Classification: Logistic Regression from Data

Machine Learning: Jordan Boyd-Graber
University of Colorado Boulder
LECTURE 3
Reminder: Logistic Regression

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

\[ P(Y = 1|X) = \frac{\exp \left[ \beta_0 + \sum_i \beta_i X_i \right]}{1 + \exp \left[ \beta_0 + \sum_i \beta_i X_i \right]} \]  \hspace{1cm} (2)

- 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

\[ \mathcal{L} \equiv \ln p(Y \mid X, \beta) = \sum_j \ln p(y^{(j)} \mid x^{(j)}, \beta) \]  
\[ = \sum_j y^{(j)} \left( \beta_0 + \sum_i \beta_i x_{i}^{(j)} \right) - \ln \left[ 1 + \exp \left( \beta_0 + \sum_i \beta_i x_{i}^{(j)} \right) \right] \]
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 $W$ and $b$
Gradient Descent (non-convex)

**Goal**
Optimize log likelihood with respect to variables $W$ and $b$
Gradient Descent (non-convex)

**Goal**
Optimize log likelihood with respect to variables $W$ and $b$
Gradient Descent (non-convex)

Goal
Optimize log likelihood with respect to variables $W$ and $b$
Gradient Descent (non-convex)

Goal

Optimize log likelihood with respect to variables $W$ and $b$
Gradient Descent (non-convex)

Goal
Optimize log likelihood with respect to variables $W$ and $b$
Gradient Descent (non-convex)

Goal
Optimize log likelihood with respect to variables $W$ and $b$
Gradient Descent (non-convex)

Goal
Optimize log likelihood with respect to variables $W$ and $b$
Gradient Descent (non-convex)

Goal
Optimize log likelihood with respect to variables $W$ and $b$
Gradient Descent (non-convex)

**Goal**

Optimize log likelihood with respect to variables $W$ and $b$
Gradient for Logistic Regression

**Gradient**

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

(5)

**Update**

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

(6)

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

(7)
Gradient for Logistic Regression

Gradient

$$\nabla_\beta \mathcal{L} (\vec{\beta}) = \left[ \frac{\partial \mathcal{L} (\vec{\beta})}{\partial \beta_0}, \ldots, \frac{\partial \mathcal{L} (\vec{\beta})}{\partial \beta_n} \right]$$  \hspace{1cm} (5)

Update

$$\Delta \beta \equiv \eta \nabla_\beta \mathcal{L} (\vec{\beta})$$  \hspace{1cm} (6)

$$\beta_i \leftarrow \beta_i' + \eta \frac{\partial \mathcal{L} (\vec{\beta})}{\partial \beta_i}$$  \hspace{1cm} (7)

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

Gradient

\[ \nabla_{\beta} \mathcal{L}(\vec{\beta}) = \begin{bmatrix} \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_0} , \ldots , \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_n} \end{bmatrix} \]  \hfill (5)

Update

\[ \Delta \beta \equiv \eta \nabla_{\beta} \mathcal{L}(\vec{\beta}) \]  \hfill (6)

\[ \beta_i \leftarrow \beta_i' + \eta \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_i} \]  \hfill (7)

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

Gradient

\[ \nabla_\beta \mathcal{L}(\vec{\beta}) = \left[ \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_0}, \ldots, \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_n} \right] \tag{5} \]

Update

\[ \Delta \beta \equiv \eta \nabla_\beta \mathcal{L}(\vec{\beta}) \tag{6} \]

\[ \beta_i \leftarrow \beta'_i + \eta \frac{\partial \mathcal{L}(\vec{\beta})}{\partial \beta_i} \tag{7} \]

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

Parameter

Objective

Parameter
Choosing Step Size

Parameter vs. Objective

Objective

Parameter
Choosing Step Size

Objective

Parameter
Choosing Step Size
Choosing Step Size

Parameter

Objective

Parameter

Objective
Remaining issues

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

Unregularized

\[ \beta^* = \arg \max_{\beta} \ln p(y^{(i)} | x^{(i)}, \beta) \]  \hspace{1cm} (8)

Regularized

\[ \beta^* = \arg \max_{\beta} \ln p(y^{(i)} | x^{(i)}, \beta) - \mu \sum_i \beta_i^2 \]  \hspace{1cm} (9)
## Regularized Conditional Log Likelihood

<table>
<thead>
<tr>
<th>Unregularized</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ \beta^* = \arg \max_{\beta} \ln \left[ p(y^{(j)}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Regularized</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ \beta^* = \arg \max_{\beta} \ln \left[ p(y^{(j)}</td>
</tr>
</tbody>
</table>

\(\mu\) 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

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

(10)

- 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

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

(10)

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

$$
\beta_i \leftarrow \beta_i' + \eta \left( -\mu \beta_i' + x_i \left[ y - p(y = 1 | \bar{x}, \bar{\beta}') \right] \right)
$$

(11)
Stochastic Gradient for Logistic Regression

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

$$
\beta_i \leftarrow \beta_i' + \eta \left( -\mu \beta_i' + x_i \left[ y - p(y = 1 | \bar{x}, \bar{\beta}') \right] \right)
$$

(11)

Examples in class.
Stochastic Gradient for Regularized Regression

\[ \mathcal{L} = \log p(y|x; \beta) - \mu \sum_j \beta_j^2 \] (12)
Stochastic Gradient for Regularized Regression

\[ \mathcal{L} = \log p(y \mid x; \beta) - \mu \sum_j \beta_j^2 \]  \hspace{1cm} (12)

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

\[ \frac{\partial \mathcal{L}}{\partial \beta_j} = (y_i - p(y_i = 1 \mid \tilde{x}_i; \beta))x_j - 2\mu \beta_j \] \hspace{1cm} (13)
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