Last time

- approximating $\theta^* = \underset{\theta \in \Theta}{\operatorname{argmin}}\ L(\theta, D)$ with $\hat{\theta} = \underset{\theta \in \Theta}{\operatorname{opt}}\ L(\theta, D)$

- Optimization challenges for 1st-order methods
  - expensive to compute $\nabla_\theta L(\theta, D)$
  - critical points (min, max, saddle points)
  - slow convergence

- Methods
  - gradient descent (GD)

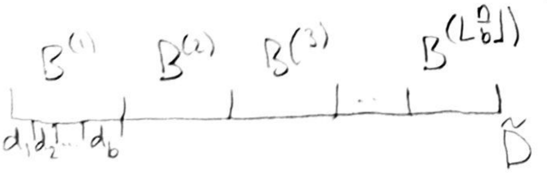
$$\boxed{\underline{\text{Update Rule}}: \quad \theta^{(t+1)} = \theta^{(t)} - \alpha^{(t+1)} \left[\nabla_\theta L(\theta, D)\right]\Big|_{\theta = \theta^{(t)}}}$$

  - stochastic gradient descent (SGD)

$$\boxed{\underline{\text{Update Rule}}: \quad \theta^{(t+1)} = \theta^{(t)} - \alpha^{(t+1)} \left[\nabla_\theta L(\theta, B^{(t+1)})\right]\Big|_{\theta = \theta^{(t)}}}$$

- $D = \{ d_i = (x_i, y_i) : i = 1, \dots, n \}$
- SGD Algorithm

$t, \hat{\theta} = 0$, random_initialization()
for epoch in [1, ..., total epochs]:
  $\tilde{D}$ = shuffle(D)
  for batch in $[1, \dots, \lfloor \frac{n}{b} \rfloor]$:
    $B^{(t+1)} = D[(\text{batch} - 1) \times b : (\text{batch}) \times b]$
    $\hat{\theta}$ = Update Rule$(\hat{\theta}, B^{(t+1)})$
    $t = t + 1$



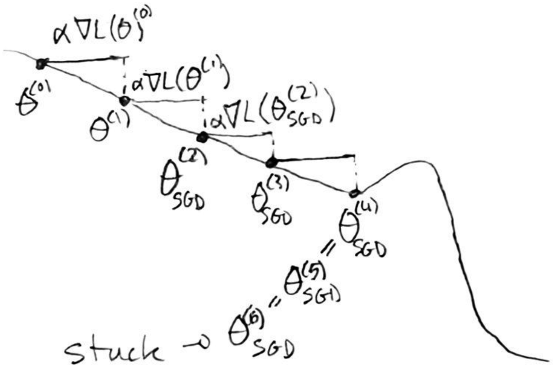
→ SGD + Momentum

Update Rule: $\theta^{(t+1)} = \theta^{(t)} - V^{(t+1)}$

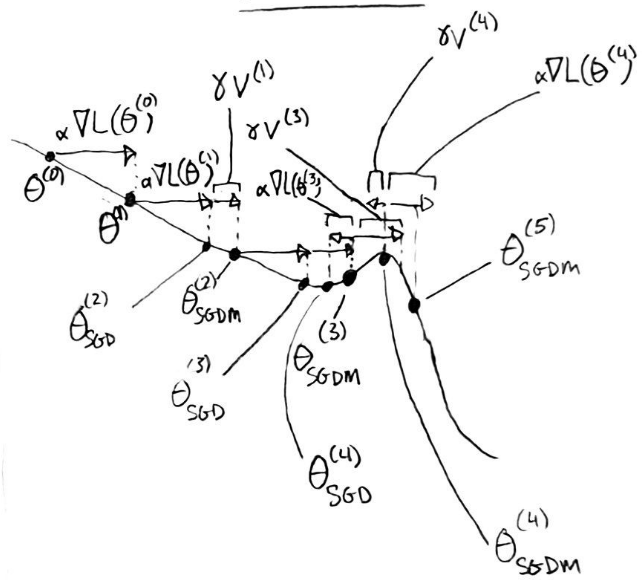
$$V^{(t+1)} = \gamma V^{(t)} + \alpha^{(t+1)} \left[\nabla_\theta L(\theta, B^{(t+1)}) \right] \Big|_{\theta = \theta^{(t)}}, \quad t > 0$$

$$V^{(0)} = 0$$

<u>SGD</u>



<u>SGD + M</u>



RMSProp (Hinton, Srivastava, Swersky)

Motivation: some informative features may encountered rarely in the training data. This "feature sparsity" can cause slow learning of these features i.e. slow convergence to the global minimum.

Example: $D =$

X_1	X_2	Y
0	10	1
-12	0	1
-12	0	1
-12	0	1
0	-10	-1
12	0	-1
12	0	-1
12	0	-1

common feature (X_1) — rare feature (X_2) — label (Y)

our model:

$$\hat{Y} = \theta_1 X_1 + \theta_2 X_2$$

ideally, we will learn

$$\theta_1 = -\frac{1}{12}$$

$$\theta_2 = \frac{1}{10}$$

if we take $L(\theta, B^{(t+1)}) = \frac{1}{b} \sum_{j=1}^{b} (\hat{Y}_j - Y_j)^2$

$\boxed{Q}$ What will the magnitude be of the gradient update for θ_2 for most batches $B^{(t+1)}$?

In general, $\dfrac{\partial L}{\partial \theta_2} = \dfrac{1}{b} \sum_{j=1}^{b} 2(\theta_1 X_{1,j} + \theta_2 X_{2,j} - Y_j) X_{2,j}$

Most batches will only contain observations for which $X_2 = 0$, thus

$$\frac{\partial L}{\partial \theta_2} = \frac{1}{b} \sum_{j=1}^{b} 2(\theta_1 X_{1,j} + \theta_2 \cdot 0 - Y_j) \cdot 0 = 0$$

so there is no update of the estimated θ_2

i.e. $\theta_2^{(t+1)} = \theta^{(t)}$

Idea: Customize the learning rate for each feature/parameter. If a particular parameter has had large updates in the past, decrease its learning rate. If the parameter has had small updates in the past, increase its learning rate.

Update Rule:

$$\theta_j^{(t+1)} = \theta_j^{(t)} - \frac{\alpha^{(t+1)}}{\sqrt{G_j^{(t+1)}} + \varepsilon} \cdot \left[\left[\nabla_\theta L(\theta, B^{(t+1)}) \right] \Big|_{\theta = \theta^{(t)}} \right]_j$$

j is the index of each univariate parameter in θ

i.e. $\theta = \{ \theta_j : j = 1, \dots p, \theta_j \in \mathbb{R} \}$

ε is a small positive number $\boxed{Q}$ purpose? prevent $x/0$

$$G_j^{(t+1)} = \gamma G_j^{(t)} + (1-\gamma) \left(\left[\left[\nabla_\theta L(\theta, B^{(t+1)}) \right] \Big|_{\theta = \theta^{(t)}} \right]_j \right)^2$$

$\gamma \in [0,1]$ ← memory parameter, $G_j^{(0)} = 0$

Adam (Kingma, Ba)

→ combination of RMSprop and momentum

Update Rule:
$$\theta_j^{(t+1)} = \theta_j^{(t)} - \frac{\alpha^{(t+1)}}{\sqrt{\hat{G}_j^{(t+1)}} + \varepsilon} \cdot \hat{V}_j^{(t+1)}$$

$$\hat{V}_j^{(t+1)} = \frac{V_j^{(t+1)}}{(1-\beta_1^{t+1})}, \quad \hat{G}_j^{(t+1)} = \frac{G_j^{(t+1)}}{(1-\beta_2^{t+1})} \quad \Bigg\} \text{ zero initialization bias correction}$$

$$V_j^{(t+1)} = \beta_1 V_j^{(t)} + (1-\beta_1)\left[\left[\nabla_\theta L(\theta, B^{(t+1)})\right]\Big|_{\theta=\theta^{(t)}}\right]_j$$

$$G_j^{(t+1)} = \beta_2 G_j^{(t)} + (1-\beta_2)\left[\left[\nabla_\theta L(\theta, B^{(t+1)})\right]\Big|_{\theta=\theta^{(t)}}\right]_j^2$$

$$V_j^{(0)} = 0, \quad G_j^{(0)} = 0$$

$\beta_0, \beta_1 \in [0,1]$ usually ≈ 0.99

Summary: four basic improvements to GD commonly used to overcome computational burden & improve convergence in practice → SGD, Momentum, RMSprop, Adam

Further reading: AdamW