Logistic regression - cs534

This lecture will introduce logistic regression - a discriminative probabilistic classifier. We will begin with the basic modeling assumption used by LR. Recall in LDA, we showed that by assuming Gaussian distributions with shared covariance matrix for each class, we have

$$\log \frac{P(\mathbf{x}, y=1)}{P(\mathbf{x}, y=0)} = \mathbf{w}^T\mathbf{x} + w_0$$

which defines a linear decision boundary. In Logistic regression, we make essentially the same assumption that

$$\log \frac{P(y=1|\mathbf{x})}{P(y=0|\mathbf{x})} = \mathbf{w}^T\mathbf{x} + w_0$$

and learn the weights directly without making the Gaussian assumptions for the class distributions.

Note that this is equivalent to assuming that

$$P(y=1|\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T\mathbf{x}+w_0)}}$$

Using the canonical representation of the data (adding a dummy feature of value 1 to each input vector), we have

$$P(y=1|\mathbf{x}) = g(\mathbf{x}, \mathbf{w}) = \frac{1}{1+e^{-\mathbf{w}^T\mathbf{x}}}$$

It follows that

$$P(y=0|\mathbf{x}) = 1 - g(\mathbf{x}, \mathbf{w}) = \frac{e^{-\mathbf{w}^T\mathbf{x}}}{1+e^{-\mathbf{w}^T\mathbf{x}}}$$

. Given a training set $\{(\mathbf{x}_i, y_i) : i = 1, \cdots, N\}$, $y_i \in \{0, 1\}$, and $\mathbf{x}_i \in R^{d+1}$, where N is the total number of training examples and d is the original feature dimension, the learning goal is to find the optimal weight vector $\mathbf{w}$.

1 Maximum Likelihood Estimation

To learn the parameters, we will again use the maximum likelihood estimation. The log likelihood function is as follows.

$$\begin{aligned}\log p(D|M) &= \sum_{i=1}^{N} \log p(\mathbf{x}_i, y_i) \\ &= \sum_{i=1}^{N} \log p(y_i|\mathbf{x}_i) p(\mathbf{x}_i)\end{aligned}$$

Note that in Logistic regression, we don't care about $p(\mathbf{x}_i)$ and only need to learn the correct $p(y|\mathbf{x})$. So we can drop $p(\mathbf{x}_i)$. Thus we have:

$$L(\mathbf{w}) = \sum_{i=1}^{N} \log p(y_i|\mathbf{x}_i) = \sum_{i=1}^{N} \log g(\mathbf{x}_i, \mathbf{w})^{y_i} (1 - g(\mathbf{x}_i, \mathbf{w}))^{1-y_i}$$

To maximize L with respect to $\mathbf{w}$, lets look at each example:

$$L_i(\mathbf{w}) = \log(\mathbf{x}_i, \mathbf{w})^{y_i} (1 - g(\mathbf{x}_i, \mathbf{w}))^{1-y_i} = y_i \log g(\mathbf{x}_i, \mathbf{w}) + (1 - y_i) \log(1 - g(\mathbf{x}_i, \mathbf{w}))$$

where $g(\mathbf{x}, \mathbf{w}) = \frac{1}{1+e^{-(\mathbf{w}^T\mathbf{x})}}$.

Taking gradient of L_i with respect to $\mathbf{w}$, we have:

$$\begin{aligned}
\nabla_{\mathbf{w}} L_i &= \frac{y_i}{g(\mathbf{x}_i, \mathbf{w}))} \nabla_{\mathbf{w}} g - \frac{1-y_i}{1-g(\mathbf{x}_i, \mathbf{w}))} \nabla_{\mathbf{w}} g \\
&= \frac{y_i}{g} g(1-g)\mathbf{x}_i - \frac{1-y_i}{1-g} g(1-g)\mathbf{x}_i \\
&= (y_i(1-g) - (1-y_i)g)\mathbf{x}_i \\
&= (y_i - y_i g - g + y_i g)\mathbf{x}_i \\
&= (y_i - g(\mathbf{x}_i, \mathbf{w}))\mathbf{x}_i
\end{aligned}$$

Consider all training examples, we have:

$$\nabla_{\mathbf{w}} L = \sum_{i=1}^{N} (y_i - g(\mathbf{x}_i, \mathbf{w}))\mathbf{x}_i$$

Note that we cannot directly solve for the optimal $\mathbf{w}$ by setting the gradient to zero. Instead, we will need to use the gradient ascent method to find the optimal $\mathbf{w}$. The update rule for the batch method is $\mathbf{w} \leftarrow \mathbf{w} + \eta \sum_{i=1}^{N} (y_i - g(\mathbf{x}_i, \mathbf{w}))\mathbf{x}_i$, where η is the learning rate. Similarly the update rule for online learning of LR is $\mathbf{w} \leftarrow \mathbf{w} + (y_i - g(\mathbf{x}_i, \mathbf{w}))\mathbf{x}_i$.

Note that the likelihood function is concave, thus Gradient ascent will find the global optimal solution.

2 Multi-class Logistic regression

For multi-class classification problems, we have a training set $\{(\mathbf{x}_i, y_i) : i = 1, \cdots, N\}$, $y_i \in \{1, 2, \cdots, K\}$. In this case, we assume that

$$p(y = k|\mathbf{x}) = \frac{e^{\mathbf{w}_k^T \mathbf{x}}}{\sum_i e^{\mathbf{w}_i^T \mathbf{x}}}$$

To write down the likelihood function, we will use the 1-of-K coding scheme in which we create a target vector $\mathbf{y}$ for each example, if $y = k$, then the vector $\mathbf{y}$ will have 1 for the k-th element and zero for all other elements. The log likelihood function for the ith example is then:

$$\begin{aligned}
L_i(\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_K) &= \log \prod_{k=1}^{K} p(k|\mathbf{x}_i)^{\mathbf{y}_i^k} \\
&= \sum_{k=1}^{K} \mathbf{y}_i^k \log p(k|\mathbf{x}_i)
\end{aligned}$$

As shown in class, taking the gradient with respect to $\mathbf{w_k}$ we have:

$$\nabla_{\mathbf{w}_k} L_i = (y_i^k - p(k|\mathbf{x}_i))\mathbf{x}_i$$

This gives us the online update rule for weight vector $\mathbf{w}_k$ is

$$\mathbf{w}_k \leftarrow \mathbf{w}_k + (y_i^k - p(k|\mathbf{x}_i))\mathbf{x}_i$$