R Nuts and Bolts

R objects and attributes

Obects

- **Five basic classes of objects:**
- **Character, numberic, integer, complex, logical**
- **The most basic object is a vector**
- **List is an exception, it can contain objects of different classes**
- **Empty vector: vector()**

Numbers

- **Numbers are treated as numberic objects**
- **NaN represents an undefine value**

Attributes

- **names, dimnames; dimensions; class; length;**
- **attributes()**

Vectors and Lists

```r
n<-c(0.5,0.6) #numeric
l_1<-c(TRUE,FALSE) #logical
l_2<-c(T,F)
ch<-c("a","b","c") #character
i<-9:29 # integer
complex<-c(1+0i,2+4i)
#Using vector() function
vec<-vector("numeric",length = 10)
#Mixing objects
m1=c(1.7,"a") # character
m2=c(TRUE,2) # numeric
m3=c("a",TRUE) # character
```

When different objects are mixed in a vector, coercion occurs that every element in the vector is of the same class.

```
#Exlpicit coercion
x = 0:6
class(x) # integer class
```

```
## [1] "integer"
```

```
#Convert it into different class
as.numeric(x)
```

```
## [1] 0 1 2 3 4 5 6
```

```
as.character(x)
```

```
## [1] "0" "1" "2" "3" "4" "5" "6"
```

```
as.logical(x)
```

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

```
#Nonsensical coersion results in NAS
y=c("a","b","c")
as.logical(y)
```

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

```
as.complex(x)
```

```
## [1] 0+0i 1+0i 2+0i 3+0i 4+0i 5+0i 6+0i
```

```
#Lists
list1=list(1,"a",TRUE,1+4i)
list1
```

```
## [[1]]
## [1] 1
##
## [[2]]
## [1] "a"
##
## [[3]]
## [1] TRUE
##
## [[4]]
## [1] 1+4i
```

Matrices

Matrices are vectors with a dimension attribute

```
m1=matrix(nrow = 2,ncol = 3)
m1
```

```
##      [,1] [,2] [,3]
## [1,]   NA   NA   NA
## [2,]   NA   NA   NA
```

```
dim(m1)
```

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

```
attributes(m1)
```

```
## $dim
## [1] 2 3
```

Matrices are contructed column-wise

```
m2=matrix(1:9,nrow = 3,ncol = 3)
m2
```

```
##      [,1] [,2] [,3]
## [1,]    1    4    7
## [2,]    2    5    8
## [3,]    3    6    9
```

```
# Created matrices from vectors by adding a dimension attribute
m3=1:10
dim(m3)=c(2,5)
m3
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    3    5    7    9
## [2,]    2    4    6    8   10
```

```
#cbind() and rbind()
x=1:3
y=10:12
cbind(x,y) #3rows and 2 columns
```

```
##      x  y
## [1,] 1 10
## [2,] 2 11
## [3,] 3 12
```

```
rbind(x,y) #2rows and 3 columns
```

```
##   [,1] [,2] [,3]
## x    1    2    3
## y   10   11   12
```

Factors

- Factors represent categorical data. - Factors can be ordered and unordered.

```
f=factor(c("yes","yes","no","yes","no"))
f
```

```
## [1] yes yes no  yes no 
## Levels: no yes
```

```
# table() create a frequency table
table(f)
```

```
## f
##  no yes 
##   2   3
```

```
# unclass() break down into an integer vector
unclass(f)
```

```
## [1] 2 2 1 2 1
## attr(,"levels")
## [1] "no"  "yes"
```

- The order of the leves can be set using the levels argument to factor(). - This is important because in linear modeling because the first level is used as the baseline level.

```
answer=factor(c("yes","yes","no","yes","no"),
              levels = c("yes","no"))
answer
```

```
## [1] yes yes no  yes no 
## Levels: yes no
```

Missing values

is.na(), is.nan()

```
x=c(1,2,NA,10,3)
is.na(x)
```

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

```
y=c(1,2,NaN,NA,4)
is.na(y)
```

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

```r
is.nan(y)
```

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

Data frames

- DF have a special attribute called row.names()

- read.table(), read.csv() create df

- convert to a matrix data.matrix()

```r
df=data.frame(itemnum=1:4,
              fruit=c("Strawberry","Bluberry","Apple","Banana"),
              vege=c("Carrot","Avacado","Asparagus","Broccoli"))
df
```

```
##   itemnum      fruit      vege
## 1       1 Strawberry    Carrot
## 2       2   Bluberry   Avacado
## 3       3      Apple Asparagus
## 4       4     Banana  Broccoli
```

```r
nrow(df)
```

```
## [1] 4
```

```r
ncol(df)
```

```
## [1] 3
```

```r
row.names(df)
```

```
## [1] "1" "2" "3" "4"
```

Names

```r
x=1:3
names(x)
```

```
## NULL
```

```r
# Give the name to each vector element
names(x)=c("foo","bar","norf")
x
```

```
##  foo  bar norf
##    1    2    3
```

```
names(x)
```

```
## [1] "foo"  "bar"  "norf"
```

```
# list can also have names
y=list(a=1,b=2,c=3)
y
```

```
## $a
## [1] 1
##
## $b
## [1] 2
##
## $c
## [1] 3
```

```
# Matrices can have names
m=matrix(1:4,nrow = 2,ncol=2)
# row name and col name
dimnames(m)=list(c("a","b"),c("c","d"))
m
```

```
##   c d
## a 1 3
## b 2 4
```

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

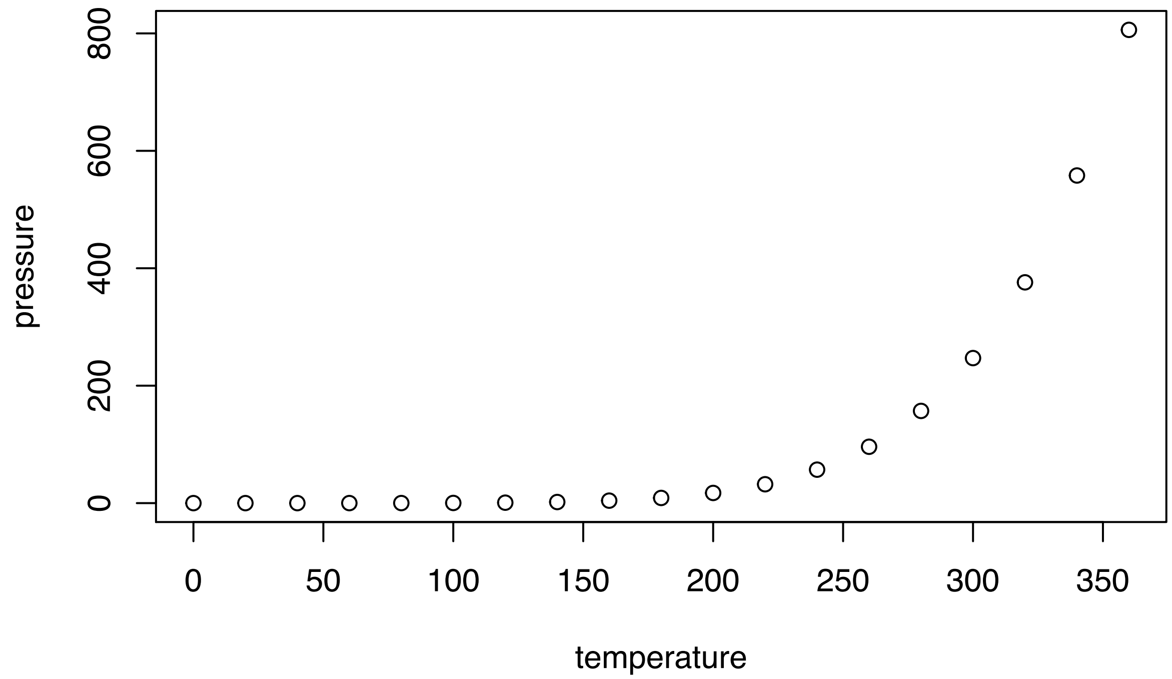
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.