



Logistic Regression

Data Science: Jordan Boyd-Graber
University of Maryland

SLIDES ADAPTED FROM WILLIAM COHEN

Gradient for Logistic Regression

To ease notation, let's define

$$\pi_i = \frac{\exp \beta^T x_i}{1 + \exp \beta^T x_i} \quad (1)$$

Our objective function is

$$\ell = \sum_i \log p(y_i | x_i) = \sum_i \ell_i = \sum_i \begin{cases} \log \pi_i & \text{if } y_i = 1 \\ \log(1 - \pi_i) & \text{if } y_i = 0 \end{cases} \quad (2)$$

Taking the Derivative

Apply chain rule:

$$\frac{\partial \ell}{\partial \beta_j} = \sum_i \frac{\partial \ell_i(\vec{\beta})}{\partial \beta_j} = \sum_i \begin{cases} \frac{1}{\pi_i} \frac{\partial \pi_i}{\partial \beta_j} & \text{if } y_i = 1 \\ \frac{1}{1-\pi_i} \left(-\frac{\partial \pi_i}{\partial \beta_j}\right) & \text{if } y_i = 0 \end{cases} \quad (3)$$

If we plug in the derivative,

$$\frac{\partial \pi_i}{\partial \beta_j} = \pi_i(1 - \pi_i)x_j, \quad (4)$$

we can merge these two cases

$$\frac{\partial \ell_i}{\partial \beta_j} = (y_i - \pi_i)x_j. \quad (5)$$

Gradient for Logistic Regression

Gradient

$$\nabla_{\beta} \ell(\vec{\beta}) = \left[\frac{\partial \ell(\vec{\beta})}{\partial \beta_0}, \dots, \frac{\partial \ell(\vec{\beta})}{\partial \beta_n} \right] \quad (6)$$

Update

$$\Delta \beta \equiv \eta \nabla_{\beta} \ell(\vec{\beta}) \quad (7)$$

$$\beta'_i \leftarrow \beta_i + \eta \frac{\partial \ell(\vec{\beta})}{\partial \beta_i} \quad (8)$$

Gradient for Logistic Regression

Gradient

$$\nabla_{\beta} \ell(\vec{\beta}) = \left[\frac{\partial \ell(\vec{\beta})}{\partial \beta_0}, \dots, \frac{\partial \ell(\vec{\beta})}{\partial \beta_n} \right] \quad (6)$$

Update

$$\Delta \beta \equiv \eta \nabla_{\beta} \ell(\vec{\beta}) \quad (7)$$

$$\beta'_i \leftarrow \beta_i + \eta \frac{\partial \ell(\vec{\beta})}{\partial \beta_i} \quad (8)$$

Why are we adding? What would we do if we wanted to do **descent**?

Gradient for Logistic Regression

Gradient

$$\nabla_{\beta} \ell(\vec{\beta}) = \left[\frac{\partial \ell(\vec{\beta})}{\partial \beta_0}, \dots, \frac{\partial \ell(\vec{\beta})}{\partial \beta_n} \right] \quad (6)$$

Update

$$\Delta \beta \equiv \eta \nabla_{\beta} \ell(\vec{\beta}) \quad (7)$$

$$\beta'_i \leftarrow \beta_i + \eta \frac{\partial \ell(\vec{\beta})}{\partial \beta_i} \quad (8)$$

η : step size, must be greater than zero

Gradient for Logistic Regression

Gradient

$$\nabla_{\beta} \ell(\vec{\beta}) = \left[\frac{\partial \ell(\vec{\beta})}{\partial \beta_0}, \dots, \frac{\partial \ell(\vec{\beta})}{\partial \beta_n} \right] \quad (6)$$

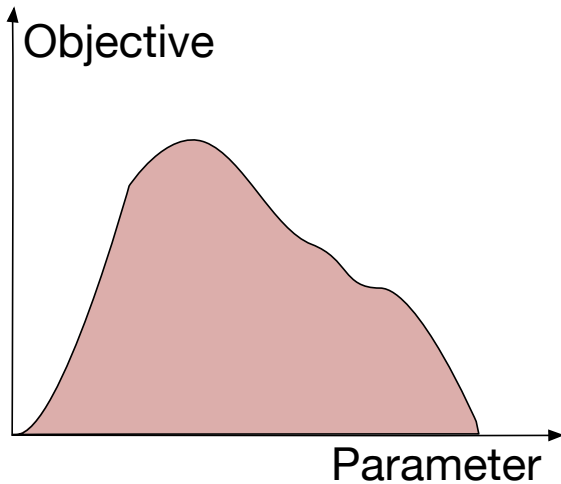
Update

$$\Delta \beta \equiv \eta \nabla_{\beta} \ell(\vec{\beta}) \quad (7)$$

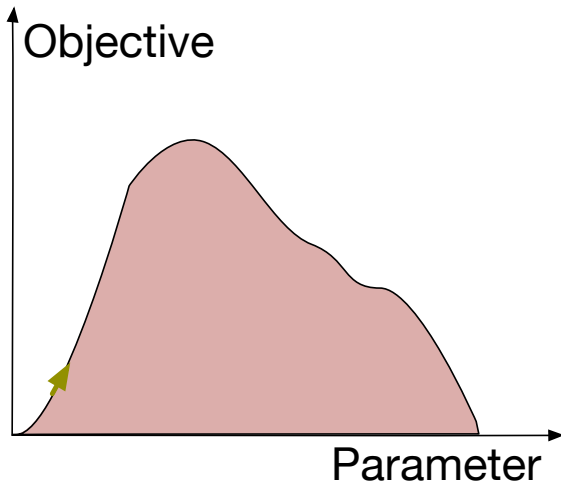
$$\beta'_i \leftarrow \beta_i + \eta \frac{\partial \ell(\vec{\beta})}{\partial \beta_i} \quad (8)$$

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

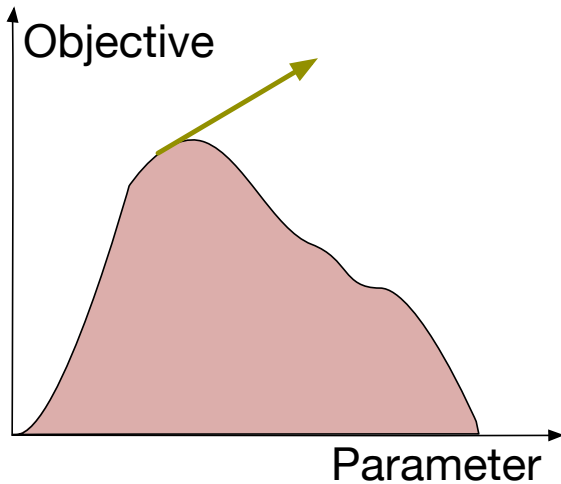
Choosing Step Size



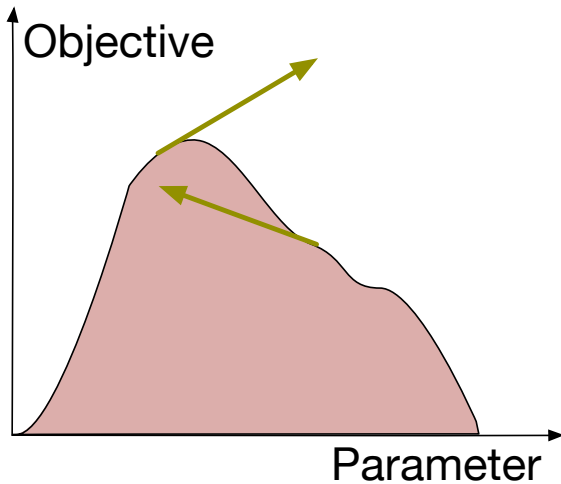
Choosing Step Size



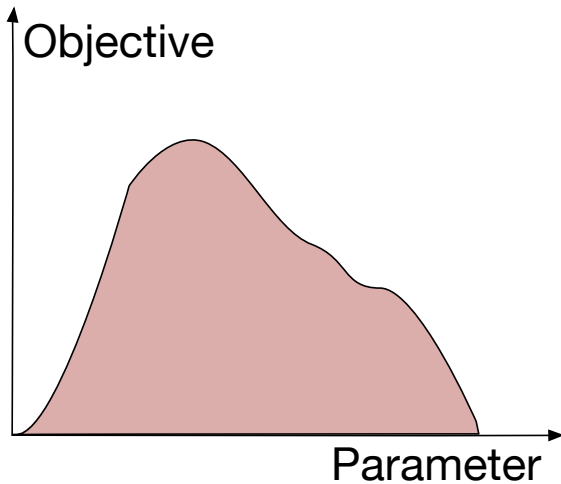
Choosing Step Size



Choosing Step Size



Choosing Step Size



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

$$\ell(\beta) \equiv \mathbb{E}_x [\nabla \ell(\beta, x)] \quad (9)$$

- 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

$$\ell(\beta) \equiv \mathbb{E}_x [\nabla \ell(\beta, x)] \quad (9)$$

- 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 **single observation** x_i chosen at random from the dataset,

$$\beta_j \leftarrow \beta'_j + \eta [y_i - \pi_i] x_{i,j} \quad (10)$$

Stochastic Gradient for Logistic Regression

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

$$\beta_j \leftarrow \beta_j' + \eta [y_i - \pi_i] x_{i,j} \quad (10)$$

Examples in class.

Algorithm

1. Initialize a vector B to be all zeros
2. For $t = 1, \dots, T$
 - 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 β_1, \dots, β_d .

Wrapup

- Logistic Regression: Regression for outputting Probabilities
- Intuitions similar to linear regression
- We'll talk about feature engineering for both next time