deeplearning.ai

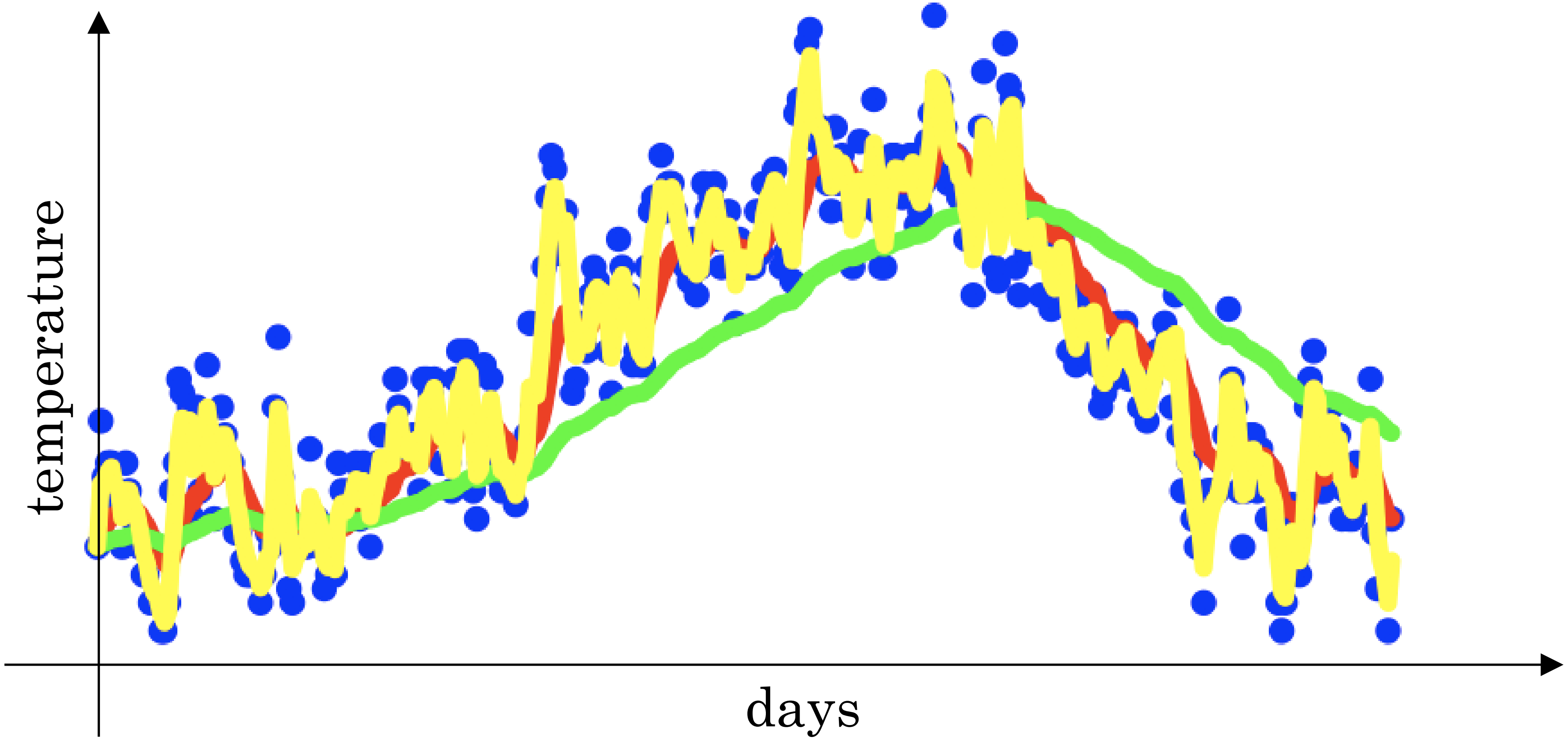
Optimization Algorithms

Understanding exponentially weighted averages

Exponentially weighted averages

$$\boxed{v_t = \beta v_{t-1} + (1-\beta)\theta_t}$$

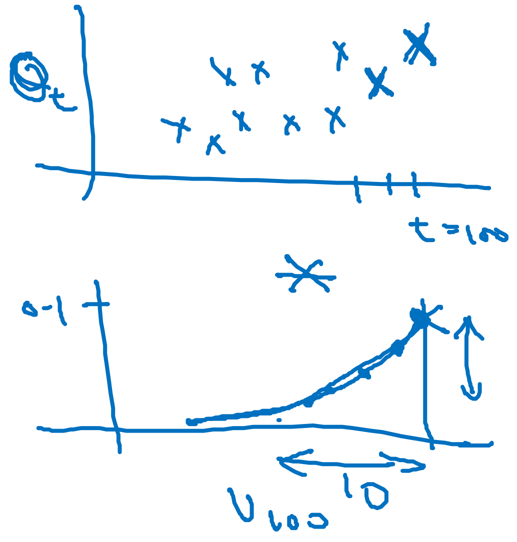
$\beta = 0.9 \qquad 0.98 \qquad 0.5$



Andrew Ng

Exponentially weighted averages

$$v_t = \beta v_{t-1} + (1-\beta)\theta_t$$



$\approx \frac{1}{1-\beta}$

$\varepsilon = 1-\beta$

$$v_{100} = 0.9 v_{99} + 0.1 \theta_{100}$$

$$\boxed{v_{99} = 0.9 v_{98} + 0.1 \theta_{99}}$$

$$\boxed{v_{98} = 0.9 v_{97} + 0.1 \theta_{98}}$$

...

$0.1\,\theta_{98} + 0.9\,v_{97}$

$$V_{100} = 0.1\,\theta_{100} + 0.9\,\cancel{V_{99}}\,(0.1\,\theta_{99} + 0.9\,\cancel{V_{98}})$$

$$= 0.1\,\theta_{100} + 0.1 \times 0.9 \cdot \theta_{99} + 0.1\,(0.9)^2\,\theta_{98} + 0.1\,(0.9)^3\,\theta_{97} + 0.1\,(0.9)^4\,\theta_{96} + \dots$$

$$0.9^{10} \approx 0.35 \approx \frac{1}{e}$$

$$\underbrace{(1-\varepsilon)}_{0.9}{}^{1/\varepsilon} = \frac{1}{e}$$

0.98 ?

$$\varepsilon = 0.02 \rightarrow 0.98^{50} \approx \frac{1}{e}$$

Andrew Ng

Implementing exponentially weighted averages

$v_0 = 0$

$v_1 = \beta v_0 + (1-\beta)\,\theta_1$

$v_2 = \beta v_1 + (1-\beta)\,\theta_2$

$v_3 = \beta v_2 + (1-\beta)\,\theta_3$

...

$V_\theta := 0$

$V_\theta := \beta v + (1-\beta)\,\theta_1$

$V_\theta := \beta v + (1-\beta)\,\theta_2$

$\vdots$

$V_\theta = 0$

Repeat {

Get next θ_t

$V_\theta := \beta V_\theta + (1-\beta)\,\theta_t$

}

Andrew Ng