Logistic Regression Optimization

Natural Language Processing: Jordan Boyd-Graber
University of Maryland

Slides adapted from Emily Fox
Reminder: Logistic Regression

\[ P(Y = 0|X) = \frac{1}{1 + \exp[\beta_0 + \sum_i \beta_i X_i]} \]  
\[ P(Y = 1|X) = \frac{\exp[\beta_0 + \sum_i \beta_i X_i]}{1 + \exp[\beta_0 + \sum_i \beta_i X_i]} \]

- Discriminative prediction: \( p(y|x) \)
- Classification uses: ad placement, spam detection
- What we didn’t talk about is how to learn \( \beta \) from data
Logistic Regression: Objective Function

\[ \mathcal{L} \equiv \ln p(Y | X, \beta) = \sum_j \ln p(y^{(j)} | x^{(j)}, \beta) \]

\[ = \sum_j y^{(j)} \left( \beta_0 + \sum_i \beta_i x_i^{(j)} \right) - \ln \left[ 1 + \exp \left( \beta_0 + \sum_i \beta_i x_i^{(j)} \right) \right] \]
Logistic Regression: Objective Function

\[ \mathcal{L} \equiv \ln p(Y \mid X, \beta) = \sum_j \ln p(y^{(j)} \mid x^{(j)}, \beta) \]  

\[ = \sum_j y^{(j)} \left( \beta_0 + \sum_i \beta_i x_i^{(j)} \right) - \ln \left[ 1 + \exp \left( \beta_0 + \sum_i \beta_i x_i^{(j)} \right) \right] \]  

Training data \((y, x)\) are fixed. Objective function is a function of \(\beta\) . . . what values of \(\beta\) give a good value.
Convexity

- Convex function
- Doesn’t matter where you start, if you “walk up” objective
Convexity

- Convex function
- Doesn’t matter where you start, if you “walk up” objective
- Gradient!
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

**Goal**

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $\beta$

Luckily, (vanilla) logistic regression is convex
Gradient for Logistic Regression

To ease notation, let’s define

\[ \pi_i = \frac{\exp \beta^T x_i}{1 + \exp \beta^T x_i} \]  \hspace{1cm} (5)

Our objective function is

\[ \mathcal{L} = \sum_i \log p(y_i | x_i) = \sum_i \mathcal{L}_i = \sum_i \left\{ \begin{array}{ll} \log \pi_i & \text{if } y_i = 1 \\ \log(1 - \pi_i) & \text{if } y_i = 0 \end{array} \right. \]  \hspace{1cm} (6)
**Taking the Derivative**

Apply chain rule:

$$\frac{\partial \mathcal{L}}{\partial \beta_j} = \sum_i \frac{\partial \mathcal{L}_i(\hat{\beta})}{\partial \beta_j} = \sum_i \left\{ \begin{array}{ll} \frac{1}{\pi_i} \frac{\partial \pi_i}{\partial \beta_j} & \text{if } y_i = 1 \\ \frac{1}{1-\pi_i} \left( -\frac{\partial \pi_i}{\partial \beta_j} \right) & \text{if } y_i = 0 \end{array} \right. \tag{7}$$

If we plug in the derivative,

$$\frac{\partial \pi_i}{\partial \beta_j} = \pi_i (1 - \pi_i) x_j, \tag{8}$$

we can merge these two cases

$$\frac{\partial \mathcal{L}_i}{\partial \beta_j} = (y_i - \pi_i) x_j. \tag{9}$$
Gradient for Logistic Regression

Gradient

\[ \nabla_\beta \mathcal{L}(\hat{\beta}) = \begin{bmatrix} \frac{\partial \mathcal{L}(\hat{\beta})}{\partial \beta_0}, \ldots, \frac{\partial \mathcal{L}(\hat{\beta})}{\partial \beta_n} \end{bmatrix} \]  

(10)

Update

\[ \Delta \beta \equiv \eta \nabla_\beta \mathcal{L}(\hat{\beta}) \]  

(11)

\[ \beta_i' \leftarrow \beta_i + \eta \frac{\partial \mathcal{L}(\hat{\beta})}{\partial \beta_i} \]  

(12)
Gradient for Logistic Regression

Gradient

\[ \nabla_\beta \mathcal{L}(\hat{\beta}) = \left[ \frac{\partial \mathcal{L}(\hat{\beta})}{\partial \beta_0}, \ldots, \frac{\partial \mathcal{L}(\hat{\beta})}{\partial \beta_n} \right] \quad (10) \]

Update

\[ \Delta \beta \equiv \eta \nabla_\beta \mathcal{L}(\hat{\beta}) \quad (11) \]

\[ \beta'_i \leftarrow \beta_i + \eta \frac{\partial \mathcal{L}(\hat{\beta})}{\partial \beta_i} \quad (12) \]

Why are we adding? What would well do if we wanted to do descent?
Gradient for Logistic Regression

**Gradient**

\[ \nabla_\beta \mathcal{L}(\vec{\beta}) = \left[ \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_0}, \ldots, \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_n} \right] \]  

(10)

**Update**

\[ \Delta \beta \equiv \eta \nabla_\beta \mathcal{L}(\vec{\beta}) \]  

(11)

\[ \beta'_i \leftarrow \beta_i + \eta \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_i} \]  

(12)

\( \eta \): step size, must be greater than zero
Gradient for Logistic Regression

Gradient

\[ \nabla_{\beta} \mathcal{L}(\vec{\beta}) = \left[ \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_0}, \ldots, \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_n} \right] \] (10)

Update

\[ \Delta \beta \equiv \eta \nabla_{\beta} \mathcal{L}(\vec{\beta}) \] (11)

\[ \beta'_i \leftarrow \beta_i + \eta \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_i} \] (12)

NB: Conjugate gradient is usually better, but harder to implement
Choosing Step Size

Parameter vs Objective

Parameter

Objective
Choosing Step Size

Objective

Parameter
Choosing Step Size

Objective

Parameter
Choosing Step Size

Parameter

Objective

Parameter
Choosing Step Size

Objective vs. Parameter

Graph showing the relationship between Objective and Parameter.
Remaining issues

- When to stop?
- What if $\beta$ keeps getting bigger?
### Regularized Conditional Log Likelihood

**Unregularized**

\[
\beta^* = \arg \max_{\beta} \ln \left[ p(y^{(j)} | x^{(j)}, \beta) \right]
\]  

(13)

**Regularized**

\[
\beta^* = \arg \max_{\beta} \ln \left[ p(y^{(j)} | x^{(j)}, \beta) \right] - \mu \sum_i \beta_i^2
\]  

(14)
Regularized Conditional Log Likelihood

Unregularized

$$\beta^* = \arg \max_{\beta} \ln p(y^{(j)} | x^{(j)}, \beta)$$

(13)

Regularized

$$\beta^* = \arg \max_{\beta} \ln p(y^{(j)} | x^{(j)}, \beta) - \mu \sum_i \beta_i^2$$

(14)

$\mu$ is “regularization” parameter that trades off between likelihood and having small parameters
Approximating the Gradient

- Our datasets are big (to fit into memory)
- ...or data are changing / streaming
Approximating the Gradient

- Our datasets are big (to fit into memory)
- ...or data are changing / streaming
- Hard to compute true gradient

\[ \mathcal{L}(\beta) \equiv \mathbb{E}_x [\nabla \mathcal{L}(\beta, x)] \]  \hspace{1cm} (15)

- Average over all observations
Approximating the Gradient

- Our datasets are big (to fit into memory)
- ... or data are changing / streaming
- Hard to compute true gradient

$$\mathcal{L}(\beta) \equiv \mathbb{E}_x [\nabla \mathcal{L}(\beta, x)]$$ (15)

- Average over all observations
- What if we compute an update just from one observation?
Getting to Union Station

Pretend it’s a pre-smartphone world and you want to get to Union Station
Stochastic Gradient for Logistic Regression

Given a \textbf{single observation} \( x_i \) chosen at random from the dataset,

\[
\beta_j \leftarrow \beta_j' + \eta \left( -\mu \beta_j' + x_{ij} [y_i - \pi_i] \right)
\]  

(16)
Stochastic Gradient for Logistic Regression

Given a **single observation** $x_i$ chosen at random from the dataset,

$$\beta_j \leftarrow \beta_j' + \eta \left( -\mu \beta_j' + x_{ij} \left[ y_i - \pi_i \right] \right)$$  \hspace{1cm} (16)

Examples in class.
Stochastic Gradient for Regularized Regression

\[ \mathcal{L} = \log p(y | x; \beta) - \mu \sum_j \beta_j^2 \]  (17)
Stochastic Gradient for Regularized Regression

\[ \mathcal{L} = \log p(y \mid x; \beta) - \mu \sum_j \beta_j^2 \]  

(17)

Taking the derivative (with respect to example \( x_i \))

\[ \frac{\partial \mathcal{L}}{\partial \beta_j} = (y_i - \pi_i) x_j - 2\mu \beta_j \]  

(18)
Algorithm

1. Initialize a vector $B$ to be all zeros
2. For $t = 1, \ldots, T$
   - For each example $\vec{x}_i, y_i$ and feature $j$:
     - Compute $\pi_i \equiv \Pr(y_i = 1 | \vec{x}_i)$
     - Set $\beta[j] = \beta[j]' + \lambda(y_i - \pi_i)x_i$
3. Output the parameters $\beta_1, \ldots, \beta_d$. 
Proofs about Stochastic Gradient

- Depends on convexity of objective and how close $\epsilon$ you want to get to actual answer
- Best bounds depend on changing $\eta$ over time and **per dimension** (not all features created equal)
In class

- Your questions!
- Working through simple example
- Prepared for logistic regression homework