Learning Rate Optimizing

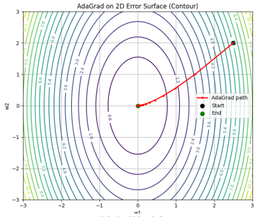
AdaGrad

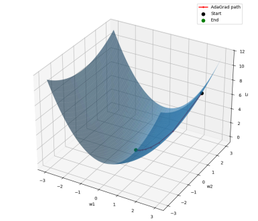
Rhyme
“Square the slope and make a stack,
Bigger the pile, the learning steps slack.
Divide by the root so updates shrink —
AdaGrad slows with every blink.”

- “Square the slope” → $(\nabla_\theta)^2$
- “make a stack” → cumulative $G = G + \ldots$
- “updates shrink” → denominator grows
- “slows with every blink” → learning rate decays

$$G = G + (\nabla_\theta)^2$$

$$\theta = \theta - \eta \frac{\nabla_\theta}{\sqrt{G} + \epsilon}$$





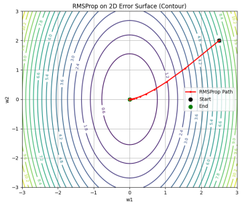
RMSProp

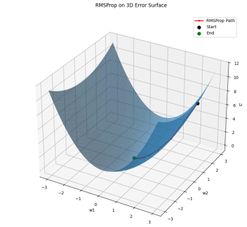
Rhyme
“Forget the past but not all the way,
Keep a soft memory with beta’s sway.
Smooth the squares and divide to drop —
That's the rhythm of RMSProp.”

- “soft memory” → moving average
- “beta’s sway” → β weighting
- “smooth the squares” → averaging squared gradients
- “divide to drop” → update rule denominator

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

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





Adam

Rhyme
“First the mean and second the square,
Moments two dance in Adam’s lair.
Bias corrected, the steps align —
Adaptive moves that learn just fine.”

- “mean” → first moment m
- “square” → second moment v
- “bias corrected” → $\hat{m}$ and $\hat{v}$
- “adaptive moves” → denominator adjusts per parameter

$$m = \beta_1 m + (1 - \beta_1)\nabla_\theta$$

$$v = \beta_2 v + (1 - \beta_2)\nabla_\theta^2$$

$$\hat{m} = \frac{m}{1 - \beta_1^t}, \quad \hat{v} = \frac{v}{1 - \beta_2^t}$$

$$\theta = \theta - \eta \frac{\hat{m}}{\sqrt{\hat{v}} + \epsilon}$$