Kernel R Code

Kang JuHyun

2026-01-02

```r
kernel_base <- function(X, q = 1 / ncol(X)) {

  n <- nrow(X)
  p <- ncol(X)

  normx <- drop((X^2) %*% rep(1, p))
  A <- X %*% t(X)
  D <- (-2 * A + normx) + outer(rep(1, n), normx)

  K <- exp(-q * D)
  K
}

# alpha select
kernel_ridge_fit <- function(K, y, lambda = 0.1) {
  n <- length(y)
  alpha <- solve(K + lambda * diag(n), y)
  alpha
}


# kernel function for prediction
kernel_cross <- function(X_train, X_new, q = 1 / ncol(X_train)) {

  norm_train <- rowSums(X_train^2)
  norm_new <- rowSums(X_new^2)

  D <- outer(norm_new, rep(1, length(norm_train))) +
       outer(rep(1, length(norm_new)), norm_train) -
       2 * X_new %*% t(X_train)

  exp(-q * D)
}

## kernel ridge function
kernel_ridge <- function(X, y, lambda = 0.1, q = 1 / ncol(X)) {

  # 1. kernel matrix
  K <- kernel_base(X, q)

  # 2. fit
  alpha <- kernel_ridge_fit(K, y, lambda)
```

```
  # 3. predict function
  predict <- function(X_new) {
    K_new <- kernel_cross(X, X_new, q)
    as.vector(K_new %*% alpha)
  }

  list(
    K = K,
    alpha = alpha,
    lambda = lambda,
    q = q,
    predict = predict
  )
}
```

Example 01

```
set.seed(1)
n <- 100
p <- 5
X <- matrix(runif(n * p), nrow = n)

f_true <- function(X) {
  sin(X[,1]) + 0.5 * X[,2]^2 + X[,3]
}
sigma <- 0.01
y <- f_true(X) + rnorm(n, sd = sigma)
c(mean(y), sd(y))
```

```
## [1] 1.0843606 0.4173145
```

```
model <- kernel_ridge(X, y, lambda = 0.05)

y_hat <- model$predict(X)
y_hat
```

```
##   [1] 0.7685623 0.6593477 1.0960091 1.4752902 0.6132164 1.2980106 1.3562629
##   [8] 0.8764704 1.2077233 0.9994596 1.5203628 0.5563030 1.4730653 1.4332427
##  [15] 1.4832060 0.7355232 1.5859043 1.7198367 1.4145972 1.2811774 1.5160191
##  [22] 0.5309174 1.0944644 0.6440888 1.4546046 1.0192078 0.4236821 0.4364951
##  [29] 1.1962538 1.3781287 0.9699990 0.6629247 0.8073954 1.0530357 1.5001021
##  [36] 1.5327439 1.6136932 0.8257787 1.5450075 0.9839754 1.2657097 1.4123725
##  [43] 1.6523557 0.7072720 1.2181153 1.0702699 0.4769460 0.8699493 1.1149926
##  [50] 1.8778524 1.4330202 1.8305530 0.9821772 0.9942846 0.6893397 0.2380645
##  [57] 0.7226027 0.9194913 1.0389153 1.3422459 1.3631718 0.9952205 0.8396395
##  [64] 1.4441869 1.6951694 0.5254515 0.4781942 1.5428536 0.9865816 1.0264751
##  [71] 0.6167586 1.1791968 1.0512697 0.5975608 0.8542160 1.3691513 1.2518391
##  [78] 0.8328379 1.4142652 1.9583951 0.5010050 1.2158604 1.6140966 0.8236210
##  [85] 1.1027478 0.3599159 1.2761317 0.5270608 0.7835057 0.5255419 0.7208592
##  [92] 0.3189168 1.5091623 1.5091101 1.2701064 1.6413095 0.5229783 0.8655073
##  [99] 0.8333165 1.7509757
```

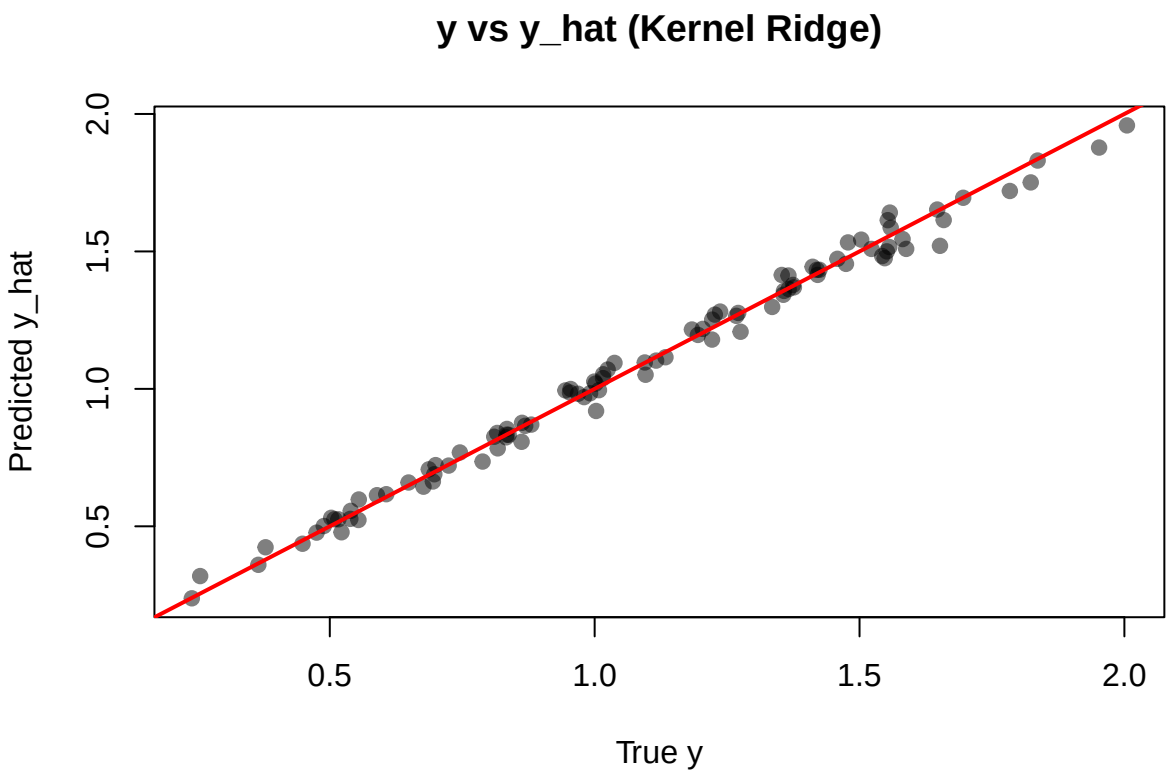
```
c(mean(y_hat), sd(y_hat))
```

```
## [1] 1.0829278 0.4084428
```

```
# mse
mean((y-y_hat)^2)
```

```
## [1] 0.001444678
```

Visualization of Ex01



Example 02

```
set.seed(1)
n <- 100
p <- 5
X <- matrix(runif(n * p), nrow = n)

f_true <- function(X) {
  X[,1] * X[,2] + sin(X[,3])
}

sigma <- 0.01
y <- f_true(X) + rnorm(n, sd = sigma)
c(mean(y), sd(y))
```

```
## [1] 0.6741035 0.3337271
```

```r
model <- kernel_ridge(X, y, lambda = 0.05)

y_hat <- model$predict(X)
y_hat
```

```
##   [1] 0.46579806 0.33477394 0.63966991 1.07198482 0.32912041 0.71523690
##   [7] 0.68628519 0.47542669 0.81728766 0.68648953 1.02523418 0.27828853
##  [13] 0.93713904 0.97489300 0.84798414 0.30447143 1.09582522 0.94761672
##  [19] 0.96872500 0.83453844 1.11185930 0.27206545 0.64967363 0.43786610
##  [25] 1.03425567 0.66083893 0.27058666 0.14613519 0.62704543 0.95460601
##  [31] 0.61375444 0.20947410 0.36822280 0.72928614 1.07578039 1.03389566
##  [37] 1.08139726 0.57153518 1.11354320 0.63359931 0.83314322 0.93959030
##  [43] 1.00285008 0.30105562 0.81656693 0.59098485 0.33121370 0.49685373
##  [49] 0.54402315 1.36540509 0.97635202 1.24051627 0.59616292 0.67261260
##  [55] 0.48633761 0.07435154 0.42632188 0.46915910 0.57074972 0.86941230
##  [61] 0.76325977 0.63022869 0.49015490 1.00923893 1.22301628 0.27532730
##  [67] 0.09989936 0.95687259 0.68361392 0.51277813 0.30834962 0.78346086
##  [73] 0.69508114 0.29241436 0.47430584 0.95503451 0.79896447 0.50449190
##  [79] 0.92589424 1.45971209 0.16975118 0.66170868 1.13821794 0.51882048
##  [85] 0.70196258 0.13336944 0.85766019 0.30005392 0.41804904 0.31477728
##  [91] 0.43079956 0.13275537 0.89450853 1.08639801 0.71355490 1.10777378
##  [97] 0.16370297 0.50525716 0.35853664 1.25285521
```

```r
c(mean(y_hat), sd(y_hat))
```

```
## [1] 0.6733848 0.3228268
```

```r
# mse
mean((y-y_hat)^2)
```

```
## [1] 0.001253911
```

Visualization of Ex02