L2 Regularisation for Perceptron

Procheta Sen

UNIVERSITY OF LIVERPOOL

L2 regularisation

- Let us denote by $L(b, \overline{W}, \mathscr{D})$ the Loss of classifying the dataset $\mathscr{D}$ using the model represented by the weight vector $\overline{W}$ and the bias term b
- We would like to impose L2 regularisation on $\overline{W}$.
- The overall objective to minimise can then be written as follows

$$J(b, \overline{W}, \mathscr{D}) = L(b, \overline{W}, \mathscr{D}) + \lambda \, | \, | \, \overline{W} \, | \, |_2^2 = L(b, \overline{W}, \mathscr{D}) + \lambda \sum_{i=1}^{d} w_i^2.$$

Here λ is called the **regularisation coefficient** and is usually set via **cross-validation**.

- The gradient of the overall objective simply becomes the sum of the loss-gradient and the scaled weight vector $\overline{W}$.

$$\nabla_{b,w_1,\ldots,w_d} J(b, \overline{W}, \mathscr{D}) = \nabla_{b,w_1,\ldots,w_d} L(b, \overline{W}, \mathscr{D}) + 2\lambda \cdot (0, w_1, \ldots, w_d)^T$$

L2 regularisation (SGD version)

$$\nabla_{b,w_1,\ldots,w_d} J(b, \overline{W}, \overline{X}, y) = \nabla_{b,w_1,\ldots,w_d} L(b, \overline{W}, \overline{X}, y) + 2\lambda \cdot (0, w_1, \ldots, w_d)^T$$

Loss function $L(b, \overline{W}, \mathscr{D})$ for Perceptron

$$L(b, \overline{W}, \mathscr{D}) = \sum_{k=1}^{n} L(b, \overline{W}, \overline{X}_k, y_k) = \sum_{k=1}^{n} h(-y_k \cdot a_k)$$

where $h(t) = \max(0,t)$

Use the gradient descent method:

$$(b, w_1, \ldots, w_d)^T \leftarrow (b, w_1, \ldots, w_d)^T - \mu \nabla_{b,w_1,\ldots,w_d} L(b, \overline{W}, \mathscr{D})$$

$$\nabla_{b,w_1,\ldots,w_d} L(b, \overline{W}, \mathscr{D}) = \sum_{k=1}^{n} \nabla_{b,w_1,\ldots,w_d} L(b, \overline{W}, \overline{X}_k, y_k) = \sum_{k=1}^{n} \nabla_{b,w_1,\ldots,w_d} h(-y_k \cdot a_k)$$

Computation of $\nabla_{b,w_1,\ldots,w_d} h(-y_k \cdot a_k)$

If $(\overline{X}_k, y_k)$ is misclassified, then

$$\nabla_{b,w_1,\ldots,w_d} h(-y_k \cdot a_k) = \left(\frac{\partial h(-y_k \cdot a_k)}{\partial b}, \frac{\partial h(-y_k \cdot a_k)}{\partial w_1}, \ldots, \frac{\partial h(-y_k \cdot a_k)}{\partial w_d} \right)^T = -y_k \cdot \left(1, x_k^{(1)}, x_k^{(2)}, \ldots, x_k^{(d)}\right)^T$$

Otherwise

$$\nabla_{b,w_1,\ldots,w_d} h(-y_k \cdot a_k) = (0,0,0,\ldots,0)^T$$

L2 regularisation for the Perceptron (SGD version)

$$\nabla_{b,w_1,\ldots,w_d} J(b, \overline{W}, \overline{X}, y) = \nabla_{b,w_1,\ldots,w_d} L(b, \overline{W}, \overline{X}, y) + 2\lambda \cdot (0, w_1, \ldots, w_d)^T$$

If $(\overline{X}, y)$ is misclassified, then

$$\nabla_{b,w_1,\ldots,w_d} L(b, \overline{W}, \overline{X}, y) = \nabla_{b,w_1,\ldots,w_d} h(-y \cdot a) = -y \cdot \left(1, x_1, \ldots, x_d\right)^T$$

Otherwise

$$\nabla_{b,w_1,\ldots,w_d} h(-y \cdot a) = (0,0,0,\ldots,0)^T$$

L2 regularisation for the Perceptron (SGD version)

$$\nabla_{b,w_1,\ldots,w_d} J(b, \overline{W}, \overline{X}, y) = \nabla_{b,w_1,\ldots,w_d} L(b, \overline{W}, \overline{X}, y) + 2\lambda \cdot (0, w_1, \ldots, w_d)^T$$

If $(\overline{X}, y)$ is misclassified, then

$$\nabla_{b,w_1,\ldots,w_d} J(b, \overline{W}, \overline{X}, y) = -y \cdot \left(1, x_1, \ldots, x_d\right)^T + 2\lambda \cdot (0, w_1, \ldots, w_d)^T$$

Otherwise

$$\nabla_{b,w_1,\ldots,w_d} J(b, \overline{W}, \overline{X}, y) = (0, 0, \ldots, 0)^T + 2\lambda \cdot (0, w_1, \ldots, w_d)^T$$

L2 regularisation for the Perceptron (SGD version)

$$\nabla_{b,w_1,\ldots,w_d} J(b, \overline{W}, \overline{X}, y) = \nabla_{b,w_1,\ldots,w_d} L(b, \overline{W}, \overline{X}, y) + 2\lambda \cdot (0, w_1, \ldots, w_d)^T$$

If $(\overline{X}, y)$ is misclassified, then

$$w_i = w_i \cdot (1 - 2\lambda) + y \cdot x_i$$

$$b = b + y$$

$$\nabla_{b,w_1,\ldots,w_d} J(b, \overline{W}, \overline{X}, y) = -y \cdot \left(1, x_1, \ldots, x_d\right)^T + 2\lambda \cdot (0, w_1, \ldots, w_d)^T$$

Otherwise

$$w_i = w_i \cdot (1 - 2\lambda)$$

$$b = b$$

$$\nabla_{b,w_1,\ldots,w_d} J(b, \overline{W}, \overline{X}, y) = (0, 0, \ldots, 0)^T + 2\lambda \cdot (0, w_1, \ldots, w_d)^T$$