```
> # HISTOGRAM ######################################################
>
> # Default
> hist(lynx)
>
> # Add some options
> hist(lynx,
+      breaks = 14,          # "Suggests" 14 bins
+      freq   = FALSE,       # Axis shows density, not freq.
+      col    = "thistle1",  # Color for histogram
+      main   = paste("Histogram of Annual Canadian Lynx",
+                     "Trappings, 1821-1934"),
+      xlab   = "Number of Lynx Trapped")
>
> # Add a normal distribution
> curve(dnorm(x, mean = mean(lynx), sd = sd(lynx)),
+       col = "thistle4",  # Color of curve
+       lwd = 2,           # Line width of 2 pixels
+       add = TRUE)        # Superimpose on previous graph
>
> # Add two kernel density estimators
> lines(density(lynx), col = "blue", lwd = 2)
> lines(density(lynx, adjust = 3), col = "purple", lwd = 2)
>
> # Add a rug plot
> rug(lynx, lwd = 2, col = "gray")
> # File:   Summary.R
> # Course: R: An Introduction (with RStudio)
>
> # INSTALL AND LOAD PACKAGES ######################################
>
> library(datasets)  # Load/unload base packages manually
>
> # LOAD DATA ######################################################
>
> head(iris)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          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          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
>
> # SUMMARY() ######################################################
>
> summary(iris$Species)       # Categorical variable
    setosa versicolor  virginica
        50         50         50
> summary(iris$Sepal.Length)  # Quantitative variable
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  4.300   5.100   5.800   5.843   6.400   7.900
> summary(iris)               # Entire data frame
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Specie
s
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :5
0
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:5
0
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :5
0
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
```

```
> # File:   Describe.R
> # Course: R: An Introduction (with RStudio)
>
> # INSTALL AND LOAD PACKAGES ##################################
>
> library(datasets)  # Load base packages manually
>
> # Installs pacman ("package manager") if needed
> if (!require("pacman")) install.packages("pacman")
Loading required package: pacman
>
> # Use pacman to load add-on packages as desired
> pacman::p_load(pacman, psych)
Installing package into 'C:/Users/Admin/AppData/Local/R/win-library/4.4'
(as 'lib' is unspecified)
also installing the dependencies 'mnormt', 'GPArotation'

Warning: unable to access index for repository http://www.stats.ox.ac.uk/pub/
RWin/bin/windows/contrib/4.4:
  cannot open URL 'http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/4.4
/PACKAGES'
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/mnormt_2.1.1.zip
'
Content type 'application/zip' length 182714 bytes (178 KB)
downloaded 178 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/GPArotation_2024
.3-1.zip'
Content type 'application/zip' length 395162 bytes (385 KB)
downloaded 385 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/psych_2.4.12.zip
'
Content type 'application/zip' length 3739383 bytes (3.6 MB)
downloaded 3.6 MB

package 'mnormt' successfully unpacked and MD5 sums checked
package 'GPArotation' successfully unpacked and MD5 sums checked
package 'psych' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\Admin\AppData\Local\Temp\Rtmps79HOh\downloaded_packages

psych installed
>
> # LOAD DATA ##################################################
>
> head(iris)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          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          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
>
> # PSYCH PACKAGE ##############################################
>
> # Get info on package
> p_help(psych)           # Opens package PDF in browser
Error: failed to load external entity "http://stat.ethz.ch/R-manual/R-patched
/library/"
> # File:   SelectingCases.R
> # Course: R: An Introduction (with RStudio)
```

```
>
> # INSTALL AND LOAD PACKAGES ##################################
>
> library(datasets)  # Load/unload base packages manually
>
> # LOAD DATA ##################################################
>
> head(iris)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          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          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
>
> # ALL DATA ###################################################
>
> hist(iris$Petal.Length)
> summary(iris$Petal.Length)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.600   4.350   3.758   5.100   6.900
>
> summary(iris$Species)  # Get names and n for each species
    setosa versicolor  virginica
        50         50         50
>
> # SELECT BY CATEGORY #########################################
>
> # Versicolor
> hist(iris$Petal.Length[iris$Species == "versicolor"],
+      main = "Petal Length: Versicolor")
>
> # Virginica
> hist(iris$Petal.Length[iris$Species == "virginica"],
+      main = "Petal Length: Virginica")
>
> # Setosa
> hist(iris$Petal.Length[iris$Species == "setosa"],
+      main = "Petal Length: Setosa")
>
> # SELECT BY VALUE ############################################
>
> # Short petals only (all Setosa)
> hist(iris$Petal.Length[iris$Petal.Length < 2],
+      main = "Petal Length < 2")
>
> # MULTIPLE SELECTORS #########################################
>
> # Short Virginica petals only
> hist(iris$Petal.Length[iris$Species == "virginica" &
+                        iris$Petal.Length < 5.5],
+      main = "Petal Length: Short Virginica")
>
> # CREATE SUBSAMPLE ###########################################
>
> # Format: data[rows, columns]
> # Leave rows or columns blank to select all
> i.setosa <- iris[iris$Species == "setosa", ]
>
> # EXPLORE SUBSAMPLE ##########################################
>
> head(i.setosa)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
```

```
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          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          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
> summary(i.setosa$Petal.Length)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.400   1.500   1.462   1.575   1.900
> hist(i.setosa$Petal.Length)
> # File:   DataFormats.R
> # Course: R: An Introduction (with RStudio)
>
> # DATA TYPES ###################################################
>
> # Numeric
>
> n1 <- 15  # Double precision by default
> n1
[1] 15
> typeof(n1)
[1] "double"
>
> n2 <- 1.5
> n2
[1] 1.5
> typeof(n2)
[1] "double"
>
> # Character
>
> c1 <- "c"
> c1
[1] "c"
> typeof(c1)
[1] "character"
>
> c2 <- "a string of text"
> c2
[1] "a string of text"
> typeof(c2)
[1] "character"
>
> # Logical
>
> l1 <- TRUE
> l1
[1] TRUE
> typeof(l1)
[1] "logical"
>
> l2 <- F
> l2
[1] FALSE
> typeof(l2)
[1] "logical"
>
> # DATA STRUCTURES ##############################################
>
> ## Vector ######################################################
>
> v1 <- c(1, 2, 3, 4, 5)
> v1
[1] 1 2 3 4 5
```

```
> is.vector(v1)
[1] TRUE
>
> v2 <- c("a", "b", "c")
> v2
[1] "a" "b" "c"
> is.vector(v2)
[1] TRUE
>
> v3 <- c(TRUE, TRUE, FALSE, FALSE, TRUE)
> v3
[1]  TRUE  TRUE FALSE FALSE  TRUE
> is.vector(v3)
[1] TRUE
>
> ## Matrix ##############################################
>
> m1 <- matrix(c(T, T, F, F, T, F), nrow = 2)
> m1
     [,1]  [,2]  [,3]
[1,] TRUE FALSE  TRUE
[2,] TRUE FALSE FALSE
>
> m2 <- matrix(c("a", "b",
+                "c", "d"),
+              nrow = 2,
+              byrow = T)
> m2
     [,1] [,2]
[1,] "a"  "b"
[2,] "c"  "d"
>
> ## Array ###############################################
>
> # Give data, then dimemensions (rows, columns, tables)
> a1 <- array(c( 1:24), c(4, 3, 2))
> a1
, , 1

     [,1] [,2] [,3]
[1,]    1    5    9
[2,]    2    6   10
[3,]    3    7   11
[4,]    4    8   12

, , 2

     [,1] [,2] [,3]
[1,]   13   17   21
[2,]   14   18   22
[3,]   15   19   23
[4,]   16   20   24

>
> ## Data frame ##########################################
>
> # Can combine vectors of the same length
>
> vNumeric   <- c(1, 2, 3)
> vCharacter <- c("a", "b", "c")
> vLogical   <- c(T, F, T)
>
> dfa <- cbind(vNumeric, vCharacter, vLogical)
> dfa  # Matrix of one data type
```

```
     vNumeric vCharacter vLogical
[1,] "1"      "a"        "TRUE"
[2,] "2"      "b"        "FALSE"
[3,] "3"      "c"        "TRUE"
>
> df <- as.data.frame(cbind(vNumeric, vCharacter, vLogical))
> df  # Makes a data frame with three different data types
  vNumeric vCharacter vLogical
1        1          a     TRUE
2        2          b    FALSE
3        3          c     TRUE
>
> ## List ###################################################
>
> o1 <- c(1, 2, 3)
> o2 <- c("a", "b", "c", "d")
> o3 <- c(T, F, T, T, F)
>
> list1 <- list(o1, o2, o3)
> list1
[[1]]
[1] 1 2 3

[[2]]
[1] "a" "b" "c" "d"

[[3]]
[1]  TRUE FALSE  TRUE  TRUE FALSE

>
> list2 <- list(o1, o2, o3, list1)  # Lists within lists!
> list2
[[1]]
[1] 1 2 3

[[2]]
[1] "a" "b" "c" "d"

[[3]]
[1]  TRUE FALSE  TRUE  TRUE FALSE

[[4]]
[[4]][[1]]
[1] 1 2 3

[[4]][[2]]
[1] "a" "b" "c" "d"

[[4]][[3]]
[1]  TRUE FALSE  TRUE  TRUE FALSE


>
> # COERCING TYPES ##########################################
>
> ## Automatic coercion #####################################
>
> # Goes to "least restrictive" data type
>
> (coerce1 <- c(1, "b", TRUE))
[1] "1"    "b"    "TRUE"
> # coerce1  # Parenthese around command above make this moot
> typeof(coerce1)
[1] "character"
```

```
>
> ## Coerce numeric to integer ################################
>
> (coerce2 <- 5)
[1] 5
> typeof(coerce2)
[1] "double"
>
> (coerce3 <- as.integer(5))
[1] 5
> typeof(coerce3)
[1] "integer"
>
> ## Coerce character to numeric ##############################
>
> (coerce4 <- c("1", "2", "3"))
[1] "1" "2" "3"
> typeof(coerce4)
[1] "character"
>
> (coerce5 <- as.numeric(c("1", "2", "3")))
[1] 1 2 3
> typeof(coerce5)
[1] "double"
>
> ## Coerce matrix to data frame ##############################
>
> (coerce6 <- matrix(1:9, nrow= 3))
     [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
> is.matrix(coerce6)
[1] TRUE
>
> (coerce7 <- as.data.frame(matrix(1:9, nrow= 3)))
  V1 V2 V3
1  1  4  7
2  2  5  8
3  3  6  9
> is.data.frame(coerce7)
[1] TRUE
> # File:   Factors.R
> # Course: R: An Introduction (with RStudio)
>
> # CREATE DATA ##############################################
>
> (x1 <- 1:3)
[1] 1 2 3
> (y  <- 1:9)
[1] 1 2 3 4 5 6 7 8 9
>
> # Combine variables
> (df1 <- cbind.data.frame(x1, y))
  x1 y
1  1 1
2  2 2
3  3 3
4  1 4
5  2 5
6  3 6
7  1 7
8  2 8
9  3 9
```

```
> typeof(df1$x1)
[1] "integer"
> str(df1)
'data.frame':  9 obs. of  2 variables:
 $ x1: int  1 2 3 1 2 3 1 2 3
 $ y : int  1 2 3 4 5 6 7 8 9
>
> # AS.FACTOR ###################################################
>
> (x2  <- as.factor(c(1:3)))
[1] 1 2 3
Levels: 1 2 3
> (df2 <- cbind.data.frame(x2, y))
  x2 y
1  1 1
2  2 2
3  3 3
4  1 4
5  2 5
6  3 6
7  1 7
8  2 8
9  3 9
> typeof(df2$x2)
[1] "integer"
> str(df2)
'data.frame':  9 obs. of  2 variables:
 $ x2: Factor w/ 3 levels "1","2","3": 1 2 3 1 2 3 1 2 3
 $ y : int  1 2 3 4 5 6 7 8 9
>
> # DEFINE EXISTING VARIABLE AS FACTOR ######################
>
> x3  <- c(1:3)
> df3 <- cbind.data.frame(x3, y)
> (df3$x3 <- factor(df3$x3,
+                   levels = c(1, 2, 3)))
[1] 1 2 3 1 2 3 1 2 3
Levels: 1 2 3
> typeof(df3$x3)
[1] "integer"
> str(df3)
'data.frame':  9 obs. of  2 variables:
 $ x3: Factor w/ 3 levels "1","2","3": 1 2 3 1 2 3 1 2 3
 $ y : int  1 2 3 4 5 6 7 8 9
>
> # LABELS FOR FACTOR ########################################
>
> x4  <- c(1:3)
> df4 <- cbind.data.frame(x4, y)
> df4$x4 <- factor(df4$x4,
+                  levels = c(1, 2, 3),
+                  labels = c("macOS", "Windows", "Linux"))
> df4
       x4 y
1   macOS 1
2 Windows 2
3   Linux 3
4   macOS 4
5 Windows 5
6   Linux 6
7   macOS 7
8 Windows 8
9   Linux 9
> typeof(df4$x4)
```

```
[1] "integer"
> str(df4)
'data.frame':  9 obs. of  2 variables:
 $ x4: Factor w/ 3 levels "macOS","Windows",..: 1 2 3 1 2 3 1 2 3
 $ y : int  1 2 3 4 5 6 7 8 9
>
> # ORDERED FACTORS AND LABELS ##################################
>
> x5  <- c(1:3)
> df5 <- cbind.data.frame(x5, y)
> (df5$x5 <- ordered(df5$x5,
+                    levels = c(3, 1, 2),
+                    labels = c("No", "Maybe", "Yes")))
[1] Maybe Yes   No    Maybe Yes   No    Maybe Yes   No
Levels: No < Maybe < Yes
> df5
     x5 y
1 Maybe 1
2   Yes 2
3    No 3
4 Maybe 4
5   Yes 5
6    No 6
7 Maybe 7
8   Yes 8
9    No 9
> typeof(df5$x5)
[1] "integer"
> str(df5)
'data.frame':  9 obs. of  2 variables:
 $ x5: Ord.factor w/ 3 levels "No"<"Maybe"<"Yes": 2 3 1 2 3 1 2 3 1
 $ y : int  1 2 3 4 5 6 7 8 9
> # File:   EnteringData.R
> # Course: R: An Introduction (with RStudio)
>
> # COLON OPERATOR ##############################################
>
> # Assigns number 0 through 10 to x1
> x1 <- 0:10
> x1
 [1]  0  1  2  3  4  5  6  7  8  9 10
>
> # Descending order
> x2 <- 10:0
> x2
 [1] 10  9  8  7  6  5  4  3  2  1  0
>
> # SEQ #########################################################
>
> ?seq  # R help on seq
>
> # Ascending values (duplicates 1:10)
> (x3 <- seq(10))
 [1]  1  2  3  4  5  6  7  8  9 10
>
> # Specify change in values
> (x4 <- seq(30, 0, by = -3))
 [1] 30 27 24 21 18 15 12  9  6  3  0
>
> # ENTER MULTIPLE VALUES WITH C ################################
>
> # c = concatenate (or combine or collect)
> ?c  # R help on c
>
```

```
> x5 <- c(5, 4, 1, 6, 7, 2, 2, 3, 2, 8)
> x5
 [1] 5 4 1 6 7 2 2 3 2 8
>
> # SCAN ##################################################
>
> ?scan  # R help on scan
>
> x6 <- scan()  # After running this command, go to console
1: # Hit return after each number
Error in scan() : scan() expected 'a real', got '#'
> # File:   HierarchicalClustering.R
> # Course: R: An Introduction (with RStudio)
>
> # INSTALL AND LOAD PACKAGES #################################
>
> library(datasets)  # Load base packages manually
>
> # Installs pacman ("package manager") if needed
> if (!require("pacman")) install.packages("pacman")
>
> # Use pacman to load add-on packages as desired
> pacman::p_load(pacman, tidyverse)
>
> # LOAD DATA #################################################
>
> ?mtcars
> head(mtcars)
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
> cars <- mtcars[, c(1:4, 6:7, 9:11)]  # Select variables
> head(cars)
                   mpg cyl disp  hp    wt  qsec am gear carb
Mazda RX4         21.0   6  160 110 2.620 16.46  1    4    4
Mazda RX4 Wag     21.0   6  160 110 2.875 17.02  1    4    4
Datsun 710        22.8   4  108  93 2.320 18.61  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.215 19.44  0    3    1
Hornet Sportabout 18.7   8  360 175 3.440 17.02  0    3    2
Valiant           18.1   6  225 105 3.460 20.22  0    3    1
>
> # COMPUTE AND PLOT CLUSTERS ################################
>
> # Save hierarchical clustering to "hc." This codes uses
> # pipes from dplyr.
> hc <- cars   %>%  # Get cars data
+   dist   %>%  # Compute distance/dissimilarity matrix
+   hclust      # Computer hierarchical clusters
>
> plot(hc)          # Plot dendrogram
>
> # ADD BOXES TO PLOT #########################################
>
> rect.hclust(hc, k = 2, border = "gray")
> rect.hclust(hc, k = 3, border = "blue")
> rect.hclust(hc, k = 4, border = "green4")
> rect.hclust(hc, k = 5, border = "darkred")
> # File:   HierarchicalClustering.R
> # Course: R: An Introduction (with RStudio)
>
```

```
> # INSTALL AND LOAD PACKAGES ################################
>
> library(datasets)  # Load base packages manually
>
> # Installs pacman ("package manager") if needed
> if (!require("pacman")) install.packages("pacman")
>
> # Use pacman to load add-on packages as desired
> pacman::p_load(pacman, tidyverse)
>
> # LOAD DATA ################################################
>
> ?mtcars
> head(mtcars)
                   mpg cyl disp  hp drat    wt  qsec vs am
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0
                  gear carb
Mazda RX4            4    4
Mazda RX4 Wag        4    4
Datsun 710           4    1
Hornet 4 Drive       3    1
Hornet Sportabout    3    2
Valiant              3    1
> cars <- mtcars[, c(1:4, 6:7, 9:11)]  # Select variables
> head(cars)
                   mpg cyl disp  hp    wt  qsec am gear carb
Mazda RX4         21.0   6  160 110 2.620 16.46  1    4    4
Mazda RX4 Wag     21.0   6  160 110 2.875 17.02  1    4    4
Datsun 710        22.8   4  108  93 2.320 18.61  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.215 19.44  0    3    1
Hornet Sportabout 18.7   8  360 175 3.440 17.02  0    3    2
Valiant           18.1   6  225 105 3.460 20.22  0    3    1
>
> # COMPUTE AND PLOT CLUSTERS ###############################
>
> # Save hierarchical clustering to "hc." This codes uses
> # pipes from dplyr.
> hc <- cars   %>%  # Get cars data
+   dist   %>%  # Compute distance/dissimilarity matrix
+   hclust      # Computer hierarchical clusters
>
> plot(hc)          # Plot dendrogram
>
> # ADD BOXES TO PLOT ########################################
>
> rect.hclust(hc, k = 2, border = "gray")
> rect.hclust(hc, k = 3, border = "blue")
> rect.hclust(hc, k = 4, border = "green4")
> rect.hclust(hc, k = 5, border = "darkred")
> # File:    R01_6_2_PrincipalComponents.R
> # Course:  R01: R: An introduction
> # Chapter: 6: Modeling data
> # Section: 2: Principal components
> # Author:  Barton Poulson, datalab.cc, @bartonpoulson
> # Date:    2016-08-04
>
> # INSTALL AND LOAD PACKAGES ################################
>
> # Packages I load every time; uses "pacman"
```

```
> pacman::p_load(pacman, dplyr, GGally, ggplot2, ggthemes,
+                ggvis, httr, lubridate, plotly, rio, rmarkdown, shiny,
+                stringr, tidyr)
>
> library(datasets)  # Load base packages manually
>
> # LOAD DATA ############################################
>
> head(mtcars)
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
> cars <- mtcars[, c(1:4, 6:7, 9:11)]  # Select variables
> head(cars)
                   mpg cyl disp  hp    wt  qsec am gear carb
Mazda RX4         21.0   6  160 110 2.620 16.46  1    4    4
Mazda RX4 Wag     21.0   6  160 110 2.875 17.02  1    4    4
Datsun 710        22.8   4  108  93 2.320 18.61  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.215 19.44  0    3    1
Hornet Sportabout 18.7   8  360 175 3.440 17.02  0    3    2
Valiant           18.1   6  225 105 3.460 20.22  0    3    1
>
> # COMPUTE PCA #########################################
>
> # For entire data frame ################################
> pc <- prcomp(cars,
+              center = TRUE,  # Centers means to 0 (optional)
+              scale = TRUE)   # Sets unit variance (helpful)
>
> # To specify variables #################################
>
> pc <- prcomp(~ mpg + cyl + disp + hp + wt + qsec + am +
+                gear + carb,
+              data = mtcars,
+              center = TRUE,
+              scale = TRUE)
>
> # EXAMINE RESULTS ######################################
>
> # Get summary stats
> summary(pc)
Importance of components:
                          PC1    PC2     PC3     PC4     PC5     PC6     PC7
PC8
Standard deviation     2.3391 1.5299 0.71836 0.46491 0.38903 0.35099 0.31714
0.24070
Proportion of Variance 0.6079 0.2601 0.05734 0.02402 0.01682 0.01369 0.01118
0.00644
Cumulative Proportion  0.6079 0.8680 0.92537 0.94939 0.96620 0.97989 0.99107
0.99750
                          PC9
Standard deviation     0.1499
Proportion of Variance 0.0025
Cumulative Proportion  1.0000
>
> # Screeplot for number of components
> plot(pc)
>
> # Get standard deviations and rotation
> pc
```

```
Standard deviations (1, .., p=9):
[1] 2.3391410 1.5299383 0.7183646 0.4649052 0.3890348 0.3509911 0.3171373 0.2
406989
[9] 0.1498962

Rotation (n x k) = (9 x 9):
            PC1         PC2         PC3        PC4         PC5           PC6
PC7
mpg  -0.4023287  0.02205294 -0.17272803 -0.1366169 -0.31654561  0.718609897
0.3633216
cyl   0.4068870  0.03589482 -0.27747610  0.1410976 -0.02066646  0.214224005
0.2099893
disp  0.4046964 -0.06479590 -0.17669890 -0.5089434 -0.21525777 -0.010052074
0.2007152
hp    0.3699702  0.26518848 -0.01046827 -0.1273173 -0.42166543  0.254229405 -
0.6741641
wt    0.3850686 -0.15955242  0.33740464 -0.4469327  0.21141143 -0.002897706
0.3392809
qsec -0.2168575 -0.48343885  0.54815205 -0.2545226 -0.05466817  0.226660704 -
0.2986852
am   -0.2594512  0.46039449 -0.19492256 -0.5354196  0.55331460  0.087616182 -
0.2135605
gear -0.2195660  0.50608232  0.34579810 -0.1799814 -0.50533262 -0.393990378
0.2484622
carb  0.2471604  0.44322600  0.53847588  0.3203064  0.25696817  0.398353829
0.1321064
            PC8         PC9
mpg  -0.1487806  0.13567069
cyl   0.7951724  0.11635839
disp -0.1346748 -0.66099594
hp   -0.1210386  0.25474680
wt   -0.1598333  0.57211273
qsec  0.4144075 -0.19671599
am    0.1897463 -0.02465169
gear  0.2614819  0.05482771
carb -0.1054553 -0.31083546
>
> # See how cases load on PCs
> predict(pc) %>% round(2)
                     PC1   PC2   PC3   PC4   PC5   PC6   PC7   PC8   PC9
Mazda RX4          -0.82  1.46 -0.21  0.32  0.85  0.01  0.25  0.07 -0.18
Mazda RX4 Wag      -0.79  1.26  0.05  0.12  0.89  0.08  0.25  0.16 -0.09
Datsun 710         -2.49  0.03 -0.32 -0.40  0.37 -0.54 -0.50 -0.03  0.11
Hornet 4 Drive     -0.29 -1.93 -0.32 -0.07 -0.21  0.05 -0.01  0.01 -0.12
Hornet Sportabout   1.56 -0.81 -1.04  0.05 -0.38  0.14  0.15  0.08 -0.15
Valiant            -0.21 -2.19  0.14 -0.07  0.08 -0.27 -0.26  0.28  0.02
Duster 360          2.73  0.29 -0.58  0.53 -0.20  0.21 -0.40 -0.36 -0.17
Merc 240D          -1.80 -1.27  1.03  0.14 -0.40 -0.22  0.54 -0.33  0.03
Merc 230           -1.90 -1.93  1.96 -0.26 -0.61  0.08 -0.39  0.34 -0.19
Merc 280            0.02 -0.06  1.06  0.74 -0.14 -0.10  0.43  0.00  0.11
Merc 280C           0.04 -0.23  1.29  0.68 -0.08 -0.19  0.25  0.17  0.01
Merc 450SE          1.82 -0.68 -0.19  0.30  0.14  0.18  0.06  0.14  0.40
Merc 450SL          1.60 -0.68 -0.27  0.40  0.01  0.31 -0.03  0.22  0.20
Merc 450SLC         1.71 -0.80 -0.07  0.37  0.12  0.11 -0.21  0.35  0.14
Cadillac Fleetwood  3.54 -0.79  0.62 -0.84  0.35 -0.14  0.17 -0.10 -0.26
Lincoln Continental 3.65 -0.73  0.64 -0.87  0.36 -0.12  0.14 -0.17 -0.04
Chrysler Imperial   3.39 -0.52  0.40 -0.82  0.07  0.39  0.26 -0.36  0.22
Fiat 128           -3.53 -0.24 -0.33 -0.52  0.03  0.62  0.11  0.03  0.21
Honda Civic        -3.44  0.33 -0.42  0.17  0.28  0.46  0.16 -0.09 -0.30
Toyota Corolla     -3.85 -0.29 -0.35 -0.41 -0.13  0.85  0.00  0.16  0.02
Toyota Corona      -1.64 -1.98  0.10  0.62 -0.05 -0.14 -0.69 -0.46 -0.07
Dodge Challenger    1.55 -0.87 -0.91  0.33  0.03 -0.35  0.19  0.20 -0.03
AMC Javelin         1.44 -0.96 -0.77  0.37  0.04 -0.33  0.05  0.33 -0.06
Camaro Z28          2.92  0.37 -0.57  0.53 -0.06  0.04 -0.31 -0.47  0.07
```

```
Pontiac Firebird     1.81 -0.90 -0.96 -0.31 -0.39  0.19  0.38 -0.04 -0.12
Fiat X1-9           -3.22 -0.06 -0.45 -0.20  0.25 -0.06 -0.19  0.06  0.01
Porsche 914-2       -2.66  1.53 -0.28 -0.21 -0.32 -0.69  0.41 -0.25  0.09
Lotus Europa        -3.19  1.69 -0.52  0.01 -0.78 -0.06  0.16 -0.22  0.02
Ford Pantera L       1.60  3.10 -0.61 -0.69 -0.69 -0.60 -0.18  0.42  0.00
Ferrari Dino        -0.25  3.18  0.73  0.51  0.24 -0.06  0.22 -0.02 -0.01
Maserati Bora        2.63  4.40  0.97 -0.01 -0.27  0.57 -0.55  0.07 -0.04
Volvo 142E          -1.94  0.28  0.19 -0.46  0.58 -0.40 -0.48 -0.19  0.14
>
> # Biplot of first two components
> biplot(pc)
> # File:   Regression.R
> # Course: R: An Introduction (with RStudio)
>
> # INSTALL AND LOAD PACKAGES ##################################
>
> library(datasets)  # Load base packages manually
>
> # Installs pacman ("package manager") if needed
> if (!require("pacman")) install.packages("pacman")
>
> # Use pacman to load add-on packages as desired
> pacman::p_load(pacman, caret, lars, tidyverse)
Installing package into 'C:/Users/Admin/AppData/Local/R/win-library/4.4'
(as 'lib' is unspecified)
Warning: unable to access index for repository http://www.stats.ox.ac.uk/pub/
RWin/bin/windows/contrib/4.4:
  cannot open URL 'http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/4.4
/PACKAGES'
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/lars_1.3.zip'
Content type 'application/zip' length 233466 bytes (227 KB)
downloaded 227 KB

package 'lars' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\Admin\AppData\Local\Temp\Rtmps79HOh\downloaded_packages

lars installed
>
> # LOAD DATA ##################################################
>
> ?USJudgeRatings
> head(USJudgeRatings)
               CONT INTG DMNR DILG CFMG DECI PREP FAMI ORAL WRIT PHYS RTEN
AARONSON,L.H.   5.7  7.9  7.7  7.3  7.1  7.4  7.1  7.1  7.1  7.0  8.3  7.8
ALEXANDER,J.M.  6.8  8.9  8.8  8.5  7.8  8.1  8.0  8.0  7.8  7.9  8.5  8.7
ARMENTANO,A.J.  7.2  8.1  7.8  7.8  7.5  7.6  7.5  7.5  7.3  7.4  7.9  7.8
BERDON,R.I.     6.8  8.8  8.5  8.8  8.3  8.5  8.7  8.7  8.4  8.5  8.8  8.7
BRACKEN,J.J.    7.3  6.4  4.3  6.5  6.0  6.2  5.7  5.7  5.1  5.3  5.5  4.8
BURNS,E.B.      6.2  8.8  8.7  8.5  7.9  8.0  8.1  8.0  8.0  8.0  8.6  8.6
> data <- USJudgeRatings
>
> # Define variable groups
> x <- as.matrix(data[, -12])
> y <- data[, 12]
>
> # REGRESSION WITH SIMULTANEOUS ENTRY ##########################
>
> # Using variable groups
> reg1 <- lm(y ~ x)
>
> # Or specify variables individually
> reg1 <- lm(RTEN ~ CONT + INTG + DMNR + DILG + CFMG +
```

```
+              DECI + PREP + FAMI + ORAL + WRIT + PHYS,
+            data = USJudgeRatings)
>
> # Results
> reg1           # Coefficients only

Call:
lm(formula = RTEN ~ CONT + INTG + DMNR + DILG + CFMG + DECI +
    PREP + FAMI + ORAL + WRIT + PHYS, data = USJudgeRatings)

Coefficients:
(Intercept)         CONT         INTG         DMNR         DILG         CFMG
   -2.11943      0.01280      0.36484      0.12540      0.06669     -0.19453
       DECI         PREP         FAMI         ORAL         WRIT         PHYS
    0.27829     -0.00196     -0.13579      0.54782     -0.06806      0.26881

> summary(reg1)  # Inferential tests

Call:
lm(formula = RTEN ~ CONT + INTG + DMNR + DILG + CFMG + DECI +
    PREP + FAMI + ORAL + WRIT + PHYS, data = USJudgeRatings)

Residuals:
     Min       1Q   Median       3Q      Max
-0.22123 -0.06155 -0.01055  0.05045  0.26079

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.11943    0.51904  -4.083 0.000290 ***
CONT         0.01280    0.02586   0.495 0.624272
INTG         0.36484    0.12936   2.820 0.008291 **
DMNR         0.12540    0.08971   1.398 0.172102
DILG         0.06669    0.14303   0.466 0.644293
CFMG        -0.19453    0.14779  -1.316 0.197735
DECI         0.27829    0.13826   2.013 0.052883 .
PREP        -0.00196    0.24001  -0.008 0.993536
FAMI        -0.13579    0.26725  -0.508 0.614972
ORAL         0.54782    0.27725   1.976 0.057121 .
WRIT        -0.06806    0.31485  -0.216 0.830269
PHYS         0.26881    0.06213   4.326 0.000146 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1174 on 31 degrees of freedom
Multiple R-squared:  0.9916,   Adjusted R-squared:  0.9886
F-statistic: 332.9 on 11 and 31 DF,  p-value: < 2.2e-16

>
> # MORE SUMMARIES ##########################################
>
> anova(reg1)            # Coefficients w/inferential tests
Analysis of Variance Table

Response: RTEN
          Df Sum Sq Mean Sq   F value    Pr(>F)
CONT       1  0.058   0.058    4.1794 0.0494903 *
INTG       1 45.096  45.096 3270.7650 < 2.2e-16 ***
DMNR       1  1.300   1.300   94.3167 6.415e-11 ***
DILG       1  2.346   2.346  170.1567 3.963e-14 ***
CFMG       1  0.503   0.503   36.5172 1.086e-06 ***
DECI       1  0.214   0.214   15.5296 0.0004306 ***
PREP       1  0.164   0.164   11.9069 0.0016353 **
FAMI       1  0.039   0.039    2.7997 0.1043449
ORAL       1  0.439   0.439   31.8608 3.385e-06 ***
```

```
WRIT       1  0.065   0.065    4.7078 0.0378096 *
PHYS       1  0.258   0.258   18.7170 0.0001464 ***
Residuals 31  0.427   0.014
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> coef(reg1)              # Coefficients (same as reg1)
 (Intercept)          CONT          INTG          DMNR          DILG          CFMG
-2.119429682  0.012796377  0.364840272  0.125399138  0.066690976 -0.194527027
        DECI          PREP          FAMI          ORAL          WRIT          PHYS
 0.278292932 -0.001960111 -0.135790972  0.547817680 -0.068061595  0.268811919
> confint(reg1)           # CI for coefficients
                   2.5 %      97.5 %
(Intercept) -3.178010347 -1.06084902
CONT        -0.039955335  0.06554809
INTG         0.101011150  0.62866939
DMNR        -0.057571651  0.30836993
DILG        -0.225031708  0.35841366
CFMG        -0.495940888  0.10688683
DECI        -0.003683181  0.56026904
PREP        -0.491456059  0.48753584
FAMI        -0.680844080  0.40926214
ORAL        -0.017628284  1.11326364
WRIT        -0.710196975  0.57407378
PHYS         0.142088434  0.39553540
> resid(reg1)             # Residuals case-by-case
  AARONSON,L.H.   ALEXANDER,J.M.   ARMENTANO,A.J.     BERDON,R.I.     BRACKEN,J.
J.
   0.1674282950    0.1599043028    0.1318188003   -0.0721243488   -0.16635135
84
     BURNS,E.B.   CALLAHAN,R.J.      COHEN,S.S.       DALY,J.J.    DANNEHY,J.
F.
   0.0344455088   -0.1228672774   -0.0359845065   -0.0414643393    0.10548491
67
      DEAN,H.H.     DEVITA,H.J.   DRISCOLL,P.J.     GRILLO,A.E.  HADDEN,W.L.J
R.
   0.0315661299    0.0279048490   -0.0066302844    0.1215116258   -0.07071694
55
    HAMILL,E.C.     HEALEY.A.H.       HULL,T.C.       LEVINE,I.   LEVISTER,R.
L.
   0.0963751277    0.0966781231    0.0587324090    0.2607914304   -0.06137839
51
    MARTIN,L.F.    MCGRATH,J.F.    MIGNONE,A.F.     MISSAL,H.M.     MULVEY,H.
M.
  -0.0105476010   -0.0926140135   -0.0964022149   -0.0479617600    0.02799992
36
     NARUK,H.J.    O'BRIEN,F.J. O'SULLIVAN,T.J.       PASKEY,L.    RUBINOW,J.
E.
  -0.0633662511   -0.0142423076   -0.1918226956    0.0253091922   -0.01797252
62
     SADEN.G.A. SATANIELLO,A.G.       SHEA,D.M.    SHEA,J.F.JR.      SIDOR,W.
J.
  -0.0144131915    0.1145104470   -0.0617147925   -0.0608608820    0.04210192
15
  SPEZIALE,J.A.     SPONZO,M.J.  STAPLETON,J.F.      TESTO,R.J. TIERNEY,W.L.J
R.
   0.1474606096    0.0421784997   -0.2212325911   -0.0375263260   -0.00075377
99
      WALL,R.A.     WRIGHT,D.B.   ZARRILLI,K.J.
  -0.0024277845   -0.1204656347   -0.0603603048
> hist(residuals(reg1))  # Histogram of residuals
>
> # ADDITIONAL MODELS ########################################
>
> # Conventional stepwise regression
```

```
> stepwise <- lars(x, y, type = "stepwise")
>
> # Stagewise: Like stepwise but with better generalizability
> forward <- lars(x, y, type = "forward.stagewise")
>
> # LAR: Least Angle Regression
> lar <- lars(x, y, type = "lar")
>
> # LASSO: Least Absolute Shrinkage and Selection Operator
> lasso <- lars(x, y, type = "lasso")
>
> # Comparison of R^2 for new models
> r2comp <- c(stepwise$R2[6], forward$R2[6],
+             lar$R2[6], lasso$R2[6]) %>%
+   round(2)
> names(r2comp) <- c("stepwise", "forward", "lar", "lasso")
> r2comp  # Show values of R^2
stepwise  forward      lar    lasso
    0.99     0.99     0.99     0.99
```