       3                         1                0
      Jobtitle
Gender Auditor I Auditor II Auditor III BI Administrator IV BI Architect II
              0          0           0                   0                0
     F        1          0           1                   0                1
     G        0          0           0                   0                0
     M        1          2           0                   1                0
      Jobtitle
Gender BI Specialist II Building Admin. Manager Business Analyst II
                      0                       0                   0
     F                0                       1                   0
     G                0                       0                   0
     M                1                       0                   2
      Jobtitle
Gender Business Analyst III Cabinet Maker II Cabinet Maker III
                          0                0                 0
     F                    1                0                 0
     G                    0                0                 0
```

```
     M                    0                1                 1
      Jobtitle
Gender Chief Executive Officer Chief Financial Officer Chief Marketing Officer
                             0                       0                       0
     F                       0                       0                       1
     G                       0                       0                       0
     M                       1                       1                       0
      Jobtitle
Gender Clerk I Concession Assistant I Concession Assistant II
             0                       0                       0
     F       2                       1                       0
     G       0                       0                       0
     M       0                       0                       1
      Jobtitle
Gender Concession Assistant III Concession Consultant I
                              0                       0
     F                        1                       2
     G                        0                       0
     M                        1                       0
      Jobtitle
Gender Concession Consultant II Concession Consultant III Concession Director
                              0                        0                    0
     F                        1                        1                    0
     G                        0                        0                    0
     M                        1                        0                    1
      Jobtitle
Gender Concession Manager Corp. Comm. Manager Corp. Comm. Specialist I
                        0                   0                        0
     F                  1                   0                        0
     G                  0                   0                        0
     M                  0                   1                        1
      Jobtitle
Gender Corp. Comm. Specialist II Director Electrician II Electrician III
                               1        0              0               0
     F                         0        0              1               1
     G                         0        0              0               0
     M                         0        3              1               0
      Jobtitle
Gender Electrician IV ETL Specialist I ETL Specialist II Events Manager
                    0                0                 0              0
     F              0                0                 1              0
     G              0                0                 0              0
     M              1                1                 0              1
      Jobtitle
Gender Finance Manager Financial Analyst II Financial Analyst III
                     0                    0                     0
     F               0                    1                     0
     G               0                    0                     0
     M               1                    0                     1
      Jobtitle
Gender Financial Controller I Financial Controller II Financial Controller III
                            0                       0                        0
     F                      1                       0                        3
     G                      0                       0                        0
```

```
     M                     0                     1                     0
      Jobtitle
Gender HR Analyst II HR Generalist I HR Generalist II HR Generalist III
                   0               0                0                 0
     F             2               0                1                 0
     G             0               0                0                 0
     M             0               1                0                 1
      Jobtitle
Gender HR Manager HR Specialist I HR Specialist II IS Administrator I
               0               0                0                  0
     F         1               1                1                  1
     G         0               0                0                  0
     M         0               1                0                  0
      Jobtitle
Gender IS Administrator II IS Administrator III IS Architect III IS Director
                         0                    0                 0           0
     F                   1                    0                 0           0
     G                   0                    0                 0           0
     M                   0                    1                 1           1
      Jobtitle
Gender Logistics Coordinator I Logistics Coordinator II Logistics Manager
                             0                       0                 0
     F                       1                       1                 0
     G                       0                       0                 0
     M                       0                       1                 2
      Jobtitle
Gender Marketing Assistant I Marketing Assistant II Marketing Assistant III
                           0                      0                       0
     F                     1                      2                       0
     G                     0                      0                       0
     M                     0                      3                       1
      Jobtitle
Gender Marketing Assistant IV Marketing Manager Office Administrator I
                            0                 0                      0
     F                      0                 1                      1
     G                      0                 0                      0
     M                      1                 2                      0
      Jobtitle
Gender Office Administrator II Office Assistant I Office Assistant II
                             0                  0                   0
     F                       1                  2                   0
     G                       0                  0                   0
     M                       0                  0                   2
      Jobtitle
Gender Office Assistant III Office Assistant IV Pricing Manager
                          0                   0               0
     F                    1                   0               1
     G                    0                   0               0
     M                    2                   1               0
      Jobtitle
Gender Pricing Specialist Project Manager Purchasing Agent I
                        0               0                  0
     F                  0               0                  3
     G                  0               0                  0
```

```
     M                  1               1                 3
      Jobtitle
Gender Purchasing Agent II Purchasing Agent III Purchasing Manager Recruiter I
                         0                    0                  0           0
     F                   4                    1                  3           1
     G                   0                    0                  0           0
     M                   1                    3                  0           0
      Jobtitle
Gender Recruiter II Recruiter III Recruitment Manager Secretary I Secretary II
                 0              0                    0           0            0
     F           1              0                    1           1            1
     G           0              0                    0           0            0
     M           0              1                    0           0            1
      Jobtitle
Gender Secretary III Secretary IV Security Guard I Security Guard II
                   0            0                0                 0
     F             2            1                0                 1
     G             0            0                0                 0
     M             0            0                2                 3
      Jobtitle
Gender Security Manager Senior Logistics Manager Senior Marketing Manager
                      0                       0                        0
     F                0                       2                        1
     G                1                       0                        0
     M                1                       0                        0
      Jobtitle
Gender Senior Project Manager Senior Strategist Service Administrator I
                           0                 0                       0
     F                     1                 0                       1
     G                     0                 0                       0
     M                     0                 1                       1
      Jobtitle
Gender Service Administrator III Service Assistant I Service Assistant II
                               0                  0                   0
     F                         0                  0                   0
     G                         0                  0                   0
     M                         1                  3                   2
      Jobtitle
Gender Services Assistant IV Services Manager Shipping Administrator I
                           0                0                       0
     F                     0                0                       1
     G                     0                0                       0
     M                     1                1                       0
      Jobtitle
Gender Shipping Agent I Shipping Agent II Shipping Agent III Shipping Manager
                      0                 0                  0                0
     F                2                 1                  2                2
     G                0                 0                  0                0
     M                1                 2                  1                3
      Jobtitle
Gender Strategist II Systems Architect II Systems Architect IV
                   0                    0                    0
     F             1                    1                    0
     G             0                    0                    0
```

```
     M             0                   0                   1
      Jobtitle
Gender Technical Manager Technician I Technician II Trainee Trainer I
                       0            0             0       0         0
     F                 1            0             0       6         1
     G                 0            0             0       0         0
     M                 0            1             1      12         1
      Jobtitle
Gender Trainer II Trainer III Training Manager Vice President
               0           0                0              0
     F         0           0                1              0
     G         0           0                0              0
     M         1           1                0              1
      Jobtitle
Gender Warehouse Assistant I Warehouse Assistant II Warehouse Assistant III
                           0                      0                       0
     F                     7                      3                       1
     G                     0                      0                       0
     M                     3                      1                       1
      Jobtitle
Gender Warehouse Assistant IV Warehouse Manager
                            0                 0
     F                      1                 0
     G                      0                 0
     M                      0                 1
```

Validation and Cleaning example VI

With all of the previous validation methods, the following is ALL you need for validation. It will execute ALL of validation for *categorical* variables

```
# install.packages("Hmisc")
library(Hmisc)
contents(nonsales)
```

```
Data frame:nonsales 235 observations and 9 variables    Maximum # NAs:1

             Storage NAs
EmployeeID   integer   1
Firstname  character   0
Lastname   character   0
Gender     character   0
Salary       integer   1
Jobtitle   character   0
Country    character   0
Birthdate  character   0
Hiredate   character   0
```

Validation and Cleaning example VII

Now the only other command needed is to look at invalid ranges of the numerical variables using `summary()`

```
with(nonsales,summary(EmployeeID))
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
 120101  120668  120740  120688  120799  121148       1
```

```
with(nonsales,summary(Salary))
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   2401   27804   34020   43955   47252  433800       1
```

Validation and Cleaning example VIII

Converting into dates and looking at `Birth_Date` and `Hire_Date` variables to see the format they are in.

```
tail(nonsales$Hiredate)
```

```
[1] "1-Jan-03" "1-Jan-74" "1-May-93" "1-Apr-06" "1-Sep-87" "1-Jan-98"
```

```
tail(nonsales$Birthdate)
```

```
[1] "2-Jul-64"  "19-Jun-44" "14-Feb-69" "9-Dec-86"  "28-May-69" "1-Jan-69"
```

Validation and Cleaning example IX

When looking at the `Birth_Date` and `Hire_Date` variables, notice that they are in quotes. That means that the variable(s) is being treated as a character rather than a date or numeric. We can create new variables by converting the vectors to dates and we also need date formats. With the formats, quotes are needed: `"%m/%d/%Y"` gives mm/dd/YYYY, etc. (web search for "R as.Date format" for more)

It appears the dates were entered as dd/mm/YYYY.

Validation and Cleaning example X

This will change the existing variables in the dataset that is in the `R` environment. **It will not change the original dataset you read in from**.

```
nonsales$Birthdate=as.Date(nonsales$Birthdate,format="%d/%m/%Y")
nonsales$Hiredate=as.Date(nonsales$Hiredate,format="%d/%m/%Y")
head(nonsales$Birthdate)
```

```
[1] NA NA NA NA NA NA
```

```
tail(nonsales$Hiredate)
```

```
[1] NA NA NA NA NA NA
```

Validation and Cleaning example XI

When dates are stored as a factor, we cannot manipulate the dates with numerical values. Now calculation of the differences between the (new) dates and check if dates are invalid (`hiredate>birthdate`).

```
nonsales$datediff=with(nonsales,difftime(Hiredate,Birthdate,units='days'))
k=which(nonsales$datediff<0)
nonsales[k,]
```

```
 [1] EmployeeID Firstname  Lastname   Gender     Salary     Jobtitle
 [7] Country    Birthdate  Hiredate   datediff
<0 rows> (or 0-length row.names)
```

Validation and Cleaning example XII

Cleaning data in R with `fix()`. The spreadsheet can be edited pretty much just like a normal spreadsheet (like Excel, etc.). The following is not going to run for this document as it is an interactive function.

```
# fix(nonsales)
```

Validation and Cleaning example XIII

Cleaning data programmatically using different functions.

```
# case tense
nonsales$Country=with(nonsales,toupper(Country))
# install.packages('DataCombine')
library(DataCombine)
ABData <- data.frame(a=c("London, UK","Oxford, UK","Berlin, DE","Hamburg, DE","Oslo, NO"),b=c(8,0.1,3,2
Replaces <- data.frame(from=c("UK","DE"),to=c("England","Germany"))
ABNewDF <- FindReplace(data=ABData,Var="a",replaceData=Replaces,from="from",to ="to",exact=F)
```

Validation and Cleaning example XIV

```
nonsales
```

```
    EmployeeID   Firstname       Lastname Gender Salary
1       120101     Patrick             Lu      M 163040
2       120104      Kareen     Billington      F  46230
3       120105         Liz          Povey      F  27110
4       120106        John        Hornsey      M     NA
5       120107      Sherie         Sheedy      F  30475
6       120108      Gladys         Gromek      F  27660
7       120108    Gabriele          Baker      F  26495
8       120110      Dennis       Entwisle      M  28615
9       120111      Ubaldo       Spillane      M  26895
10      120112       Ellis      Glattback      F  26550
11      120113         Riu         Horsey      F  26870
12      120114   Jeannette        Buddery      G  31285
13      120115        Hugh      Nichollas      M   2650
14          NA      Austen        Ralston      M  29250
15      120117        Bill       Mccleary      M  31670
16      120118      Darshi      Hartshorn      M  28090
17      120119         Lal        Elleman      M  30255
18      120120     Krishna         Peiris      F  27645
19      120190        Ivor     Czernezkyi      M  24100
20      120191     Jannene    Graham-Rowe      F   2401
21      120192     Anthony      Nichollas      M  26185
22      120193     Russell         Streit      M  24515
```

23	120194	Reece	Harwood	M	25985
24	120195	Jina	Fiocca	F	24990
25	120196	Merle	Hieds	F	24025
26	120197	Kerrin	Dillin	F	25580
27	120259	Anthony	Miller	M	433800
28	120260	Christine	Fletcher	F	207885
29	120262	Max	Crown	M	268455
30	120263	Bobby	Cleverley	M	42605
31	120264	Latonya	Croome	F	37510
32	120265	Wanda	Branly	F	51950
33	120266	Bao	Krafve	F	31750
34	120267	Belanda	Rink	F	28585
35	120268	Jacques	Villeneuve	M	76105
36	120269	Shrimatee	Kagolanu	F	52540
37	120270	Grezegorz	Nuss	M	48435
38	120271	Kenisha	Winge	F	43635
39	120272	Febin	Flow	M	34390
40	120273	Doris	Antonini	F	28455
41	120274	Angela	Landry	F	26840
42	120275	Brandy	Lattimer	F	32195
43	120276	Nicholas	Plybon	M	28090
44	120277	Wesley	Shirts	F	32645
45	120278	Binit	Jongleux	M	27685
46	120279	Kareema	Dunlap	F	32925
47	120280	Jaime	Laurer	F	36930
48	120656	Salley	Amos	F	42570
49	120657	Theresa	Weisbarth	F	36110
50	120658	Kenneth	Kennedy	M	42485
51	120659	Jay	Havasy	M	161290
52	120660	Robert	Smith	M	61125
53	120661	Cynthia	Racine	F	85495
54	120662	Lemonica	Burroughs	M	27045
55	120663	Anglar	Kornblith	F	56385
56	120664	Brock	Senchak	M	47605
57	120665	Jill	Leacock	F	80070
58	120666	John	Onuscheck	M	64555
59	120667	Edwin	Droste	M	29980
60	120668	Thyland	Dolan	M	47785
61	120669	Ronald	Hill	M	36370
62	120670	Odudu	Zisek	M	65420
63	120671	William	Latty	M	40080
64	120672	Verne	Guscott	M	60980
65	120673	Pearl	Santomaggio	F	35935
66	120677	Suad	Sochacki	F	65555
67	120678	Lucretta	Octetree	F	40035
68	120679	Chrisy	Cutucache	F	46190
69	120680	Raymondria	Desaulniers	F	27295
70	120681	Elery	Tolbet	M	30950
71	120682	Barbara	Kennedy	F	26760
72	120683	Deven	Kochneff	F	36315
73	120684	Suzon	Woyach	F	26960
74	120685	Anita	Howard	F	25130
75	120686	Berether	Tucker	F	26690
76	120687	Freda	Dannin	F	26800

77	120688	Lisa	Carcaterra	F	25905
78	120689	Katherine	Pongor	F	27780
79	120690	Taronda	Langston	F	25185
80	120691	Sek	Habres	M	49240
81	120692	Rit	Tregonning	M	32485
82	120693	Diaz	Tellam	M	26625
83	120694	Sharon	Leazer	F	27365
84	120695	Trent	Moffat	M	28180
85	120696	Peter	Pettolino	M	26615
86	120697	Madelaine	Fouche	F	29625
87	120698	Geoff	Kistanna	M	26160
88	120710	Timothy	Baltzell	M	54840
89	120711	Gloria	Drew	F	59130
90	120712	Elisabeth	Motashaw	F	63640
91	120713	Carston	Campbell	M	31630
92	120714	Robert	Dinley	M	62625
93	120715	Angelia	Neal	F	28535
94	120716	Kenneth	Juif	M	53015
95	120717	Jon	Sleva	M	30155
96	120718	Charles	Hennington	M	29190
97	120719	Roya	Ridley	F	87420
98	120720	John	Spingola	M	46580
99	120721	Dlutomi	Knust	F	29870
100	120722	Ishmar	Sheffield	M	32460
101	120723	Deanna	Olsen		33950
102	120724	Hampie	Brown	M	63705
103	120725	Robert	Whitlock	M	29970
104	120726	Lutezenia	Obermeyer	F	27380
105	120727	Donald	Marples	M	34925
106	120728	Kathryn	Borge	F	35070
107	120729	Kimberly	Howell	F	31495
108	120730	Woodson	Burt	M	30195
109	120731	Robert	Lerew	M	34150
110	120732	Kent	Uenking	M	35870
111	120733	Michael	Bezinque	M	31760
112	120734	Svein	Saylor	M	34270
113	120735	Brenda	Bilobran	F	61985
114	120736	Parie	Kiemle	F	63985
115	120737	Brenner	Toner	F	63605
116	120738	Huilun	Swaiti	F	30025
117	120739	Bryon	Cooper	M	36970
118	120740	Lisa	Koonce	F	35110
119	120741	Keisha	Court	F	36365
120	120742	Ronald	Shewitz	M	31020
121	120743	Chimena	Harrison	F	34620
122	120744	Alden	Feigenbaum	F	33490
123	120745	Barbara	Harvill	F	31365
124	120746	Kevie	Kimmerle	M	46090
125	120747	Zashia	Farthing	F	43590
126	120748	Nahliah	Post	F	48380
127	120749	Kevin	Niemann	M	26545
128	120750	Connie	Woods	F	32675
129	120751	AzaviOus	Mea	M	58200
130	120752	Jean-Claude	Van Damme	M	30590

131	120753	Ralph	Ferrari	M	47000
132	120754	John	Atkins	M	34760
133	120755	Elizabeth	Thoits	F	36440
134	120756	Wendy	Asta	F	52295
135	120757	Paul	Knopfmacher	M	38545
136	120758	Sal	Voltz	M	34040
137	120759	Nishan	Apr	M	36230
138	120760	Pamela	Miller	F	53475
139	120761	Tameaka	Akinfolarin	F	30960
140	120762	Marvin	Leone	M	30625
141	120763	Ramond	Capps	M	45100
142	120764	Steven	Worton	M	40450
143	120765	Nikeisha	Kokoszka	F	51950
144	120766	Janelle	Kempster	F	53400
145	120767	Legette	Terricciano	M	32965
146	120768	Roland	Rayburn	M	44955
147	120769	Abelino	Lightbourne	M	47990
148	120770	Julia	Pascoe	F	43930
149	120771	Wei	Xing Moore	F	36435
150	120772	Erich	Overdorff	M	27365
151	120773	Entrisse	Horne	F	27370
152	120774	Sue	El-Amin	F	45155
153	120775	Tanya	Thompson	F	41580
154	120776	Ratna	Silverthorne	M	32580
155	120777	Kary	Sacher	M	40955
156	120778	Angela	Gardner	F	43650
157	120779	Jennifer	Eggleston	F	43690
158	120780	Kimberly	Walcott	F	62995
159	120781	Sarah	Sitnik	F	32620
160	120782	Rilma	Sines	F	63915
161	120783	Davis	Karp	M	42975
162	120784	Jennifer	Pinol	F	35715
163	120785	Damesha	Donnell	F	48335
164	120786	Chris-Anne	Delafuente	F	32650
165	120787	Carl	Peachey	M	34000
166	120788	Smitty	Lisowe	M	33530
167	120789	Julius	Denhollem	M	39330
168	120790	Tara	O'Toole	F	53740
169	120791	Richard	Chiseloff	M	61115
170	120792	Omeba	Horne	F	54760
171	120793	William	Mamo	M	47155
172	120794	Samantha	Cross	F	51265
173	120795	David	Deacon	M	49105
174	120796	Philip	Kellis	M	47030
175	120797	Sherrie	Jones	F	43385
176	120798	Elizabeth	Ardskin	F	80755
177	120799	Jeffery	Stefandonovan	M	29070
178	120800	Fred	Benyami	M	80210
179	120801	Kathryn	Kennedy	F	40040
180	120802	U'Vonda	Parker	F	65125
181	120803	Victor	Droste	M	43630
182	120804	Ahmed	Zied	M	55400
183	120805	Robert	Walker	M	58530
184	120806	Lorna	Ousley	F	47285

```
185     120807    Gerlinde        Peppers      F  43325
186     120808      Marcel         Dupree      M  44425
187     120809    Chiorene         Marion      F  47155
188     120810       Loyal       Esguerra      M  58375
189     120811        Dale Bergeron-Jeter      M  43985
190     120812      Fauver         Arruza      M  45810
191     120813        John       Heinsler      M  50865
192     120814      Victor       Scroggin      M  59140
193     120815       Craig         Honore      M  31590
194     120816      Tessia           Hart      F  30485
195     120992        Lisa          Kicak      F  26940
196     120993    Lorraine      Boatright      F  26260
197     120994     Danelle       Sergeant      F  31645
198     120995    Lily-Ann          Gordo      F  34850
199     120996    Johannes           Wade      M  32745
200     120997        Mary       Donathan      F  27420
201     120998   Tondelayo      Benedicto      F  26330
202     120999    Sherelyn       Heilmann      F  27215
203     121000      Herman         Supple      M  48600
204     121001        Tony          House      M  43615
205     121002   Terry-Ann          Clark      F  26650
206     121003  Troyce Van      Der Wiele      M  26000
207     121004      Kellen          Smith      M  30895
208     121005    Yuh-Lang         Mclamb      M  25020
209     121006     Bernard        Bolster      M  26145
210     121007        John       Banaszak      M  27290
211     121008        Eron       Mckenzie      M  27875
212     121009      Robert        Goodwin      M  32955
213     121010      Donald           Lamp      M  25195
214     121011      Steven         Banchi      M  25735
215     121012     Carmelo         Broome      M  29575
216     121013        Seco       Hargrave      M  26675
217     121014     Donelle        Liguori      F  28510
218     121015      Wilson      Elmoslamy      M  26140
219     121016   Lutezenia       Sullivan      F  48075
220     121017     Gilbert      Arizmendi      M  29225
221     121125     Michael           Holt      M  25315
222     121126       James        Penhale      M  26015
223     121127       Keyna        Mangini      F  25435
224     121128      Glacia          Nazar      F  25405
225     121129       Yusef         Hafley      M  30945
226     121130        Gary        Herndon      M  25255
227     121131     William       Pantages      M  25445
228     121132   Shia-Ling      Digiorgio      M  24390
229     121133       Peter       Pringley      M  25405
230     121134        Paul         Tacosa      M  25585
231     121141    Henri Le           Bleu      M 194885
232     121142    Reginald        Steiber      M 156065
233     121146   Julieanne     Sangiorgio      F  29320
234     121147   Christine          Sneed      F  29145
235     121148       Shane          Sadig      M  52930
                    Jobtitle Country Birthdate Hiredate datediff
1                   Director      AU      <NA>     <NA>  NA days
2      Administration Manager      AU      <NA>     <NA>  NA days
```

3	Secretary I	AU	<NA>	<NA>	NA days
4	Office Assistant II	AU	<NA>	<NA>	NA days
5	Office Assistant III	AU	<NA>	<NA>	NA days
6	Warehouse Assistant II	AU	<NA>	<NA>	NA days
7	Warehouse Assistant I	AU	<NA>	<NA>	NA days
8	Warehouse Assistant III	AU	<NA>	<NA>	NA days
9	Security Guard II	AU	<NA>	<NA>	NA days
10		AU	<NA>	<NA>	NA days
11	Security Guard II	AU	<NA>	<NA>	NA days
12	Security Manager	AU	<NA>	<NA>	NA days
13	Service Assistant I	AU	<NA>	<NA>	NA days
14	Service Assistant II	AU	<NA>	<NA>	NA days
15	Cabinet Maker III	AU	<NA>	<NA>	NA days
16	Cabinet Maker II	AU	<NA>	<NA>	NA days
17	Electrician IV	AU	<NA>	<NA>	NA days
18	Electrician II	AU	<NA>	<NA>	NA days
19	Trainee	AU	<NA>	<NA>	NA days
20	Trainee	AU	<NA>	<NA>	NA days
21	Trainee	AU	<NA>	<NA>	NA days
22	Trainee	AU	<NA>	<NA>	NA days
23	Trainee	AU	<NA>	<NA>	NA days
24	Trainee	AU	<NA>	<NA>	NA days
25	Trainee	AU	<NA>	<NA>	NA days
26	Trainee	AU	<NA>	<NA>	NA days
27	Chief Executive Officer	US	<NA>	<NA>	NA days
28	Chief Marketing Officer	US	<NA>	<NA>	NA days
29	Chief Financial Officer	US	<NA>	<NA>	NA days
30	Financial Analyst III	US	<NA>	<NA>	NA days
31	Financial Analyst II	US	<NA>	<NA>	NA days
32	Auditor III	US	<NA>	<NA>	NA days
33	Secretary IV	US	<NA>	<NA>	NA days
34	Secretary III	US	<NA>	<NA>	NA days
35	Senior Strategist	US	<NA>	<NA>	NA days
36	Strategist II	US	<NA>	<NA>	NA days
37	Concession Director	US	<NA>	<NA>	NA days
38	Concession Manager	US	<NA>	<NA>	NA days
39	Concession Consultant II	US	<NA>	<NA>	NA days
40	Concession Assistant III	US	<NA>	<NA>	NA days
41	Concession Assistant I	US	<NA>	<NA>	NA days
42	Concession Consultant II	US	<NA>	<NA>	NA days
43	Concession Assistant II	US	<NA>	<NA>	NA days
44	Concession Consultant I	US	<NA>	<NA>	NA days
45	Concession Assistant III	US	<NA>	<NA>	NA days
46	Concession Consultant I	US	<NA>	<NA>	NA days
47	Concession Consultant III	US	<NA>	<NA>	NA days
48	Logistics Coordinator II	US	<NA>	<NA>	NA days
49	Logistics Coordinator I	US	<NA>	<NA>	NA days
50	Logistics Coordinator II	US	<NA>	<NA>	NA days
51	Director	US	<NA>	<NA>	NA days
52	Logistics Manager	US	<NA>	<NA>	NA days
53	Senior Logistics Manager	US	<NA>	<NA>	NA days
54	Secretary II	US	<NA>	<NA>	NA days
55	Pricing Manager	US	<NA>	<NA>	NA days
56	Pricing Specialist	US	<NA>	<NA>	NA days

57	Senior Logistics Manager	US	<NA>	<NA>	NA days
58	Logistics Manager	US	<NA>	<NA>	NA days
59	Office Assistant III	US	<NA>	<NA>	NA days
60	Services Manager	US	<NA>	<NA>	NA days
61	Services Assistant IV	US	<NA>	<NA>	NA days
62	Shipping Manager	US	<NA>	<NA>	NA days
63	Shipping Agent III	US	<NA>	<NA>	NA days
64	Shipping Manager	AU	<NA>	<NA>	NA days
65	Shipping Agent II	AU	<NA>	<NA>	NA days
66	Shipping Manager	US	<NA>	<NA>	NA days
67	Shipping Agent III	US	<NA>	<NA>	NA days
68	Shipping Manager	US	<NA>	<NA>	NA days
69	Shipping Agent I	US	<NA>	<NA>	NA days
70	Shipping Agent II	US	<NA>	<NA>	NA days
71	Shipping Agent I	US	<NA>	<NA>	NA days
72	Shipping Agent III	US	<NA>	<NA>	NA days
73	Warehouse Assistant I	US	<NA>	<NA>	NA days
74	Warehouse Assistant I	US	<NA>	<NA>	NA days
75	Warehouse Assistant II	US	<NA>	<NA>	NA days
76	Warehouse Assistant I	US	<NA>	<NA>	NA days
77	Warehouse Assistant I	US	<NA>	<NA>	NA days
78	Warehouse Assistant III	US	<NA>	<NA>	NA days
79	Warehouse Assistant I	US	<NA>	<NA>	NA days
80	Shipping Manager	AU	<NA>	<NA>	NA days
81	Shipping Agent II	AU	<NA>	<NA>	NA days
82	Shipping Agent I	AU	<NA>	<NA>	NA days
83	Warehouse Assistant I	AU	<NA>	<NA>	NA days
84	Warehouse Assistant II	AU	<NA>	<NA>	NA days
85	Warehouse Assistant I	AU	<NA>	<NA>	NA days
86	Warehouse Assistant IV	AU	<NA>	<NA>	NA days
87	Warehouse Assistant I	AU	<NA>	<NA>	NA days
88	Business Analyst II	US	<NA>	<NA>	NA days
89	Business Analyst III	US	<NA>	<NA>	NA days
90	Marketing Manager	US	<NA>	<NA>	NA days
91	Marketing Assistant III	US	<NA>	<NA>	NA days
92	Marketing Manager	US	<NA>	<NA>	NA days
93	Marketing Assistant II	US	<NA>	<NA>	NA days
94	Events Manager	US	<NA>	<NA>	NA days
95	Marketing Assistant II	US	<NA>	<NA>	NA days
96	Marketing Assistant II	US	<NA>	<NA>	NA days
97	Senior Marketing Manager	US	<NA>	<NA>	NA days
98	Corp. Comm. Manager	US	<NA>	<NA>	NA days
99	Marketing Assistant II	US	<NA>	<NA>	NA days
100	Corp. Comm. Specialist I	US	<NA>	<NA>	NA days
101	Corp. Comm. Specialist II	US	<NA>	<NA>	NA days
102	Marketing Manager	US	<NA>	<NA>	NA days
103	Marketing Assistant II	US	<NA>	<NA>	NA days
104	Marketing Assistant I	US	<NA>	<NA>	NA days
105	Marketing Assistant IV	US	<NA>	<NA>	NA days
106	Purchasing Agent II	US	<NA>	<NA>	NA days
107	Purchasing Agent I	US	<NA>	<NA>	NA days
108	Purchasing Agent I	US	<NA>	<NA>	NA days
109	Purchasing Agent II	US	<NA>	<NA>	NA days
110	Purchasing Agent III	US	<NA>	<NA>	NA days

111	Purchasing Agent I	US	<NA>	<NA>	NA days
112	Purchasing Agent III	US	<NA>	<NA>	NA days
113	Purchasing Manager	US	<NA>	<NA>	NA days
114	Purchasing Manager	US	<NA>	<NA>	NA days
115	Purchasing Manager	US	<NA>	<NA>	NA days
116	Purchasing Agent I	US	<NA>	<NA>	NA days
117	Purchasing Agent III	US	<NA>	<NA>	NA days
118	Purchasing Agent II	US	<NA>	<NA>	NA days
119	Purchasing Agent III	US	<NA>	<NA>	NA days
120	Purchasing Agent I	US	<NA>	<NA>	NA days
121	Purchasing Agent II	US	<NA>	<NA>	NA days
122	Purchasing Agent II	US	<NA>	<NA>	NA days
123	Purchasing Agent I	US	<NA>	<NA>	NA days
124	Account Manager	US	<NA>	<NA>	NA days
125	Financial Controller I	US	<NA>	<NA>	NA days
126	Building Admin. Manager	US	<NA>	<NA>	NA days
127	Office Assistant II	US	<NA>	<NA>	NA days
128	Accountant I	US	<NA>	<NA>	NA days
129	Finance Manager	US	<NA>	<NA>	NA days
130	Accountant I	US	<NA>	<NA>	NA days
131	Financial Controller II	US	<NA>	<NA>	NA days
132	Accountant II	US	<NA>	<NA>	NA days
133	Accountant III	US	<NA>	<NA>	NA days
134	Financial Controller III	US	<NA>	<NA>	NA days
135	Accountant III	US	<NA>	<NA>	NA days
136	Accountant II	US	<NA>	<NA>	NA days
137	Accountant II	US	<NA>	<NA>	NA days
138	Financial Controller III	US	<NA>	<NA>	NA days
139	Accountant I	US	<NA>	<NA>	NA days
140	Accountant I	US	<NA>	<NA>	NA days
141	Auditor II	US	<NA>	<NA>	NA days
142	Auditor I	US	<NA>	<NA>	NA days
143	Financial Controller III	US	<NA>	<NA>	NA days
144	Auditing Manager	US	<NA>	<NA>	NA days
145	Accountant I	US	<NA>	<NA>	NA days
146	Accountant II	US	<NA>	<NA>	NA days
147	Auditor II	US	<NA>	<NA>	NA days
148	Auditor I	US	<NA>	<NA>	NA days
149	Accountant II	US	<NA>	<NA>	NA days
150	HR Generalist I	US	<NA>	<NA>	NA days
151	HR Generalist II	US	<NA>	<NA>	NA days
152	HR Specialist II	US	<NA>	<NA>	NA days
153	HR Analyst II	US	<NA>	<NA>	NA days
154	HR Generalist III	US	<NA>	<NA>	NA days
155	HR Specialist I	US	<NA>	<NA>	NA days
156	HR Specialist I	US	<NA>	<NA>	NA days
157	HR Analyst II	US	<NA>	<NA>	NA days
158	HR Manager	US	<NA>	<NA>	NA days
159	Recruiter I	US	<NA>	<NA>	NA days
160	Recruitment Manager	US	<NA>	<NA>	NA days
161	Recruiter III	US	<NA>	<NA>	NA days
162	Recruiter II	US	<NA>	<NA>	NA days
163	Training Manager	US	<NA>	<NA>	NA days
164	Trainer I	US	<NA>	<NA>	NA days

165	Trainer II	US	<NA>	<NA>	NA days
166	Trainer I	US	<NA>	<NA>	NA days
167	Trainer III	US	<NA>	<NA>	NA days
168	ETL Specialist II	US	<NA>	<NA>	NA days
169	Systems Architect IV	US	<NA>	<NA>	NA days
170	Systems Architect II	US	<NA>	<NA>	NA days
171	ETL Specialist I	US	<NA>	<NA>	NA days
172	Applications Developer IV	US	<NA>	<NA>	NA days
173	Applications Developer II	US	<NA>	<NA>	NA days
174	Applications Developer II	US	<NA>	<NA>	NA days
175	Applications Developer I	US	<NA>	<NA>	NA days
176	Senior Project Manager	US	<NA>	<NA>	NA days
177	Office Assistant III	US	<NA>	<NA>	NA days
178	IS Director	US	<NA>	<NA>	NA days
179	Applications Developer I	US	<NA>	<NA>	NA days
180	Applications Developer IV	US	<NA>	<NA>	NA days
181	Applications Developer I	US	<NA>	<NA>	NA days
182	IS Administrator III	US	<NA>	<NA>	NA days
183	BI Administrator IV	US	<NA>	<NA>	NA days
184	IS Administrator II	US	<NA>	<NA>	NA days
185	IS Administrator I	US	<NA>	<NA>	NA days
186	BI Specialist II	US	<NA>	<NA>	NA days
187	BI Architect II	US	<NA>	<NA>	NA days
188	IS Architect III	US	<NA>	<NA>	NA days
189	Applications Developer I	US	<NA>	<NA>	NA days
190	Applications Developer II	US	<NA>	<NA>	NA days
191	Applications Developer IV	US	<NA>	<NA>	NA days
192	Project Manager	US	<NA>	<NA>	NA days
193	Service Administrator III	US	<NA>	<NA>	NA days
194	Service Administrator I	US	<NA>	<NA>	NA days
195	Office Assistant I	US	<NA>	<NA>	NA days
196	Office Assistant I	US	<NA>	<NA>	NA days
197	Office Administrator I	US	<NA>	<NA>	NA days
198	Office Administrator II	US	<NA>	<NA>	NA days
199	Office Assistant IV	US	<NA>	<NA>	NA days
200	Shipping Administrator I	US	<NA>	<NA>	NA days
201	Clerk I	US	<NA>	<NA>	NA days
202	Clerk I	US	<NA>	<NA>	NA days
203	Administration Manager	US	<NA>	<NA>	NA days
204	Warehouse Manager	US	<NA>	<NA>	NA days
205	Warehouse Assistant II	US	<NA>	<NA>	NA days
206	Warehouse Assistant I	US	<NA>	<NA>	NA days
207	Security Manager	US	<NA>	<NA>	NA days
208	Security Guard I	US	<NA>	<NA>	NA days
209	Security Guard I	US	<NA>	<NA>	NA days
210	Security Guard II	US	<NA>	<NA>	NA days
211	Security Guard II	US	<NA>	<NA>	NA days
212	Service Administrator I	US	<NA>	<NA>	NA days
213	Service Assistant I	US	<NA>	<NA>	NA days
214	Service Assistant I	US	<NA>	<NA>	NA days
215	Service Assistant II	US	<NA>	<NA>	NA days
216	Electrician II	US	<NA>	<NA>	NA days
217	Electrician III	US	<NA>	<NA>	NA days
218	Technician I	US	<NA>	<NA>	NA days

```
219         Technical Manager      US      <NA>     <NA>  NA days
220             Technician II      US      <NA>     <NA>  NA days
221                   Trainee      US      <NA>     <NA>  NA days
222                   Trainee      US      <NA>     <NA>  NA days
223                   Trainee      US      <NA>     <NA>  NA days
224                   Trainee      US      <NA>     <NA>  NA days
225                   Trainee      US      <NA>     <NA>  NA days
226                   Trainee      US      <NA>     <NA>  NA days
227                   Trainee      US      <NA>     <NA>  NA days
228                   Trainee      US      <NA>     <NA>  NA days
229                   Trainee      US      <NA>     <NA>  NA days
230                   Trainee      US      <NA>     <NA>  NA days
231            Vice President      US      <NA>     <NA>  NA days
232                  Director      US      <NA>     <NA>  NA days
233             Secretary III      US      <NA>     <NA>  NA days
234              Secretary II      US      <NA>     <NA>  NA days
235       Business Analyst II      US      <NA>     <NA>  NA days
```

```
ABNewDF
```

```
                 a   b
1  London, England 8.0
2  Oxford, England 0.1
3  Berlin, Germany 3.0
4 Hamburg, Germany 2.0
5         Oslo, NO 1.0
```

Sorting

To sort a data frame in R, use the `order()` function. By default, sorting is *ascending*. Prepend the sorting variable by a minus sign to indicate *descending* order.

```
data(mtcars); attach(mtcars)
mtcars
```

```
                   mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Duster 360        14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Merc 240D         24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230          22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 280          19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Merc 280C         17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Merc 450SE        16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
Merc 450SL        17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
Merc 450SLC       15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
Fiat 128          32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
```

```
Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

Sort by mpg

```
newdata <- mtcars[order(mpg),]
newdata
```

```
                     mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
Duster 360          14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
Merc 450SLC         15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
Merc 450SE          16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
Merc 450SL          17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
Merc 280C           17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Valiant             18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Hornet Sportabout   18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Merc 280            19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Mazda RX4           21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag       21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Hornet 4 Drive      21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Datsun 710          22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Merc 230            22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 240D           24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Fiat 128            32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
```

```
Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
```

Sort by mpg and cyl

```
newdata1 <- mtcars[order(mpg,cyl),]
newdata1
```

```
                     mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
Duster 360          14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
Merc 450SLC         15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
Merc 450SE          16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
Merc 450SL          17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
Merc 280C           17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Valiant             18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Hornet Sportabout   18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Merc 280            19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Mazda RX4           21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag       21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
Hornet 4 Drive      21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Datsun 710          22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Merc 230            22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 240D           24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Fiat 128            32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
```

Sort by mpg (ascending) and cyl (descending)

```
newdata2 <- mtcars[order(mpg,-cyl),]
newdata2
```

```
                     mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
Duster 360          14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
```

```
Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
Merc 450SLC         15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
Merc 450SE          16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
Merc 450SL          17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
Merc 280C           17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Valiant             18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Hornet Sportabout   18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
Merc 280            19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Mazda RX4           21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag       21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Hornet 4 Drive      21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Datsun 710          22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Merc 230            22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 240D           24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Fiat 128            32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
```

```r
detach(mtcars)
```

Combining data frames

Appending adds the observations in the second dataset directly to the end of the original dataset

Concatenating copies all observations from the first dataset and then copies all observations from one or more successive datasets into a new dataset

Merging involves combining observations from two data frames by one or more common variables. Observations can be merged based on their positions in the original datasets or merged by one or more common variables

Appending with `c()`

```r
# Create a, b, c, d variables
a <- c(10,20,30,40)
b <- c('book','pen','textbook','pencil_case')
c <- c(TRUE,FALSE,TRUE,FALSE)
d <- c(2.5,8,10,7)
# Join the variables to create a data frame
df <- data.frame(a,b,c,d); df
```

```
   a           b     c    d
1 10        book  TRUE  2.5
2 20         pen FALSE  8.0
```

```
3 30    textbook  TRUE 10.0
4 40 pencil_case FALSE  7.0
```

Appending with c()

```
names(df) <- c('ID','items','store','price'); df
```

```
  ID       items store price
1 10        book  TRUE   2.5
2 20         pen FALSE   8.0
3 30    textbook  TRUE  10.0
4 40 pencil_case FALSE   7.0
```

```
str(df)  # Print the structure
```

```
'data.frame':   4 obs. of  4 variables:
 $ ID   : num  10 20 30 40
 $ items: chr  "book" "pen" "textbook" "pencil_case"
 $ store: logi  TRUE FALSE TRUE FALSE
 $ price: num  2.5 8 10 7
```

Appending with c()

Create a new vector to add to the data frame. Use the two-level name so that the data frame has that variable in it; the two-level name: `datasetname$variablename`. The following code cannot be run because it will produce the following error: `Error in `$<-.data.frame`(`*tmp*`, quantity, value = c(10, 35, 40)) : replacement has 3 rows, data has 4`.

```
quantity <- c(10,35,40)
# Add variable quantity to the df data frame (not)
df$quantity <- quantity; df
```

Appending with c()

```
quantity <- c(10,35,40,5)
df$quantity <- quantity; df
```

```
  ID       items store price quantity
1 10        book  TRUE   2.5       10
2 20         pen FALSE   8.0       35
3 30    textbook  TRUE  10.0       40
4 40 pencil_case FALSE   7.0        5
```

Concatenation

Often `rbind()` or `cbind()` is used, as long as the data structures are alike with no duplicates. However, if there are duplicate values, then the duplicates will be displayed

Concatenation with `rbind()`

```
df1 <- data.frame(name=c("tim","tim","tim","ron"),val=1:4)
df1
```

```
  name val
1  tim   1
2  tim   2
3  tim   3
4  ron   4
```

Concatenation with `rbind()`

```
df2 <- data.frame(name=c("tim","tim","ron","ron"),val=1:4)
df2
```

```
  name val
1  tim   1
2  tim   2
3  ron   3
4  ron   4
```

Concatenation with `rbind()`

```
rbind(df1,df2)
```

```
  name val
1  tim   1
2  tim   2
3  tim   3
4  ron   4
5  tim   1
6  tim   2
7  ron   3
8  ron   4
```

`merge()`

Merging two data frames by one or more common variables

General form:

`merge(x,y,by=by,by.x=by,by.y=by,sort=T,...)`

`x`, `y`: data frames to be merged
`by`, `by.x`, `by.y`: variables to merge (and/or sort if option is used) by sort logical. Should the result be sorted by the by variable (TRUE or FALSE (T or F))
`...`: other options

Note two things: one is that you do not have to have presorted data and that the sorting can happen during the merge rather than before. Second, notice that this is for combining just 2 datasets. If you want to combine more than two, then you would have to stack the commands until you were finished merging.

merge() logistics

`merge()` works when the vectors are of equal length. You will get an error if they are not. The best way to deal with unequal vector lengths is to input "missing" values in the data frame before merging. Not always easy but it is a way to get it done.

merge()

```
merge(df1,df2,all.x=T,all.y=T)
```

```
  name val
1  ron   3
2  ron   4
3  tim   1
4  tim   2
5  tim   3
```