Gradients of logistic regression with squared error and binary cross-entropy loss functions

Pawel Wocjan

February 17, 2019

Abstract

We derive the gradients for logistic regression with (a) mean squared error and (b) binary cross-entropy as loss functions. You will need these results for the first homework.

1 Logistic regression

Let

$$\boldsymbol{w} = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} \in \mathbb{R}^n \quad \text{and} \quad b \in \mathbb{R} \tag{1}$$

be the weight vector and the bias of a neuron, respectively. Let $a : \mathbb{R} \to \mathbb{R}$ denote its activation function a. The neuron outputs

$$\hat{y} = a\left(\sum_{j=1}^{n} w_j x_j + b\right) \tag{2}$$

when given a feature vector $\boldsymbol{x} = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$ as input. We use z to denote

$$z = \sum_{j=1}^{n} w_j x_j + b\,. \tag{3}$$

The function implemented by the neuron consists of two steps:

$$\boldsymbol{x} \quad \mapsto \quad z = \boldsymbol{w}^T \boldsymbol{x} + b = \sum_{j=1}^{n} w_j x_j + b \quad \mapsto \quad a = a(z). \tag{4}$$

For logistic regression the activation function $a(z)$ is equal to the sigmoid function $\sigma(z)$, which is defined by

$$\sigma(z) = \frac{1}{1+e^{-z}}. \tag{5}$$

The sigmoid function σ maps $\mathbb{R}$ to the open interval $(0,1)$. It satisfies the following properties:

$$\begin{aligned}
\sigma(0) &= \frac{1}{2} && (6)\\
\sigma(-z) &= 1-\sigma(z) && (7)\\
\lim_{z\to-\infty} &= 0 && (8)\\
\lim_{z\to\infty} &= 1. && (9)
\end{aligned}$$

Its derivative $\sigma'(z)$ is obtained by applying the chain rule and simple algebraic manipulations:

$$\begin{aligned}
\sigma'(z) &= -\frac{1}{(1+e^{-z})^2}\cdot e^{-z}\cdot(-1) && (10)\\
&= \frac{1}{1+e^{-z}}\cdot\frac{e^{-z}}{1+e^{-z}} && (11)\\
&= \frac{1}{1+e^{-z}}\cdot\frac{1+e^{-z}-1}{1+e^{-z}} && (12)\\
&= \sigma(z)\cdot(1-\sigma(z)) && (13)
\end{aligned}$$

Logistic regression can be used for binary classification, which is the task of classifying the feature vectors into two classes, denoted by 0 and 1. The activation $\sigma(z)$ can be interpreted as the probability that the neuron assigns to the class 1. Using this probability, we make a prediction as follows:

$$\begin{aligned}
&\text{class } 0 \quad \text{if} \quad \sigma(z) < \frac{1}{2} && (14)\\
&\text{class } 1 \quad \text{if} \quad \sigma(z) \geq \frac{1}{2} && (15)
\end{aligned}$$

2 Squared error and binary entropy loss functions

We consider two loss functions for logistic regression: squared error and binary cross-entropy. Let $\boldsymbol{x} \in \mathbb{R}^n$ be a feature vector and $y \in \{0,1\}$ its correct label.

- The squared error loss $\mathcal{L}_{\text{se}}$ is defined by

$$\mathcal{L}_{\text{se}} = \frac{1}{2}(a-y)^2\,. \tag{16}$$

  Its derivative with respect to a is equal to

$$\frac{\mathrm{d}\mathcal{L}_{\text{se}}}{\mathrm{d}a} = a-y. \tag{17}$$

- The (binary) cross-entropy loss is defined by

$$\mathcal{L}_{\text{ce}} = -y \log a - (1-y)\log(1-a)\,. \tag{18}$$

Its derivative with respect with a is equal to

$$\frac{\mathrm{d}\mathcal{L}_{\text{be}}}{\mathrm{d}a} = -\frac{y}{a} + \frac{1-y}{1-a}. \tag{19}$$

I will explain the intuition behind the cross-entropy loss in class in detail. For the analysis, consider the following two cases:

- If the true label is $y = 1$, then the loss is equal to $-\log a$, which is a strictly decreasing function on the open interval $(0,1)$. The loss tends to ∞ as $a \to 0$ and to 0 as $a \to 1$, respectively.
- If the true label is $y = 0$, then the loss is equal to $-\log(1-a)$, which is a strictly increasing function of the open interval $(0,1)$. The loss tends to ∞ as $a \to 1$ and to 0 as $a \to 0$, respectively.

3 Gradient of squared error and binary cross-entropy loss functions

We have compute the partial derivatives of the loss functions with respect to w_j and b to be able to apply stochastic gradient descent. This is done by applying the chain rule multiple times according to the following computational graph:

$$w_1, \dots, w_n, b \to z \to a \to \mathcal{L}, \tag{20}$$

where $\to$ indicates that the variables on the LHS influence those on the RHS.

Recall that derivative of the activation function a is $a' = (1-a)$ because the sigmoid function is used as the activation function for logistic regression.

- The partial derivatives of the squared error loss $\mathcal{L}_{\text{se}}$ are derived as follows:

$$\frac{\partial \mathcal{L}_{\text{se}}}{\partial w_j} = \frac{\mathrm{d}\mathcal{L}_{\text{se}}}{\mathrm{d}a} \cdot \frac{\partial a}{\partial z} \cdot \frac{\partial z}{\partial w_j} = (a-y) \cdot a' \cdot x_j \tag{21}$$

$$\frac{\partial \mathcal{L}_{\text{se}}}{\partial b} = \frac{\partial \mathcal{L}_{\text{se}}}{\partial b} \cdot \frac{\partial a}{\partial z} \cdot \frac{\partial z}{\partial b} = (a-y) \cdot a' \tag{22}$$

Note that $a' \approx 0$ whenever z is either very small or very large. In either case, we say that the neuron is saturated. The problem is that a saturated neuron learns slowly when a is very far from y. This can be improved by using the binary cross entropy loss function.

- The partial derivatives of the cross entropy loss $\mathcal{L}_{\mathrm{ce}}$ are derived as follows:

$$
\begin{aligned}
\frac{\partial \mathcal{L}_{\mathrm{ce}}}{\partial w_j} &= \frac{\mathrm{d}\mathcal{L}_{\mathrm{ce}}}{\mathrm{d}a} \cdot \frac{\partial a}{\partial z} \cdot \frac{\partial z}{\partial w_j} && (23)\\
&= \left(-\frac{y}{a} + \frac{1-y}{1-a}\right) \cdot a' \cdot x_j && (24)\\
&= \left(-\frac{y}{a} + \frac{1-y}{1-a}\right) \cdot a \cdot (1-a) \cdot x_j && (25)\\
&= \Big(-y \cdot (1-a) + (1-y) \cdot a\Big) \cdot x_j && (26)\\
&= (a-y) \cdot x_j && (27)
\end{aligned}
$$

$$
\frac{\partial \mathcal{L}_{\mathrm{be}}}{\partial b} = \frac{\mathrm{d}\mathcal{L}_{\mathrm{ce}}}{\mathrm{d}a} \cdot \frac{\partial a}{\partial z} \cdot \frac{\partial z}{\partial b} = a - y. \tag{28}
$$

Here it is essential that we expand a' as $a \cdot (1-a)$ to cancel a and $1-a$ is the denominators.