COMP527
Data Mining and Visualisation
Problem Set 2

Danushka Bollegala

Question 1 Let us consider the hinge loss $h(y) = \max(0, y)$. Given a train dataset $\mathcal{D} = \{(\boldsymbol{x}_i, t_i)\}_{i=1}^{N}$, we define the loss of classifying an instance (x_n, t_n) by $h(-t_n \boldsymbol{w}^\top \boldsymbol{x}_n)$. Here, $t_n \in \{1, -1\}$ is the target label of the instance $\boldsymbol{x}_n$. Answer the following questions about the derivation of the perceptron update rule.

A. Plot the hinge loss as a function of y.

B. Compute the differential $h'(y) = \frac{dh(y)}{dy}$.

C. Let us define the loss associated with a single instance to be $L(\boldsymbol{x}_n, t_n) = h(-t_n \boldsymbol{w}^\top \boldsymbol{x}_n)$. Show that this loss function reflects the *error-driven learning* approach on which perceptron is based.

D. Write the stochastic gradient descent rule for obtaining a new vector $\boldsymbol{w}^{(t+1)}$ from the current weight vector $\boldsymbol{w}^{(t)}$ after observing a train instance $(\boldsymbol{x}_n, t_n)$. Assume learning rate to be η.

E. Show that when $\eta = 1$ the update rule you derived in part D becomes the perceptron update rule.

F. How does regularization prevent overfitting?

G. Let us now add an ℓ_2 regularizer $||\boldsymbol{w}||^2 = \boldsymbol{w}^\top \boldsymbol{w}$ to our loss function to design the following objective function.

$$L(\boldsymbol{x}_n, t_n) = h(-t_n \boldsymbol{w}^\top \boldsymbol{x}_n) + \lambda ||\boldsymbol{w}||2$$

Derive the perceptron update rule for this case.

H. Write the update rule for the logistic regression classifier.

I. Comparing part E and G, write the update rule for the regularised logistic regression.

Answers

A. Hinge loss function is shown in Figure 1.

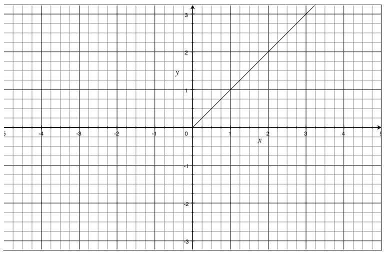

Figure 1: Hinge loss function.

B.

$$h'(y) = \begin{cases} 0 & \text{if } y < 0 \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

C. If t and $\boldsymbol{w}^\top \boldsymbol{x}_n$ are of opposite signs then we have an error. When this happens $-t\boldsymbol{w}^\top \boldsymbol{x}_n > 0$, and $h(-t\boldsymbol{w}^\top \boldsymbol{x}_n) = -t\boldsymbol{w}^\top \boldsymbol{x}_n > 0$. However, if there is no classification error, then $-t\boldsymbol{w}^\top \boldsymbol{x}_n < 0$ and $h(-t\boldsymbol{w}^\top \boldsymbol{x}_n) = 0$. Therefore, we will have a non-zero loss value only when there is a classification error.

D. Let us first compute the gradient of the loss function w.r.t. $\boldsymbol{w}$.

$$\frac{\partial L}{\partial w} = \underbrace{h'(-t_n \boldsymbol{w}^\top \boldsymbol{x}_n)}_{=1} \underbrace{\frac{\partial}{\partial \boldsymbol{w}} \left(-t_n \boldsymbol{w}^\top \boldsymbol{x}_n\right)}_{=-t_n \boldsymbol{x}_n} = -t_n \boldsymbol{x}_n \tag{2}$$

The SGD update rule is

$$\begin{aligned} \boldsymbol{w}^{(k+1)} &= \boldsymbol{w}^{(k)} - \eta \frac{\partial L}{\partial \boldsymbol{w}} \\ &= \boldsymbol{w}^{(k)} + \eta t_n \boldsymbol{x}_n \end{aligned} \tag{3}$$

E. When we set $\eta = 1$ in (3) we get the update rule for the perceptron which is,

$$\boldsymbol{w}^{(k+1)} = \boldsymbol{w}^{(k)} + t_n \boldsymbol{x}_n$$

F. Regularization methods such as ℓ_2 regularization impose a penalty on the length of the weight vector. Therefore, if we minimize both the loss and the regularization term, we obtain a weight vector that not only correctly classifies the train instances but also has lesser number of non-zero parameters. If the weight vector has most elements set to zero (or nearly zero), it can be considered as a simpler model compared to a weight vector that does not demonstrate this property. Therefore, from the Occam's razor principle we should prefer the simpler weight vector to avoid overfitting.

G. The gradient of the objective L w.r.t. $\boldsymbol{w}$ in this case will be

$$\frac{\partial L}{\partial \boldsymbol{w}} = \underbrace{-t_n \boldsymbol{x}_n}_{\text{from (2)}} + \lambda \underbrace{\frac{\partial}{\partial \boldsymbol{w}} \boldsymbol{w}^\top \boldsymbol{w}}_{=2\boldsymbol{w}} = -t_n \boldsymbol{x}_n + 2\lambda \boldsymbol{w}$$

The update rule in this case will be

$$\begin{aligned} \boldsymbol{w}^{(k+1)} &= \boldsymbol{w}^{(k)} - \eta \frac{\partial L}{\partial \boldsymbol{w}} \\ &= \boldsymbol{w}^{(k)} - \eta \left(-t_n \boldsymbol{x}_n + 2\lambda \boldsymbol{w}^{(k)} \right) \\ &= \boldsymbol{w}^{(k)}(1 - 2\eta\lambda) + \eta t_n \boldsymbol{x}_n \qquad (4) \end{aligned}$$

H. Update rule for the logistic regression classifier was

$$\boldsymbol{w}^{(k+1)} = \boldsymbol{w}^{(k)} - \eta (y_n - t_n) \boldsymbol{x}_n$$

where,

$$y_n = \frac{1}{1 + \exp(-\boldsymbol{w}^{(k)} \boldsymbol{x}_n)}$$

.

I. Comparing the update rules in (3) and (4) we see that the effect of adding an ℓ_2 regularization term is adding a $2\lambda\boldsymbol{w}$ term to the gradient of the objective function. Therefore, the update rule for the regularized logistic regression will be,

$$\begin{aligned} \boldsymbol{w}^{(k+1)} &= \boldsymbol{w}^{(k)} - \eta \left((y_n - t_n) \boldsymbol{x}_n + 2\lambda \boldsymbol{w}^{(k)} \right) \\ &= \boldsymbol{w}^{(k)}(1 - 2\eta\lambda) - \eta (y_n - t_n) \boldsymbol{x}_n \end{aligned}$$