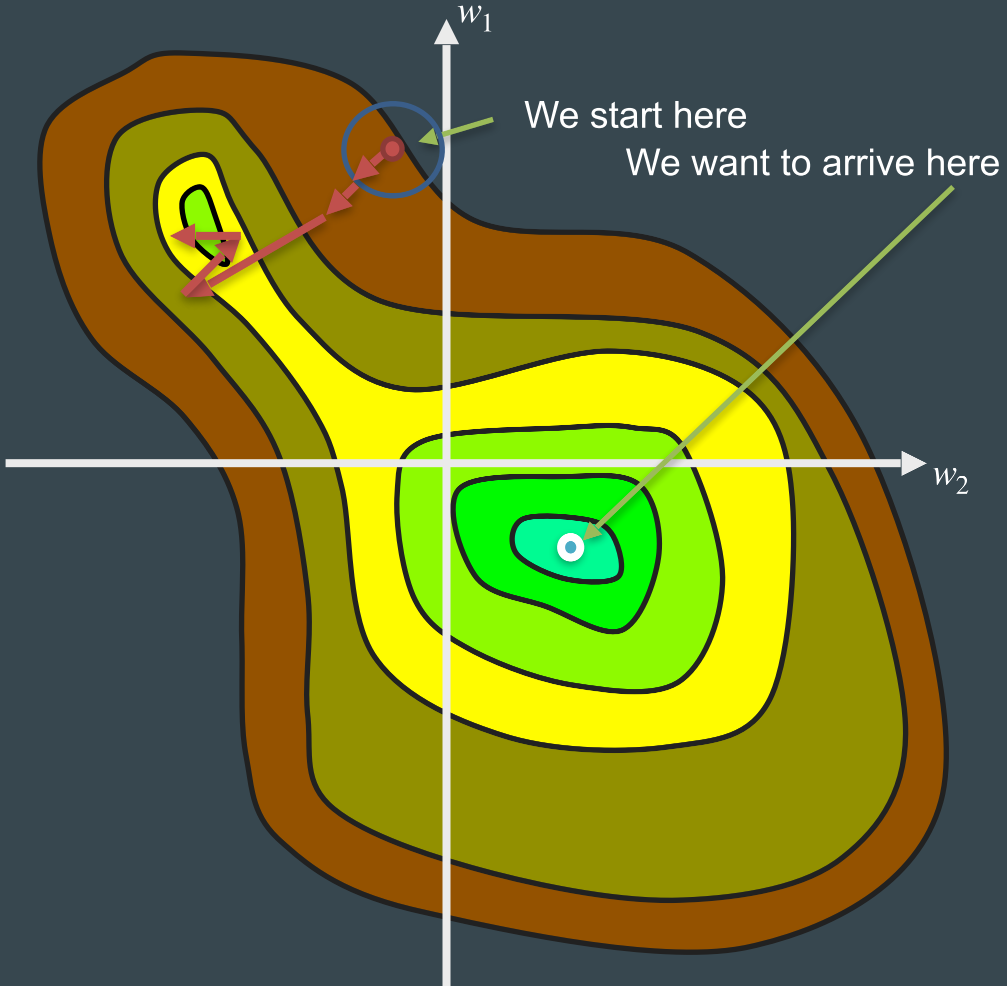
Part 4: Optimisers

Mikhail Romanov

Gradient Descent

Gradient Descent

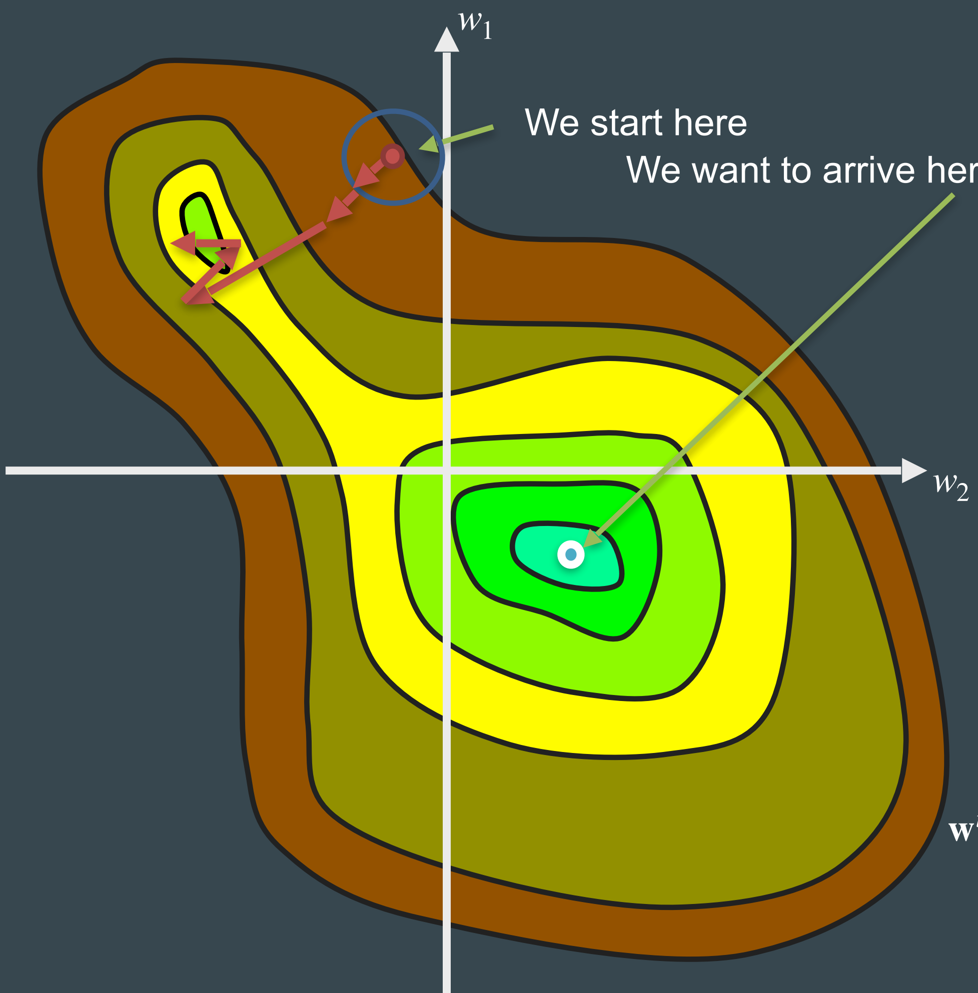


$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_p \end{bmatrix} \qquad \nabla_{\mathbf{w}} L = \begin{bmatrix} \frac{\partial L}{\partial w_0} \\ \frac{\partial L}{\partial w_1} \\ \vdots \\ \frac{\partial L}{\partial w_p} \end{bmatrix}$$

$$\mathbf{w}^{t+1} = \mathbf{w}^{t} - \alpha \nabla_{\mathbf{w}} L^{t}_{total}$$

$$L^{t=0}_{total} = \frac{1}{S} \sum_{s=1}^{S} L(Net_{\mathbf{w^{t=0}}}(x_s), y_s)$$

Stochastic Gradient Descent



$$\nabla L = \nabla(L_1 + L_2 + \ldots + L_S) = \nabla L_1 + \nabla L_2 + \ldots + \nabla L_S$$

$$\begin{array}{cccccc} & 1 & 2 & 3 & & S \\ L^{t=0}_{total} = & L_5 + & L_3 + & L_9 + & \ldots + & L_5 \end{array}$$

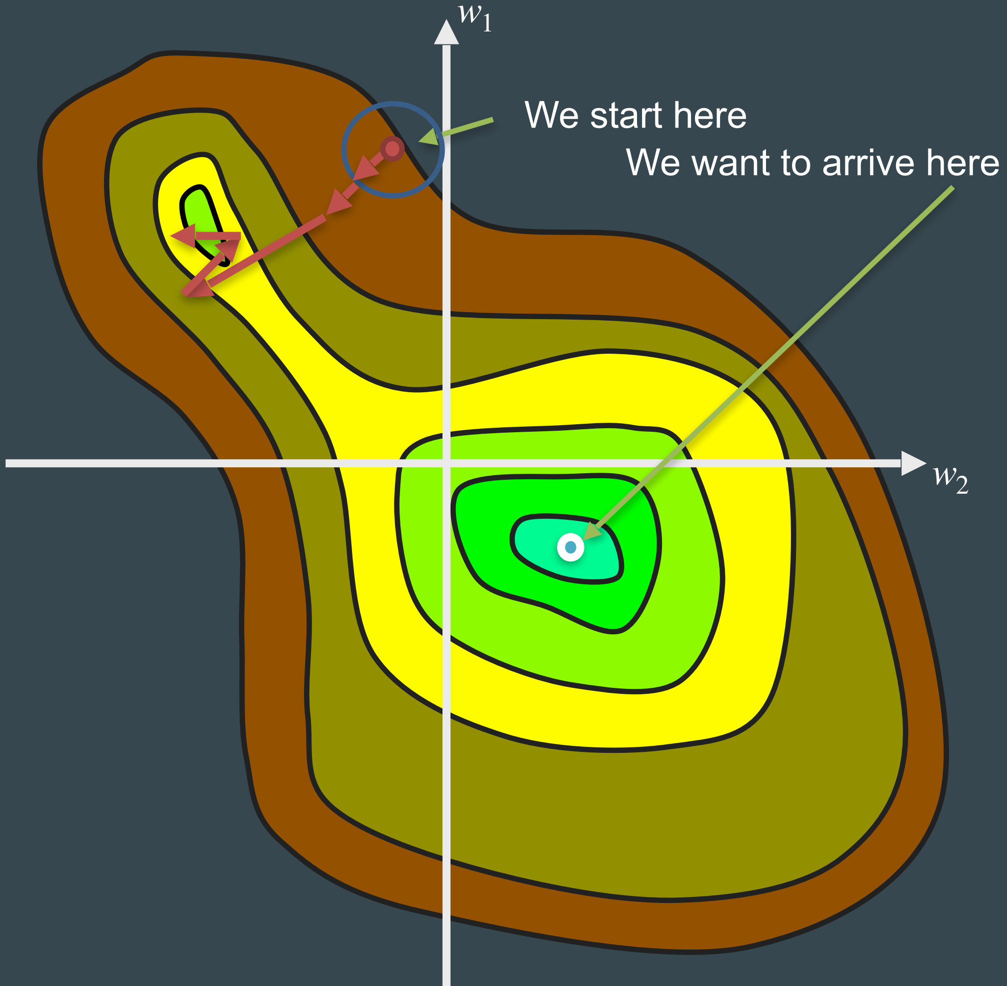
$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_p \end{bmatrix} \qquad \nabla_{\mathbf{w}} L = \begin{bmatrix} \frac{\partial L}{\partial w_0} \\ \frac{\partial L}{\partial w_1} \\ \vdots \\ \frac{\partial L}{\partial w_p} \end{bmatrix}$$

$$\mathbf{w}^{t=1} = \mathbf{w}^{t=0} - \alpha \nabla_{\mathbf{w}} L_1^{t=0}$$

Gradient is calculated for one samplele

$$\mathbf{w}^{t=1} = \mathbf{w}^{t=0} - \alpha \nabla L_1 - \alpha \nabla L_2 - \ldots - \alpha \nabla L_S \approx \mathbf{w} - \alpha \nabla L$$

Stochastic Gradient Descent



$$L_{total} = \overset{b_1}{\boxed{L_5 + L_3}} + \overset{b_2}{\boxed{L_9 +}} \ldots \overset{b_N}{\boxed{+ L_5}}$$

$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_p \end{bmatrix} \qquad \nabla_{\mathbf{w}} L = \begin{bmatrix} \frac{\partial L}{\partial w_0} \\ \frac{\partial L}{\partial w_1} \\ \vdots \\ \frac{\partial L}{\partial w_p} \end{bmatrix}$$

$$\mathbf{w}^{t=1} = \mathbf{w}^{t=0} - \alpha \nabla_{\mathbf{w}} L_{b_1}^{t=0}$$

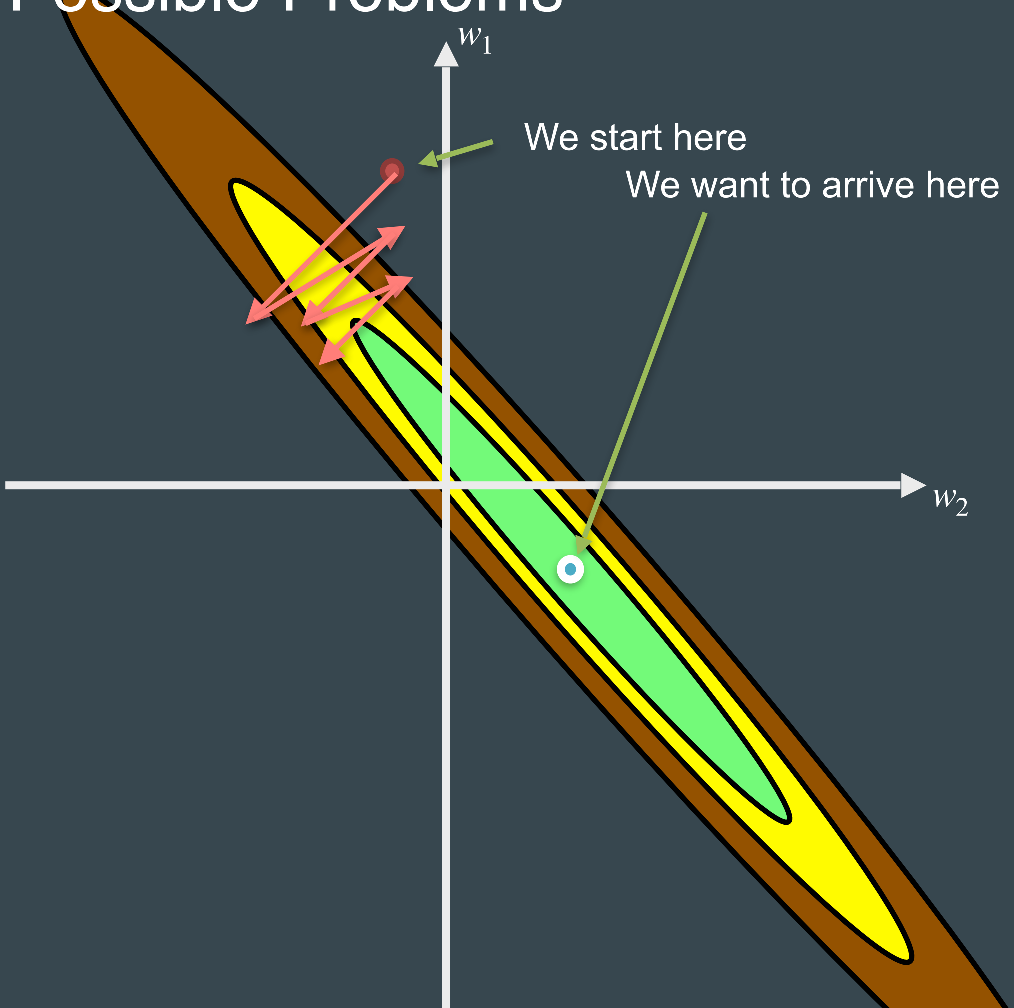
$$\mathbf{w}^{t=2} = \mathbf{w}^{t=1} - \alpha \nabla_{\mathbf{w}} L_{b_2}^{t=1}$$

$$\ldots$$

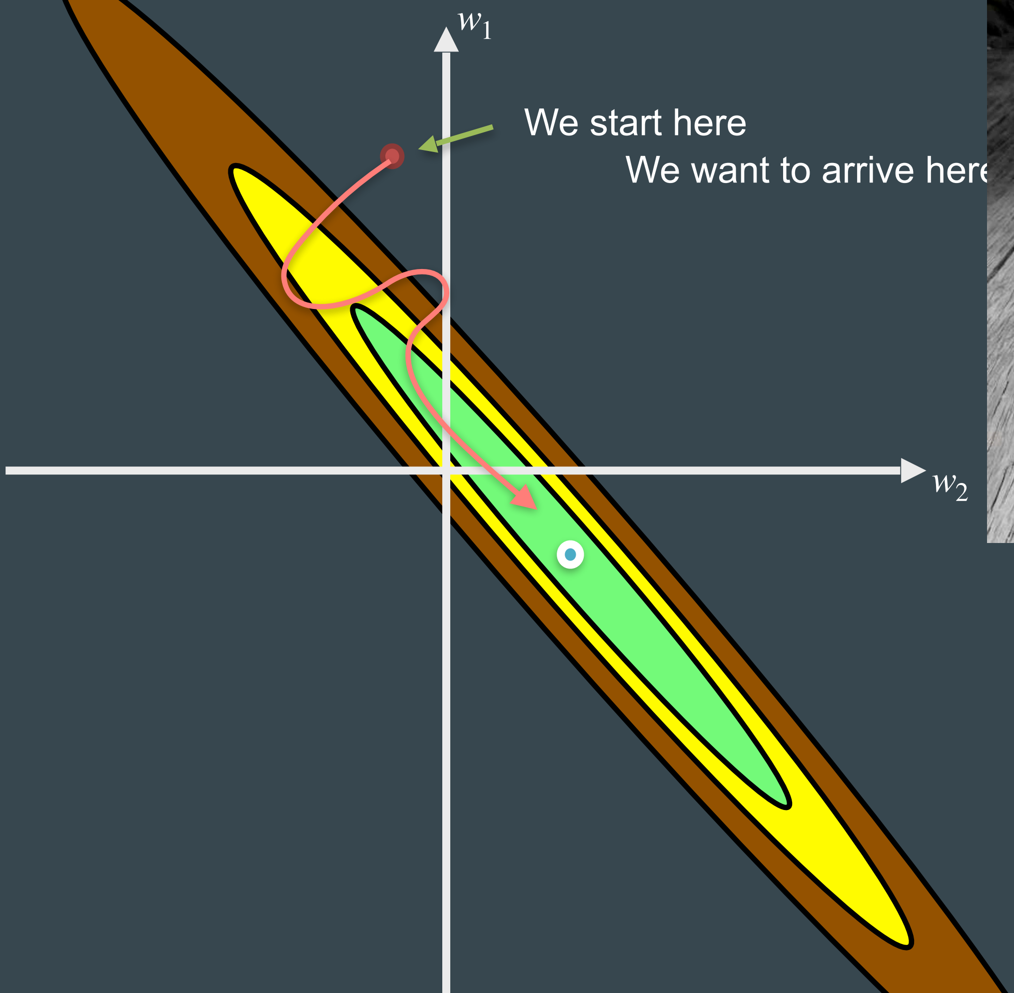
$$\mathbf{w}^{t+1} = \mathbf{w}^{t} - \alpha \nabla_{\mathbf{w}} L_{b_t}^{t}$$

Gradient is calculated for one Batch

Possible Problems

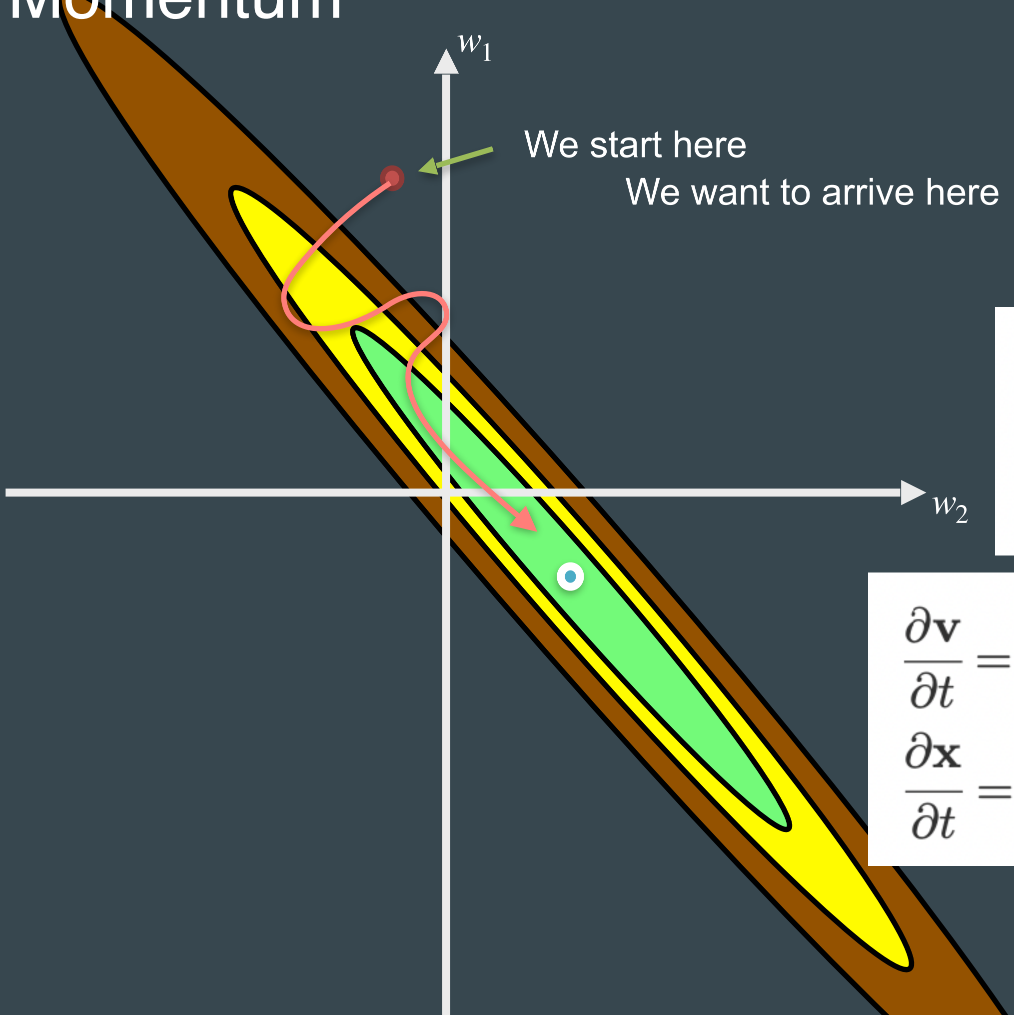


Momentum



$$\begin{cases} \dfrac{\partial \mathbf{v}}{\partial t} = \dfrac{1}{m}(\mathbf{F} + \mathbf{F}_{\text{тр}}) \\ \dfrac{\partial \mathbf{x}}{\partial t} = \mathbf{v} \end{cases}$$

Momentum



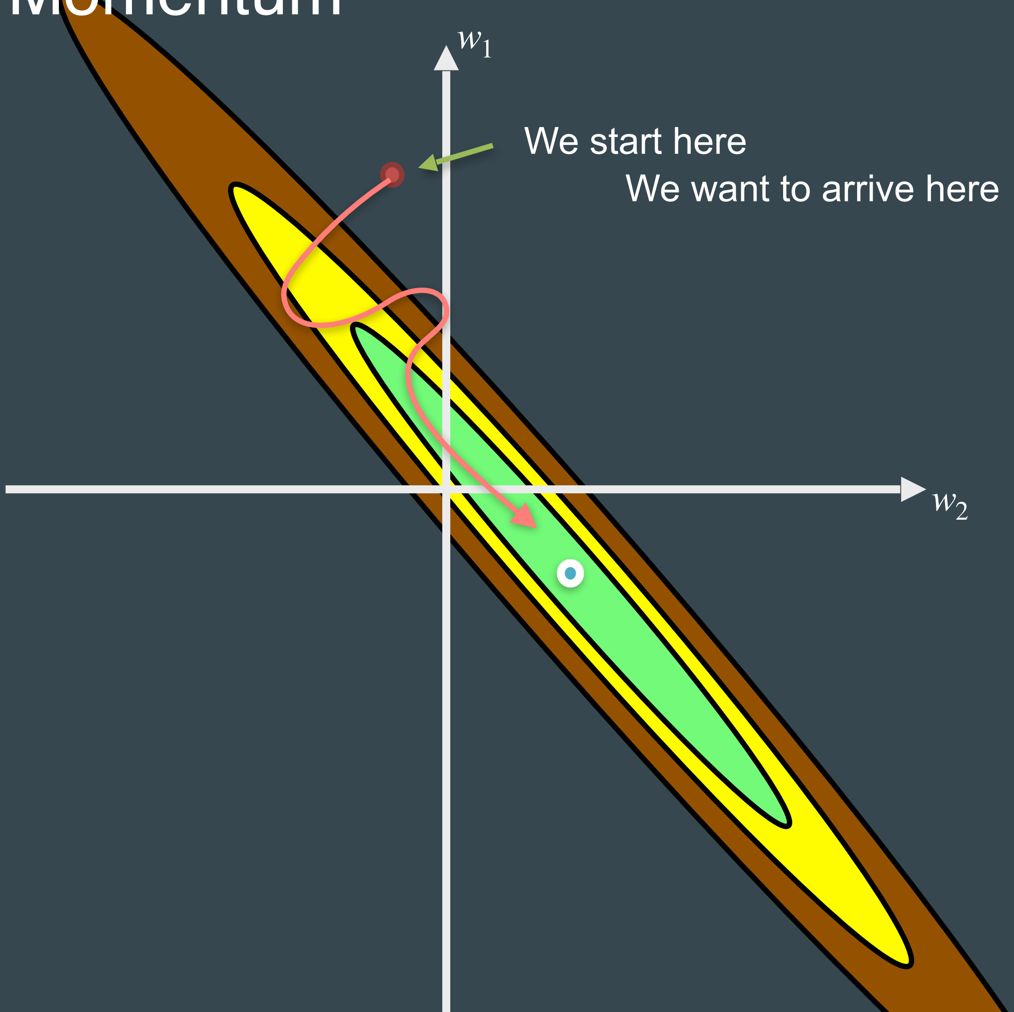
$$\begin{cases} \dfrac{\partial \mathbf{v}}{\partial t} = \dfrac{1}{m}(\mathbf{F} + \mathbf{F}_{\text{TP}}) \\ \dfrac{\partial \mathbf{x}}{\partial t} = \mathbf{v} \end{cases}$$

$$\begin{cases} \dfrac{\partial \mathbf{v}}{\partial t} = \dfrac{1}{m}(\mathbf{F} + \mathbf{F}_{\text{TP}}) = -\dfrac{1}{m}\boldsymbol{\nabla} L - \dfrac{1}{m}\gamma \mathbf{v} \\ \dfrac{\partial \mathbf{x}}{\partial t} = \mathbf{v} \end{cases}$$

$$\dfrac{\partial \mathbf{v}}{\partial t} = \dfrac{\mathbf{v}^{t+1} - \mathbf{v}^t}{\Delta t}$$
$$\dfrac{\partial \mathbf{x}}{\partial t} = \dfrac{\mathbf{x}^{t+1} - \mathbf{x}^t}{\Delta t}$$

$$\begin{cases} \mathbf{v}^{t+1} = -\alpha \boldsymbol{\nabla} L(\mathbf{w}^t) - \beta \mathbf{v}^t \\ \mathbf{w}^{t+1} = \mathbf{w}^t + \mathbf{v}^t \end{cases}$$

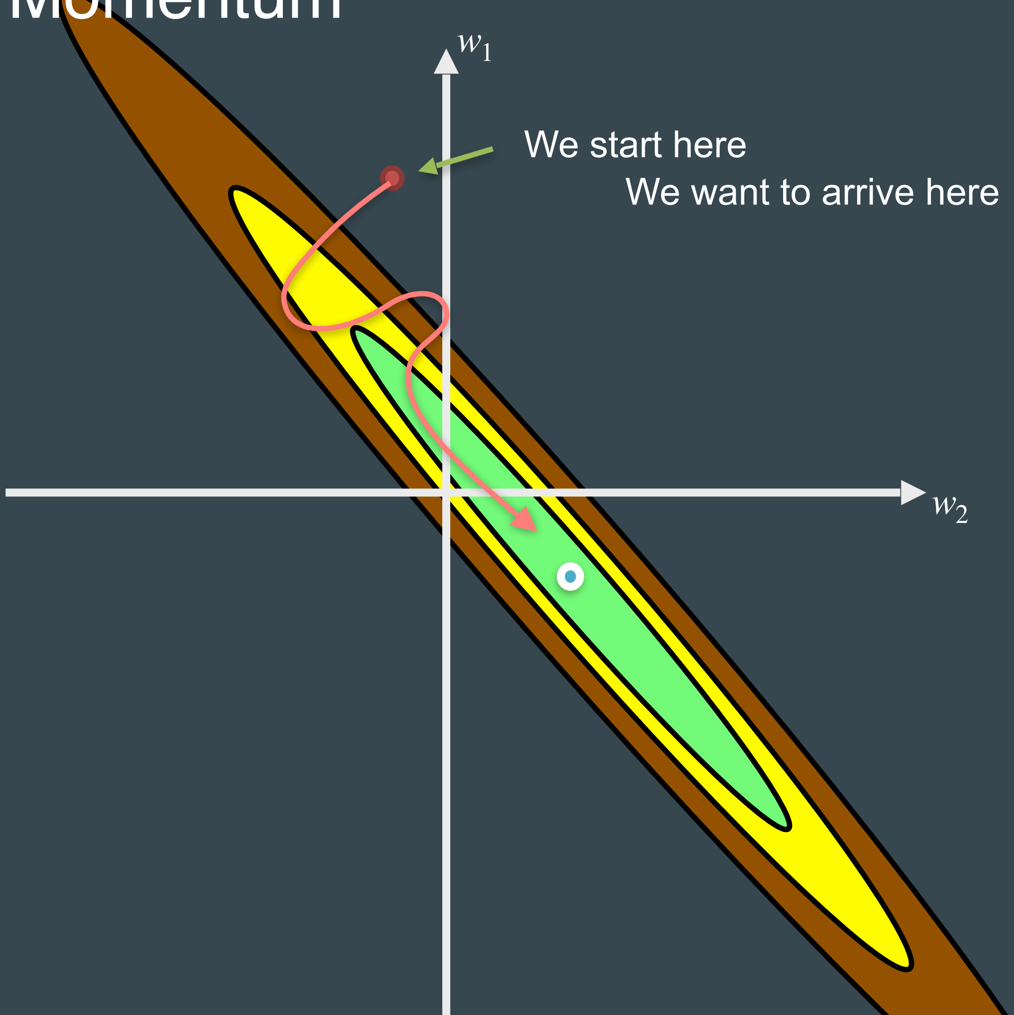
Momentum



$$\begin{cases} \dfrac{\partial \mathbf{v}}{\partial t} = \dfrac{1}{m}(\mathbf{F} + \mathbf{F}_{\text{тр}}) = -\dfrac{1}{m}\boldsymbol{\nabla} L - \dfrac{1}{m}\gamma \mathbf{v} \\ \dfrac{\partial \mathbf{x}}{\partial t} = \mathbf{v} \end{cases}$$

$$\begin{cases} \mathbf{v}^{t+1} = -\alpha \boldsymbol{\nabla} L\left(\mathbf{w}^t + \mathbf{v}^t\right) - \beta \mathbf{v}^t \\ \mathbf{w}^{t+1} = \mathbf{w}^t + \mathbf{v}^{t+1} \end{cases}$$

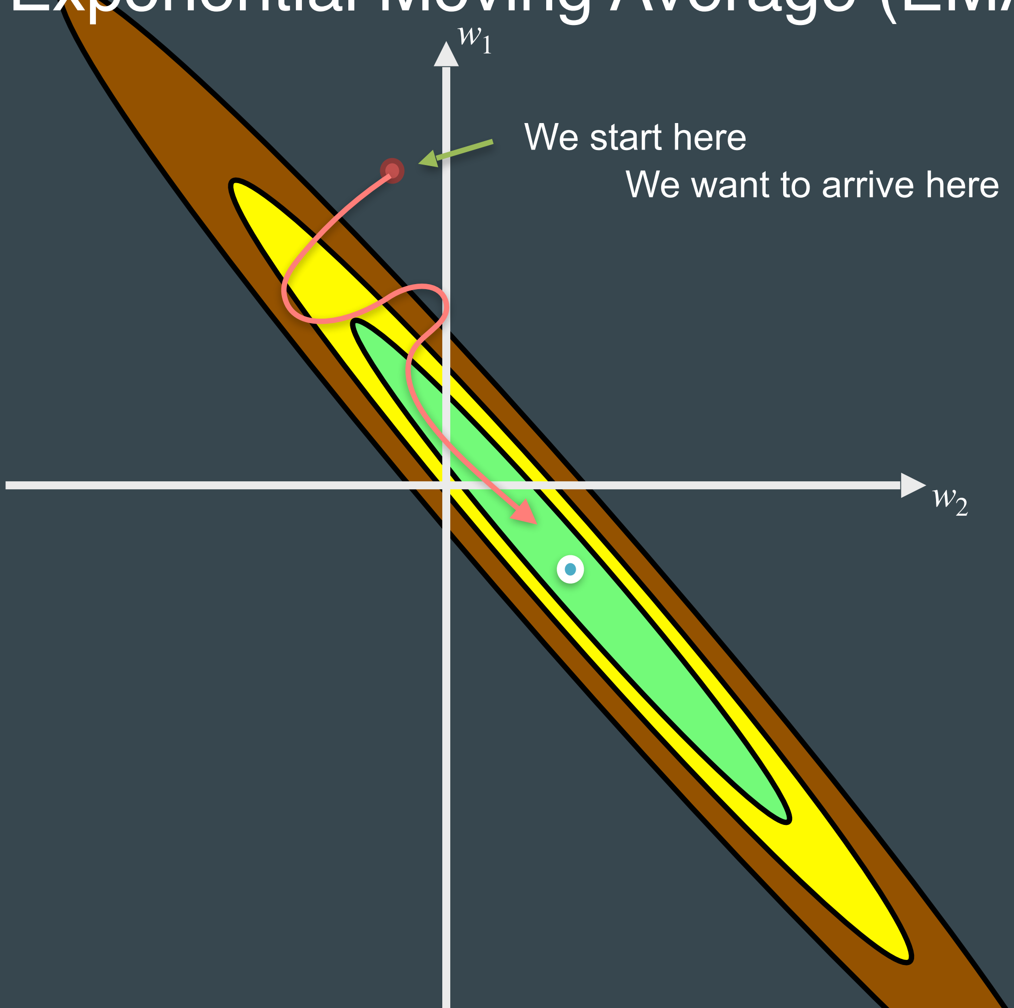
Momentum



$$\mathbf{w}^{t+1} = \mathbf{w}^{t} - \alpha EMA_{\beta}^{t}(\nabla L)$$

$$EMA_{\beta}^{t}(\nabla L) = (1-\beta)\,\nabla L^{t} + \\ +\beta EMA_{\beta}^{t-1}(\nabla L)$$

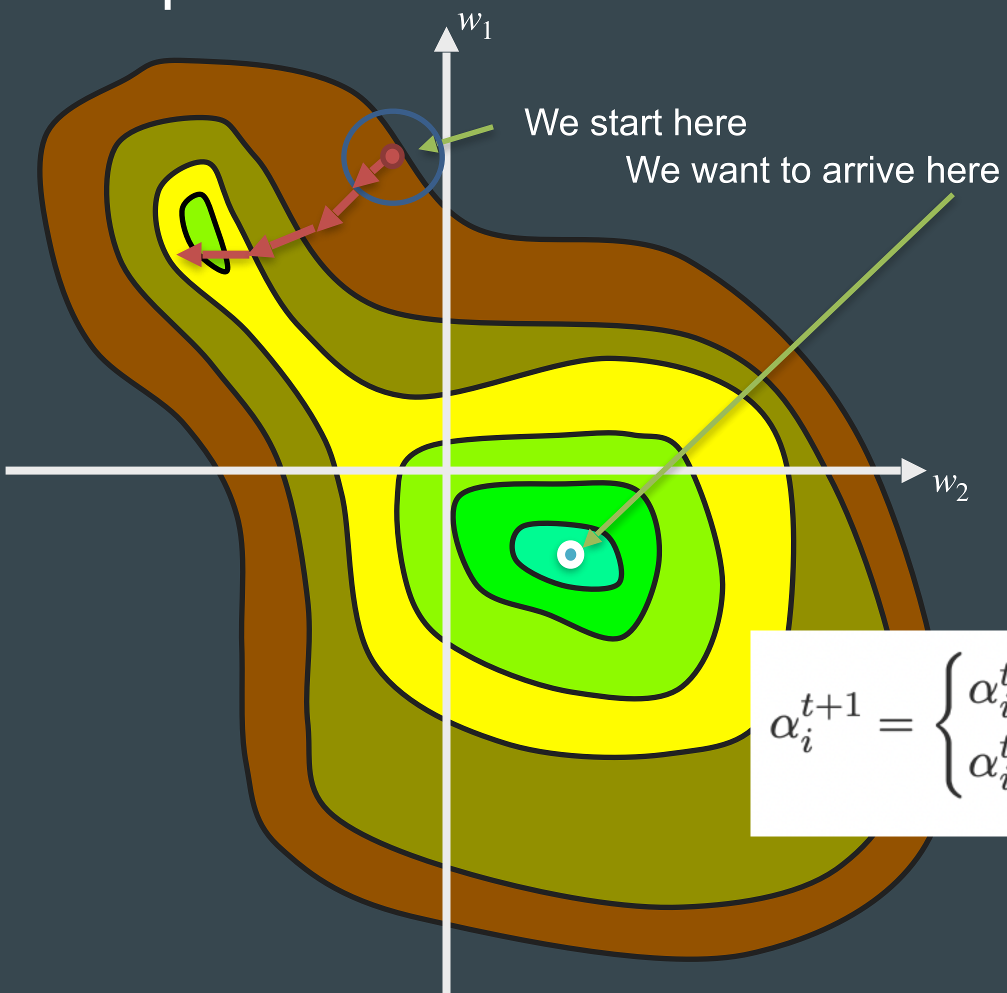
Exponential Moving Average (EMA)



$$\mathbf{w}^{t+1} = \mathbf{w}^{t} - \alpha EMA_{\beta}^{t}(\nabla L)$$

$$EMA_{\beta}^{t}(\nabla L) = (1-\beta)\, \nabla L^{t} + \\ +\beta EMA_{\beta}^{t-1}(\nabla L)$$

RProp



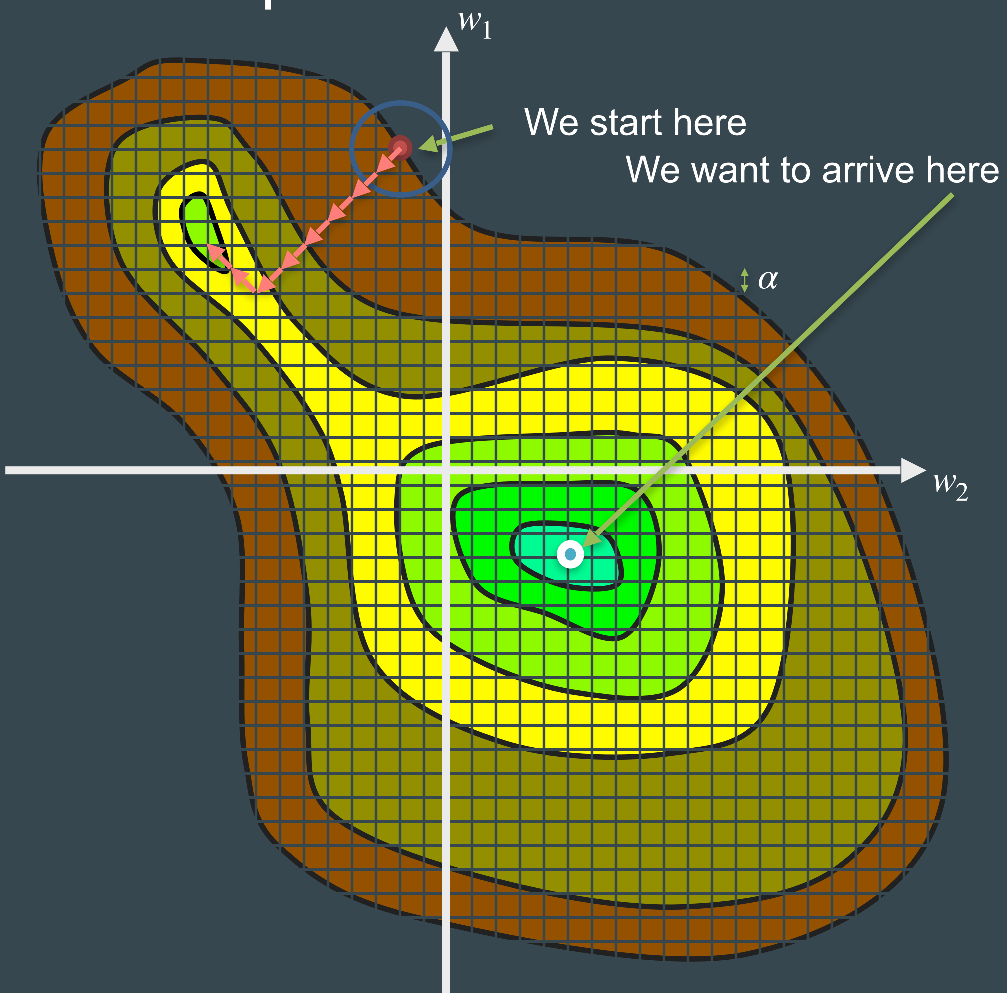
- SGD has equal learning rates for all the parameters

- What if we make individual LRs?
- We will only take into account signs of gradients
- We will initialise all the LRs with equal values
- And then we will adjust them

$$\mathbf{w}_i^t = \mathbf{w}_i^{t-1} - \alpha_i^t \cdot sign\left(\nabla_i L\left(\mathbf{w}^{t-1}\right)\right)$$

$$\alpha_i^{t+1} = \begin{cases} \alpha_i^t \cdot 1.2 \text{ if } sign\left(\nabla_i L\left(\mathbf{w}^t\right) \cdot \nabla_i L\left(\mathbf{w}^{t-1}\right)\right) > 0 \\ \alpha_i^t \cdot 0.6 \text{ if } sign\left(\nabla_i L\left(\mathbf{w}^t\right) \cdot \nabla_i L\left(\mathbf{w}^{t-1}\right)\right) \leqslant 0 \end{cases}$$

RMSProp



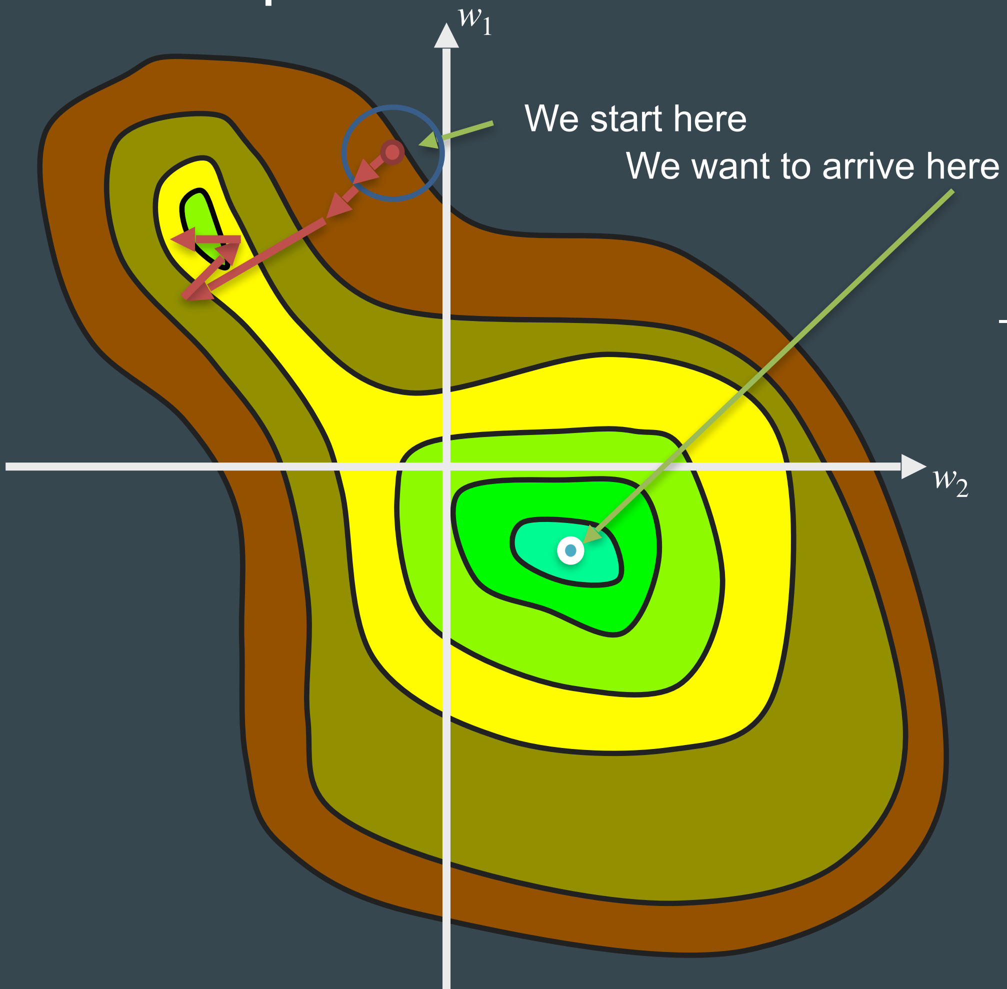
- RMSProp == RProp, no LR adjusting
- Equal step lengths for all parameters and gradients
- Like grid search

$$\mathbf{w}^{t+1} = \mathbf{w}^t - \alpha \frac{\nabla L}{\nabla L}$$

$$\frac{\nabla (L_1 + L_2)}{\nabla (L_1 + L_2)} \neq \frac{\nabla L_1}{\nabla L_1} + \frac{\nabla L_2}{\nabla L_2}$$

$$\frac{\nabla (L_1 + L_2)}{\nabla (L_1 + L_2)} = \frac{\nabla L_1}{\nabla (L_1 + L_2)} + \frac{\nabla L_2}{\nabla (L_1 + L_2)}$$

RMSProp



$$\nabla L = \nabla L_1 + \nabla L_2 + \ldots + \nabla L_S$$

$$\frac{\nabla L}{\nabla L} \neq \frac{\nabla L_1}{\nabla L_1} + \frac{\nabla L_2}{\nabla L_2} + \ldots + \frac{\nabla L_S}{\nabla L_S}$$

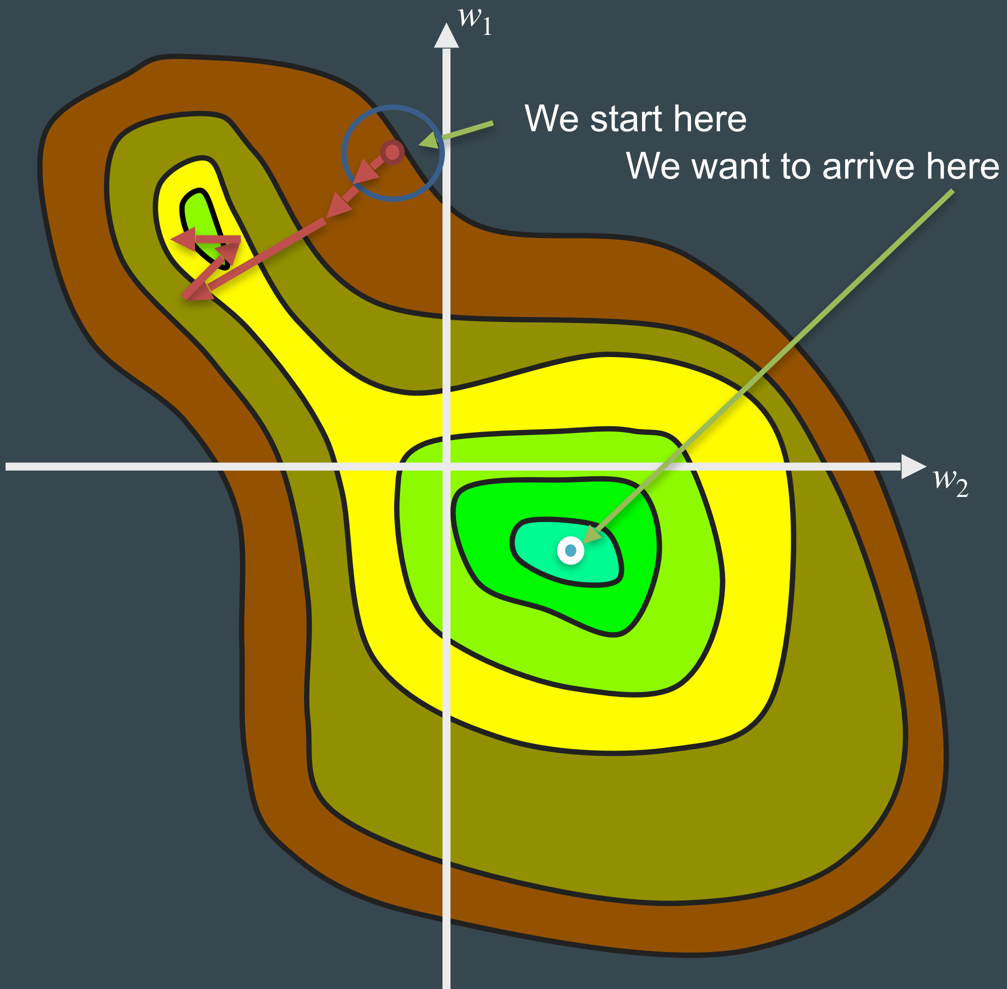
$$\frac{\nabla L}{\nabla L} = \frac{\nabla (L_1 + L_2 + \ldots + L_s)}{\nabla L} =$$

$$\frac{\nabla (L_1 + L_2 + \ldots + L_s)}{\sqrt{\nabla L^2}} =$$

$$\frac{\nabla L_1}{\sqrt{\nabla L^2}} + \frac{\nabla L_2}{\sqrt{\nabla L^2}} + \ldots + \frac{\nabla L_S}{\sqrt{\nabla L^2}}$$

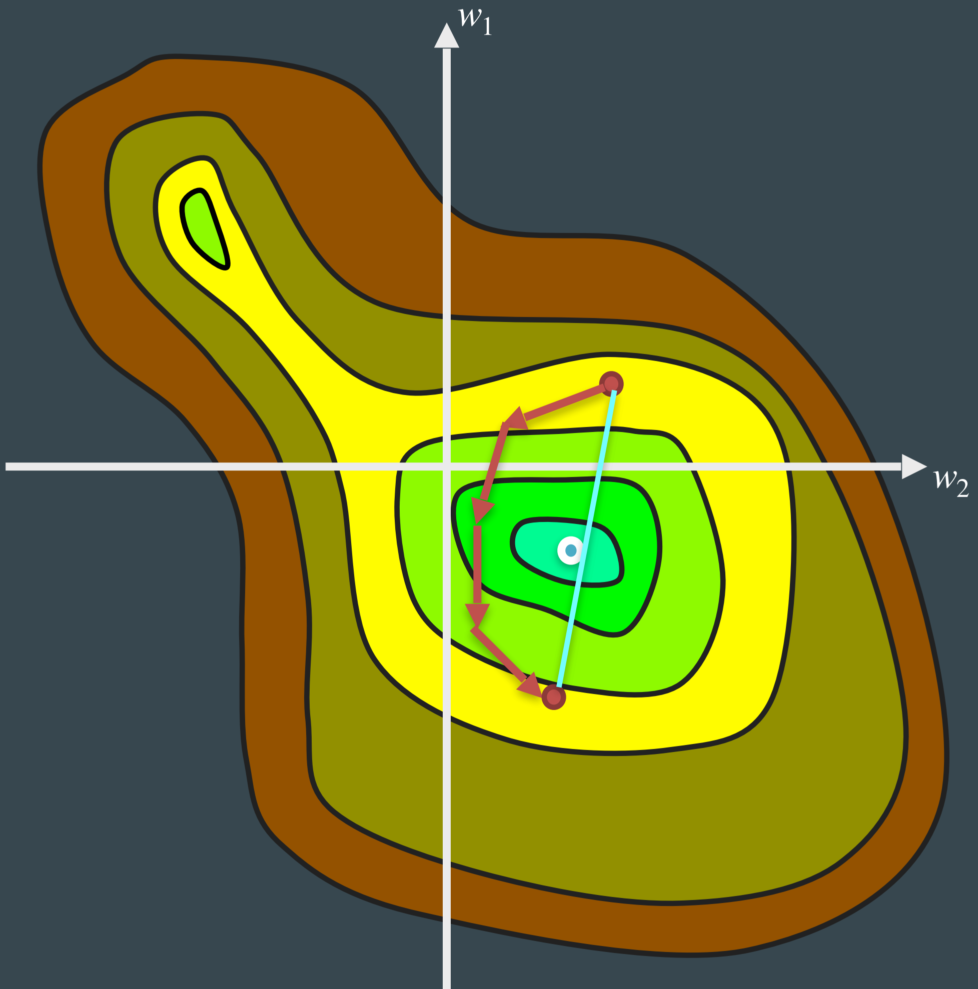
$$\mathbf{w}^{t+1} = \mathbf{w}^t - \alpha \frac{\nabla L}{\sqrt{EMA^t_{\beta_2} \nabla L^2} + \varepsilon}$$

Adam



$$\mathbf{w}^{t+1} = \mathbf{w}^{t} - \alpha \frac{EMA^{t}_{\beta_1} \nabla L}{\sqrt{EMA^{t}_{\beta_2} \nabla L^2} + \varepsilon}$$

Look Ahead



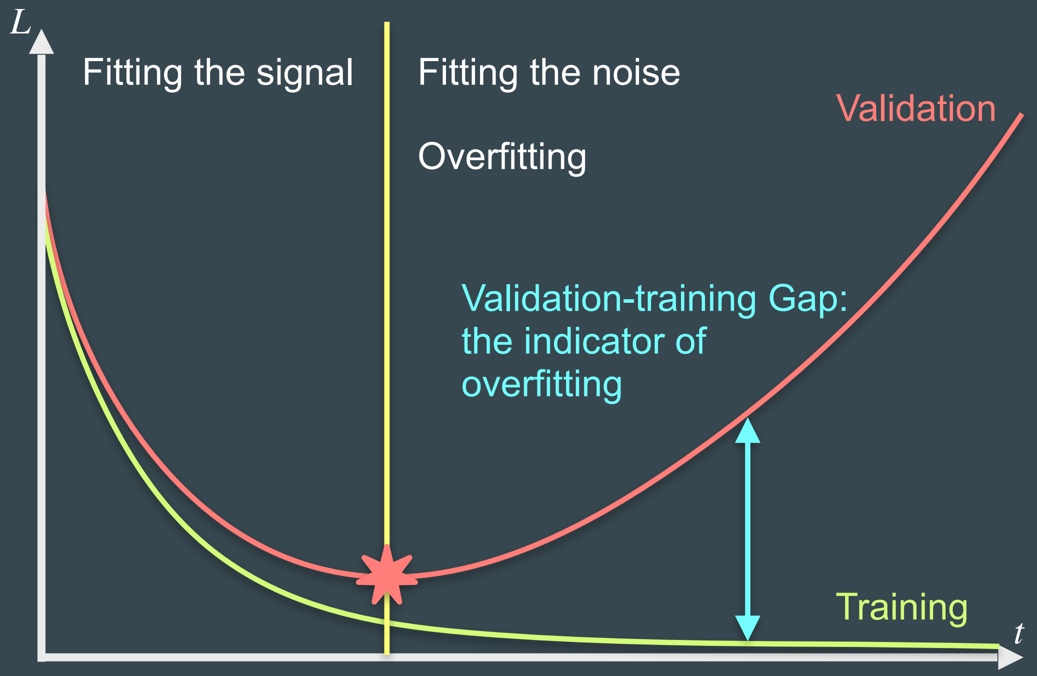
$$\mathbf{w}^*_{end} = (1-\gamma)\mathbf{w}_{start} + \gamma\,\mathbf{w}_{end}$$

Two ways:

- γ Is constant
- γ Is adjusted on-the-fly

Scheduling

Training procedure: how the process goes



The main objective is not to make a zero Train-Valid Gap!

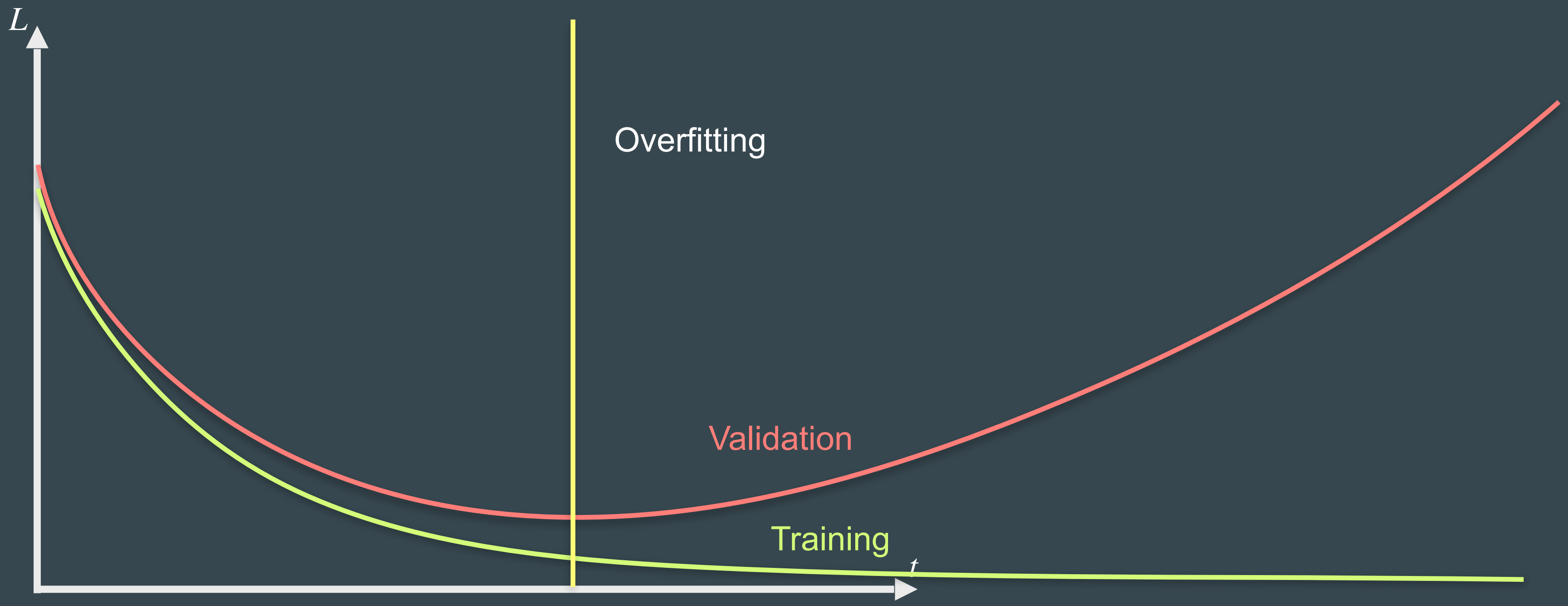
It is to get the best loss/metric value for validation

But the Gap is a compass:

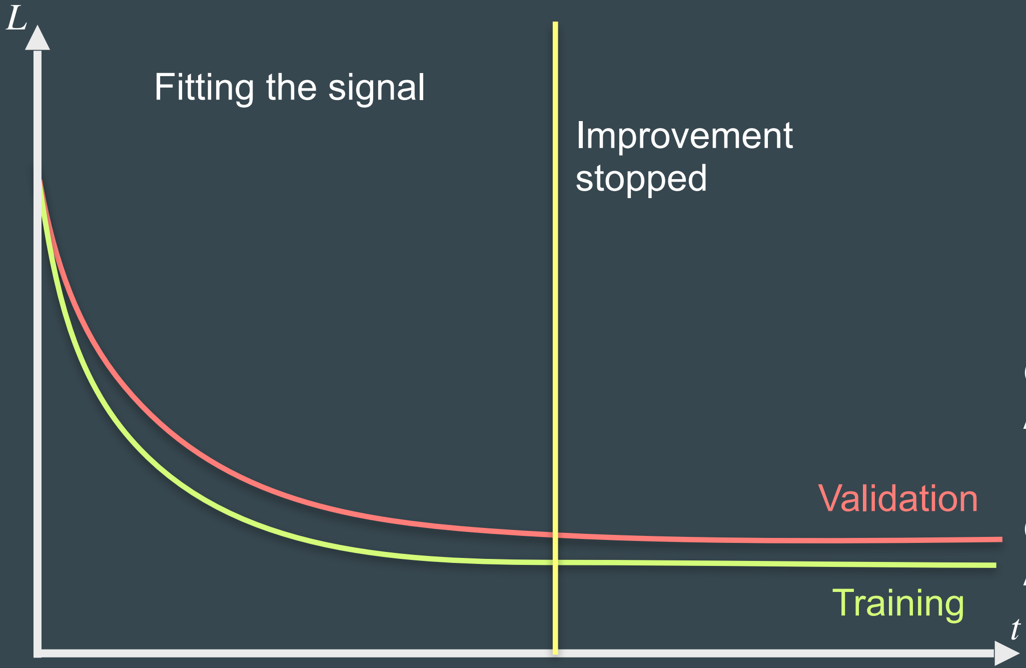
- No gap = underfitting
- Big gap = overfitting

Q: How to stop at the best point?
A: We can save best model and last model

What if we reduce LR?



What if we increase LR?



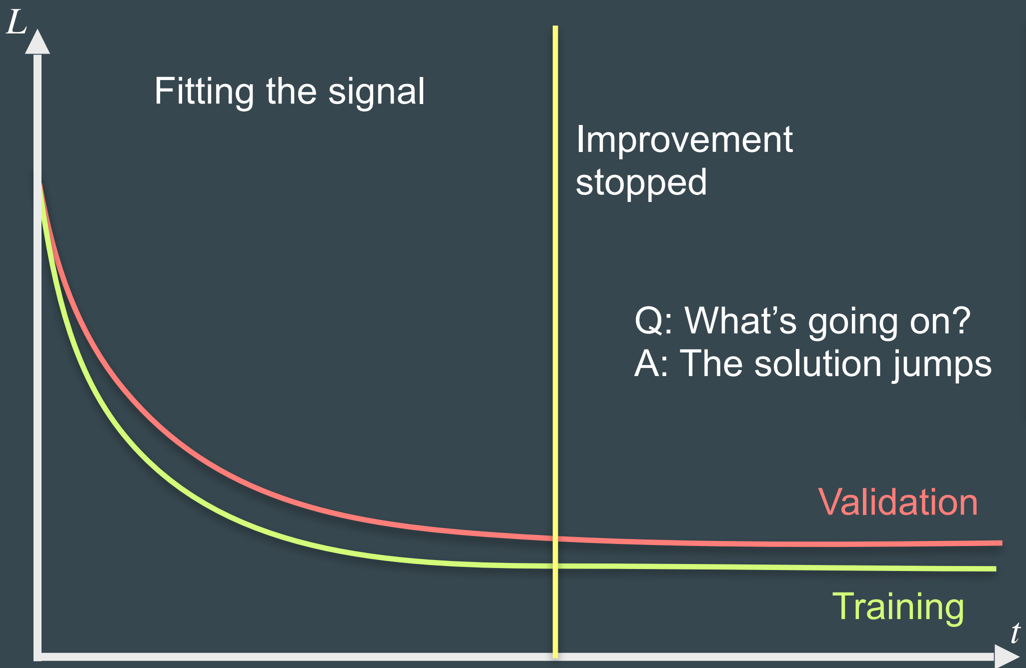
Q: What’s going on?
A: The solution jumps

Q: What’s going on?
A: The solution jumps

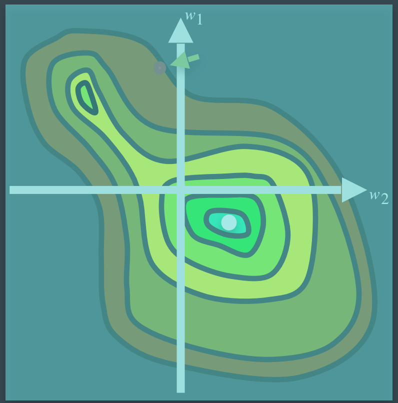
What if we increase LR?

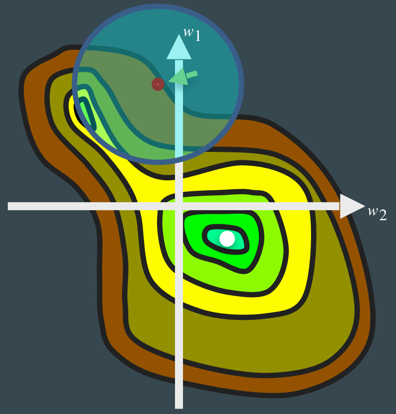
$$\alpha_t = \frac{\alpha_0}{t} \qquad \alpha_t = \frac{\alpha_0}{t^2} \qquad \alpha_t = \frac{\alpha_0}{2^t}$$

$$\sum_{t=1}^{\infty} \alpha_t = \infty \qquad \sum_{t=1}^{\infty} \alpha_t = C \qquad \sum_{t=1}^{\infty} \alpha_t = C$$

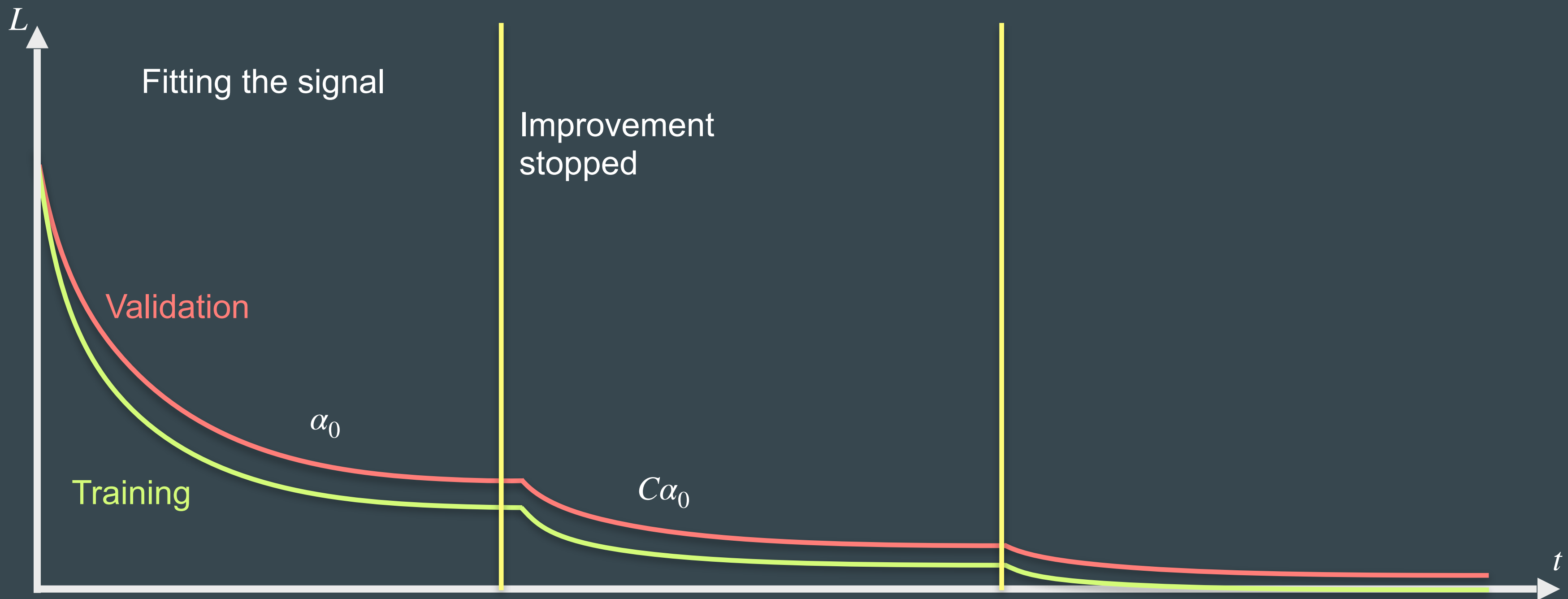


Q: What's going on?
A: The solution jumps





ReduceLR On Plateau



Summary

- Gradient Descent:
  - GD
  - SGD
  - SGD with momentum
- RMSprop
  - GD with equal steps
  - With momentum: Adam
- Look Ahead
- Scheduling
  - ReduceLROnPlateau