On the computation of the p-d multivariate normal density

Yanfei Kang

2023-10-26

Contents

Parameter settings

```r
N = 10000
p = 100
# Setting mu and sigma
mu = rep(0, p)
set.seed(20231025)
sigma = cov(matrix(rnorm(N * p), N, p))
```

Evaluating $f(x)$ using cholesky decomposition

```r
# Evaluating f(x) using cholesky decomposition
mn_cholesky = function(x, mu, sigma) {
    R = chol(sigma)
    det_sigma = prod(diag(R))^2
    quad_term = crossprod(solve(t(R), x - mu))
    log_den1 = (-p/2) * log(2 * pi) - 0.5 * log(det_sigma) - 0.5 * (quad_term)
    return(log_den1)
}
```

Evaluating $f(x)$ directly

```r
mn_direct = function(x, mu, sigma) {
    log_den2 = (-p/2) * log(2 * pi) - 0.5 * log(det(sigma)) - 0.5 * t(x - mu) %*%
        solve(sigma) %*% (x - mu)
    return(log_den2)
}
```

Let's evaluate $f(x)$

```r
set.seed(20231025)
x = rnorm(p)
```

```
print(mn_cholesky(x, mu, sigma))
```

```
##            [,1]
## [1,] -136.9197
```

```
print(mn_direct(x, mu, sigma))
```

```
##            [,1]
## [1,] -136.9197
```

```
print(mvtnorm::dmvnorm(x, mu, sigma, log = TRUE))
```

```
## [1] -136.9197
```

Comparison of computing time

```
library(microbenchmark)
microbenchmark(mn_cholesky(x, mu, sigma), mvtnorm::dmvnorm(x, mu, sigma, log = TRUE),
    mn_direct(x, mu, sigma))
```

```
## Unit: microseconds
##                                        expr      min        lq      mean
##                   mn_cholesky(x, mu, sigma)  396.846  453.2685  612.9111
##  mvtnorm::dmvnorm(x, mu, sigma, log = TRUE)  366.059  436.9205  774.9825
##                     mn_direct(x, mu, sigma) 1013.850 1087.1035 1421.1759
##     median        uq      max neval
##   495.2430  546.7740 10258.09   100
##   581.2935  721.6605 13752.34   100
##  1187.7070 1313.0580 20625.22   100
```