Discussion 4: Logistic Regression
Probabilistic Machine Learning, Fall 2016

1 Formulation and Coordinate Descent

1.1 Formulation

For dataset $\mathbf{x}_i \in \mathbb{R}^D$, $y_i \in \{0, 1\}$, $i = 1, ..., N$, prove the negative log-likelihood of logistic regression

$$\mathcal{L}(\boldsymbol{\theta}) = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i \log \frac{1}{1+e^{-\boldsymbol{\theta}^\top \mathbf{x}_i}} + (1-y_i)\log \frac{e^{-\boldsymbol{\theta}^\top \mathbf{x}_i}}{1+e^{-\boldsymbol{\theta}^\top \mathbf{x}_i}}\right)$$

is convex w.r.t. $\boldsymbol{\theta}$.

1.2 Interpretation

We change notation slightly and assume $y_i \in \{\pm 1\}$. In this case, the logistic regression model is defined by

$$p(y \,|\, \mathbf{x};\, \boldsymbol{\theta}) = \sigma(y\boldsymbol{\theta}^\top \mathbf{x}),$$

where σ is the logistic sigmoid function defined by

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

The *log odds* of $y = 1$ conditioned on $\mathbf{x}$ is defined as

$$\log \frac{p(+1 \,|\, \mathbf{x};\, \boldsymbol{\theta})}{p(-1 \,|\, \mathbf{x};\, \boldsymbol{\theta})}.$$

(a) Prove that the log odds is equal to the simple expression $\boldsymbol{\theta}^\top \mathbf{x}$.

(b) In light of (a), give an interpretation for each $\boldsymbol{\theta}_i$. For example, if we increase the i-th component of $\mathbf{x}$ while holding the others constant, what effect does this have on the log odds.

1.3 ℓ_2-regularization and Coordinate Descent

(a) If we assume $y_i \in \{-1, 1\}$ ($0 \to -1$), the negative log-likelihood will be $\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N} \log\left(1 + e^{-y_i \boldsymbol{\theta}^\top \mathbf{x}_i}\right)$. We also add an ℓ_2 regularization on parameter $\boldsymbol{\theta}$, we have the regularized loss function $\tilde{\mathcal{L}}(\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta}) + \frac{\lambda}{2}\boldsymbol{\theta}^\top \boldsymbol{\theta}$. Is this still a convex function? Also, what is the advantage of a ℓ_2-regularization for Logistic Regression?

(b) Suppose we use coordinate descent to optimize the regularized loss function above. Can you come up with a closed-form update for the j-th coordinate of $\boldsymbol{\theta}$?

(c) Now let's try coordinate descent on the quadratic approximation to the objective function. First derive the quadratic approximation by Taylor expansion, and then try to find a closed-form update for the j-th coordinate of $\boldsymbol{\theta}$. (Hint: optimizing the j-th coordinate should just be a quadratic function, which has a closed-form minimum.)

(d) (Separable Data) Suppose that our data are linearly separable, i.e. there exists some parameter vector $\boldsymbol{\theta}_*$ such that $y_i \boldsymbol{\theta}_*^\top \mathbf{x}_i \geq \delta > 0$ for all i (where $\delta > 0$ is some positive constant). Prove that there exists some sequence $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \ldots$ such that

$$\lim_{i \to \infty} \mathcal{L}(\boldsymbol{\theta}_i) = 0.$$

What does this imply about the existence of a maximum likelihood estimate for $\boldsymbol{\theta}$ in this case? (Hint: consider scalar multiples of $\boldsymbol{\theta}_*$.)

2 Practice and Gradient Descent

2.1 Gradient Descent

In practice, we often use *gradient descent* (or related variants) to learn the parameters in logistic regression: your optimization trajectory moves in the direction of the negative gradient at each step, which is the direction of steepest local descent. Is gradient descent going to perform better than coordinate descent when the same step is chosen?

2.2 Large N Scenario

In the scenario of "big data", *i.e.,* N is very large, running gradient descent can be very slow. What is the main reason? Could you suggest some solutions to speed up? Hint: *stochastic gradient descent.*

3 KKT

3.1 Inequality Constraint

Let $\alpha \in \mathbb{R}$ and $\boldsymbol{a} \in \mathbb{R}^n$ with $\boldsymbol{a} \neq 0$. Define the halfspace $H = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^T \boldsymbol{x} + \alpha \geq 0\}$. Consider the problem of finding the point in H with the smallest Euclidean norm.

(a) Formulate this problem as a constrained optimization problem.

(b) Solve the problem with the help of the KKT conditions. (Hint: you should consider different cases based on if α is negative or nonnegative.)

3.2 Linear Programming

A *linear program* is an optimization problem that has a linear objective function and linear constraints. Myriad important problems can be formulated as a linear program, so it is important to be able to solve such problems efficiently. We can pose any linear program in the following standard form:

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} \boldsymbol{c}^T \boldsymbol{x}, \qquad \text{subject to } \mathbf{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq 0,$$

where $\boldsymbol{c} \in \mathbb{R}^n$, $\boldsymbol{b} \in \mathbb{R}^m$, and $\mathbf{A}$ is an $m \times n$ matrix. Here, $\boldsymbol{x} \geq 0$ means that each component of the vector $\boldsymbol{x}$ must be nonnegative. Prove that the KKT conditions for this problem are

$$\begin{aligned} \mathbf{A}^T \lambda + \boldsymbol{\mu} &= \boldsymbol{c} \\ \mathbf{A}\boldsymbol{x} &= \boldsymbol{b} \\ \boldsymbol{x} &\geq 0 \\ \boldsymbol{\mu} &\geq 0 \\ \mu_i x_i &= 0, \ i = 1, 2, \ldots, n. \end{aligned}$$

An important class of optimization algorithms known as *primal-dual interior point methods* seek to solve the linear program by directly finding solutions to the KKT conditions.