

# iqr.R

davidharvey

Sat Aug 25 09:12:33 2018

```
# format is name = function(arguments passed to the code) {code}

iqr = function(x) {

# x is a vector that contains our data; we need to create a new
# vector sorted from smallest-to-largest value

  x_sorted = sort(x)

# we also need the vector's length

  x_length = length(x)

# how we find the median depends on whether the vector has an
# even or an odd number of elements; the modulus operator (%)
# tests if division by 2 has a remainder of zero; if it does the
# vector has an even number of elements and the IF statement
# divides the sorted vector into its lower and its upper halves;
# if the vector has an odd number of elements, then the ELSE
# statement divide it into its lower and its upper halves

  if(x_length %% 2 == 0) {
    mid = x_length/2
    lower = x_sorted[1:mid]
    upper = x_sorted[(mid + 1):x_length]
  } else {
    mid = (x_length/2) + 0.5
    lower = x_sorted[1:(mid - 1)]
    upper = x_sorted[(mid + 1):x_length]
  }

# now we find the median for each half and calculate the iqr

  f.lower = median(lower)
  f.upper = median(upper)
  iqr = f.upper - f.lower

# the print() command returns the iqr to the console, but we also
# create a list of values to return if the function is assigned
# to an object

  print(iqr)
  out = list("iqr" = iqr, "F_U" = f.upper, "F_L" = f.lower)
  invisible(out)
}
```