SOS 385 : Exercise 03

G.D. & Back tracking line search.

To find an inexact approximation for the stepsize α, we use the wolf condition.

wolf condition is a combination of 2 different condition.

(1) Armijo Condition : (Sufficient decrease condition)

$$F(x_k + \alpha p_k) \leq F(x_k) + c_1 \alpha \nabla f_k^t p_k$$

$$= L(\beta_k) + \alpha \nabla(\beta_k) \leq c_1 \alpha \|\nabla \beta_k\|^2$$

$[x_k \Rightarrow \beta_k]$

L = loss function, β_k = coefficient vector for k^{th} step.

c_1 = chosen $\forall$ $c_1 \in (0,1)$ [usually close to 10^{-4}]

$\nabla(\beta_k)$ = diff. of β_k. = direction.

(2) Curvature condition : It ensures that stepsizes are not too short.

$$\nabla F(x_k + \alpha_k p_k)' p_k \geq c_2 \nabla f_k' p_k \qquad [x_k = \beta_k \ \& \ p_k = \nabla(\beta_k)]$$

$$\rightarrow \nabla(\beta_k + \alpha_k \nabla(\beta_k))' \nabla(\beta_k) \geq c_2 \nabla(\beta_k) \quad \forall c_2 \in (c_1, 1)$$

It is so that, if slope is very negative, the direction will have negative slope later as well.

Sudocode:

(1) choose initial step size, a_0. Set $a = a_0$ [1 in this case]

(2) Choose $\rho \in (0,1)$.

(3) choose a constant, $c \in (0,1)$

(4) Repeat till $\ell(\beta_k - a\nabla(\ell(\beta_k))) \leq \ell(\beta_k) + ca\|\nabla(\ell(\beta_k))\|^2$

(5) Return $a_k = a =$ Stepsize.

PROBLEM 2: Quasi-Newton METHOD :

MOTIVATION: Calculating Hessian is expensive, we want something cheaper.

SOLUTION : SECANT CONDITION :

Hessian is the second derivative of the loss function, to make an appropriate approximation, we can use the definition of derivative.

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h} \quad \text{-- (1)}$$

, how small is h ??

It really depends on us How accurate we want our $f'(x)$ to be.

B = Hessian = Second derivate = 1st derivative of gradient

$$= \nabla(l(\beta_{k+1}))' = \frac{\nabla l(\beta_{k+1}) - \nabla l(\beta_k)}{\beta_{k+1} - \beta_k} \quad \text{-- (2)}$$

Here $\beta_{k+1} - \beta_k = h$ = stepsize , equation (1) & (2) are equivalent.

$$\Rightarrow \boxed{H_{k+1} = \frac{\nabla(l(\beta_{k+1})) - \nabla(l(\beta_k))}{\beta_{k+1} - \beta_k}}$$

It is a good Approximation for the Hessian, found from the secant condition

$$\Rightarrow H_{k+1} s_k = z_k \quad \Big| \quad s_k = \beta_{k+1} - \beta_k$$

$$z_k = \nabla(l(\beta_{k+1})) - \nabla(l(\beta_k))$$

PSEUDO-CODE : (using Limited Memory BFGS)

(I) (a) Initialize, β_1 (truncated normal), H_1 = [Identity matrix]

(b) Calculate loss (-ve log likelyhood)

(c) Calculate gradient (g_1)

(II)

(1) Compute $p = -(H_{i-1})\, g_{i-1}$

(2) α = step size from line search

(3) Update $\beta_i = \beta_{i-1} + \alpha * p$

(4) Calculate loss

(5) Update gradient

(6) Update hessian $H_i = (1 - \rho_i)(\beta_{i-1})(1 - \rho_i^T) + \rho_i * s_i s_i^T$

where $\rho_i = s_i * s_i z_i^T$

$\rho_i = \frac{1}{z_i^T s_i}$

$z_i = g_i - g_{i-1}$

$s = \alpha * p$

g_i = gradient at i^{th} step

(7) check for convergence, if not, go to step 1 of (II) part.

if Yes, Return β_i