STAT 383C: Statistical Modeling I **Fall 2016**

Lecture 11 — October 1

Lecturer: Purnamrita Sarkar *Scribe: Jesse Miller*

Disclaimer: These scribe notes have been slightly proofread and may have typos etc.

Note: The latex template was borrowed from EECS, U.C. Berkeley.

11.1 L_∞ norm regularization

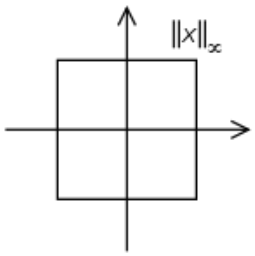

Figure 11.1. L_∞ norm penalization

Lets talk a bit about infinity norm penalization. Above you have the feasible set for $\|\beta\|_\infty \leq t$, where the corners are at (t, t), $(t, -t)$ etc. Note that this is convex, so if you add this to the Least squares objective you will have a convex problem, but note that here the corners are not on the axis. So, the solutions will give you lots of coefficients shrunk to have absolute value t, (if you used the alternative formulation) but not sparse solutions.

11.2 Logistic regression

In Naive Bayes, we saw that we modeled the distribution of the X's given the Y's and then we used Bayes rule to get the distribution of Y given X. This is why Naive Bayes is also called a *Generative* model for classification. In the discrete case, we modeled X as Gaussian random variables (with different parameters for different classes), whereas for the continuous case, we modeled X as Multinomial random variables. This raises the question, what if we went straight for the distribution of Y given X and made no assumptions whatsoever about the distribution of X? That brings us to Logistic Regression, which is also known as a *Discriminative* model for classification. In particular we will model $p := P(Y = 1|X)$ as follows:

$$\log \frac{p}{1-p} = X^T \beta$$

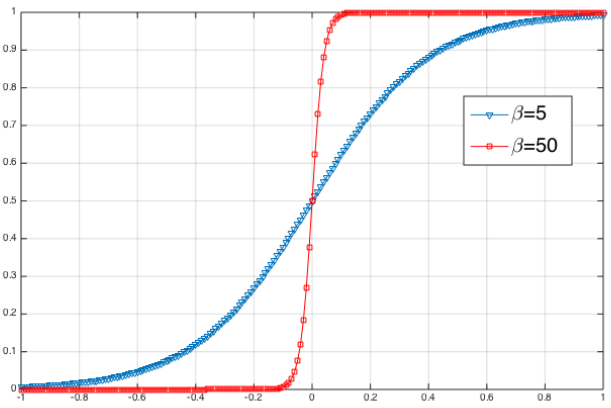

Figure 11.2. $\frac{\exp(\beta x)}{1+\exp(\beta x)}$ as x is growing along the x axis

Whats so special about this form? Well, note that we now have a function (the logit function) which takes a real variable as an argument and spits out a number between zero and one. This is great, because we are trying to model the probability of $Y = 1$. What does this function look like? See figure 11.2. An interesting point that will come in handy later: as one increases β, the function in 11.2 looks increasingly like a step function.

Now, let's think about the MLE for β.

11.2.1 Maximum Likelihood Estimation for Logistic Regression

First, note that

$$P(y_i = 1|x_i, \beta) = \frac{e^{x_i^T \beta}}{1 + e^{x_i^T \beta}}.$$

Now, we write the **conditional** log likelihood:

$$P(\boldsymbol{y}|\boldsymbol{X};\boldsymbol{\beta}) = \prod_i p_i^{y_i}(1-p_i)^{1-y_i} = \prod_i \left(\frac{p_i}{1-p_i}\right)^{y_i} \prod_i (1-p_i)$$
$$\ell := \log P(\boldsymbol{y}|\boldsymbol{X};\boldsymbol{\beta}) = \sum_i y_i \boldsymbol{\beta}^T \boldsymbol{x}_i - \sum_i \log(1 + \exp(\boldsymbol{\beta}^T \boldsymbol{x}_i))$$

We start trying to find the maximum first by differentiating the log likelihood:

$$\frac{\partial \ell}{\partial \beta_r} = \sum_i y_i x_{ir} - \sum_i x_{ir} \frac{\exp(\boldsymbol{\beta}^T \boldsymbol{x}_i)}{1 + \exp(\boldsymbol{\beta}^T \boldsymbol{x}_i)} = \sum_i x_{ir}(y_i - p_i)$$
$$\frac{\partial \ell}{\partial \boldsymbol{\beta}} = \boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{p}),$$

where $\boldsymbol{p}$ is the vector of probabilities $p_i = P(Y_i = 1|X_i; \boldsymbol{\beta})$.

We cannot find a closed form solution by setting this to zero in this case and so we will have to resort to iterative methods like the gradient ascent or Newton Raphson. For gradient ascent, we could keep taking small steps in the direction of the current gradient, but the steps would need to be small and this could take a while. For Newton Raphson, we use second-order information via the Hessian:

$$H_{rs} = \frac{\partial^2 \ell}{\partial \boldsymbol{\beta}_s \partial \boldsymbol{\beta}_r} = \frac{\partial}{\partial \beta_s} \sum_i x_{ir}(y_i - p_i) = -\sum_i x_{ir} \frac{\partial p_i}{\partial \beta_s} = -\sum_i x_{ir} x_{is} p_i(1 - p_i)$$
$$\boldsymbol{H} = -\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X},$$

where $\boldsymbol{W}$ is a diagonal matrix with $W_{ij} = p_i(1-p_i)\mathbf{1}(i = j)$. Recall that the Newton Raphson update is given by:

$$\begin{aligned}
\boldsymbol{\beta}^{t+1} &= \boldsymbol{\beta}^t - \boldsymbol{H}^{-1} \frac{\partial \ell}{\partial \boldsymbol{\beta}}\bigg|_{\boldsymbol{\beta}=\boldsymbol{\beta}^t} = \boldsymbol{\beta}^t + (\boldsymbol{X}^T \boldsymbol{W}^t \boldsymbol{X})^{-1} \boldsymbol{X}^T (\boldsymbol{y} - \boldsymbol{p}^t) \\
&= (\boldsymbol{X}^T \boldsymbol{W}^t \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{W}^t \underbrace{\left(\boldsymbol{X}\boldsymbol{\beta}^t + (\boldsymbol{W}^t)^{-1}(\boldsymbol{y} - \boldsymbol{p}^t)\right)}_{:= \mathbf{z}^t} \\
&= \arg\min_{\boldsymbol{\beta}'} (\mathbf{z}^t - \boldsymbol{X}\boldsymbol{\beta}')^T \boldsymbol{W}^t (\mathbf{z}^t - \boldsymbol{X}\boldsymbol{\beta}')
\end{aligned}$$

Compare this to the gradient ascent with equal steps λ; here we allow our step sizes to change. So at each step we solve a weighted least squares problem. But let's take a detour on weighted least squares.

11.2.2 Weighted Least Squares

Remember our model for OLS? All the y_i's had the same variance. What if each random variable had a different variance:

$$\boldsymbol{y} = \boldsymbol{X}\beta + \epsilon, \ \epsilon \sim N(\mathbf{0}, \Sigma)$$

where Σ is a diagonal matrix where the diagonal terms are different, i.e. $\Sigma_{ii} = \sigma_i^2$. Now the log likelihood is proportional to $-1/2(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^T \Sigma^{-1} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})$. And so the MLE would minimize

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}'} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}')^T \Sigma^{-1} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}') = \arg\min_{\boldsymbol{\beta}'} \sum_i (y_i - \boldsymbol{x}_i^T \boldsymbol{\beta})^2 / \sigma_i^2$$

Setting the first derivative to zero gives:

$$\boldsymbol{X}^T \Sigma^{-1} (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}) = 0 \to \hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T \Sigma^{-1} \boldsymbol{X})^{-1} \boldsymbol{X}^T \Sigma^{-1} \boldsymbol{y}.$$

Note that this is the same as with OLS, in which case $\Sigma = I$. In our case, we are solving a weighted regression at each step. This is why the algorithm is also called **Iteratively Reweighted Least Squares**.

Now,

$$\begin{aligned}\log\left(\frac{y_i}{1-y_i}\right) &\approx \log\left(\frac{p_i}{1-p_i}\right) + (y_i - p_i) \times \frac{d}{dy}\log\left(\frac{y}{1-y}\right)\Bigg|_{y_i=p_i} \\ &= \log\left(\frac{p_i}{1-p_i}\right) + \frac{y_i - p_i}{p_i(1-p_i)} \\ &= x_i^t\beta + \frac{y_i - p_i}{p_i(1-p_i)} := z_i\end{aligned}$$

and therefore

$$\mathrm{Var}(z_i) = \frac{\mathrm{Var}(y_i)}{p_i^2(1-p_i)^2} = \frac{1}{p_i(1-p_i)}.$$

Now suppose that $\boldsymbol{z}^t \sim N(\boldsymbol{X}\beta^{t+1}, (\boldsymbol{W}^t)^{-1})$. Then

$$z^t = \boldsymbol{X}\beta^t + (\boldsymbol{W}^t)^{-1}(\boldsymbol{y} - \boldsymbol{p}^t),$$

just as before, and we again get

$$\begin{aligned}\hat{\beta}^{t+1} &= (\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X})^{-1}(\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{z}^t) \\ &= (\boldsymbol{X}^T\boldsymbol{W}^t\boldsymbol{X})^{-1}(\boldsymbol{X}^T\boldsymbol{W}^t\boldsymbol{z}^t).\end{aligned}$$