

### Introducing Adagrad

How do we convert the adaptive learning rate intuition into an equation?

1. **Intuition:** Decay the learning rate for parameters in proportion to their update history (fewer updates, lesser decay)
2. The Adagrad (Adaptive Gradient) is an algorithm which satisfies the above intuition
3. Adagrad
  - a.  $v_t = v_{t-1} + (\nabla \omega_t)^2$ 
    - i. This value increments based on the gradient of that particular iteration, i.e. the value of the feature is non-zero.
    - ii. In the case of dense features, it increments for most iterations, resulting in a larger  $v_t$  value
    - iii. For sparse features, does not increment much as the gradient value is often 0, leading to a lower  $v_t$  value.
  - b.  $\omega_{t+1} = \omega_t - \frac{\eta}{\sqrt{(v_t)} + \epsilon} \nabla \omega_t$ 
    - i. The denominator term  $\sqrt{(v_t)}$  serves to regulate the learning rate  $\eta$
    - ii. For dense features,  $v_t$  is larger,  $\sqrt{(v_t)}$  becomes larger thereby lowering  $\eta$
    - iii. For sparse features,  $v_t$  is smaller,  $\sqrt{(v_t)}$  becomes smaller and lowers  $\eta$  to a smaller extent.
    - iv. The  $\epsilon$  term is added to the denominator  $\sqrt{(v_t)} + \epsilon$  to **prevent a divide-by-zero error** from occurring in the case of very sparse features i.e. where all the data points yield zero up till the measured instance.