# Block coordinate descent algorithm for L1/L2 Regularized Logistic Regression


**Abstract**

The strucutre learning algorithm proposed for discrete graphical models described in Ravikumar et al.[(3)] involves solving an L1/L2 Regularized Logistic Regression problem. In this note, we describe the Block coordinate descent algorithm being used for solving this problem. This algorithm was proposed in Meier et al.[(1)].


## 1 Notation

Let us recall the notation used in Ravikumar et al.[(3)] and the dropbox note[(2)]. Consider a discrete pairwise graphical model describing a probability distribution over $p$ variables, each of which can take one of $m$ discrete values. Let $D = \{1, \ldots, m-1\}$ denote the set of the first $m-1$ values.

$$Pr(x) \quad \propto \quad \exp(\sum_{s \in V} \phi_s(x_s) + \sum_{(s,t) \in E} \phi_{st}(x_s, x_t)). \tag{1}$$

Using indicator variables, any set of potential functions can then be written as

$$\begin{aligned} \phi_s(x_s) \quad &= \quad \sum_{j \in D} \theta^*_{s;j} \; I[x_s = j] \quad \text{for } s \in V \text{, and} \\ \phi_{st}(x_s, x_t) \quad &= \quad \sum_{(j,k) \in D^2} \theta^*_{st;jk} \; I[x_s = j,\, x_t = k] \quad \text{for } (s,t) \in E. \end{aligned}$$

Thus, the Markov random field can be parameterized in terms of the vector $\theta^*_s \in \mathbf{R}^{m-1}$ for each $s \in V$, and the vector $\theta^*_{st} \in \mathbf{R}^{(m-1)^2}$ associated with each edge.

The conditional probability distribution of values taken by node $r$ is given by:

$$\mathbb{P}_\Theta\big[X_r = m \mid X_{\backslash r} = x_{\backslash r}\big] = \frac{1}{1 + \sum_\ell \exp(\theta^*_{r;\ell} + \sum_{t \in V \backslash \{r\}} \sum_k \theta^*_{rt;\ell k} I[x_t = k])} \tag{2}$$

and, for $j \in \{1, ..m-1\}$:

$$\mathbb{P}_\Theta\big[X_r = j \mid X_{\backslash r} = x_{\backslash r}\big] = \frac{\exp(\theta^*_{r;j} + \sum_{t \in V \backslash \{r\}} \sum_k \theta^*_{rt;jk} I[x_t = k])}{1 + \sum_\ell \exp(\theta^*_{r;\ell} + \sum_{t \in V \backslash \{r\}} \sum_k \theta^*_{rt;\ell k} I[x_t = k])}. \tag{3}$$

In the above expression, it is assumed that $\forall t \notin \Gamma(r)$, $\theta_{rt}$ are zero vectors. Let $\Theta$ be the set of all parameters involved in Equation 3. Given a set $S = \left\{x^{(1)}..x^{(n)}\right\}$ of $n$ sample-points, we can deduce the neighborhood $N(r)$ of $r$ by estimating the parameter vectors $\forall t \in V \backslash \{r\} : \theta^*_{rt}$. In particular, we solve the problem:

$$\min_\Theta -n^{-1} \sum_{i=1}^n \log P_\Theta(x_r^{(i)} | x_{\backslash r}^{(i)}) + \lambda \sum_{v \in V \backslash \{r\}} \|\theta_{rv}\|_2 .$$

# 2 Solving the logistic regression problem

The algorithm we use for solving $l1/l2$ regularized logistic regression works best when the design matrix is group-orthogonalized. So, we find it convenient to describe this algorithm in general terms, rather than in terms of parameters $\Theta$ introduced earlier.

## 2.1 Problem setting

We now introduce the $l1/l2$ regularized logistic regression in general terms. Let $Y$ be the response variable, and $X$ be the predictor variables whose relationship is being modelled using a multi-class logistic model. Further, suppose that any predictor vector $x \in R^{p'+1}$ includes the intercept; that is $x_1 = 1$ always. Suppose that the $Y$ takes values in the set $\{1..m\}$, and that each predictor $X_i$ takes values in the set $\{1..m\}$. Then, the logistic model we deal with is described below:

$$\mathbb{P}^*_\beta\big[Y = m \mid X = x\big] = \frac{1}{1 + \sum_{\ell \in \{1..m-1\}} \exp(\beta_\ell^{*T} x)}$$

and, for $j \in \{1, ..m-1\}$:

$$\mathbb{P}^*_\beta\big[Y = j \mid X = x\big] = \frac{\beta_j^{*T} x}{1 + \sum_{\ell \in \{1..m-1\}} \exp(\beta_\ell^{*T} x)}. \tag{4}$$

Let $\{\beta^*_0, \beta^*_1, ..\beta^*_G\}$ be a partition of the parameters $\beta^*$, which need not coincide with the partitioning $\{\beta^*_\ell | \ell \in \{1..m-1\}\}$ used in Equation 4. We work with the prior belief that $\beta^*$ is group-sparse: that is, we assume that most of the vectors in $\{\beta^*_1, ..\beta^*_G\}$ are actually zero vectors. So, given $n$ observations $\left\{(x^{(i)}, y^{(i)})\right\}$, to estimate $\beta^*$, we will solve the problem:

$$\min_\beta n^{-1} \sum_{i=1}^n -\log P_\beta(y^{(i)} | x^{(i)}) + \lambda \sum_{g \in \{1..G\}} \|\beta_g\|_2 .$$

### 2.2 Details of some computations

The algorithm to solve this problem will involve computation of the negative log likelihood function $nll(\beta|(x^{(i)}, y^{(i)})) = -\log P_\beta(y^{(i)}|x^{(i)})$, its gradient, and the diagonal of its Hessian. The negative log likelihood given the observation $(x, y)$ and its gradient are computed by evaluating the following expressions [1]:

$$
\begin{aligned}
nll(\beta|(x, y = m)) &= \log(1 + \sum_{\ell \in \{1..m-1\}} \exp(\beta_\ell^T x)) \\
\frac{\partial nll(\beta|(x, y = m))}{\partial \beta_{i,j}} &= (1 + \sum_{\ell \in \{1..m-1\}} \exp(\beta_\ell^T x))^{-1} \exp(\beta_i^T x) x_j.
\end{aligned}
$$

and for $q \in \{1, ..m' - 1\}$:

$$
\begin{aligned}
nll(\beta|(x, y = q)) &= -\beta_q^T x + \log(1 + \sum_{\ell \in \{1..m-1\}} \exp(\beta_\ell^T x)) \\
\frac{\partial nll(\beta|(x, y = m))}{\partial \beta_{i=q,j}} &= -x_j + (1 + \sum_{\ell \in \{1..m-1\}} \exp(\beta_\ell^T x))^{-1} \exp(\beta_q^T x) \\
\frac{\partial nll(\beta|(x, y = m))}{\partial \beta_{i \neq q,j}} &= (1 + \sum_{\ell \in \{1..m-1\}} \exp(\beta_\ell^T x))^{-1} \exp(\beta_i^T x) x_j.
\end{aligned}
$$

The diagonal of the Hessian is computed by evaluating the following expression:

$$
\begin{aligned}
\frac{\partial^2 nll(\beta|(x, y = m))}{\partial \beta_{i,j}^2} &= (1 + \sum_{\ell \in \{1..m-1\}} \exp(\beta_\ell^T x))^{-1} \exp(\beta_i^T x) x_j^2 \\
&\quad -(1 + \sum_{\ell \in \{1..m-1\}} \exp(\beta_\ell^T x))^{-2} \exp(2\beta_i^T x) x_j^2.
\end{aligned}
$$

Given $n$ observations $S = \left\{(x^{(i)}, y^{(i)})\right\}$, we define

$$
nll(\beta|S) = n^{-1} \sum_{i=1:n} nll(\beta|(x^{(i)}, y^{(i)})).
$$

### 2.3 The block coordinate descent algorithm

The algorithm is specified as Algorithm 1.

## References


[1] Lukas Meier, Sara van de Geer, and Peter Buhlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71, February 2008. ISSN 1369-7412. doi: 10.1111/j.1467-9868.2007.00627.x. URL `http://dx.doi.org/10.1111/j.1467-9868.2007.00627.x`.


[1] Here we return to the partitioning $\left\{\beta_\ell^* | \ell \in \{1..m-1\}\right\}$ used in Equation 4.

**Algorithm 1** Block coordinate descent algorithm

Input: $\beta^{(0)}$, Sample set $S = \left\{(x^{(i)}, y^{(i)})\right\}$ of $n$ points, $\lambda$, $tol$.
Output: $\beta$.

$\beta \leftarrow \beta^{(0)}$
**loop**
  **for all** $g \in \{0, ..G\}$ **do**
    Compute $nll(\beta|S), \nabla nll(\beta|S)_g, diag(\nabla^2 nll(\beta|S)_{gg})$, where the subscripts indicate that we refer to the portions of the gradient and hessian corresponding to the vairables $\beta_g$.
    $h_g \leftarrow -\max\left\{diag(\nabla^2 nll(\beta|S)_{gg}), 10^{-5}\right\}$.
    $d \leftarrow 0$.
    **if** $g = 0$ **then**
      $d_g \leftarrow \nabla nll(\beta|S)/h_g$.
    **else**
      $z \leftarrow -\nabla nll(\beta|S)_g - h_g \beta_g$.
      **if** $\|z\|_2 \leq \lambda$ **then**
        $d_g \leftarrow -\beta$.
      **else**
        $d_g \leftarrow -h_g^{-1}[-\nabla nll(\beta|S)_g - \lambda \frac{z}{\|z\|_2}]$.
      **end if**
    **end if**
    **if** $\max|d| \geq tol$ **then**
      Get $\alpha$ from line search along $d$ with $\alpha_0 = 2, \delta = 0.75, \sigma = .01$.
      $\beta_g \leftarrow \beta_g + \alpha d_g$.
    **end if**
  **end for**
  Return if decrease in objective value is less than $tol$.
**end loop**


[2] Pradeep Ravikumar. Learning Discrete Graphical Models [Dropbox note], 2010.

[3] Pradeep Ravikumar, M. J. Wainwright, and J. Lafferty. High-dimensional ising model selection using l1-regularized logistic regression. *Annals of Statistics*, 2009.