Lecture Notes: Binary logistic regression

CSCI 375: NLP / Williams College / Fall 2024

1 Deriving NLL for one document

Let's derive our loss function[1] for binary logistic regression: negative log likelihood.[2] We will find weights, θ, that minimize the loss function and use them in our logistic regression classifier.

Notation. We begin with a bit of notation. Our *training data* consists of a matrix, X, the featurized version of the input text data, and $\vec{y}$, the labels (typically annotations on the documents by humans).

We index into a single element of these (a single document and a single label) with i, resulting in $(\vec{x}_i, y_i)$ where $y_i \in \{0, 1\}$. In a bag-of-words feature representation, $|\vec{x}| = |V|$ where V is the set of vocabulary words.

Set-up. Recall, the weights in our logistic regression model are θ and our model gives a prediction for the probability of the positive class given the input

$$\hat{p}_i := P_\theta(y = 1|x_i) = \frac{1}{1 + e^{-x_i \cdot \theta}} \tag{1}$$

Now let's use the fact that y_i is binary to write a generic probability that incorporates the case in which $y_i = 1$ and $y_i = 0$. We can rewrite this as

$$p_\theta(y_i|x_i) = \hat{p}_i^{y_i}(1 - \hat{p}_i)^{1-y_i} \tag{2}$$

Why does this work? Well suppose the true label $y_i = 1$ then we have

$$\begin{aligned} p_\theta(y_i|x_i) &= \hat{p}_i^1(1 - \hat{p}_i)^{1-1} && (3)\\ &= \hat{p}_i && (4) \end{aligned}$$

If instead $y_i = 0$ then

$$\begin{aligned} p_\theta(y_i|x_i) &= \hat{p}_i^0(1 - \hat{p}_i)^{1-0} && (5)\\ &= 1 - \hat{p}_i && (6) \end{aligned}$$

Both of these are true for how we defined $\hat{p}_i$ in Equation 1.

Loss function.

We'll use principles of maximum likelihood estimation to define a loss function. We want to set parameters θ that maximize the likelihood of the data, $P_\theta(y_i|x_i)$. For reasons we'll see later, this is equivalent to saying we want to minimize the negative likelihood (NL):

$$\begin{aligned} NL(\hat{p}_i, y_i) &= -P_\theta(y_i|x_i) && (7)\\ &= -\hat{p}_i^{y_i}(1 - \hat{p}_i)^{1-y_i} && (8) \end{aligned}$$

by substituting Equation 2.

[1] Other books might call this a "objective function" or a "cost function".
[2] Note, our book calls this "cross entropy loss".

Like we've seen before, it'll be easier to use logs in implementation so we take the log of both sides and this becomes the negative log likelihood (NLL)

$$NLL(\hat{p}_i, y_i) = -\log\left(\hat{p}_i^{y_i}(1-\hat{p}_i)^{1-y_i}\right) \tag{9}$$

$$= -y_i \log \hat{p}_i - (1-y_i)\log(1-\hat{p}_i) \tag{10}$$

Intution. Why does the negative log likelihood work?

Suppose the true label is $y_i = 1$ and our model predicts $\hat{p}_i = 0.99$.

Then from Equation 10 we have

$$NLL(0.99, 1) = -1\log(0.99) - (1-1)\log(1-0.99) \tag{11}$$

$$= -1\log(0.99) \tag{12}$$

$$= -1 * (-0.01) \tag{13}$$

$$= 0.01 \tag{14}$$

which is very close to zero and intuitively we consider this "successful."

If instead we predicted for this same example i, $\hat{p}_i = 0.6$ we have

$$NLL(0.6, 1) = -1\log(0.6) - (1-1)\log(1-0.6) \tag{15}$$

$$= -1\log(0.6) \tag{16}$$

$$= -1 * (-0.51) \tag{17}$$

$$= 0.51 \tag{18}$$

this gives us a higher number (which is worse) indicating that we were not as successful with our model.

Try out similar examples for yourself for $y_i = 0$.

2 Deriving NLL for entire training set

Let's be even more explicit about why we want this negative log likelihood for the entire training data, X and y. First we assume that our examples are independent and identically distributed (iid).

Recall, two variables, let's call them A and B, are independent if an only iff $P(A, B) = P(A), P(B)$

Under this i.i.d. assumption, the log likelihood of the entire training data is

$$\log p(y|X) = \log p(y_1, y_2, \ldots, y_n|\vec{x}_1, \vec{x}_2, \ldots \vec{x}_n) = \log \prod_{i=1,2,\cdots,n} p(y_i|x_i) \tag{19}$$

$$= \sum_{i=1,2,\cdots,n} \log p(y_i|x_i) \tag{20}$$

$$= -\sum_{i=1,2,\cdots,n} \text{NLL}(\theta, x_i, y_i) \tag{21}$$

3 Gradient of NLL

We use **gradient descent** to find the weights θ that minimize the negative log likelihood objective function. Let's keep things simple and just look at this gradient for a single document i

$$\hat{\theta} = \underset{\theta}{\text{argmin}}\ NLL(\hat{p}_i, y_i) \tag{22}$$

$$= \underset{\theta}{\text{argmin}} \Big(- y_i \log \hat{p}_i - (1 - y_i) \log(1 - \hat{p}_i) \Big) \tag{23}$$

$$= \underset{\theta}{\text{argmin}} \Big(- y_i \log \big(\frac{1}{1 + e^{-x_i \cdot \theta}}\big) - (1 - y_i) \log \big(1 - \frac{1}{1 + e^{-x_i \cdot \theta}}\big) \Big) \tag{24}$$

Now we need the gradient of the NLL with respect to $\vec{\theta}$,

$$\nabla_\theta \text{NLL}(\vec{\theta}) := [\frac{\partial}{\partial \theta_1} \text{NLL}(\theta), \frac{\partial}{\partial \theta_2} \text{NLL}(\theta), \ldots, \frac{\partial}{\partial \theta_k} \text{NLL}(\theta),] \tag{25}$$

Let's start by taking the partial derivative (gradient in one dimension) of θ_1 for a single exmaple i. Then

$$\frac{\partial}{\partial \theta_1} \Bigg(- y_i \log \left(\frac{1}{1 + e^{-x_i \cdot \theta}}\right) - (1 - y_i) \log \left(1 - \frac{1}{1 + e^{-x_i \cdot \theta}}\right) \Bigg) \tag{26}$$

First, we showed the derivative of the logistic function, $\sigma(\cdot)$ in HW0

$$\frac{\partial}{\partial u} \sigma(u) = \frac{\partial}{\partial u} \left(\frac{1}{1 + e^{-u}}\right) \tag{27}$$

$$= \frac{\partial}{\partial u} (1 + e^{-u})^{-1} \tag{28}$$

$$= (-1)(1 + e^{-u})^{-2}(e^{-u})(-1) \tag{29}$$

$$= \frac{e^{-u}}{(1 + e^{-u})^2} \tag{30}$$

Now a *clever algebra trick*, let's do some rearranging of the following expression

$$1 - \sigma(u) = 1 - \frac{1}{1 + e^{-u}} \tag{31}$$

$$= \frac{1 + e^{-u}}{1 + e^{-u}} - \frac{1}{1 + e^{-u}} \tag{32}$$

$$= \frac{1 + e^{-u} - 1}{1 + e^{-u}} \tag{33}$$

$$= \frac{e^{-u}}{1 + e^{-u}} \tag{34}$$

Using Equation 34 in Equation 30, we have

$$\frac{\partial}{\partial u} \sigma(u) = \frac{e^{-u}}{(1 + e^{-u})^2} \tag{35}$$

$$= \frac{e^{-u}}{1 + e^{-u}} \cdot \frac{1}{1 + e^{-u}} \tag{36}$$

$$= (1 - \sigma(u))\sigma(u) \tag{37}$$

Back to the derivative of our negative log likelihood equation. Let's chunk it off and first look at

$$\frac{\partial}{\partial \theta_1}\Big(y_i \log(\sigma(x_i \cdot \theta))\Big) \tag{38}$$

Recall, the **chain rule** of taking the derivative of a function f with respect to a variable v,

$$\frac{\partial f}{\partial v} = \frac{\partial f}{\partial u} \cdot \frac{\partial u}{\partial v} \tag{39}$$

Now call $u = x_i \cdot \theta$ and let's first look at

$$\frac{\partial}{\partial u}\Big(y_i \log(\sigma(u))\Big) = y_i(\sigma(u))^{-1}(1-\sigma(u))\sigma(u) \tag{40}$$

$$= y_i(1-\sigma(u)) \tag{41}$$

Then

$$\frac{\partial u}{\partial \theta_1}(x_i \cdot \theta) = x_{i,1}\theta_1 + x_{i,2}\theta_2 + \cdots + x_{i,n}\theta_n = x_{i,1} \tag{42}$$

since we are are taking the partial derivative with respect to θ_1 and (by the definition of partial derivatives), every other variable is a constant.

Altogether,

$$\frac{\partial}{\partial \theta_1}\Big(y_i \log(\sigma(x_i \cdot \theta))\Big) = y_i(1-\sigma(x_i \cdot \theta))x_i \tag{43}$$

$$= x_{i,1}y_i(1-\sigma(x_i \cdot \theta)) \tag{44}$$

Let's now expand the second part of the NLL. *(Note, we did not go over this in lecture, but I encourage you to try it on your own and then check your answer.)*

$$\frac{\partial}{\partial \theta_1}\Big((1-y_i)\log(1-\sigma(x_i \cdot \theta))\Big) = \frac{\partial}{\partial \theta}\Big(\log(1-\sigma(x_i \cdot \theta)) - y_i \log(1-\sigma(x_i \cdot \theta))\Big) \tag{45}$$

Let's break these up into terms again, the left-most term we have

$$\frac{\partial}{\partial \theta_1}\Big(\log(1-\sigma(x_i \cdot \theta))\Big) = (1-\sigma(x_i \cdot \theta))^{-1}(-1)(1-\sigma(x_i \cdot \theta))\sigma(x_i \cdot \theta)x_{i_1} \tag{46}$$

$$= -x_{i,1}\sigma(x_i \cdot \theta) \tag{47}$$

for the right-most term we have

$$\frac{\partial}{\partial \theta_1}\Big(-y_i \log(1-\sigma(x_i \cdot \theta))\Big) = y_i\sigma(x_i \cdot \theta)x_{i,1} \tag{48}$$

Let's now combine all these terms (paying attention to when we dropped negative signs) and for simplicity

call $x_{i,1} = x$, $y_i = y$ and $\sigma(x_i \cdot \theta) = \sigma$, then

$$\frac{\partial}{\partial \theta_1} \text{NLL}(\theta, x, y) = -xy(1-\sigma) - \Big(-x\sigma + xy\sigma \Big) \tag{49}$$

$$= -xy + xy\sigma + x\sigma - xy\sigma \tag{50}$$

$$= -xy + x\sigma \tag{51}$$

$$= (\sigma - y)x \tag{52}$$

$$= (\sigma(x_i \cdot \theta) - y_i)x_{i,1} \tag{53}$$

If we then repeated this whole process with $\frac{\partial}{\partial \theta_2}$, we would find

$$\frac{\partial}{\partial \theta_2} \text{NLL}(\theta, x_i, y_i) = (\sigma(x_i \cdot \theta) - y_i)x_{i,2} \tag{54}$$

Notice, the only term that changes is the final $x_{i,2}$ (which is coming from the chain rule and derivative of $\vec{x}\vec{\theta}$).

On all the training documents and for the gradient of the entire vector, $\vec{theta}$, we have

$$\nabla \text{NLL}(\theta_{\text{Train}}, X_{\text{Train}}, y_{\text{Train}}) = [\sum_{i=1}^{n}(\sigma(x_i \cdot \theta) - y_i)x_{i,1}, \sum_{i=1}^{n}(\sigma(x_i \cdot \theta) - y_i)x_{i,2}, \ldots, \sum_{i=1}^{n}(\sigma(x_i \cdot \theta) - y_i)x_{i,k}] \tag{55}$$

The vectorized version of this is

$$\nabla \text{NLL}(\theta_{\text{Train}}, X_{\text{Train}}, y_{\text{Train}}) = (\sigma(X \cdot \theta) - y)^T \cdot X \tag{56}$$

Typically, we normalized by the number of examples, n, that we are taking the gradient with respect to. In mini-batch stochastic gradient descent, this is the number of examples in the batch.

$$\nabla \text{NLL}(\theta_{\text{batch}}, X_{\text{batch}}, y_{\text{batch}}) = \frac{1}{n_{\text{batch}}}(\sigma(X \cdot \theta) - y)^T \cdot X \tag{57}$$