

# Workshop 6 Solution

*Veerasak Kritsanapraphan*

## 1. Loop practice

(a) Write a function called `calculateRowMeans` that uses a **for loop** to calculate the row means of a matrix `x`.

```
# calculateRowMeans computes the row means of a matrix x
# input: matrix x
# output: vector of length nrow(x) giving row means of x
calculateRowMeans <- function(x) {
  row.means <- numeric(nrow(x))
  for(i in 1:nrow(x)) {
    row.means[i] <- mean(x[i,])
  }
  row.means
}
```

(b) Try out your function on the random matrix `fake.data` defined below.

```
set.seed(12345) # Set seed of random number generator
fake.data <- matrix(rnorm(500), ncol=5)
calculateRowMeans(fake.data)
```

```
## [1] 0.104711160 -0.212815850 0.080097701 -0.104435950 0.168726034
## [6] -0.276477625 -0.062646728 0.366882193 -0.433799415 -0.004952225
## [11] -0.448751356 -0.122943620 0.775054241 1.140645559 -0.032540068
## [16] 0.058354128 -0.105059472 -0.752380885 0.460506777 0.122805690
## [21] 0.817373014 0.277099615 -0.003069567 0.078224338 -0.237097230
## [26] 0.364344827 -0.145794603 0.283380408 0.105536231 0.062912786
## [31] 0.626789243 0.547661893 0.652108071 0.069632813 0.918556841
## [36] -0.033380220 0.028618498 0.161430138 0.764679323 0.449573986
## [41] 0.351826141 -0.783320802 -0.172279252 -0.016911857 0.376613887
## [46] 0.626161998 -0.461096173 -0.063753735 -0.024108330 -0.133152285
## [51] -0.392474023 0.887754214 -0.266413458 -0.681254718 0.203629848
## [56] 0.302400050 0.764454675 -0.021683984 0.191472161 -0.944533784
## [61] -0.257833724 -0.371427022 0.122738610 -0.095647016 -0.039965777
## [66] -0.496087157 0.173478440 0.318222961 -0.060915512 -0.480218084
## [71] 0.154699485 0.306528684 -0.440338582 0.433748573 0.291561192
## [76] 0.258839366 0.777379894 -0.639731351 0.099893526 0.368316779
## [81] -0.456305624 0.185681802 -0.268762250 1.023826194 0.306709808
## [86] 0.472252016 0.776952506 0.354263312 0.060452361 0.177687987
## [91] -0.242639463 -0.144272519 0.472164602 -0.414840660 -0.122722673
## [96] 0.431275348 -0.380115931 0.110312883 -0.391319986 -0.380656883
```

(b) Use the `apply()` function to calculate the row means of the matrix `fake.data`

```
apply(fake.data, MARGIN=1, FUN=mean)
```

```
## [1] 0.104711160 -0.212815850 0.080097701 -0.104435950 0.168726034
## [6] -0.276477625 -0.062646728 0.366882193 -0.433799415 -0.004952225
## [11] -0.448751356 -0.122943620 0.775054241 1.140645559 -0.032540068
## [16] 0.058354128 -0.105059472 -0.752380885 0.460506777 0.122805690
## [21] 0.817373014 0.277099615 -0.003069567 0.078224338 -0.237097230
```

```
## [26] 0.364344827 -0.145794603 0.283380408 0.105536231 0.062912786
## [31] 0.626789243 0.547661893 0.652108071 0.069632813 0.918556841
## [36] -0.033380220 0.028618498 0.161430138 0.764679323 0.449573986
## [41] 0.351826141 -0.783320802 -0.172279252 -0.016911857 0.376613887
## [46] 0.626161998 -0.461096173 -0.063753735 -0.024108330 -0.133152285
## [51] -0.392474023 0.887754214 -0.266413458 -0.681254718 0.203629848
## [56] 0.302400050 0.764454675 -0.021683984 0.191472161 -0.944533784
## [61] -0.257833724 -0.371427022 0.122738610 -0.095647016 -0.039965777
## [66] -0.496087157 0.173478440 0.318222961 -0.060915512 -0.480218084
## [71] 0.154699485 0.306528684 -0.440338582 0.433748573 0.291561192
## [76] 0.258839366 0.777379894 -0.639731351 0.099893526 0.368316779
## [81] -0.456305624 0.185681802 -0.268762250 1.023826194 0.306709808
## [86] 0.472252016 0.776952506 0.354263312 0.060452361 0.177687987
## [91] -0.242639463 -0.144272519 0.472164602 -0.414840660 -0.122722673
## [96] 0.431275348 -0.380115931 0.110312883 -0.391319986 -0.380656883
```

(c) Compare this to the output of the `rowMeans()` function to check that your calculation is correct.

```
identical(calculateRowMeans(fake.data), apply(fake.data, MARGIN=1, FUN=mean))
```

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