# Optimization

# Outline

- Recap minibatch SGD
- Discuss semester timeline
- Some prerequisites (SGD, contour plots, EMA)
- Momentum
- Adaptive learning rate
  1. AdaGrad (Adaptive Gradients)
  2. RMSProp (Root-Mean-Square Backpropagation)
  3. Adam (Adaptive Moment Estimation)

# Recap: Minibatch SGD

- Take five minutes to draw
  - Whatever will help you remember (no correct or incorrect drawings)
  - You'll keep a running drawing log the rest of the semester

# Semester vs. Topic Timelines

## Foundations

- Math
  - Calculus
  - Linear algebra
- Programming
  - Python
  - Notebooks
  - Libraries
- Computing
  - HPC
  - CLI

## NN Basics

- Terminology
- History
- Ethics
- Neurons
- Networks
- Backpropagation
- Autodiff
- Gradient descent

Done!

## NN Intermediate

- Optimization
- Initialization
- Normalization
- Overfitting
- Convolutions
- Recurrent

## NN Advanced

- Transfer learning
- Inference
- GANs
- Reinforcement
- Attention
- Transformers
- Stable diffusion

# MB-SGD Equations

W, b
parameters
↓
$\theta$

updated params
current params
parameter gradients
$g_t$

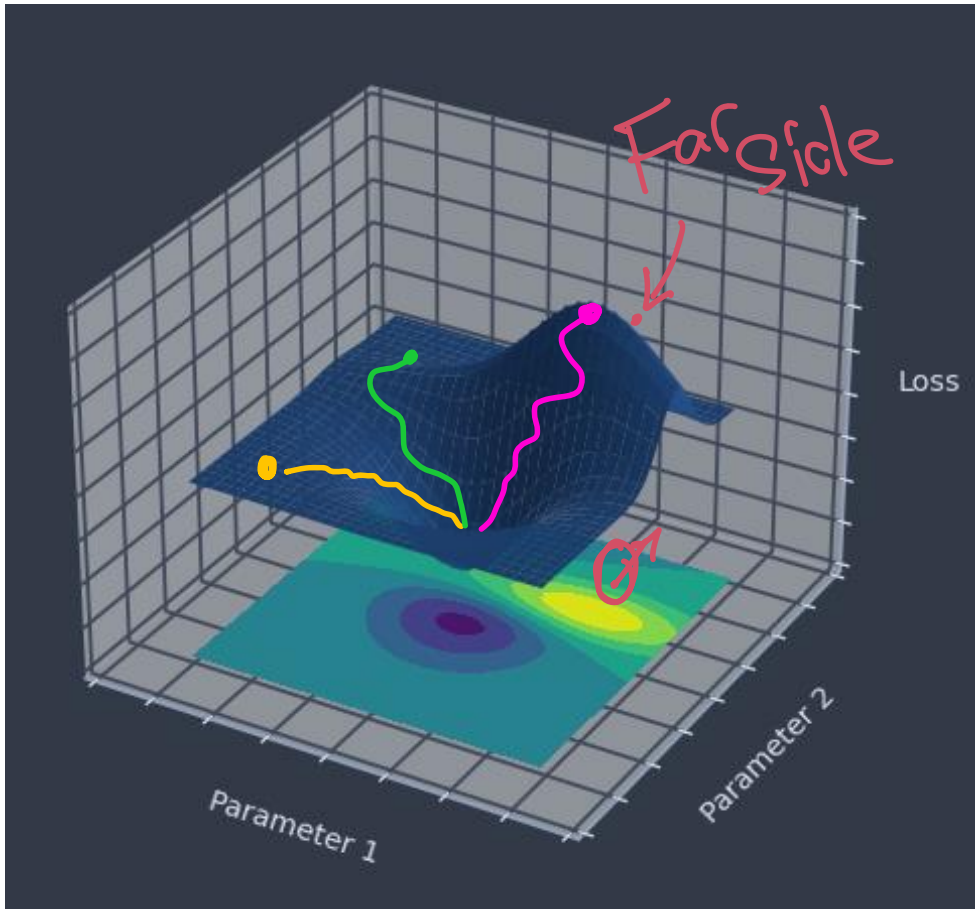
$$\theta_{t+1} := \theta_t - \alpha \frac{\nabla \mathcal{L}(\hat{y}_b, y_b)}{\nabla \theta_t}$$

assignment
learning rate

```
param -= learning_rate * param.grad
```

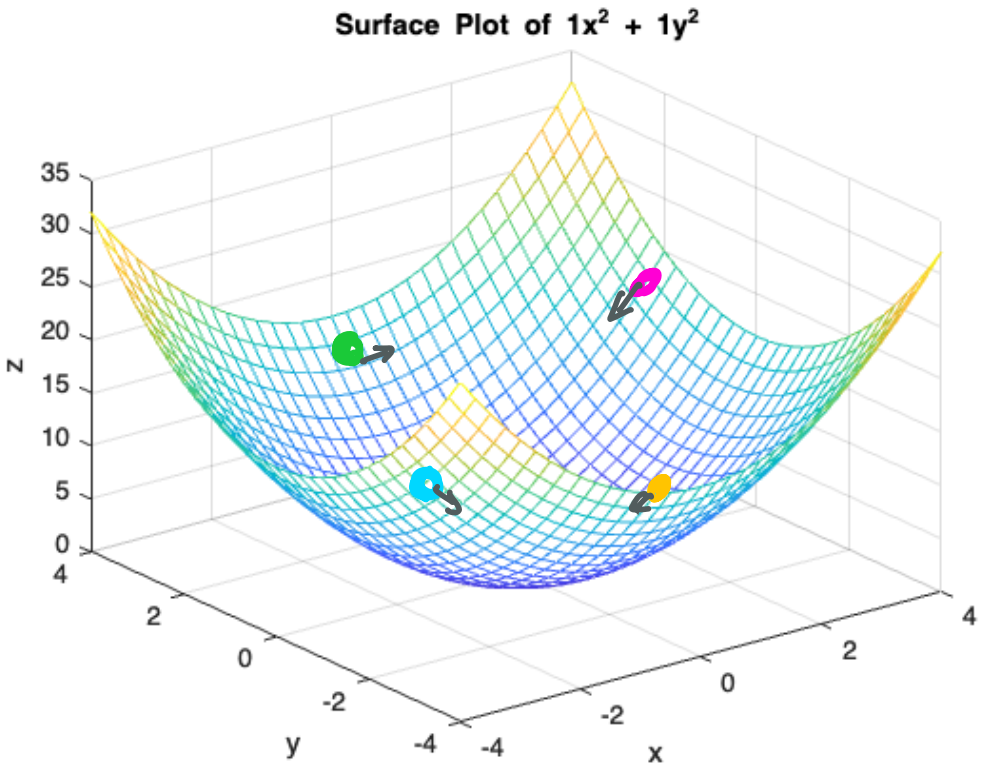
How do we choose a learning rate?

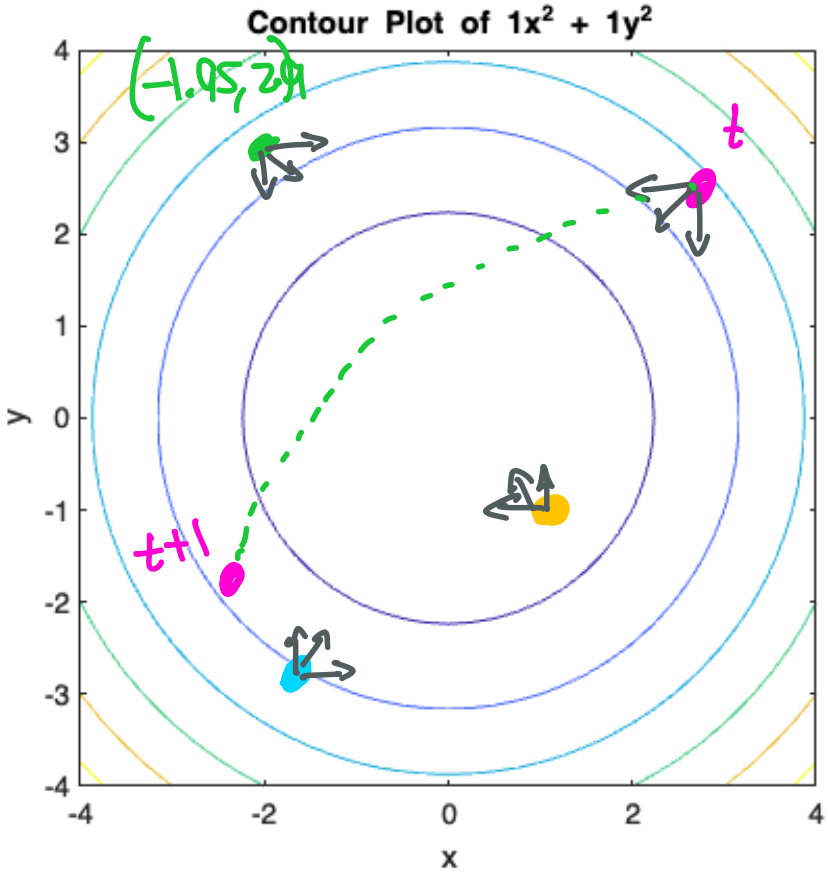
# Contour Plots



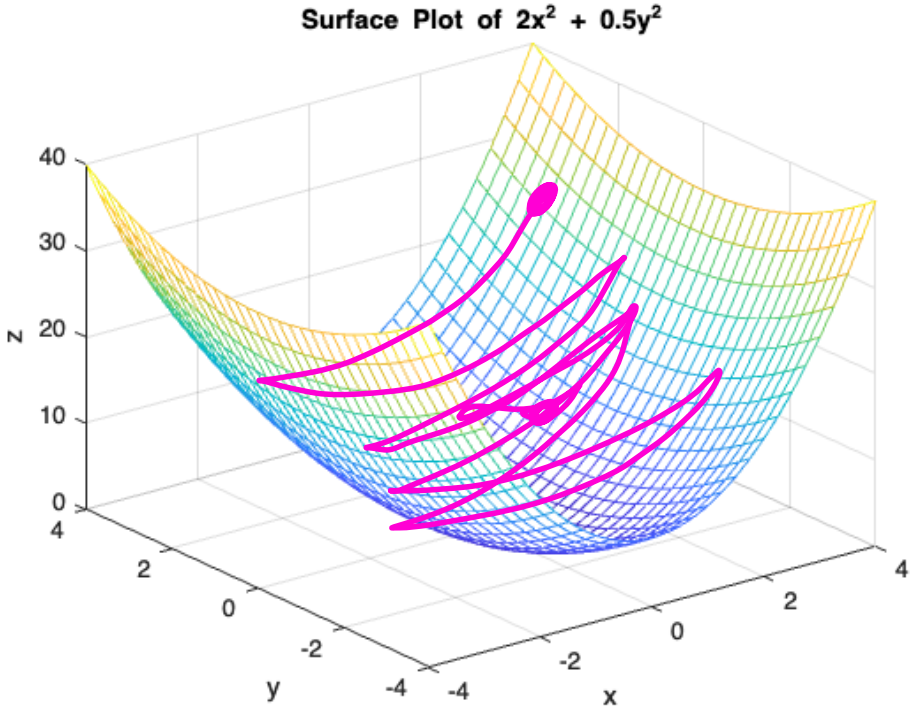
# Uniform Gradients

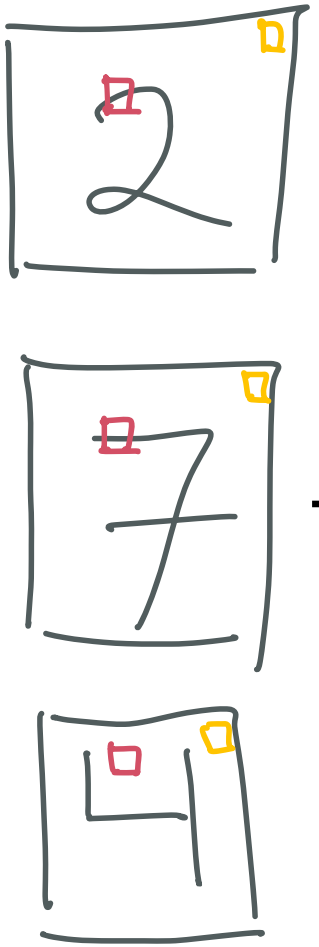
$\theta_{t+1} := \theta_t - \alpha g_t$



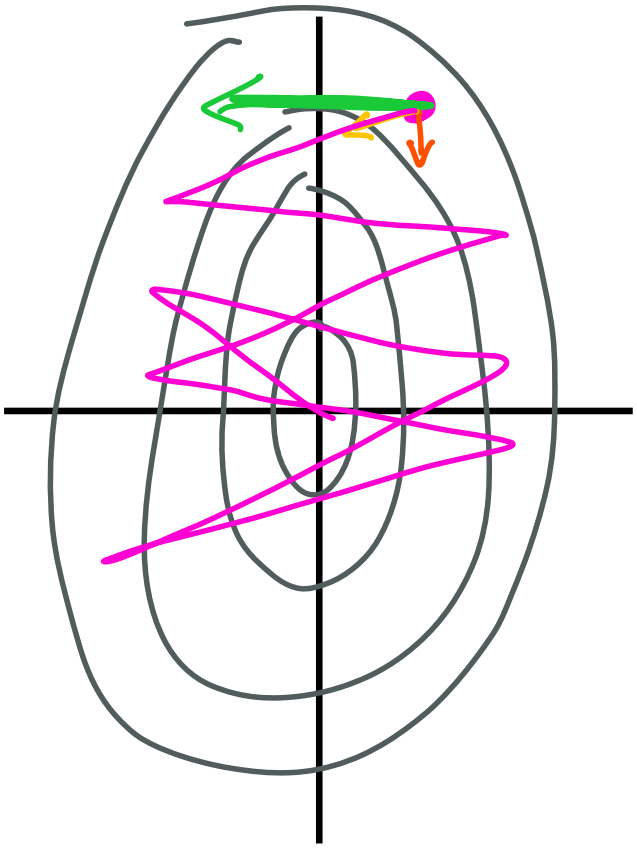



# Contour Plots



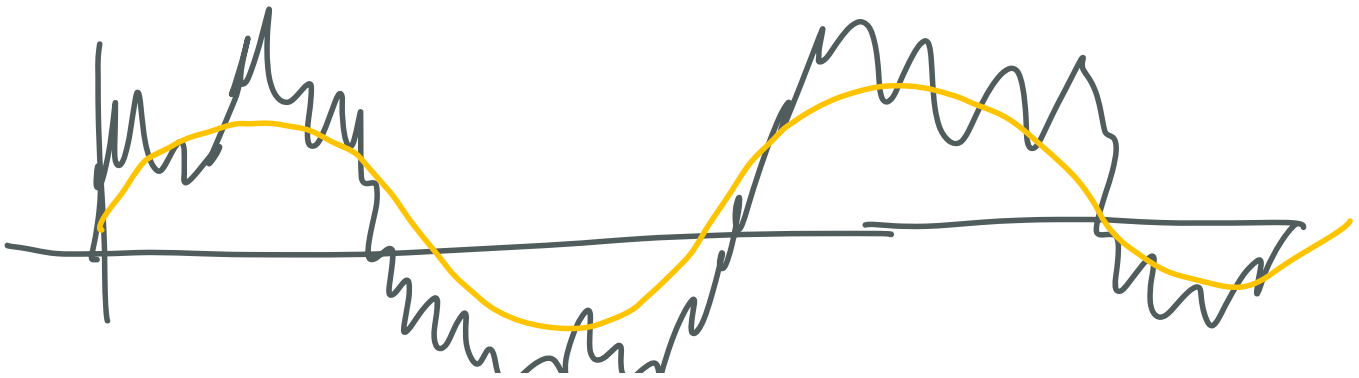
Contour Plot



# Exponential Moving Average (EMA)

- A simple moving average is the unweighted mean of the previous k values
- EMA is a rule-of-thumb technique for smoothing time series data

smoothed value → $\bar{y}_{t+1} := \beta \bar{y}_t + (1-\beta) y_t$ ← next value in the sequence

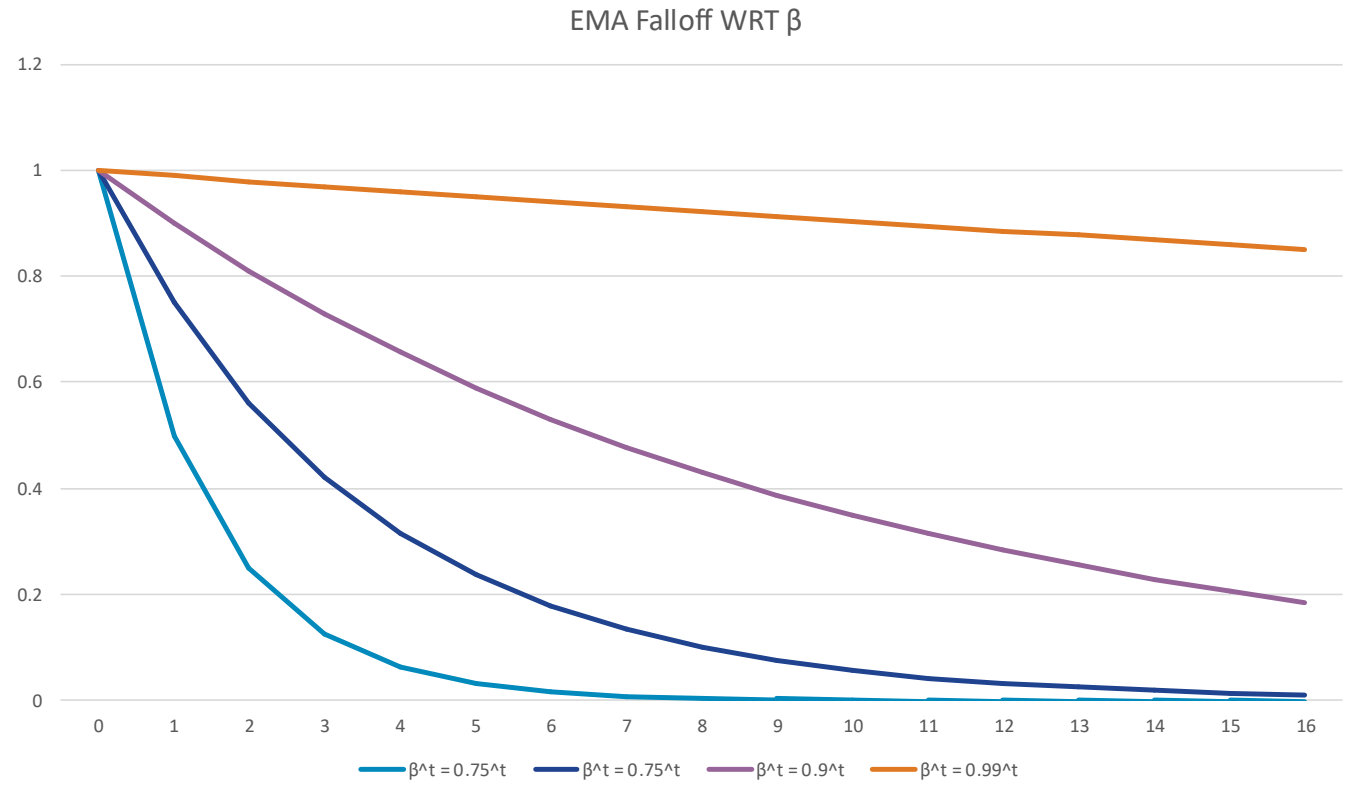


# Exponential Moving Average (EMA)

- https://cs.pomona.edu/classes/cs181r/book/19-SensorFusion.html

# Exponential Moving Average (EMA)

- A simple moving average is the unweighted mean of the previous k values
- EMA is a rule-of-thumb technique for smoothing time series data

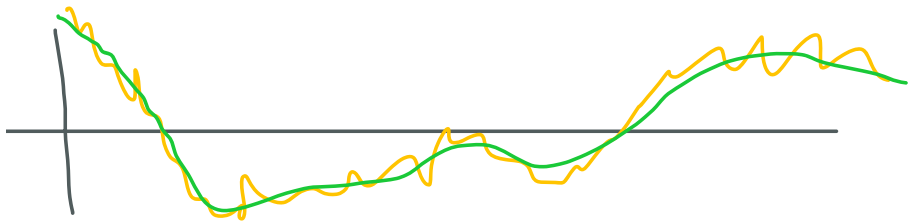


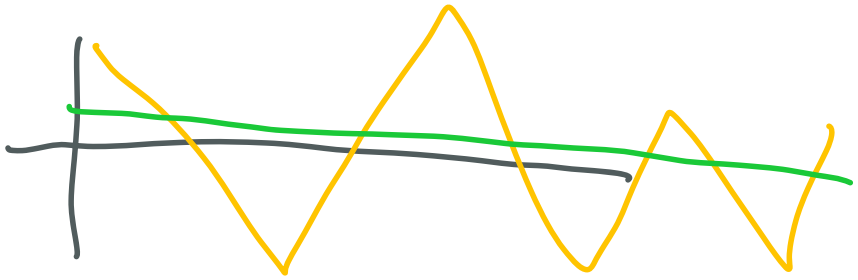
$$\bar{y}_{t+1} := \beta \bar{y}_t + (1-\beta) y_t \qquad 0 < \beta < 1$$

$$\bar{y}_{t+1} := \beta\left(\beta \bar{y}_{t-1} + (1-\beta) y_{t-1}\right) + (1-\beta) y_t$$

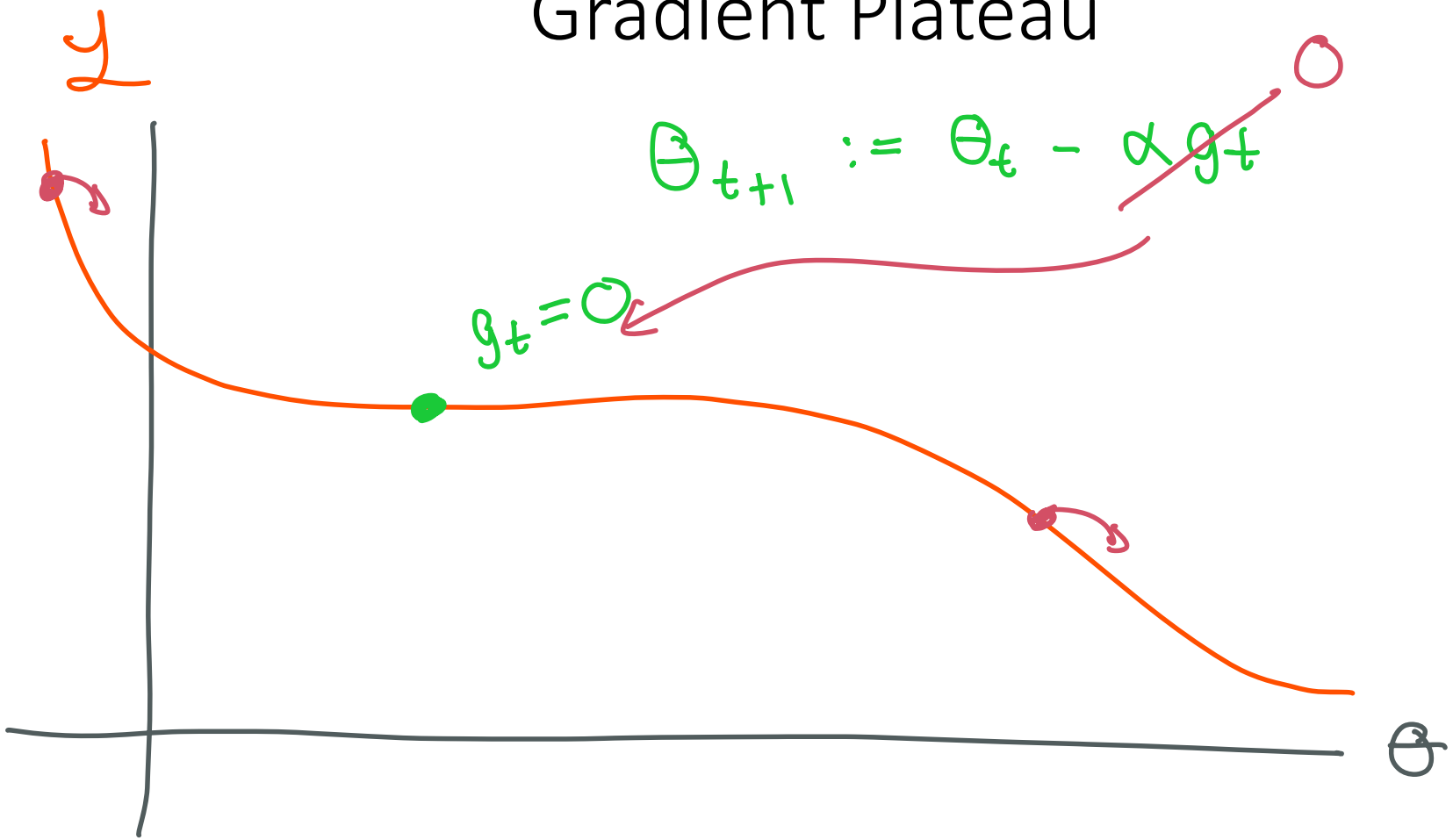
$$\bar{y}_{t+1} := \beta\left(\beta\left(\beta \bar{y}_{t-2} + (1-\beta) y_{t-2}\right) + (1-\beta) y_{t-1}\right) + (1-\beta) y_t$$

$$:= \beta^3 \bar{y}_{t-2} + \beta^2 (1-\beta) y_{t-2} + \beta (1-\beta) y_{t-1} + (1-\beta) y_t$$

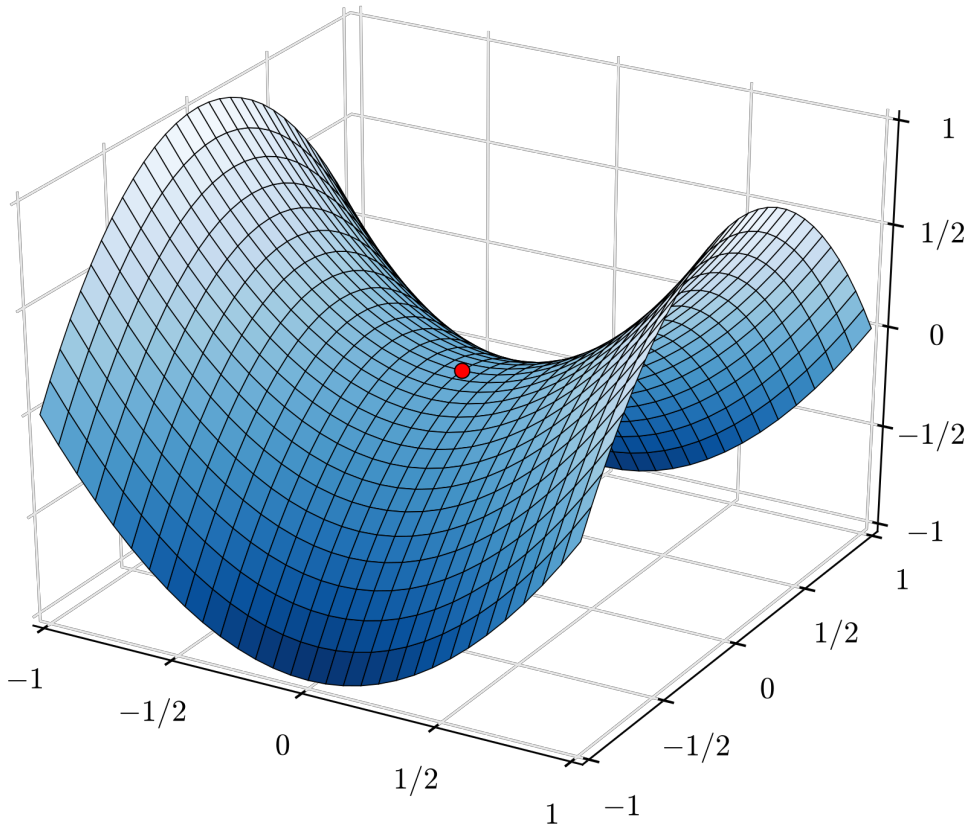




# Gradient Plateau



# Saddle Points



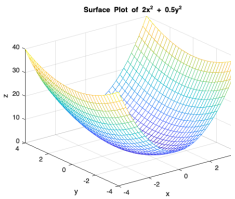
# Momentum

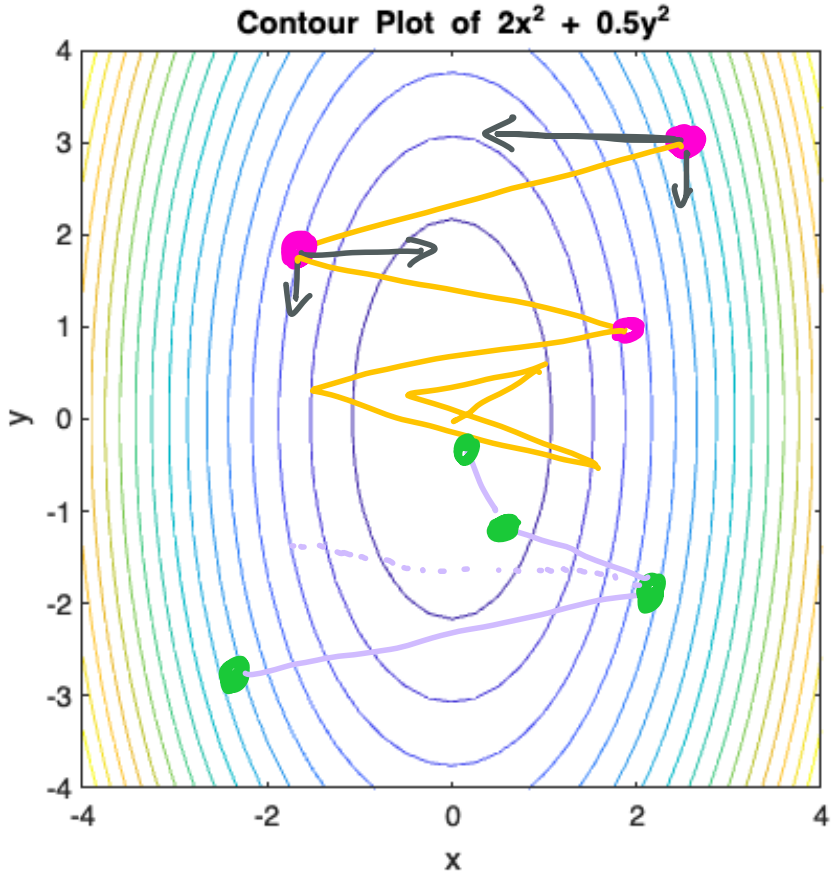
exponential moving average

$$m_{t+1} := \beta_m m_t + (1-\beta_m) g_t$$

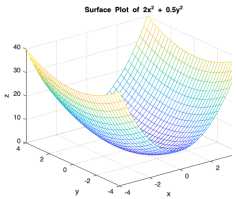
$$\beta_m m_{t-1} + (1-\beta_m) g_{t-1}$$

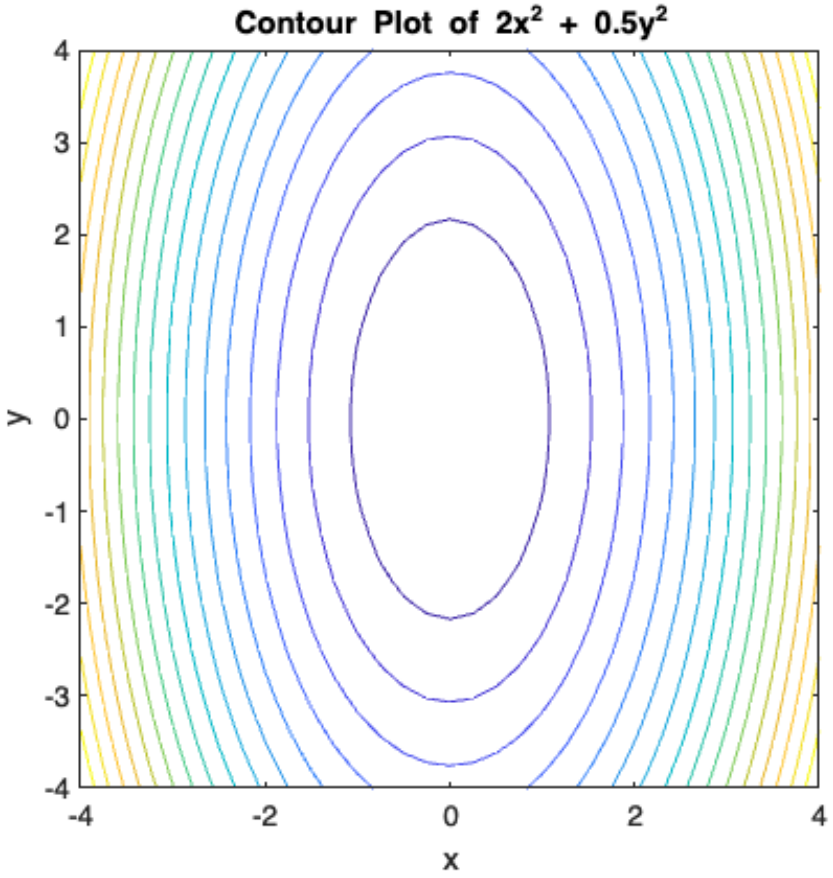
$$\theta_{t+1} := \theta_t - \alpha m_{t+1}$$



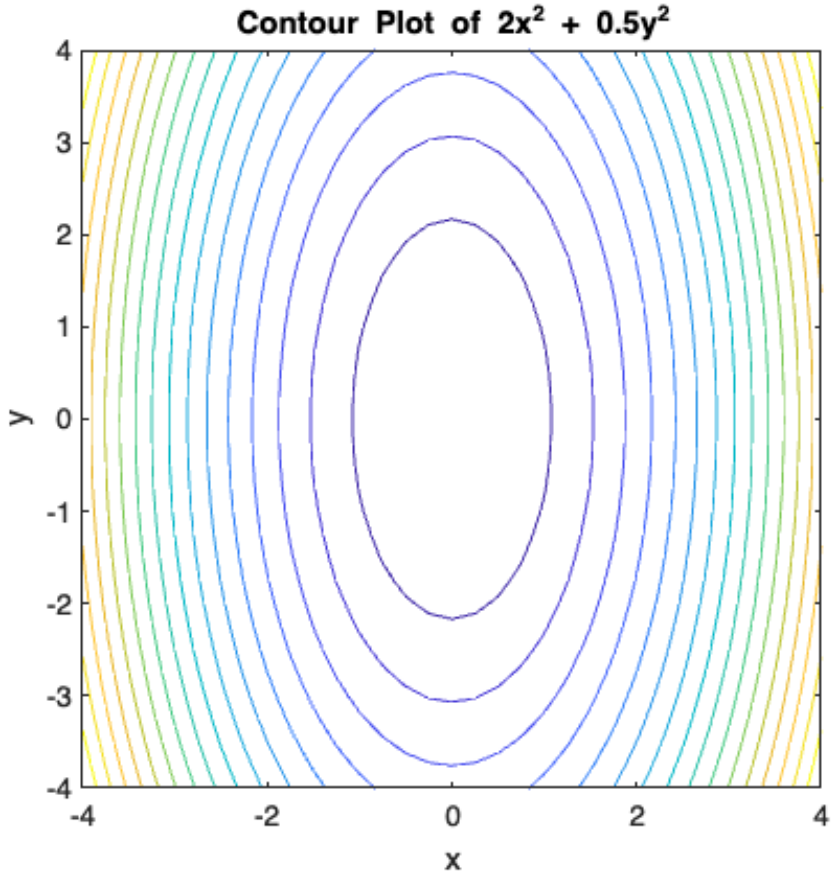



# Momentum



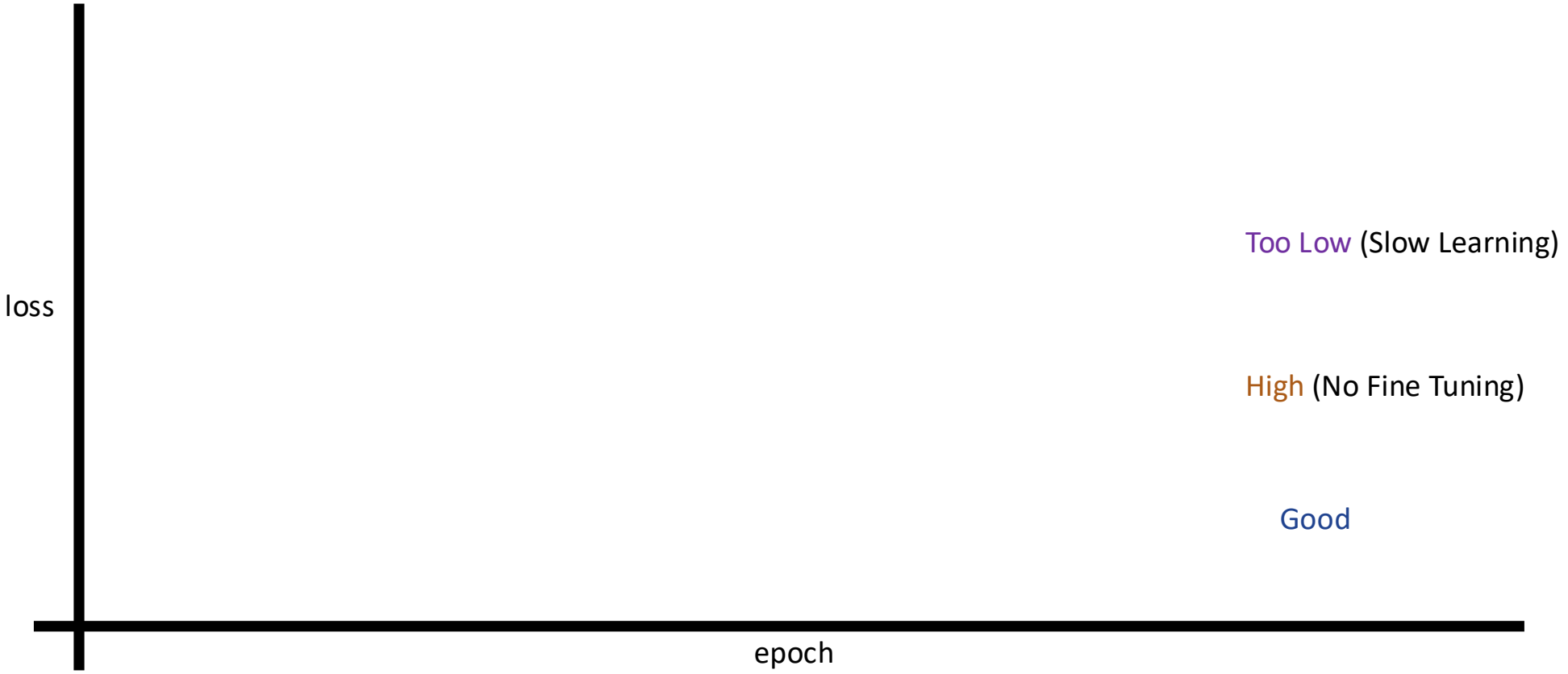

Without Momentum




With Momentum

# Learning Rates

Too High (No Learning)



# AdaGrad

# RMSProp

$Z = A w^T + b \rightarrow$ Linear

$A = \sigma(Z) \rightarrow$ Sigmoid

# Adam

# Recent Optimizers

- SGD (1951)
- SGD+Momentum (1999)
- AdaGrad (2011)
- AdaDelta (2012)
- RMSProp (2013)
- Adam (2014)
- NADAM (2015)
- AdamW (2017)
- AdaShift (2018)
- AggMo (2018)
- LAMB (2019)
- AMSGrad (2019)
- Adabelief (2020)
- MADGRAD (2021)
- AdaSmooth (2022)

Just a sample

See: https://johnchenresearch.github.io/demon/ for more information