Lecture 8 - scribbles

Friday, September 28, 2018 13:34

today: - logistic regression
- numerical optimization
IRLS

Logistic regression model:

$Y = \{0, 1\}$

$p(y=1|x) = \sigma(w^T x)$

$p(y=0|x) = 1 - \sigma(w^T x) = \sigma(-w^T x)$

[if $Y = \{\pm 1\}$

encode $p(y|x) = \sigma(y w^T x)$

i.e. $Y|X=x$ is Bernoulli$(\sigma(w^T x))$

$p(y|x) = \sigma(w^T x)^y \sigma(-w^T x)^{1-y}$

given $(x_i, y_i)_{i=1}^n$, maximum conditional likelihood:

$$\ell(w) = \sum_{i=1}^n \log p(y_i|x_i; w) = \sum_{i=1}^n \left[y_i \log \sigma(w^T x_i) + (1-y_i) \log \sigma(-w^T x_i) \right]$$

derivative

$\sigma'(z) = \sigma(z)\sigma(-z)$

$\nabla_w \sigma(w^T x_i) = x_i \left[\sigma(w^T x_i) \sigma(-w^T x_i) \right]$ let $v_i \triangleq w^T x_i$

$$\nabla \ell(w) = \sum_{i=1}^n x_i \left[y_i \frac{\sigma(v_i)\sigma(-v_i)}{\sigma(v_i)} - (1-y_i) \frac{\sigma(v_i)\sigma(-v_i)}{\sigma(-v_i)} \right]$$

$$\left[y_i \left[\sigma(-v_i) + \sigma(v_i) \right] - \sigma(v_i) \right]$$

1

$$\nabla \ell(w) = \sum_{i=1}^{n} x_i \left[y_i - \sigma(w^T x_i) \right]$$

solve for $\nabla \ell(w) = 0$ $\Rightarrow$ need to solve a transcendental eq.

$\Rightarrow$ need to use numerical methods

because $\frac{1}{1+\exp(-w^T x_i)} (\dots) = 0$

need to solve for this

contrast to least square regression:

$$\nabla \ell(w) = \sum_{i=1}^{n} x_i \left[y_i - w^T x_i \right]$$

linear in w

numerical optimization

want to minimize $f(w)$

$$\min_{s.t.\ w \in \mathbb{R}^d} f(w)$$

1) **gradient descent** (1st order method)

start at w_0

$$w_{t+1} = w_t - \gamma_t \nabla f(w_t)$$

γ_t: step-size

step-size rules:

a) constant step-size $\gamma_t = \frac{1}{L}$ ← Lipschitz constant of ∇f

$$\|\nabla f(w) - \nabla f(w')\| \leq L \|w - w'\|$$

b) decreasing step-size rule: $\gamma_t = \frac{C}{t}$ ← constant

usually want: $\sum_t \gamma_t = +\infty$ $\sum_t \gamma_t^2 < \infty$

c) choose γ_t by "line search": $\min_{\gamma \in \mathbb{R}} f(w_t + \gamma d_t)$ (direction for update, e.g. $-\nabla f(w_t)$)

↓ costly in general

instead do approximate search

e.g. Armijo line search (see Boyd's book)

2) Newton's method (2nd order method)

motivation: minimizing a quadratic approximation:

Hessian $[H(w_t)]_{ij} = \frac{\partial^2 f(w)}{\partial w_i \partial w_j}$

Taylor expansion: $f(w) = f(w_t) + \nabla f(w_t)^T (w - w_t) + \frac{1}{2} (w - w_t)^T H(w_t) (w - w_t) + O(\|w - w_t\|^3)$

(Taylor's remainder)

$$\simeq Q_t(w) + O(\|w - w_t\|^3)$$

↳ quadratic model approximation

$w_{t+1} \rightsquigarrow$ minimizing $Q_t(w)$

$\nabla_w Q_t(w) = 0 \qquad \nabla f(w_t) + H(w_t)(w - w_t) = 0$

$$\Rightarrow \quad w - w_t = -H^{-1}(w_t)\nabla f(w_t)$$

$$\boxed{w_{t+1} = w_t - H^{-1}(w_t)\nabla f(w_t)}$$ Newton's update

inverse Hessian $\rightsquigarrow O(d^3)$ time to compute in general $O(d^2)$ space

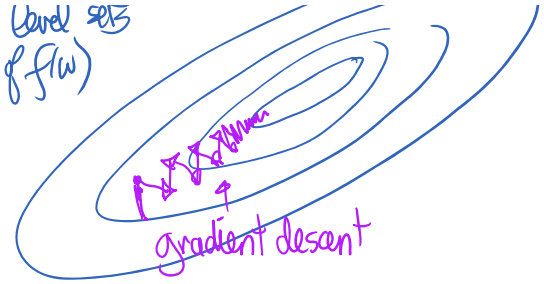
damped Newton: you add a stepsize to stabilize Newton's method

$$w_{t+1} = w_t - \underbrace{\gamma}_{\text{step-size}} H^{-1}(w_t)\nabla f(w_t)$$

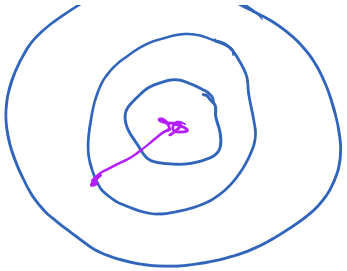
why Newton's method?

- much faster convergence in # of iterations vs. gradient descent
- affine invariant $\rightarrow$ invariant to rescaling of variables

but iterations are costly $O(d^3)$ time $O(d^2)$ memory vs. $O(d)$ for G.D.



using Hessian to make it "well-conditioned"



$z = H^{-1}x$

level sets of $f(z)$

$\nabla f(z) = H^{-1}\nabla f(x)$

$\frac{1}{2}x^T H x = c$

$\frac{1}{2}x^T P^T \Sigma P x = c$

look at "quadratic forms"

(side note on implicit regularization properties of optimization method)

Newton's method for logistic regression : IRLS

recall for log. reg., $\nabla \ell(w) = \sum_{i=1}^{n} x_i [y_i - \sigma(w^T x_i)]$

$$H(\ell(w)) = -\sum_{i=1}^{n} x_i x_i^T \sigma(w^T x_i)\sigma(-w^T x_i)$$

$$v^T H v = -\sum_{i=1}^{n} \underbrace{(v^T x_i)(x_i^T v)}_{(v^T x_i)^2 \geq 0} \underbrace{\sigma(-)\sigma(-)}_{\geq 0}$$

i.e. $v^T H v \leq 0 \quad \forall v \in \mathbb{R}^d$

$\Rightarrow H \leq 0$

i.e. concave function

notation: recall $X = \begin{pmatrix} \vdots \\ -x_i^T- \\ \vdots \end{pmatrix}_{n \times d}$

let $\mu_i \triangleq \sigma(w^T x_i) \in]0,1[$

Let $\mu_i \triangleq \sigma(w^T x_i) \in]0,1[$

$$\nabla l(w) = \sum_{i=1}^{n} x_i [y_i - \mu_i] = X^T (y - \mu)$$

Hessian : $= -\sum_{i=1}^{n} x_i x_i^T \mu_i (1-\mu_i) = -X^T D(w) X$ where $D_{ii} = \mu_i (1-\mu_i)$

[note: D depends]

Newton's update (here Newton's updates are maximizing log-likeli. because it is concave)

Newton's update:

$$w_{t+1} = w_t - (-X^T D_t X)^{-1} X^T (y - \mu_t)$$

$$= (X^T D_t X)^{-1} \left[(X^T D_t X) w_t + X^T (y - \mu_t) \right]$$

$$w_{t+1} = (X^T D_t X)^{-1} \left[X^T D_t z_t \right] \quad \text{where} \quad z_t \triangleq X w_t + D_t^{-1} (y - \mu_t)$$

this is a solution to "weighted least square problem"

$$\min_w \| D^{1/2} (z - Xw) \|^2$$

z: new target

$$\sum_{i=1} \frac{(z_i - x_i^T w)^2}{d_{ii}^{-1}}$$

compare with Gaussian noise model for L.S. $\sum_i \frac{(y_i - x_i^T w)^2}{2\sigma^2}$

Gaussian noise model for L.S.

data dependent noise

Newton's method for logistic regression

= "iterative reweighted least square" (IRLS)

2 comments for assignment:

1) stopping criterion: $\|\nabla f(w)\| \leq \varepsilon$

2) to compute $A^{-1}v$ $\quad A \backslash v$

Big data logistic regression: suppose $f(w) = \frac{1}{n}\sum_{i=1}^{n} f_i(w)$

- cannot do $O(d^2)$ or $O(d^3)$ operations $\Rightarrow$ first order methods
- if n is huge, you cannot do batch method $\nabla f(w) = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(w)$ $O(nd)$ time

("gradient of one fct." / "batch" gradient)

instead you "incremental gradient methods"

e.g. **Stochastic gradient descent (SGD):** $w_{t+1} = w_t - \gamma_t \nabla f_{i_t}(w_t)$ $O(d)$ time

where i_t is picked randomly

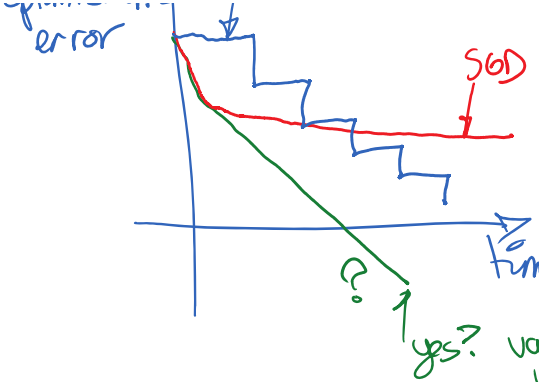
SGD $\rightarrow$ cheap updates, but slower convergence per iteration

batch gradient $\rightarrow$ expensive, but fast " " "

log optimization error

batch gradient

SGD



SAG : stochastic averaged gradient [2012]

G.D. : $w_{t+1} = w_t - \gamma_t \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w_t)$

SAG : $w_{t+1} = w_t - \gamma_t \frac{1}{n} \sum_{i=1}^{n} v_i$ & memory

where $v_i = \nabla f_i(w_{old(i)})$

at each t, update only one $v_{i_t} \triangleq \nabla f_{i_t}(w_t)$

SAGA : [2014] $w_{t+1} = w_t - \gamma \left(\nabla f_{i_t}(w_t) + \frac{1}{n} \sum_{j=1}^{n} v_j - v_{i_t} \right)$

variance reducing correction

(default method for logistic regression in Scikit-learn)