SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY OF CHINA

Intro to Big Data Science: Assignment 4

Due Date: April 14, 2022

✏ **Exercise 1**

Log into "cookdata.cn", and enroll the course "数据科学导引". Finish the online exercise there.

✏ **Exercise 2**

The soft margin support vector classifier (SVC) is to solve the optimization problem:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i, \qquad s.t. \quad y_i(\mathbf{w}^T\mathbf{x}_i + b) \geqslant 1 - \xi_i, \quad \xi_i \geqslant 0, \quad i = 1, \ldots, n \tag{1}$$

1. Show that the KKT condition is

$$\begin{cases} \alpha_i \geqslant 0, \\ y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i \geqslant 0, \\ \alpha_i[y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i] = 0, \\ \mu_i \geqslant 0, \\ \xi_i \geqslant 0, \\ \mu_i\xi_i = 0, \\ \sum_{i=1}^{n}\alpha_i y_i = 0, \\ \mathbf{w} = \sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i, \\ \alpha_i + \mu_i = C, \end{cases}$$

where α_i and μ_i are the Lagrange multiplier for the constraints $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geqslant 1 - \xi_i$ and $\xi_i \geqslant 0$, respectively.

2. Show that the dual optimization problem is

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j - \sum_{i=1}^{n} \alpha_i,$$
$$s.t. \quad \sum_{i=1}^{n} \alpha_i y_i = 0, \quad 0 \leqslant \alpha_i \leqslant C, \quad i = 1, \ldots, n$$

3. Properties of Kernel:

   a) Using the definition of kernel functions in SVM, prove that the kernel $K(\mathbf{x}_i, \mathbf{x}_j)$ is symmetric, where $\mathbf{x}_i$ and $\mathbf{x}_j$ are the feature vectors for i-th and j-th examples.

   b) Given n training examples $(\mathbf{x}_i, \mathbf{x}_j)$ for $(i, j = 1, \ldots, n)$, the kernel matrix A is an $n \times n$ square matrix, where $A(i, j) = K(\mathbf{x}_i, \mathbf{x}_j)$. Prove that the kernel matrix A is semi-positive definite.

✏ **Exercise 3** (Linear Classifiers) We can also use linear function $f_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$ to make classification. The idea is as follows: if $f_{\mathbf{w}}(\mathbf{x}) > 0$, we assign 1 to label y; if $f_{\mathbf{w}}(\mathbf{x}) < 0$, we assign -1 to label y. This can be regarded as 0/1-loss minimization:

$$\min_{\mathbf{w}} \sum_{i=1}^{n} \frac{1}{2}(1 - y_i \operatorname{sign}(f_{\mathbf{w}}(\mathbf{x}_i))).$$

1. Given a two-class data set $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, we assume that there is a vector $\mathbf{w}$ satisfying $y_i \operatorname{sign}(f_{\mathbf{w}}(\mathbf{x}_i)) > 0$ for $i = 1, \ldots, n$. Show that the 0/1-loss minimization can be formulated as a linear programming problem:

$$\min_{\mathbf{w}} 0, \quad \text{subject to} \quad \mathbf{A}\mathbf{w} \geqslant \mathbf{1},$$

   where $A_{ij} = y_i x_{ij}$, $\mathbf{1} = (1, \ldots, 1)^T \in \mathbb{R}^n$, and the objective is dummy which means we don't have to minimize it.

2. Another way to solve 0/1-loss minimization is to replace it by l_2-loss (sometimes this is also called surrogate loss):

$$\min_{\mathbf{w}} \sum_{i=1}^{n} (1 - y_i f_{\mathbf{w}}(\mathbf{x}_i))^2 = \min_{\mathbf{w}} \sum_{i=1}^{n} (y_i - f_{\mathbf{w}}(\mathbf{x}_i))^2.$$

   Please give the analytical formula of the solution.

3. So far we have introduced two loss functions: $L_{0/1}(y, f) = \frac{1}{2}(1 - y \operatorname{sign} f)$ and $L_2(y, f) = (1 - yf)^2$. Show that the SVM can also be written as a loss minimization problem with the hinge loss function $L(y, f) = [1 - yf]_+ = \max\{1 - yf, 0\}$ (the positive part of the function $1 - yf$). Please also plot these three loss functions in the same figure and check their differences.

✏ **Exercise 4** (Logistic Regression)

We consider the following models of logistic regression for a binary classification with a sigmoid function $g(z) = \frac{1}{1+e^{-z}}$.

- Model 1: $P(Y=1|\mathbf{X}, w_1, w_2) = g(w_1 X_1 + w_2 X_2)$;
- Model 2: $P(Y=1|\mathbf{X}, w_1, w_2) = g(w_0 + w_1 X_1 + w_2 X_2)$.

We have three training examples:

$$\begin{aligned} \mathbf{x}^{(1)} &= (1,1)^T, & \mathbf{x}^{(2)} &= (1,0)^T, & \mathbf{x}^{(3)} &= (0,0)^T \\ y^{(1)} &= 1, & y^{(2)} &= -1, & y^{(3)} &= 1 \end{aligned}$$

1. Does it matter how the third example is labeled in Model 1? i.e., would the learned value of $\mathbf{w} = (w_1, w_2)$ be different if we change the label of the third example to -1? Does it matter in Model 2? Briefly explain your answer. (Hint: think of the decision boundary on 2D plane.)
2. Now, suppose we train the logistic regression model (Model 2) based on the n training examples $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}$ and labels $y^{(1)}, \dots, y^{(n)}$ by maximizing the penalized log-likelihood of the labels:

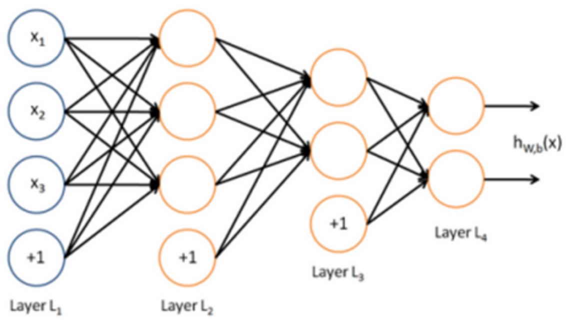
$$\sum_i \log P(y^{(i)}|\mathbf{x}^{(i)}, \mathbf{w}) - \frac{\lambda}{2}\|\mathbf{w}\|^2 = \sum_i \log g(y^{(i)}\mathbf{w}^T\mathbf{x}^{(i)}) - \frac{\lambda}{2}\|\mathbf{w}\|^2$$

   For large λ (strong regularization), the log-likelihood terms will behave as linear functions of w:

$$\log g(y^{(i)}\mathbf{w}^T\mathbf{x}^{(i)}) \approx \frac{1}{2} y^{(i)}\mathbf{w}^T\mathbf{x}^{(i)}.$$

   Express the penalized log-likelihood using this approximation (with Model 1), and derive the expression for MLE $\hat{\mathbf{w}}$ in terms of λ and training data $\{\mathbf{x}^{(i)}, y^{(i)}\}$. Based on this, explain how $\mathbf{w}$ behaves as λ increases. (We assume each $\mathbf{x}^{(i)} = (x_1^{(i)}, x_2^{(i)})^T$ and $y^{(i)}$ is either 1 or -1)

✏ **Exercise 5** (Back propagation in neural network) In a neural network, we have one layer of input $\mathbf{x} = \{x_i\}$, several hidden layers of hidden units $\{(z_j^{(l)}, a_j^{(l)})\}$, and a final layer of outputs y. Let $w_{ij}^{(l)}$ be the weight connecting unit j in layer l to unit i in layer $l+1$, $z_i^{(l)}$ and $a_i^{(l)}$ be the input and output of unit i in layer l before and after activation respectively, $b_i^{(l)}$ be the bias (intercept) of unit i in layer $l+1$. For an L-layer network with an



input $\mathbf{x}$ and an output y, the forward propagating network is established according to the weighted sum and nonlinear activation f:

$$\begin{aligned} z^{(l+1)} =& W^{(l)} a^{(l)} + b^{(l)}, \quad a^{(l+1)} = f(z^{(l+1)}), \quad \text{for} \quad l = 1, \dots, L-1 \\ a^{(1)} =& \mathbf{x}, \quad \text{and} \quad h_{W,b}(\mathbf{x}) = a^{(L)} \end{aligned}$$

We use the square error as our loss function:

$$J(W, b; \mathbf{x}, y) = \frac{1}{2} \| h_{W,b}(\mathbf{x}) - y \|^2,$$

then the sample mean of loss functions after penalization is

$$J(W, b) = \frac{1}{n} \sum_{i=1}^{n} J(W, b; \mathbf{x}, y) + \frac{\lambda}{2} \sum_{l=1}^{L-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} (w_{ji}^{(l)})^2.$$

1. In order to optimize the parameters W and b, we need to use gradient descent method to update their values:

$$w_{ij}^{(l)} \leftarrow w_{ij}^{(l)} - \alpha \frac{\partial}{\partial w_{ij}^{(l)}} J(W, b), \quad b_i^{(l)} \leftarrow b_i^{(l)} - \alpha \frac{\partial}{\partial b_i^{(l)}} J(W, b).$$

   The key point is to compute the partial derivatives $\frac{\partial}{\partial w_{ij}^{(l)}} J(W, b; \mathbf{x}, y)$ and $\frac{\partial}{\partial b_i^{(l)}} J(W, b; \mathbf{x}, y)$. Show that these two partial derivatives can be written in terms of the residual $\delta_i^{(l+1)} = \frac{\partial}{\partial z_i^{(l+1)}} J(W, b; \mathbf{x}, y)$:

$$\frac{\partial}{\partial w_{ij}^{(l)}} J(W, b; \mathbf{x}, y) = a_j^{(l)} \delta_i^{(l+1)} \quad \text{and} \quad \frac{\partial}{\partial b_i^{(l)}} J(W, b; \mathbf{x}, y) = \delta_i^{(l+1)}$$

2. Show that the residuals can be updated according to the following backward rule:

$$\delta_i^{(L)} = -(y_i - a_i^{(L)}) f'(z_i^{(L)}), \quad \text{and} \quad \delta_i^{(l)} = (\sum_{j=1}^{s_{l+1}} w_{ji}^{(l)} \delta_j^{(l+1)}) f'(z_i^{(l)}), \quad \text{for} \quad l = L-1, \dots, 2.$$