Kernelized Ridge Regression Derivation

Alvin Wan

1 Kernelized Closed Form

Let K denote our kernel matrix. Take the objective function and solution for ridge regression.

$$\text{minimize}_w \|Xw - y\|_2^2 + \lambda \|w\|_2^2$$
$$w^* = (X^T X + \lambda I)^{-1} X^T y$$

Applying the matrix inversion lemma, we have

$$w^* = X^T (XX^T + \lambda I)^{-1} y$$

Finally, replace XX^T with the $n \times n$ kernel matrix K and its associated kernel function $k(\cdot, \cdot)$.

$$w^* = X^T (K + \lambda I)^{-1} y$$

Note that the weight update is the following, as we will draw comparisons with it later on.

$$\frac{\partial L}{\partial w} = X^T (Xw - y) + 2\lambda w$$

2 Inversion as Optimization

We now model computing $(K + \lambda I)^{-1}y$ as an optimization problem. Let B be our desired quantity.

$$B = (K + \lambda I)^{-1}y$$

Rearranging, we have the following expression. Note this is the typical formulation for least-squares. We substitute $\Lambda = K + \lambda I$.

$$(K + \lambda I)B = y$$
$$\Lambda B = y$$

Solving for B is equivalent to the following minimization problem.

$$\text{minimize}_B \|\Lambda B - y\|_2^2$$

Thus, we consider a weight update for stochastic gradient descent. Ignoring constant coefficients, we have the following:

$$\frac{\partial L}{\partial B} = \Lambda^T(\Lambda B - y)$$

Note that this update is identical to the weight update identified in part 1, where $\lambda = 0, X = \Lambda, w = B$. As in streaming stochastic gradient descent, our X (Λ in this case) is too large to fit in memory. As a result, we can simply run streaming stochastic gradient descent on this second optimization problem.

3 Prediction

Method 1

After computing B, we then have the following w^*.

$$w^* = X^T B$$

Finally, to classify X_{test}, we use the following

$$\hat{y} = X_{test} w^* = X_{test} X^T B$$

Method 2

We can alternatively leverage the second optimization problem. Using the terminology above, we have $\hat{y} = \Lambda B$. In terms of K, we have

$$\hat{y} = (K + \lambda I)B$$