EE5841 - Machine Learning: HW #7

Due on Tuesday, April 14, 2015

Havens TR 3:35pm

Mike Grimes

Problem 1

Kernelized Logistic Regression Derivation

Problem 1 continued on next page...

Starting with (1.1) (Keerthi 2002) which is identical the equation given in the homework

$$\min_{w,b} \frac{\lambda}{2}||w||^2 + \sum_i g(-y_i(w \cdot z_i - b)) \tag{1}$$

Where g is given by:

$$g(\xi) = log(1 + e^{\xi}) \tag{2}$$

We can rewrite 1.1 as:

$$\min_{w,b} \frac{1}{2}||w||^2 + C\sum_i g(\xi_i) \tag{3}$$

subject to:

$$\xi_i = -y_i(w \cdot z_i - b) \;\; \forall i \tag{4}$$

Taking the Lagrangian for this optimziation problem results in:

$$L = \frac{1}{2}||w||^2 + C\sum_i g(\xi_i) + \sum_i \alpha_i[-\xi_i - y_i(w \cdot z_i - b)] \tag{5}$$

Where the KKT conditions are:

$$\nabla_w L = w - \sum_i \alpha_i y_i z_i = 0 \tag{6}$$

$$\frac{\partial L}{\partial b} = \sum_i \alpha_i y_i = 0 \tag{7}$$

$$\frac{\partial L}{\partial \xi_i} = Cg'(\xi_i) - \alpha_i = 0 \;\; \forall i \tag{8}$$

We can rewrite w and ξ_i as functions of α_i using (7) and (8):

$$w(\alpha) = \sum_i \alpha_i y_i z_i, \;\; \xi_i(\alpha_i) = g'^{-1}(\frac{\alpha_i}{C}) \tag{9}$$

Let $\delta = \frac{\alpha_i}{C}$. Because ξ_i can be put in terms of α_i, consider the function

$$G(\delta) = \delta\xi_i - g(\xi_i) \tag{10}$$

You've seen all this already so you know eventually we get to

$$G(\delta) = \delta log\delta + (1 - \delta)log(1 - \delta). \tag{11}$$

Once we have this, we can turn 3 into the dual form:

$$\min f(a) = \frac{1}{2}||w(\alpha)||2 + C\sum_i G(\frac{\alpha_i}{C} \tag{12}$$

$$s.t. \;\; \sum_i \alpha_i y_i = 0 \tag{13}$$

Expanded out, with the kernel function:

$$\min f(a) = \frac{1}{2}||\sum_i \sum_j \alpha_i \alpha_j y_i y_j K(x_i, x_j)||^2 + C\sum_i G(\frac{\alpha_i}{C}) \tag{14}$$

Which is implemented in the optwrapper.m Matlab code file.

Problem 2

Kernelized Logistic Regression Implementation

Kernelized Logistic Regression was implemented using Matlab. Using k-fold cross- validation ($k = 5$), the training, validation, and testing errors were recored for 7 different λ values ($\lambda \in \{0.01, 0.05, 0.25, 1, 5, 25, 100\}$). Among the RBF kernel ($\sigma = 0.1$), a number of other kernels were tested (trigonometric, and polynomial). Running a few times didn't seem to show any trends for certain values of lambda. Perhaps I implemented something incorrectly, but I can't tell.

Overall, the kernelized logistic regression using the RBF kernel performs much better than a linear logistic regression (less than 10% testing error for most lambdas).

Matlab code is attached.

errors-eps-converted-to.pdf