ML ERASMUS, NOV 14, 2022

optimization part

$C(\beta)$ for classification

$$\frac{\partial C}{\partial \beta} = -X^T(y-p) = 0$$

$$\frac{\partial C}{\partial \beta_0} = -\sum_{i=0}^{n-1}\left(y_i - p(x_i)\right) = g_0 = 0$$

$$\frac{\partial C}{\partial \beta_1} = -\sum_{i=0}^{n-1} x_i\left(y_i - p(x_i)\right) = g_1 = 0$$

$p, y \in \mathbb{R}^n$

$X^T \in \mathbb{R}^{p \times n}$

$$\frac{\partial^2 C}{\partial \beta \partial \beta^T} = X^T W X = H$$

= hessian matrix

$$W_{ii} = p(x_i)(1-p(x_i))$$

How to find optimal $\hat{\beta}$?

Make Taylor expansion of $c(\beta)$ around $\hat{\beta} - \beta^{(n)}$

$$\beta = \hat{\beta} \quad (\beta^{(n+1)})$$

$$\|\beta^{(n+1)} - \beta^{(n)}\|_2 \leq \delta \sim 10^{-8}$$

(convergence criterion)

$$c(\beta) = c(\hat{\beta}) = c(\beta^{(n+1)})$$

$$= c(\beta^{(n)}) + g^{T(n)}(\beta - \beta^{(n)})$$

$$+ \frac{1}{2}(\beta - \beta^{(n)})^T H (\beta - \beta^{(n)}) + \dots$$

$$b^{(n)} = \beta - \beta^{(n)}$$

$$c(\beta) \simeq \underbrace{c(\beta^{(n)})}_{c_0} + g^{T(n)} b^{(n)}$$

$$+ \frac{1}{2} b^{T(m)} H^{(m)} b^{(m)}$$

This is of the form

$$c(\beta) \Rightarrow f(x) = c + g^T x + \frac{1}{2} x^T A x$$

x is unknown

$$\frac{\partial f}{\partial x} = 0 = Ax + g = 0$$

$$\Rightarrow \quad Ax = -g \qquad \left(g^{(m)} \triangleq g(\beta^{(m)}) \right)$$

$$H^{(m)} \triangleq H(\beta^{(m)})$$

$$\left(x = \beta - \beta^{(m)} \right)$$

$$x = A^{-1} g$$

$$x^{(m+1)} = x^{(m)} - A^{-1}(x^{(m)})\, g^{(m)}$$

$$\beta^{(m+1)} = \beta^{(m)} - H^{-1}(\beta^{(m)})\, g(\beta^{(m)})$$

- need $g^{(m)}$
- need $H^{-1}(\beta^{(m)})$

replace $H^{-1}(\beta^{(m)})$ with a parameter called the learning rate $\gamma^{(m)}$

Two main issues

1. evaluation of gradients
   - Gradient descent (GD)
   - Stochastic gradient descent (SGD)
   - GD/SGD with momentum

2. optimization $\gamma^{(m)}$
   - different scheduler (no gradient info)
   - adaptive learning rates

- Adagrad
- RMS prop
- ADAM

$$\gamma^{(m)} \simeq H^{-1}(\beta^{(m)})$$

For logistic regression with β_0 and β_1

$$H(\beta) = \begin{bmatrix} \frac{\partial g_0}{\partial \beta_0} & \frac{\partial g_0}{\partial \beta_1} \\ \frac{\partial g_1}{\partial \beta_0} & \frac{\partial g_1}{\partial \beta_1} \end{bmatrix}$$

Taylor expand

$$\beta^{(m+1)} = \beta^{(m)} - \gamma g^{(m)} \quad (GD)$$

$$C(\beta^{(m)} - \gamma g^{(m)}) \simeq$$

$$C(\beta^{(m)}) - \gamma g^{T(m)} g^{(m)}$$

$$+ \frac{1}{2}\gamma^2 g^{T(m)} H g^{(m)}$$

$C(\beta^{(n)})$ is not a function of γ

$$\frac{\partial C}{\partial \gamma} = 0 \Rightarrow$$

$$\gamma = \frac{g^{T(n)} g^{(n)}}{g^{T(n)} H^{(n)} g^{(n)}}$$

$$Hg = \lambda g \quad \wedge \quad g^T g = 1$$

$$\gamma = \frac{1}{\lambda}$$

λ is an eigenvalue of H

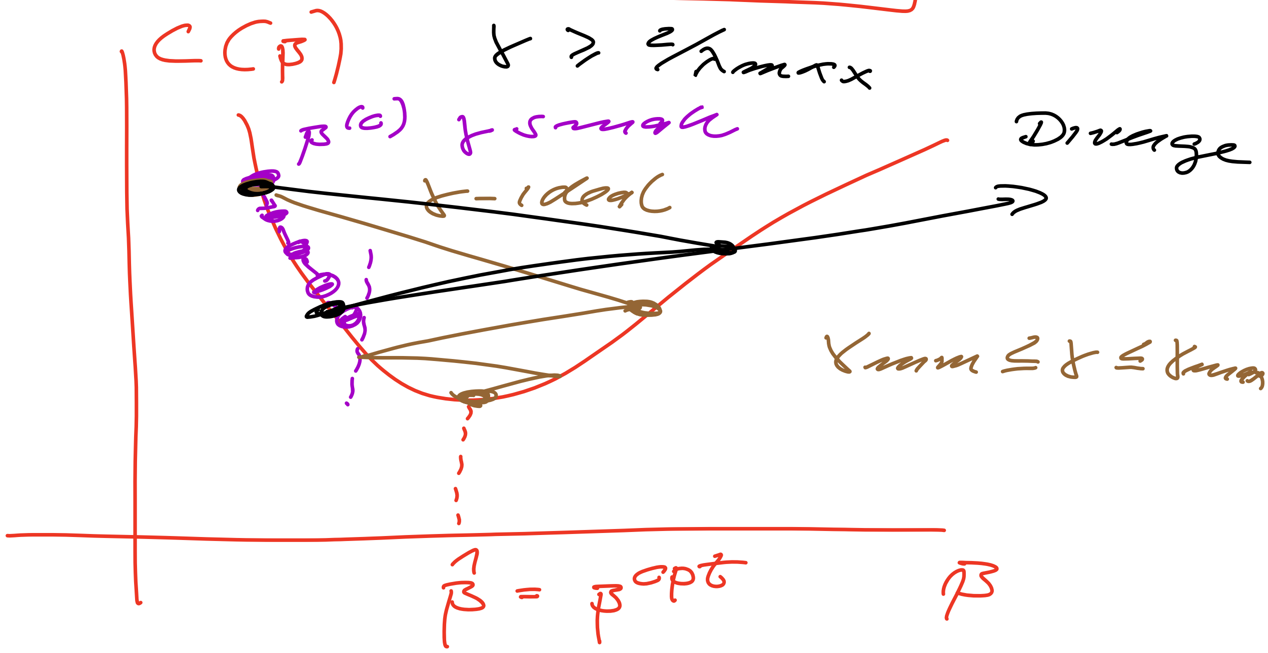
smallest learning rate

$$\gamma_{min} = \frac{1}{\lambda_{max}}$$

$$\gamma_{max} = \frac{1}{\lambda_{min}}$$

in general

$$\boxed{\gamma < \frac{2}{\lambda_{max}}}$$



Simple GD

$$\beta^{(n+1)} = \beta^{(n)} - \gamma \underbrace{g^{(n)}(\beta^{(n)})}_{\text{Full Gradient}}$$

OLS $\quad g^{(n)} = X^T(y - \beta^{(n)})$