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

Pawel Wocjan

January 30, 2019

Abstract

We derive the gradients for logistic regression with (a) mean square error and (b) binary cross-entropy as loss functions.

1 Logistic regression

Let

$$\boldsymbol{w} = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} \in \mathbb{R}^n$$

be the weight vector and $b \in \mathbb{R}$ the bias of a neuron with the activation function a. When given the feature vector $\boldsymbol{x}$ as input it produces the output

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

We use z to denote

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

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

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

See Colab notebook for graph of sigmoid function. It maps $\mathbb{R}$ to the open interval $(0, 1)$ and has the following properties:

$$\begin{aligned}
\sigma(0) &= \frac{1}{2} && (2)\\
\sigma(-z) &= 1 - \sigma(z) && (3)\\
\lim_{z\to-\infty} &= 0 && (4)\\
\lim_{z\to\infty} &= 1. && (5)
\end{aligned}$$

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

$$\begin{aligned}
\sigma'(z) &= -\frac{1}{(1+e^{-z})^2} \cdot e^{-z} \cdot (-1) && (6)\\
&= \frac{1}{1+e^{-z}} \cdot \frac{e^{-z}}{1+e^{-z}} && (7)\\
&= \frac{1}{1+e^{-z}} \cdot \frac{1+e^{-z}-1}{1+e^{-z}} && (8)\\
&= \sigma(z) \cdot (1 - \sigma(z)) && (9)
\end{aligned}$$

Logistic regression can be used binary classification. The activation $\sigma(z)$ can be interpreted as the probability that the neuron assigns to the class 1. Using this probability, we predict the label as follows:

$$\begin{aligned}
&0 \quad \text{if} \quad \sigma(z) < \frac{1}{2} && (10)\\
&1 \quad \text{if} \quad \sigma(z) \geq \frac{1}{2} && (11)
\end{aligned}$$

2 Loss functions

We consider two loss functions for logistic regression: squared error and binary cross-entropy. Let a denote the activation of the neuron and y the correct label. The squared error loss is defined by

$$\mathcal{L}_{\text{se}} = (a - y)^2$$

where a is the activation of the neuron and y is the correct label. The (binary) cross-entropy loss is defined by

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

I will explain the intuition behind the cross entropy loss in class in detail. Note that if the true label is $y = 1$, then the loss is equal to $-\log a$, which is a strictly decreasing function of the open interval $(0, 1)$. The loss increases unboundedly as $a \to 0$ and goes to 0 as $a \to 1$.

In contrast, 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 $(0, 1)$. Thus, the loss increases unboundedly as $a \to 1$ and goes to 0 as $a \to 0$.

To avoid cumbersome notation we omit that the loss function depends on the weights w_j's and the biases, that a depends on z, and z in turn depends on the weights w_j's and the bias b.

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.

The partial derivatives for the squared error loss are:

$$\begin{aligned}
\frac{\partial \mathcal{L}_{\text{se}}}{\partial w_j} &= \frac{\mathrm{d}\mathcal{L}_{\text{se}}}{\mathrm{d}a} \cdot \frac{\partial a}{\partial w_j} && (12)\\
&= (a-y) \cdot \frac{\partial a}{\partial w_j} && (13)\\
&= (a-y) \cdot \frac{\partial a}{\partial z} \cdot \frac{\partial z}{\partial w_j} && (14)\\
&= (a-y) \cdot \sigma'(z) \cdot \frac{\partial z}{\partial w_j} && (15)\\
&= (a-y) \cdot \sigma(z)(1-\sigma(z)) \cdot \frac{\partial z}{\partial w_j} && (16)\\
&= (a-y) \cdot \sigma(z)(1-\sigma(z)) \cdot x_j && (17)\\
\frac{\partial \mathcal{L}_{\text{se}}}{\partial b} &= (a-y) \cdot \sigma(z)(1-\sigma(z)). && (18)
\end{aligned}$$

The partial derivatives for the cross-entropy loss are:

$$\begin{aligned}
\frac{\partial \mathcal{L}_{\text{ce}}}{\partial w_j} &= \frac{\mathrm{d}\mathcal{L}_{\text{ce}}}{\mathrm{d}a} \cdot \frac{\partial a}{\partial w_j} && (19)\\
&= \left(-\frac{y}{a} - (1-y)\frac{1}{1-y}\right) \cdot \frac{\partial a}{\partial w_j} && (20)\\
&= \left(-\frac{y}{a} - (1-y)\frac{1}{1-y}\right) \cdot \sigma(z)(1-\sigma(z)) \cdot \frac{\partial z}{\partial w_j} && (21)\\
\frac{\partial \mathcal{L}_{\text{be}}}{\partial w_j} &= \left(-\frac{y}{a} - (1-y)\frac{1}{1-y}\right) \cdot \sigma(z)(1-\sigma(z)). && (22)
\end{aligned}$$