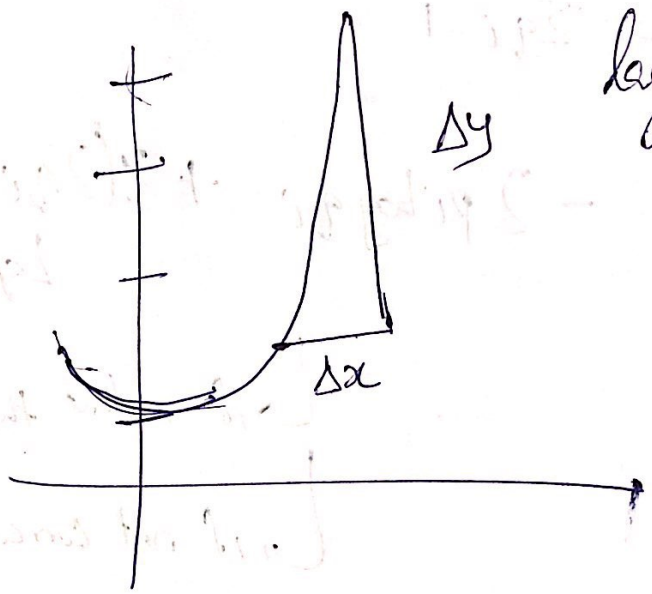
Lecture 5

Learning Parameters, Gradient Descent

Part 5.1 & 5.2

$f(x) = x^2 + 1$

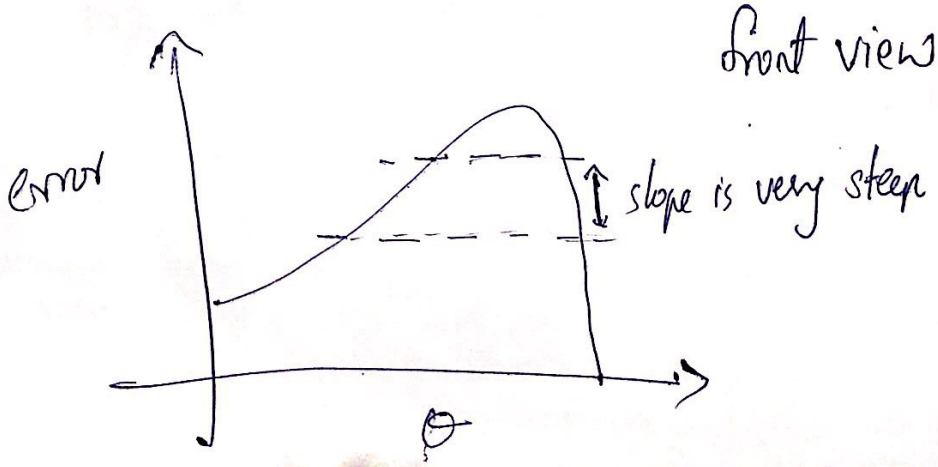


large change

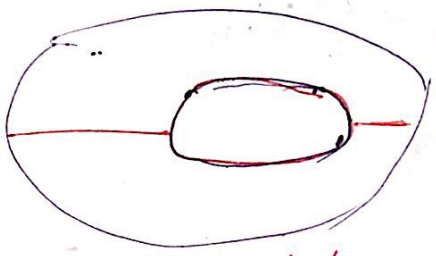
Curve is steep if the gradient $\left(\frac{\Delta y_1}{\Delta x_1}\right)$ is large.

Curve is gentle the gradient $\left(\frac{\Delta y_2}{\Delta x_2}\right)$ is small.

5.3 Contour Maps



Top view



A small distance b/w the contours indicates a steep slope along that direction

A large distance b/w the contours indicates a gentle slope along that direction

5.4 Momentum Based Gradient Descent

Lot of time to navigate regions having gentle slope.

more people pointing in one direction, keep moving in the direction with a bigger step.

Update rule

$$update_t = \gamma \cdot update_{t-1} + \eta \nabla w_t$$

(history; current gradient)

$$w_{t+1} = w_t - update_t$$

$$w_{t+1} = w_t - \gamma \cdot update_{t-1} - \eta \nabla w_t$$ {larger step}

$$update_t = \gamma \cdot update_{t-1} + \eta \nabla w_t$$

$$w_{t+1} = w_t - update_t$$

(-) Oscillations

$update_0 = 0$

$update_1 = \eta \nabla w_1$

$update_2 = \gamma \eta \nabla w_1 + \eta \nabla w_2$

$update_3 = \gamma^2 \eta \nabla w_1 + \gamma \eta \nabla w_2 + \eta \nabla w_3$

5.5 Nesterov Accelerated Gradient Descent

To reduce the oscillations

* Look before you leap
* Recall that $update_t = \gamma \cdot update_{t-1} + \eta \nabla w_t$

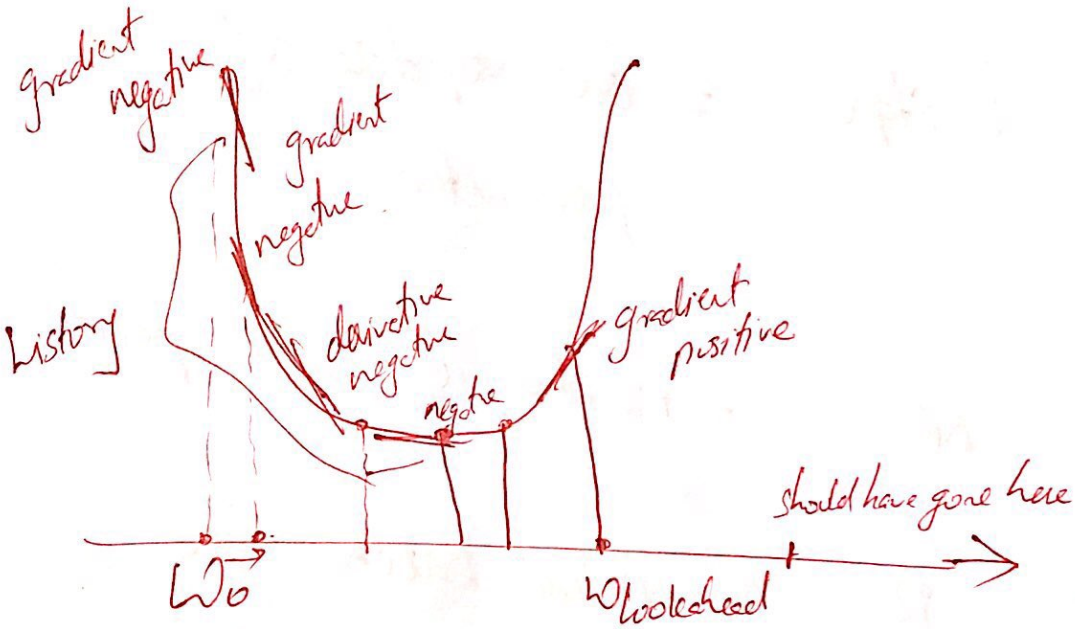
$$w_{t-1} = w_t - \gamma \cdot update_{t-1} - \eta \nabla w_t$$

Rule for NAG

$$W_{look-ahead} = W_t - \gamma \cdot update_{t-1}$$

$$update_t = \gamma \cdot update_{t-1} + \eta \nabla W_{t\,look-ahead}$$

$$W_{t-1} = W_t - update_t$$



5.6 Stochastic & Mini-Batch Gradient Descent

→ Updates the parameter for every single point.

→ Appropriate Gradient (Stochastic)

→ Oscillations (Greedy Decisions)

Higher the value of 'k', the more accurate are the estimates

1 epoch = one pass over the entire data

1 step = one update of the parameters

N = number of data points

B = Mini Batch Size

Algorithm	# no. of steps in 1 epoch
Vanilla (Batch Gradient Descent)	1
Stochastic Gradient Descent	N
Mini-Batch Gradient Descent	N/B

Lecture-5

Module 5.7

Tips for Adjusting Learning Rate and Momentum.

Adaptive O

gentle step - fast

else - slow

Try learning rate

log scale 0.0001, 0.001, 0.01, 0.1

Step Decay

→ Halve the learning rate after every 5 epochs.

→

Exponential Decay

$\eta = \eta_0^{-kt}$, where η_0 and k are hyperparameters, t is the step number

1/t Decay : $\eta = \frac{\eta_0}{1+kt}$

Tips for momentum

5-8

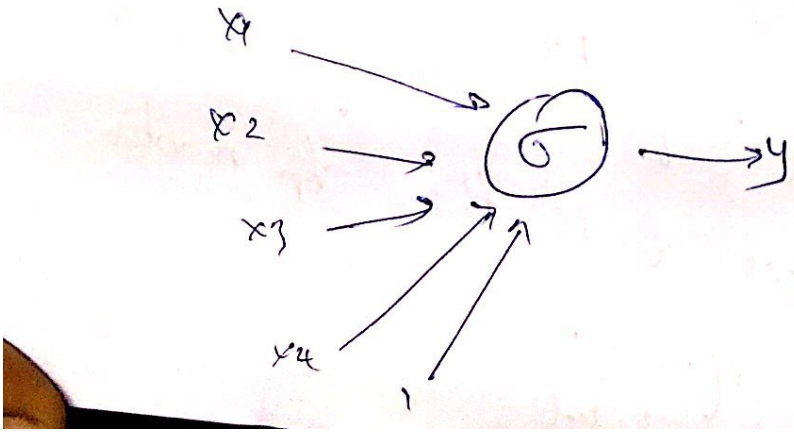
Line Search

Using many learning rate at once compute the for all the learning rates and check it.

If does a lot of computation in one step.

5-9

Gradient Descent with Adaptive Learning Rate



$$y = f(x) = \frac{1}{1+e^{-(wx+b)}}$$

$$x = \{x^1, x2, x3, x4\}$$

$$w = \{w1, w2, w3, w4\}$$

Decay the learning rate for parameters in proportion to their update history.

Rule for Adagrad

$$V_t = V_{t-1} + (\nabla w_t)^2$$

$$w_{t+1} = w_t - \frac{\eta}{(\sqrt{V_t + \epsilon})} * \nabla w_t$$

→ divided by the history.

for gradient descent $w_{t+1} = w_t - \eta * \Delta w_t$

Adagrad decays the learning rate very aggressively.

The frequent parameters will start receiving very small updates because of the decayed learning rate.

$$V_t = \beta * V_{t-1} + (1-\beta) \nabla w_t^2$$

$\beta > 0.95$

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{V_t + \epsilon}} * \nabla w_t$$

RMS prop overcomes Adagrd's problem by being less aggressive on the decay.

Adam → moment + tailing the learning rate

→ accumulate history

→ (moving average of the gradient descent)

$$m_t = \beta_1 * m_{t-1} + (1-\beta_1) * \nabla w_t$$

→ accumilative history

$$V_t = \beta_2 * V_{t-1} + (1-\beta_2) * (\nabla w_t)^2$$

Bias correction

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \qquad \hat{V}_t = \frac{V_t}{1-\beta_2^t}$$

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{\hat{V}_t + \epsilon}} * \hat{m}_t$$

→ close to the mean of the distribution

↳ accumilative descent

↳ divide the learning rate by accumilative history of gradients

Adaptive Momentum

Adam

$\beta_1 = 0.9$

$\beta_2 = 0.999$

$\epsilon = 1e-8$

Sequence generation problems $\eta = 0.01, 0.0001$.
works best

- SGD with momentum (Nesterov)

  $\eta = 0.001, 0.0001$

Best choice would be Adam.

$$m_t = \beta m_{t-1} + (1-\beta) g_t \rightarrow \nabla w_t$$

$m_0 = 0, \quad m_1 = (1-\beta) g_1$

$m_2 = \beta(1-\beta) g_1 + (1-\beta) g_2$

$m_3 = \beta^2 (1-\beta) g_1 + \beta(1-\beta) g_2 + (1-\beta) g_3$

$$m_t = 1 - \beta \sum_{i=1}^{t} \beta^{t-i} g^i$$

$$E[m_t] = E\left[1 - \beta \sum_{i=1}^{t} \beta^{t-i} g^{i}\right]$$

$$= E[g_t] \cdot (1-\beta) \sum_{i=1}^{t} \beta^{t-1}$$

$$\hookrightarrow \frac{1-\beta^t}{1-\beta}$$

$$E[m_t] = (1-\beta^t)\, E[g_t]$$

$$E[m_t] = E[g_t]$$

Lec 5.9 Bias correction in Adam

Update equations for Adam

running average of gradient

$$m_t = \beta_1 * m_{t-1} + (1-\beta_1) * \nabla w_t$$

$$v_t = \beta_2 * v_{t-1} + (1-\beta_2) * (\nabla w_t)^2$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \qquad \hat{v}_t = \frac{v_t}{1-\beta_2^t}$$

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} * \hat{m}_t$$

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} * \hat{m}_t$$

~~Ideal~~

Ideal

$$E[m_t] = E[\nabla w_t]$$

∇w_t as g_t

$$m_t = \beta_1 * m_{t-1} + (1-\beta_1) * g_t$$

$$m_0 = 0$$

$$m_1 = \beta m_0 + (1-\beta) g_1$$

$$= (1-\beta) g_1$$

$$m_2 = \beta (1-\beta) g_1 + (1-\beta) g_2$$

$$m_3 = \beta m_2 + (1-\beta) g_3$$

$$= \beta \left(\beta (1-\beta) g_1 + (1-\beta) g_2 \right) + (1-\beta) g_3$$

$$= \beta^2 (1-\beta) g_1 + \beta (1-\beta) g_2 + (1-\beta) g_3$$

$$= (1-\beta) \sum_{i=1}^{3} \beta^{3-i} g_i$$

$$m_t = (1-\beta) \sum_{i=1}^{t} \beta^{t-i} g_i$$

$$E[m_t] = E\left[(1-\beta) \sum_{i=1}^{t} \beta^{t-i} g_i\right]$$

$$E[m_t] = (1-\beta)\, E\left[\sum_{i=1}^{t} \beta^{t-i} g_i\right]$$

$$= (1-\beta) \sum_{i=1}^{t} \beta^{t-i} E[g_i]$$

All g_i's same distribution

$$E[g_i] = E[g] \quad \forall i$$

$$E[m_t] = (1-\beta) \sum_{i=1}^{t} \beta^{t-i} E[g]$$

$$= E[g] (1-\beta) \sum_{i=1}^{t} \beta^{t-i}$$

$$= E[g] (1-\beta) \frac{1-\beta^t}{1-\beta^t}$$

$$E[m_t] = E[g](1-\beta^t)$$

$$E\left[\frac{m_t}{(1-\beta^t)}\right] = E[g]$$