Lab 9

AUTHOR
Breyonne Williams

Problem 1: Vectorization

```
fun1 <- function(n = 100, k = 4, lambda = 4){
  x <- NULL
  for (i in 1:n) {
    x <- rbind(x, rpois(k, lambda))
  }
  return(x)
}

fun1alt <- function(n = 100, k = 4, lambda = 4){
  x <- matrix(rpois(n * k, lambda), n, k)
  return(x)
}

library(microbenchmark)

microbenchmark::microbenchmark(
  fun1(),
  fun1alt()
)
```

```
Warning in microbenchmark::microbenchmark(fun1(), fun1alt()): less accurate
nanosecond times to avoid potential integer overflows

Unit: microseconds
      expr     min       lq      mean   median       uq      max neval
    fun1() 156.620 176.8945 192.47655 178.9855 185.5045 1363.660   100
 fun1alt()  12.013  12.9560  21.97764  13.7760  14.2270  832.218   100
```

```
library(matrixStats)

set.seed(1234)
x <- matrix(rnorm(1e4), nrow = 10)

fun2 <- function(x){
  apply(x, 2, max)
}

fun2alt <- function(x){
  colMax <- colSums2(x * (x == matrixStats::rowMaxs(x)))
  return(colMax)
}

library(microbenchmark)

microbenchmark::microbenchmark(
  fun2(x)
)
```

```
Unit: microseconds
    expr     min       lq     mean  median       uq      max neval
 fun2(x) 462.521 485.3375 526.9111 499.749 525.3945 1332.623   100
```

```
microbenchmark::microbenchmark(
  fun2alt(x)
)
```

```
Unit: microseconds
       expr    min      lq     mean median     uq      max neval
 fun2alt(x) 50.225 58.8555 70.52246 59.819 62.279 1026.968   100
```

Problem 3: Parallelization

```
library(parallel)

my_boot <- function(dat, stat, R, ncpus = 1L){
  n <- nrow(dat)
  idx <- matrix(sample.int(n, n*R, TRUE), nrow = n, ncol = R)
  cl <- makeCluster(ncpus)
  ans <- mclapply(seq_len(R), function(i){
    stat(dat[idx[,i],, drop = FALSE])
  }, mc.cores = ncpus)
  stopCluster(cl)
  ans <- do.call(rbind, ans)
  return(ans)
}

my_stat <- function(d) coef(lm(y ~ x, data = d))

set.seed(1)
n <- 500; R <- 1e4
x <- cbind(rnorm(n)); y <- x*5 + rnorm(n)

ans0 <- confint(lm(y~x))
ans1 <- my_boot(dat = data.frame(x, y), my_stat, R = R, ncpus = 2L)

t(apply(ans1, 2, quantile, c(.025, .975)))
```

```
                2.5%      97.5%
(Intercept) -0.1386903 0.04856752
x            4.8685162 5.04351239
```

```
ans0
```

```
                2.5 %     97.5 %
(Intercept) -0.1379033 0.04797344
x            4.8650100 5.04883353
```

```
system.time(my_boot(dat = data.frame(x, y), my_stat, R = 4000, ncpus = 1L))
```

```
   user  system elapsed
  0.994   0.022   1.099
```

```
system.time(my_boot(dat = data.frame(x, y), my_stat, R = 4000, ncpus = 2L))
```

```
   user  system elapsed
  1.082   0.061   0.677
```