TTIC 31020: Introduction to Machine Learning
Autumn 2022

Problem Set #2

Collaborated with Haichuan Wang

Regularization

We will consider general L_p norm regularization for a convex loss function L

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} L(\mathbf{w}, \mathbf{x}_i, y_i) + \lambda \sum_{j=1}^{d} |w_j|^p \right\} \quad (1)$$

where $\mathbf{w}$ is a $d+1$-dimensional parameter vector, including the bias term (constant feature coefficient) w_0 which we do not regularize, and N is the size of the training set. $L(\mathbf{w}, \mathbf{x}_i, y_i)$ denotes the loss of the predictor parameterized by $\mathbf{w}$ on the training example $(\mathbf{x}_i, y_i)$. This is a general formulation, that covers, among others, least squares regression or log-loss classification.

Problem 1 **[15 points]**

Show that the objective in (1) is equivalent to

$$\begin{aligned} \mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} & \sum_{i=1}^{N} L(\mathbf{w}, \mathbf{x}_i, y_i) \\ \text{subject to} & \sum_{j=1}^{d} |w_j|^p \le \tau \end{aligned} \quad (2)$$

for an appropriate τ , which may depend in some way on the data and on λ. In other words, if you find $\mathbf{w}^*$ according to (1), then find $\mathbf{w}_1^*$ according to (2) with the appropriate τ , the two values of $\mathbf{w}_2^*$ will be the same.

- Let Γ_1 be (1)'s solution set and Γ_2 be (2)'s.

  As L is convex and $|w_j|^p$ is convex ($p \geq 1$), we can assume (1) has single solution. So $\Gamma_1 = \{w_1^*\}$.

- Choose $\tau = \sum_{j=1}^{d} |w_{1\,j}^*|^p$, pick a $w_2^* \in \Gamma_2$.

  then $\begin{cases} w_2^* = \underset{w}{\operatorname{argmin}} \sum_{i=1}^{N} L(w, x_i, y_i) \\ \sum_{j=1}^{d} |w_{2\,j}^*|^p \le \sum_{j=1}^{d} |w_{1\,j}^*|^p \end{cases}$

  $\Rightarrow \begin{cases} \sum_{i=1}^{N} L(w_2^*, x_i, y_i) \le \sum_{i=1}^{N} L(w_1^*, x_i, y_i) \\ \sum_{j=1}^{d} |w_{2\,j}^*|^p \le \sum_{j=1}^{d} |w_{1\,j}^*|^p \end{cases}$

- If any of the "≤" above is strictly "<", we would have

  $\sum_{i=1}^{N} L(w_2^*, x_i, y_i) + \lambda \sum_{j=1}^{d} |w_{2\,j}^*|^p < \sum_{i=1}^{N} L(w_1^*, x_i, y_i) + \lambda \sum_{j=1}^{d} |w_{1\,j}^*|^p$

  But w_1^* is supposed to be minimum solution, therefore, the two "≤" can only be "=".

- This tells us that w_2^*, when pluged into (1), also minimizes (1). So, $w_2^* \in \Gamma_1$. As Γ_1 has only only element, w_1^*, we have $w_1^* = w_2^*$.

Loss and Noise

Now we are going to look at a noise model which is a bit different from the i.i.d. Gaussian noise model described in class. Suppose that for every $\mathbf{x}$, the noise that affects y is still an additive zero-mean Gaussian, but the variance of this noise depends on $\mathbf{x}$:

$$y = \mathbf{w} \cdot \mathbf{x} + \nu, \quad \nu \sim \mathcal{N}(0, \sigma_{\mathbf{x}}^2) \tag{3}$$

Problem 2 [10 points]

Without knowing anything else besides the assumptions in (3), can we compute the maximum likelihood estimate for the linear regression parameters $\mathbf{w}^*$ from a given data set under this noise model? If yes, describe the procedure as precisely as you can; if not, explain why not.

- No.
- $p(y|X; w, \sigma)$

$$= \prod_{i=1}^{N} \frac{1}{\sigma_x \sqrt{2\pi}} \exp\left(-\frac{(y_i - w \cdot x_i)^2}{2\sigma_x^2}\right)$$

- $\frac{1}{N} \log p(y|X; w, \sigma)$

$$= \frac{1}{N} \sum_{i=1}^{N} \left(-\log(\sigma_{x_i} \sqrt{2\pi}) - \frac{(y_i - w \cdot x_i)^2}{2\sigma_{x_i}^2}\right)$$

- As the σ_{x_i} are unknown, we cannot solve to get

$$w^* = \arg\max_{w} \left\{ \frac{1}{N} \sum_{i=1}^{N} -\log(\sigma_{x_i} \sqrt{2\pi}) - \frac{(y_i - w x_i)^2}{2\sigma_{x_i}^2} \right\}$$

Problem 3 [10 points]

Now suppose we know the value of the noise variance $\sigma_{\mathbf{x}_i}^2$ at every training input $\mathbf{x}_i$ for $i = 1, \cdots, N$ (perhaps because each was obtained with a different sensor with known accuracy). With this additional assumption, can we compute the maximum likelihood estimate for linear regression parameters $\mathbf{w}^*$ from a given data set? If yes, describe the procedure as precisely as you can; if not, explain why not.

- Yes.
- We want to find w^* to maximize the loglikelihood. Take derivative:

$$\cdot \frac{\partial}{\partial w} \frac{1}{N} \sum_{i=1}^{N} -\log(\sigma_i \sqrt{2\pi}) - \frac{(y_i - w \cdot x_i)^2}{2\sigma_i^2}$$

$$= \frac{\partial}{\partial w} \frac{1}{N} \sum_{i=1}^{N} - \frac{y_i^2 - 2 y_i w x_i + (w x_i)^2}{2\sigma_i^2}$$

$$= \frac{1}{N} \sum_{i=1}^{N} - \frac{-2 y_i x_i + 2 (w x_i) x_i}{2\sigma_i^2} \quad \rightarrow \; = x_i (x_i^T w) = (x_i x_i^T) w$$

$$= \frac{1}{N} \sum_{i=1}^{N} - \frac{-2 y_i x_i + (x_i x_i^T) w}{2\sigma_i^2}$$

- Set the derivative $= 0$,

$$\frac{1}{N} \sum_{i=1}^{N} - \frac{-y_i x_i + (x_i x_i^T) w}{2\sigma_i^2} = 0$$

$$\left[\sum_{i=1}^{N} \frac{x_i x_i^T}{\sigma_i^2}\right] w = \left[\sum_{i=1}^{N} \frac{y_i x_i}{\sigma_i^2}\right]$$

We can solve for w.

Softmax

In this section we will consider a discriminative model for a multi-class setup, in which the class labels take values in $\{1, \cdots, C\}$. A principled generalization of the logistic regression model to this setup is the softmax model. It requires that we maintain a separate parameter vector $\mathbf{w}_c$ for each class c. Under this model, the estimate for the posterior for class c, $c = 1, \cdots, C$ is

$$\hat{p}(y = c|\mathbf{x}; \mathbf{W}) = \text{softmax}(\mathbf{w}_c \cdot \mathbf{x}) \triangleq \frac{\exp(\mathbf{w}_c \cdot \mathbf{x})}{\sum_{y=1}^{C} \exp(\mathbf{w}_y \cdot \mathbf{x})} \tag{4}$$

where $\mathbf{W}$ is a $C \times d$ matrix, the c^{th} row of which is a vector $\mathbf{w}_c$ associated with class c. We will assume throughout the problem set that $\mathbf{x}$ is the feature vector associated with an input example, including the constant feature $x_0 = 1$.

Problem 4 **[15 points]**
Show that the softmax model as stated in (4) is over-parameterized, that is, show that for any value $\mathbf{w}_c$ for $c = 1, \cdots, C$ there is a different value that yields exactly the same $p(y|\mathbf{x})$ for every $\mathbf{x}$. Then explain how this implies that we only need $C - 1$ trainable parameter vectors for softmax, and not C. Explain how this understanding also shows that for $C = 2$ softmax is equivalent to the logistic regression derived in class for binary classification.

- If $w_1, w_2, \dots, w_C$ maximize the likelihood

then $w_1 - w_1, w_2 - w_1, \dots, w_C - w_1$ also maximize the likelihood, b/c $\frac{e^{(w_c - w_1)\cdot x}}{\sum e^{(w_y - w_1)x}} = \frac{e^{w_c x} \cdot \cancel{e^{-w_1 x}}}{\sum e^{w_y x}\, \cancel{e^{-w_1 x}}} = \frac{e^{w_c x}}{\sum e^{w_y x}}$.

Then, we can define a new set of weights

$$\begin{array}{ccccc} w_1 - w_1, & w_2 - w_1, & \dots, & w_C - w_1 \\ \| & \| & & \| \\ 0, & w_2', & \dots, & w_C' \end{array}$$

which uses only $C-1$ variable.

- when $C=2$, let the new set of weights be $(w_1 - w_1, w_2 - w_1) = (0, w_2')$. Then,

$$p(y=1 \mid X; w) = \frac{e^{0\cdot x}}{e^{0\cdot x} + e^{w_2' \cdot x}} = \frac{1}{1 + e^{w_2' x}}$$

which is the logistic regression for binary classification.

Problem 5 **[10 points]**

Write down (precisely and as simplified as you can) the expression for (a) the log-loss of the linear C-way softmax classification model using feature mapping $\phi)\mathbf{x})$, and (b) its gradient with respect to $\mathbf{W}$. You can work in the stochastic gradient descent setting, i.e., compute the loss and the gradient for a single training example $\mathbf{x}$ with label $y \in [C]$.

- Assume x is a single point:

- log loss: $-\log p(y=c \mid x; W) = -\log \frac{e^{w_c \cdot \phi(x)}}{\sum_{y=1}^{C} e^{w_y \cdot \phi(x)}}$

w_c is a vector

- gradient: $\frac{\partial}{\partial w_c} -\log \frac{e^{w_c \cdot \phi(x)}}{\sum_{y=1}^{C} e^{w_y \phi(x)}}$

$$= \frac{\partial}{\partial w_c} - \left(w_c \phi(x) - \log \sum_{y=1}^{C} e^{w_y \phi(x)} \right)$$

$$= -\phi(x) + \frac{\phi(x) \cdot e^{w_c \phi(x)}}{\sum_{y=1}^{C} e^{w_y \cdot \phi(x)}}$$