COMP5211: Machine Learning

Lecture 3

Minhao Cheng

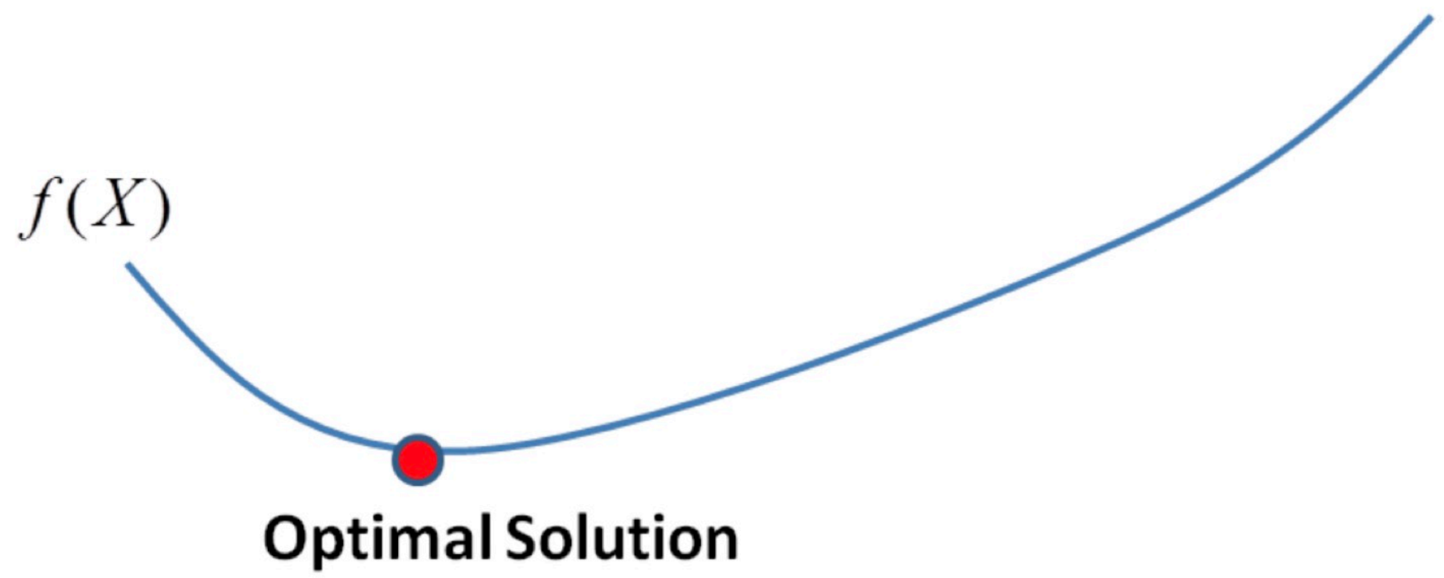
Logistics

- Form your group
    - Group registration: Due next Monday
        - Submit your team members & project title & project abstract
- Homework 1 will release this week

Optimization

Goal

- Goal: find the minimizer of a function
  - $min_w f(w)$
- For now we assume f is twice differentiable

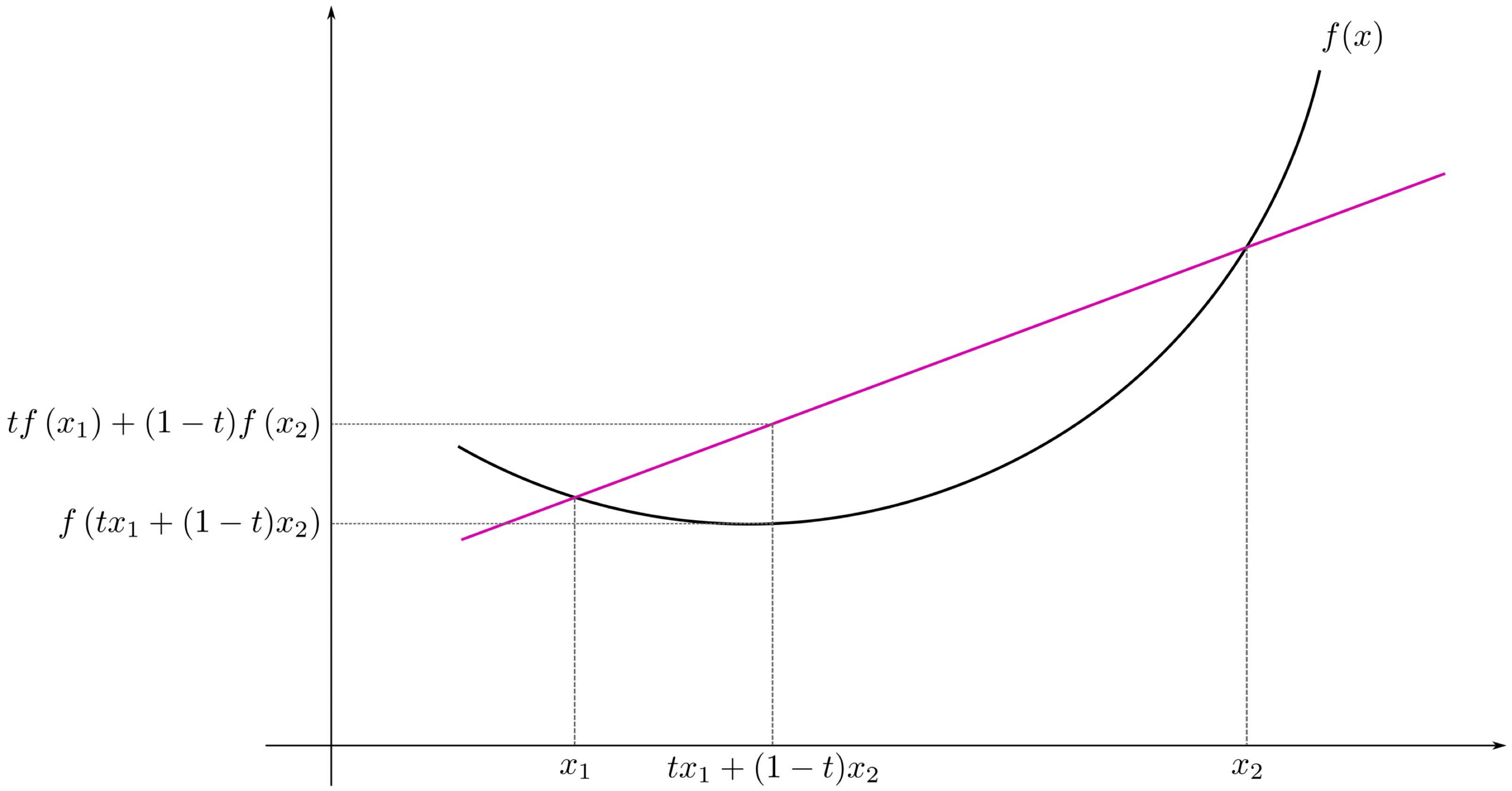


Optimization

Convex function

- A function $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function

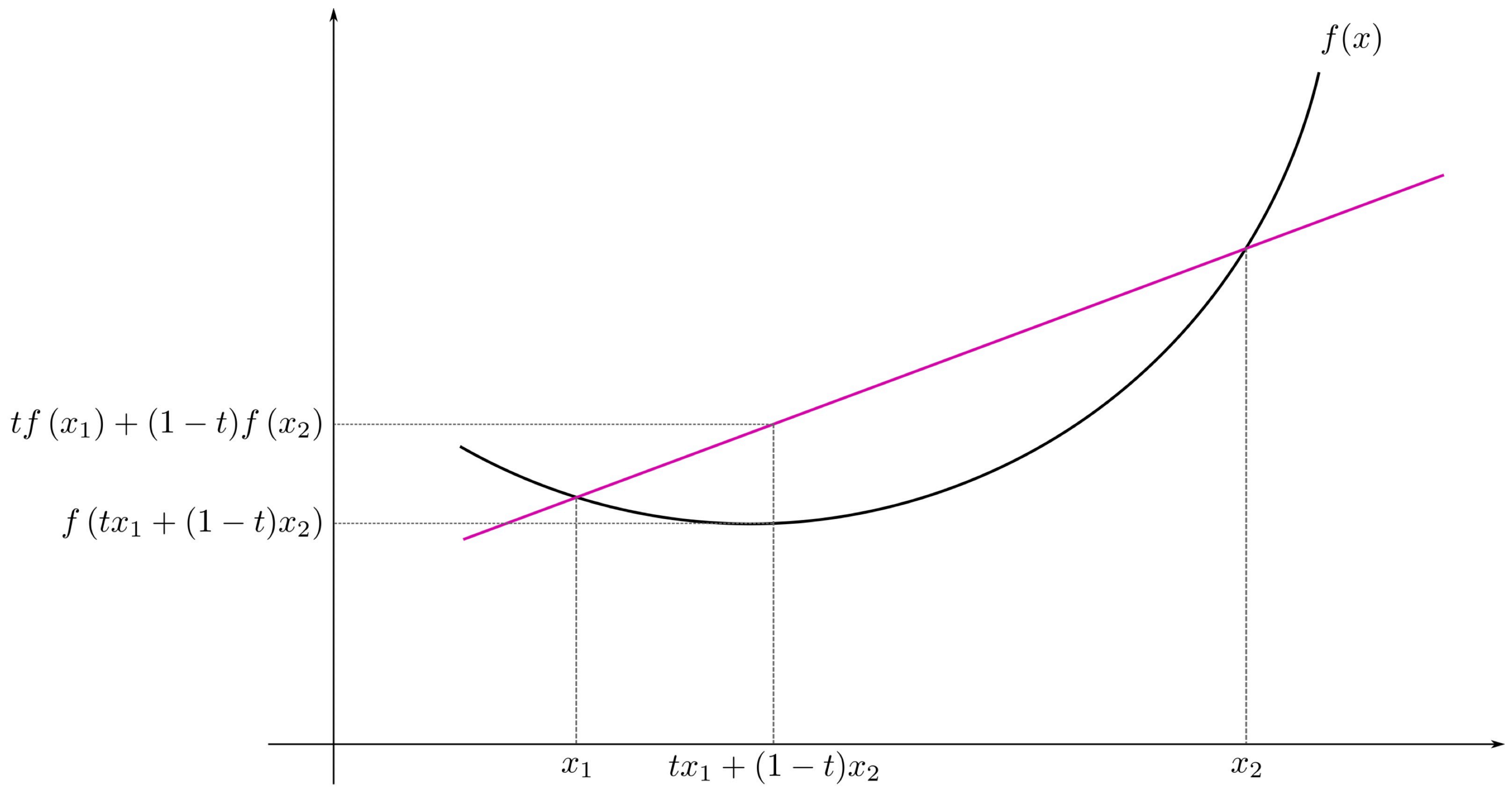
- $\Leftrightarrow$ the function f is below any line segment between two points on f:
    - $\forall x_1, x_2, \forall t \in [0,1]$,
    - $f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2)$



Optimization

Convex function

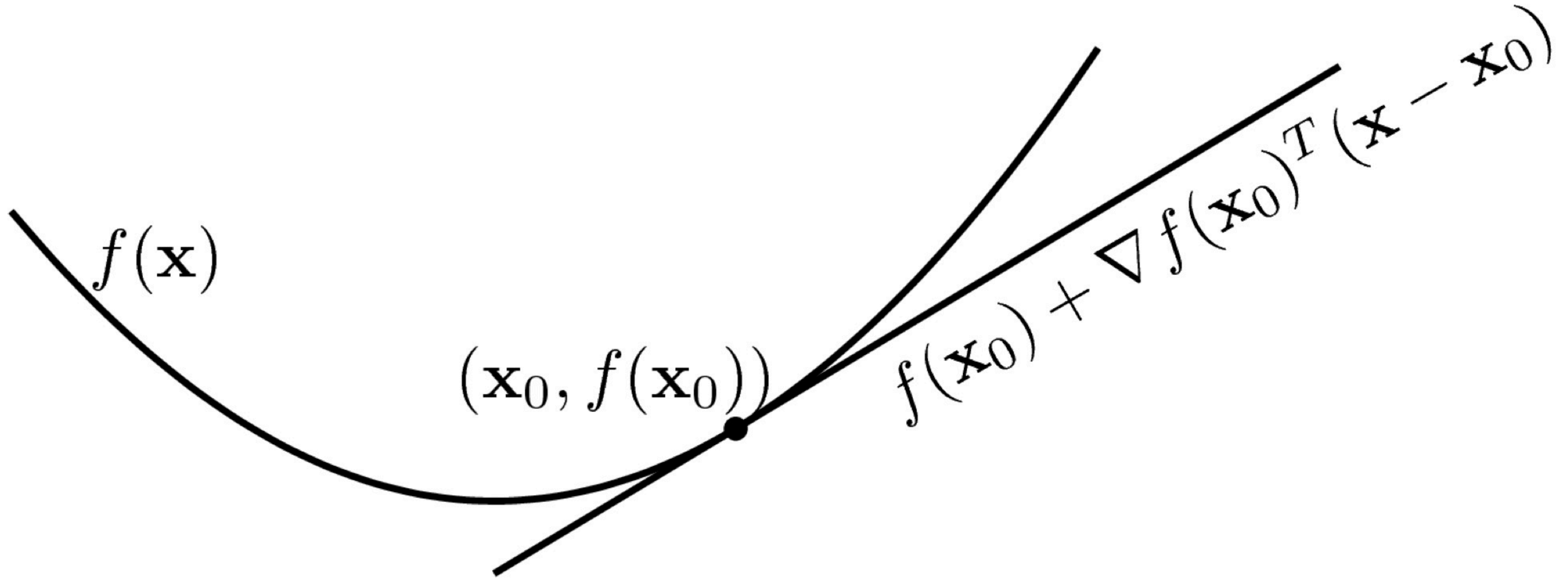
- A function $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function
- $\Leftrightarrow$ the function f is below any line segment between two points on f:
    - $\forall x_1, x_2, \forall t \in [0,1]$,
    - $f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2)$
- Strictly convex: $f(tx_1 + (1-t)x_2) < tf(x_1) + (1-t)f(x_2)$



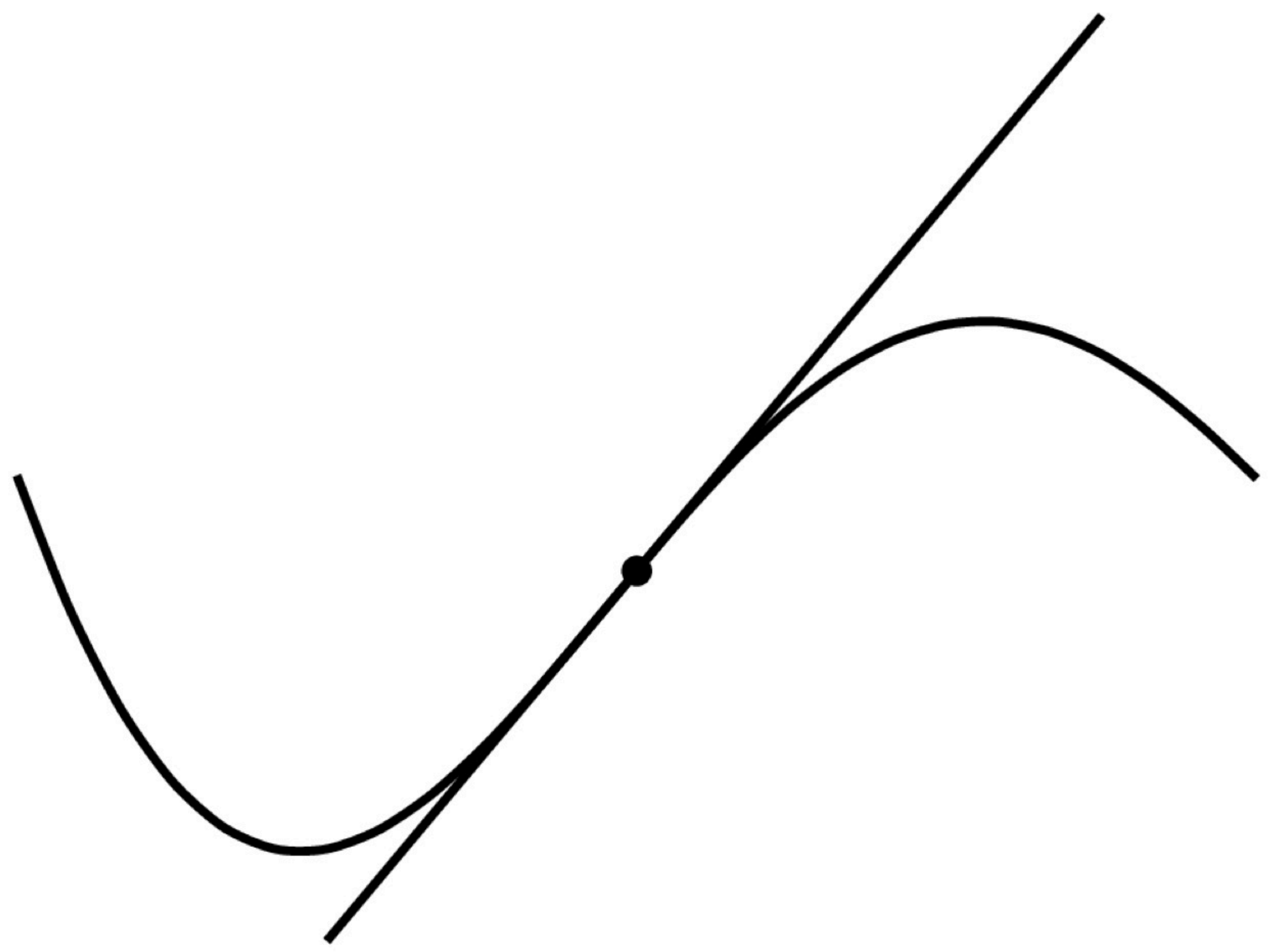
Optimization

Convex function

- Another equivalent definition for differentiable function:
  - f is convex if and only if $f(x) \geq f(x_0) + \nabla f(x_0)^T(x - x_0), \forall x, x_0$

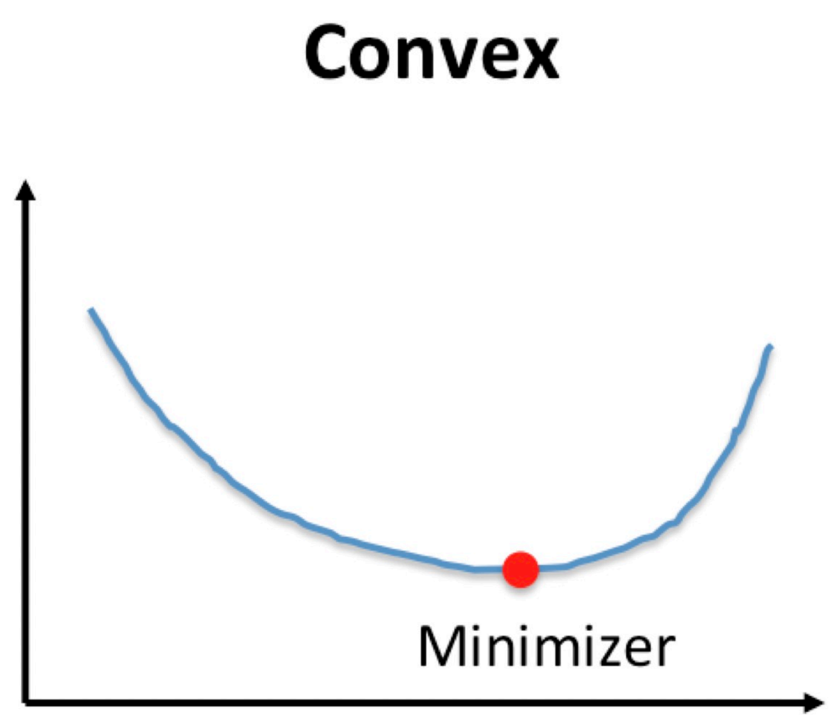


convex function



nonconvex function

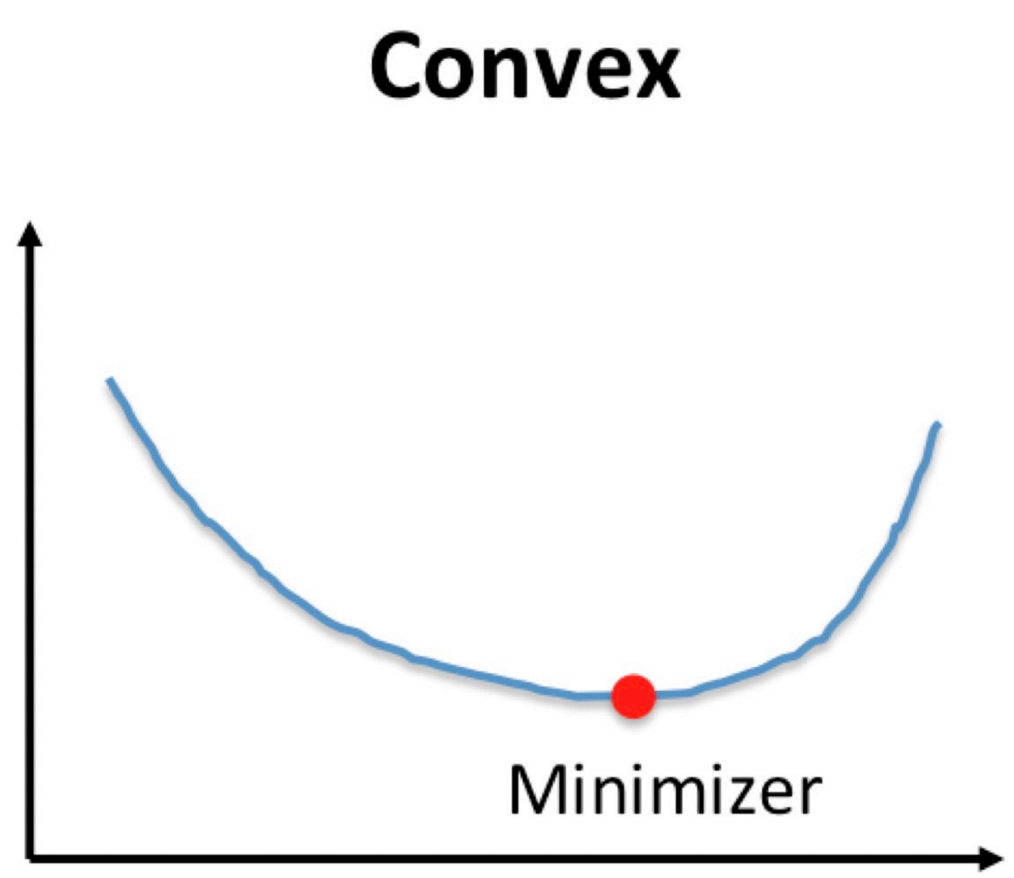
Optimization
Convex function



- Convex function:
  - (For differentiable function) $\nabla f(w^*) = 0 \Leftrightarrow w^*$ is a global minimum
  - If f is twice differentiable $\Rightarrow$
    - F is convex if and only if $\nabla^2 f(w)$ is **positive semi-definite**
    - Example: linear regression, logistic regression, …

Optimization

Convex function

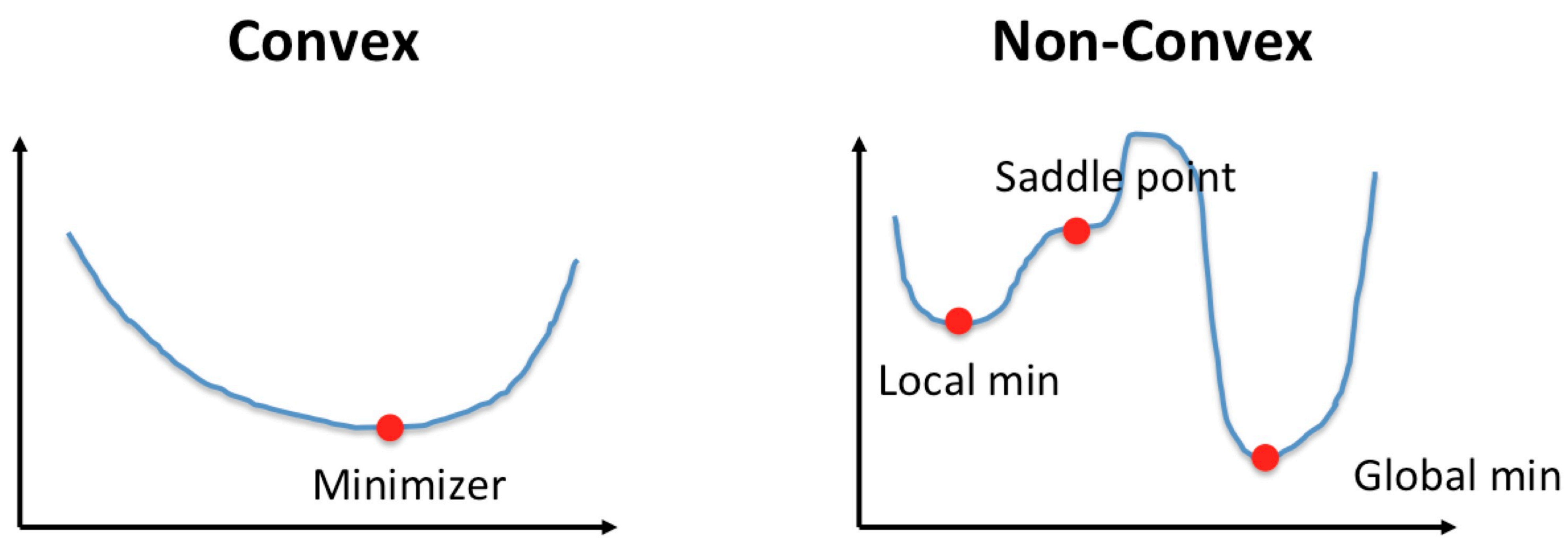


- Strict convex function:
  - $\nabla f(w^*) = 0 \Leftrightarrow w^*$ is the unique global minimum
    - Most algorithms only converge to gradient=0
    - Example: Linear regression when $X^T X$ is invertible

Optimization

Convex vs Nonconvex

- Convex function:
  - $\nabla f(x) = 0 \longleftrightarrow$ Global minimum
  - A function is convex if $\nabla^2 f(x)$ is positive definite
  - Example: linear regression, logistic rgression, …
- Non-convex function:
  - $\nabla f(x) = 0 \longleftrightarrow$ Global min, local min, or saddle point
    - Most algorithms only converge to gradient =0
    - Example: neural network, …



Optimization

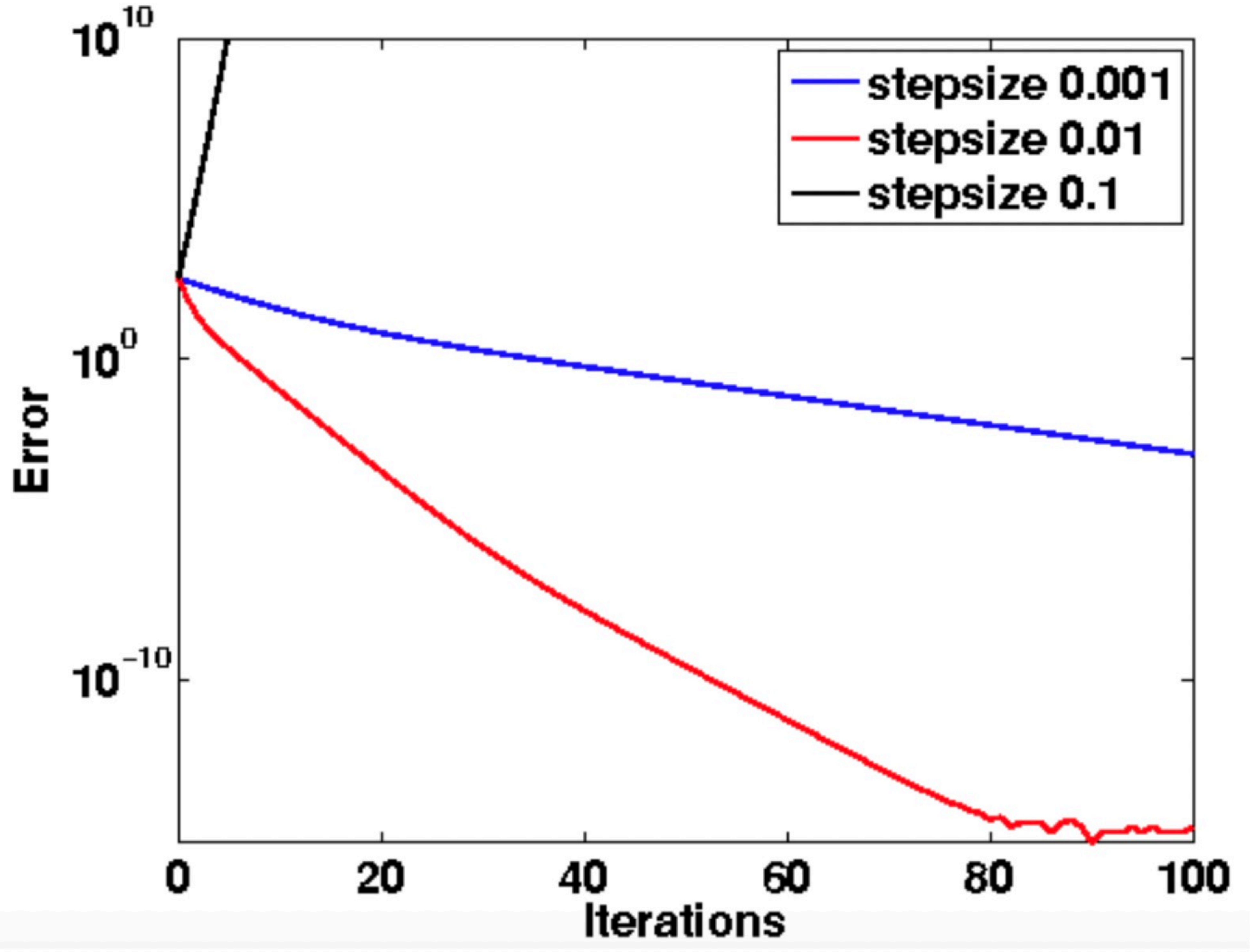
Gradient descent

- Gradient descent: repeatedly do
  - $w^{t+1} \leftarrow w^t - \alpha \nabla f(w^t)$
  - $\alpha > 0$ is the step size
- Generate the sequence $w^1, w^2, \ldots$
  - Converge to stationary points ($\lim_{t\to\infty} ||\nabla f(w^t)|| = 0$)

Optimization

Gradient descent

- Gradient descent: repeatedly do
  - $w^{t+1} \leftarrow w^t - \alpha \nabla f(w^t)$
  - $\alpha > 0$ is the step size
- Generate the sequence $w^1, w^2, \dots$
  - Converge to stationary points ($\lim_{t \to \infty} \|\nabla f(w^t)\| = 0$)
  - Step size too large $\Rightarrow$ diverge;
  - too small $\Rightarrow$ slow convergence



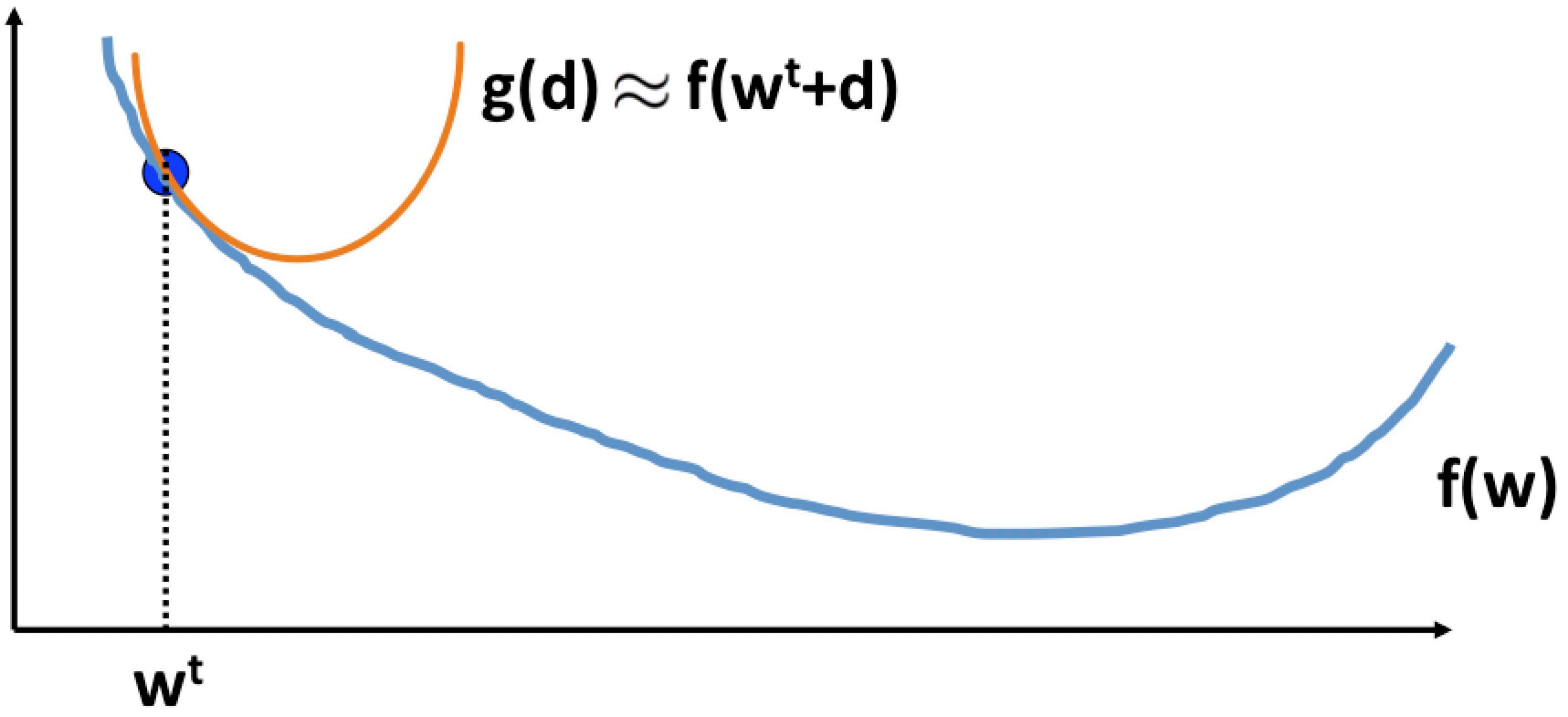
Optimization

Why gradient descent

- At each iteration, form a approximation function of $f(\cdot)$:
  - $f(w+d) \approx g(d) := f(w^t) + \nabla f(w^t)d + \frac{1}{2\alpha}\|d\|^2$
- Update solution by $w^{t+1} \leftarrow w^t + d*$
- $d* = \arg\min_d g(d)$
  - $\nabla g(d*) = 0 \Rightarrow \nabla f(w^t) + \frac{1}{\alpha}d* = 0 \Rightarrow d* = -\alpha \nabla f(w^t)$
- $d*$ will decrease $f(\cdot)$ if α (step size) is sufficiently small

Optimization

Illustration of gradient descent

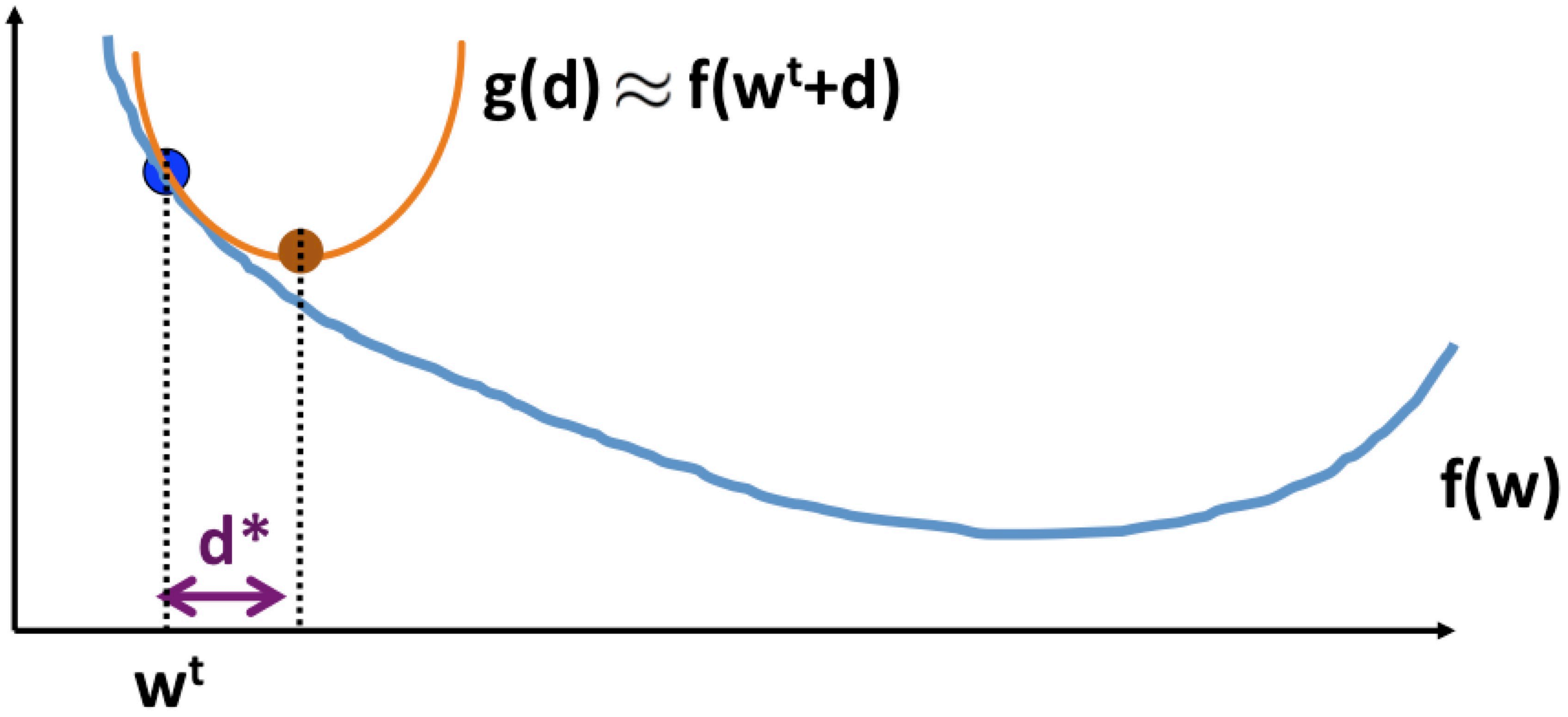


- Form a quadratic approximation

  - $f(w+\color{red}{d}) \approx g(\color{red}{d}) := f(w^t) + \nabla f(w^t)\color{red}{d} + \frac{1}{2\alpha}\|\color{red}{d}\|^2$

Optimization

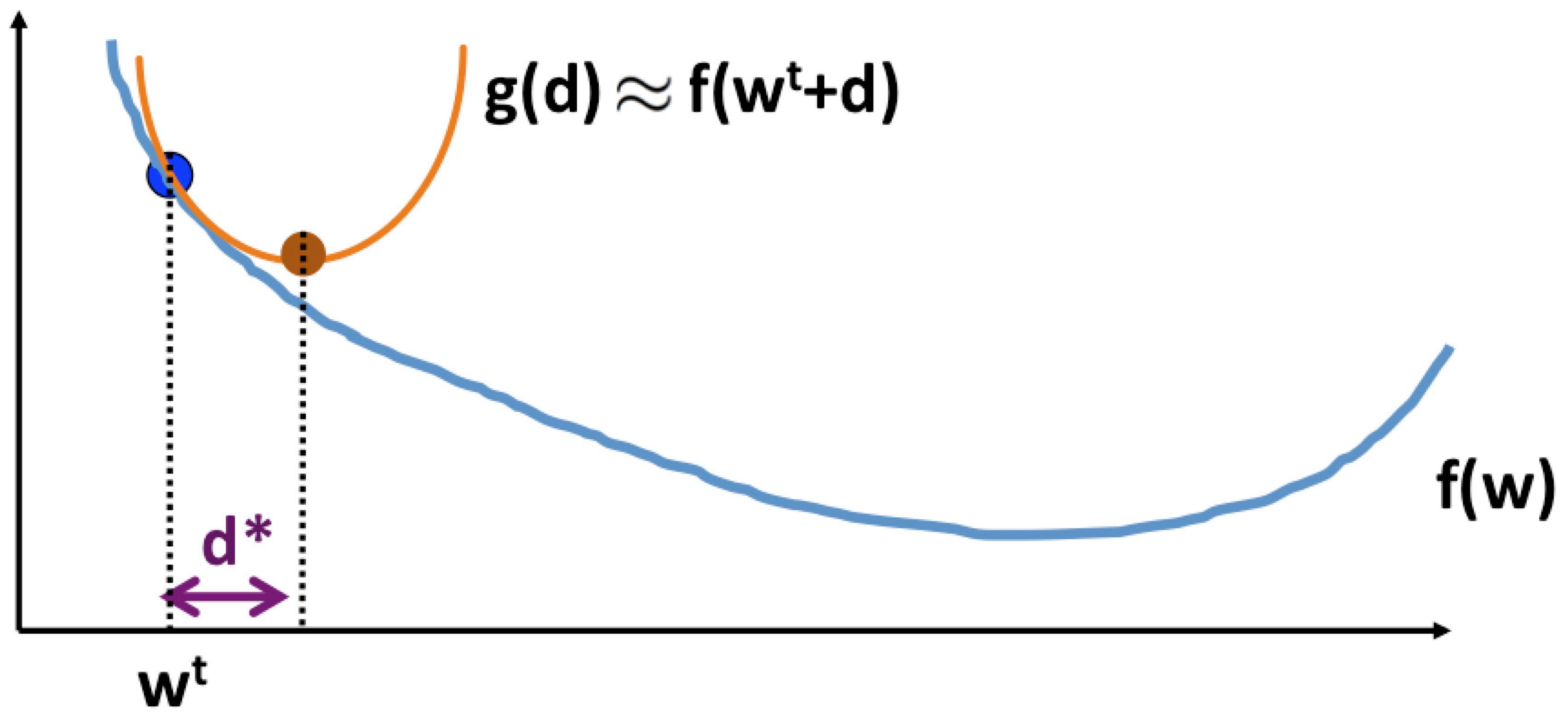
Illustration of gradient descent



- Minimize $g(d)$
  - $\nabla g(d^*) = 0 \Rightarrow \nabla f(w^t) + \frac{1}{\alpha} d^* = 0 \Rightarrow d^* = -\alpha \nabla f(w^t)$

Optimization

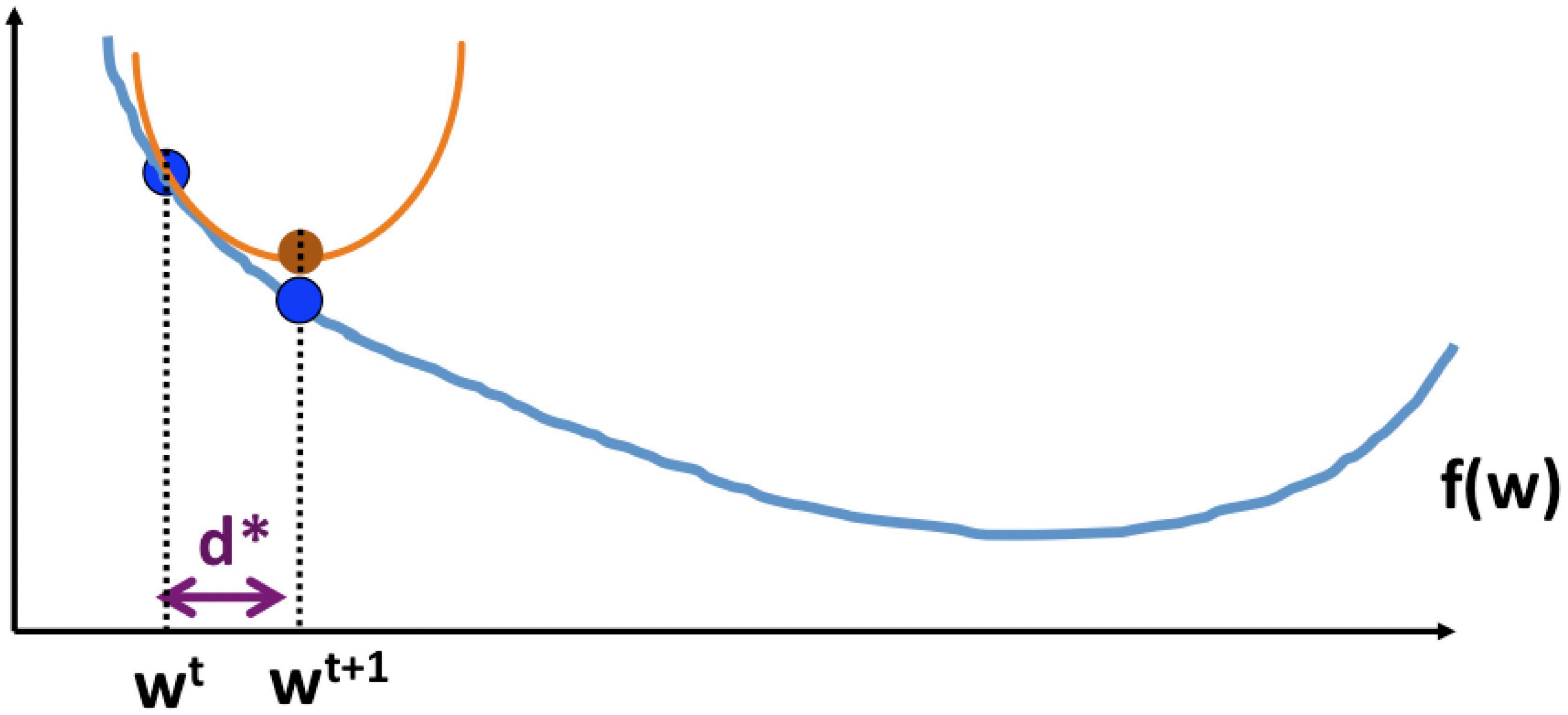
Illustration of gradient descent



- Update w
    - $w^{t+1} = w^t + d^* = w^t - \alpha \nabla f(w^t)$

Optimization

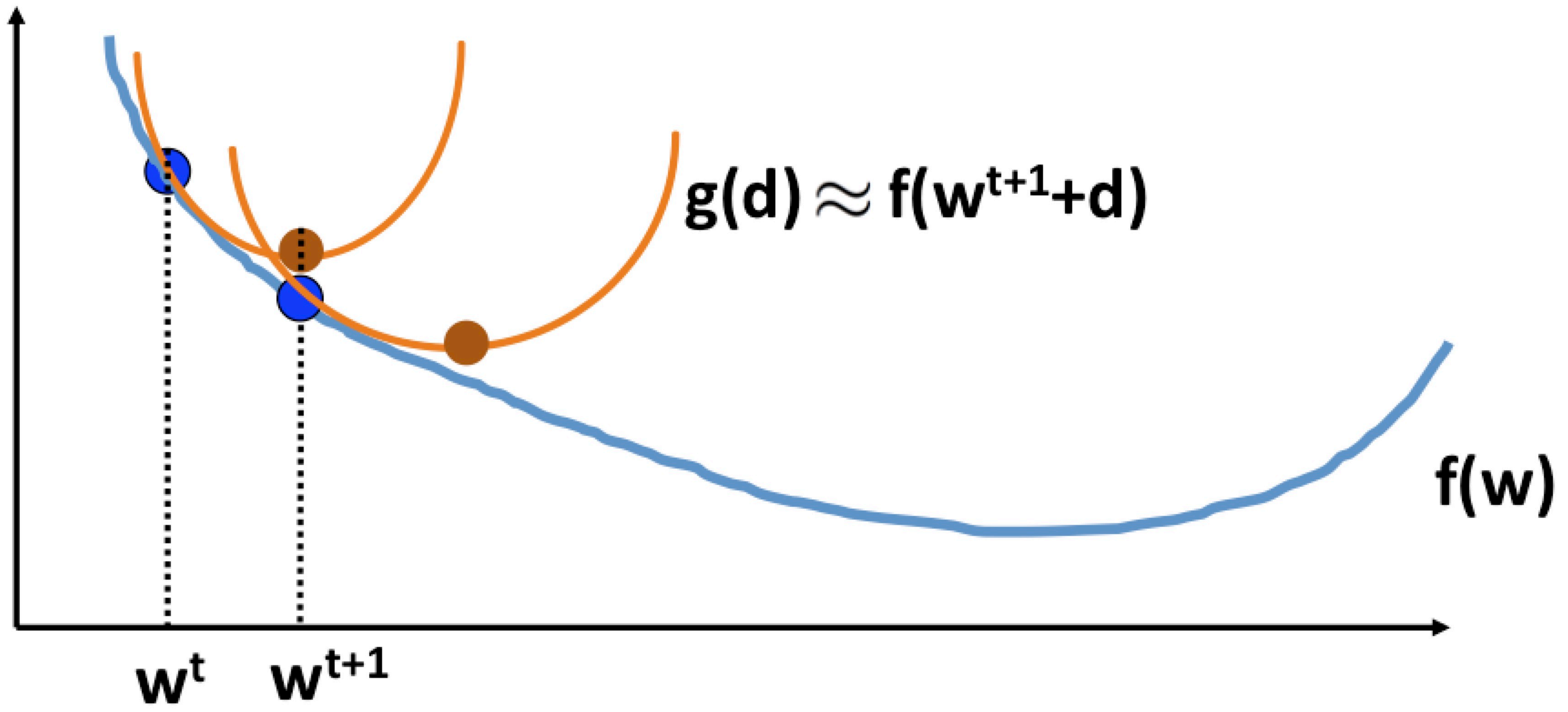
Illustration of gradient descent



- Update w
  - $w^{t+1} = w^t + d* = w^t - \alpha \nabla f(w^t)$

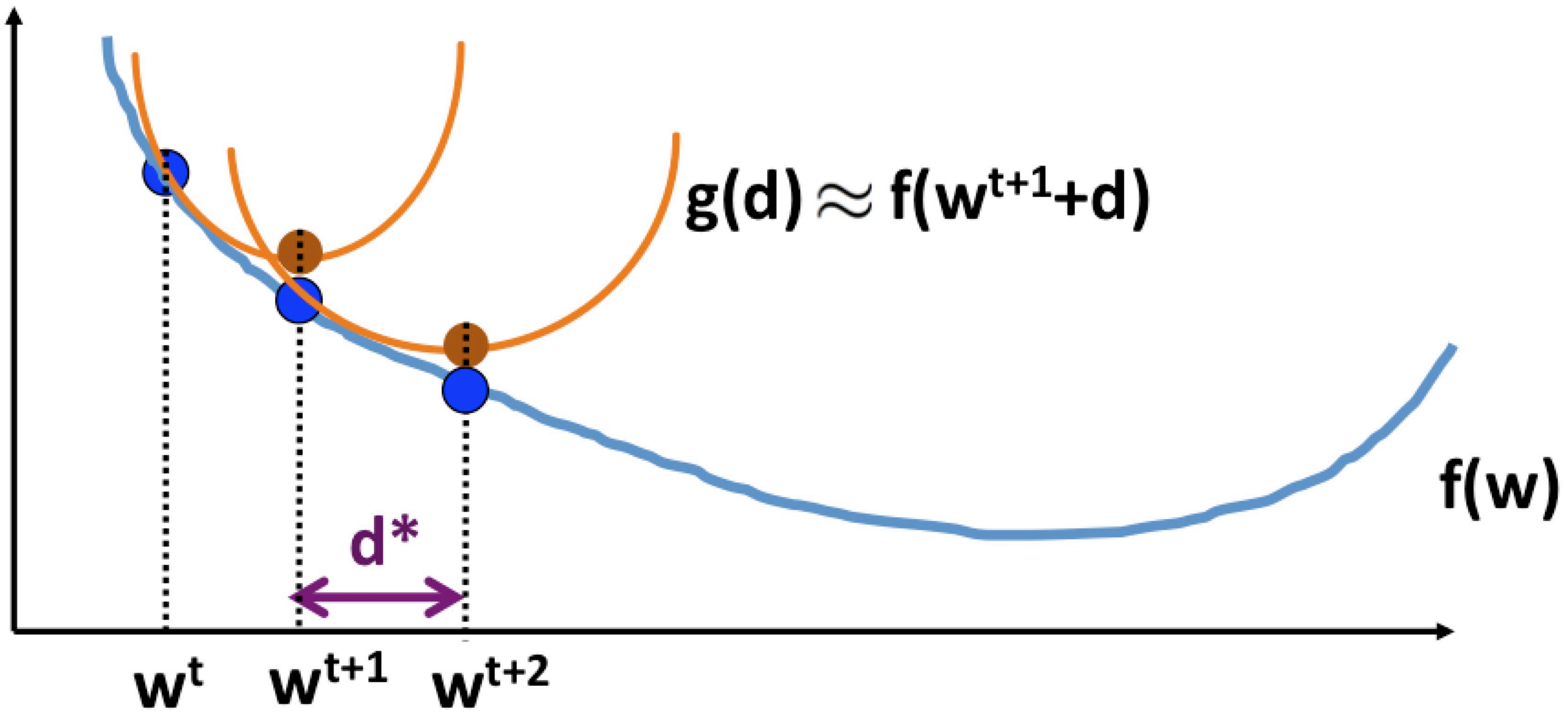
Optimization

Illustration of gradient descent



Optimization

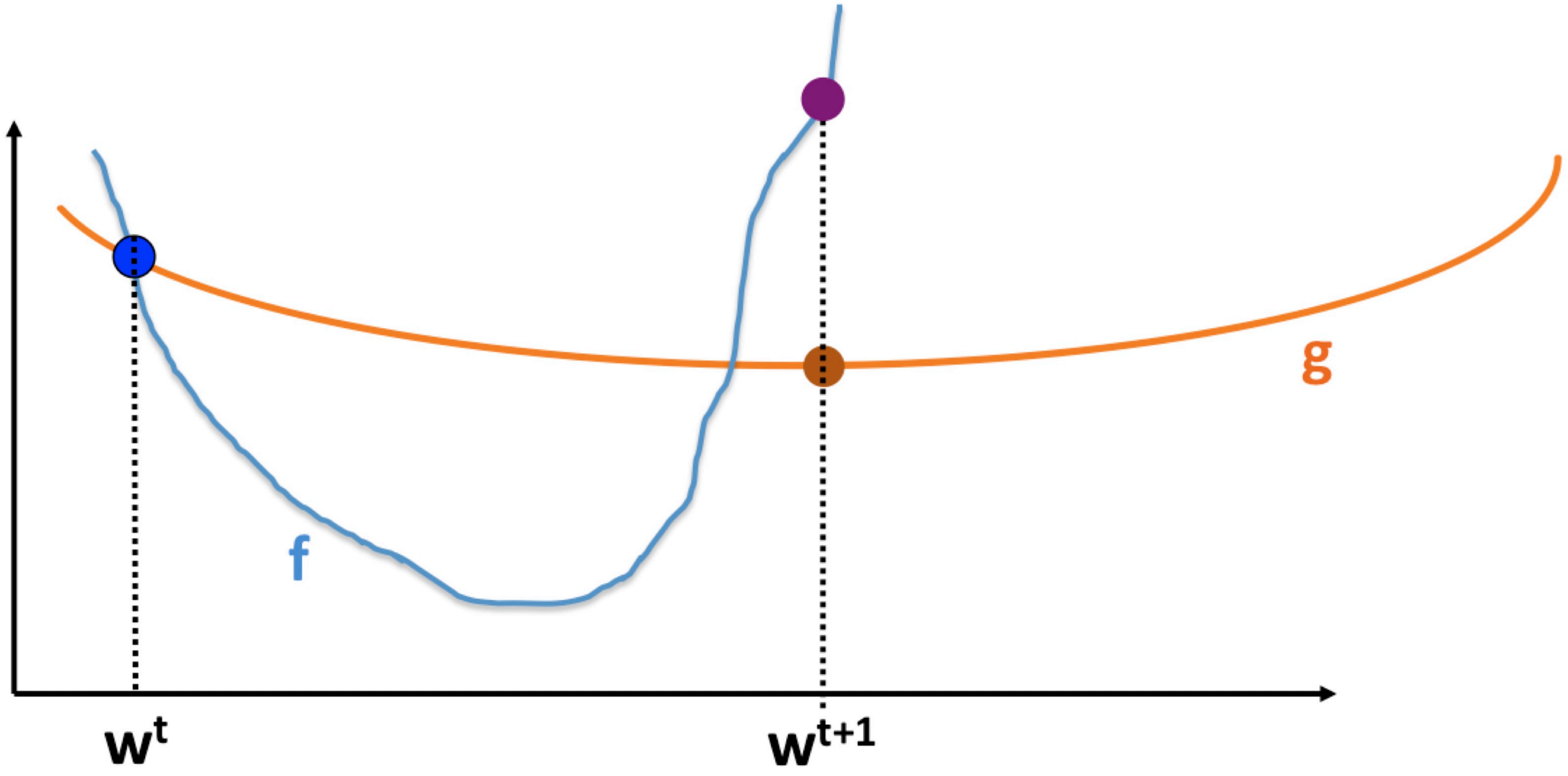
Illustration of gradient descent



Optimization

When will it diverge

Can diverge ($f(w^t) < f(w^{t+1})$) if g is not an upper bound of f

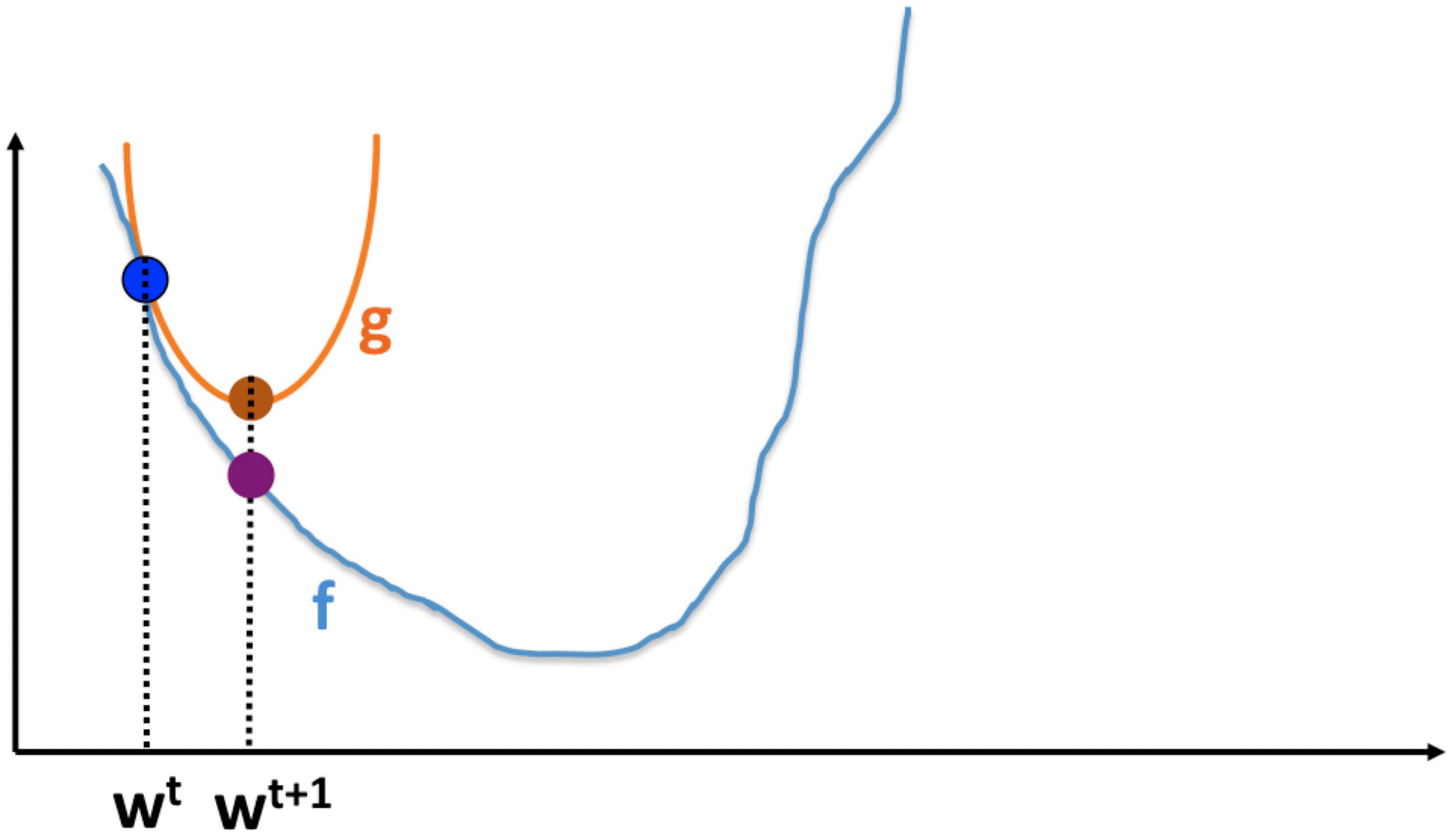


$f(w^t) < f(w^{t+1})$, diverge because g's curvature is too small

Optimization

When will it converge

Always converge ($f(w^t) > f(w^{t+1})$) if g is an upper bound of f



$f(w^t) > f(w^{t+1})$, converge when g's curvature is large enough

Optimization

Convergence

- A differential function f is said to be L-Lipschitz continuous:
  - $\|f(x_1) - f(x_2)\|_2 \leq L\|x_1 - x_2\|_2$
- A differential function f is said to be L-smooth: its gradient are Lipschitz continuous:
  - $\|\nabla f(x_1) - \nabla f(x_2)\|_2 \leq L\|x_1 - x_2\|_2$
  - And we could get
    - $\nabla^2 f(x) \preceq LI$
    - $f(y) \leq f(x) + \nabla f(x)^T(y - x) + \frac{1}{2}L\|y - x\|^2$

Optimization

Convergence

- Let L be a Lipchitz constant ($\nabla^2 f(x) \preceq LI$ for all x)
- Theorem: gradient descent converges if $\alpha < \dfrac{1}{L}$
- In practice, we do not know L …
  - Need to tune step size when running gradient descent

Optimization

Convergence

- Let L be a Lipchitz constant ($\nabla^2 f(x) \preceq LI$ for all x)
- Theorem: gradient descent converges if $\alpha < \dfrac{1}{L}$
- Why?

Optimization

Convergence

- Let L be a Lipchitz constant ($\nabla^2 f(x) \preceq LI$ for all x)
- Theorem: gradient descent converges if $\alpha < \dfrac{1}{L}$
- Why?
  - When $\alpha < 1/L$, for any d,
  - $$\begin{aligned} g(d) &= f(w^t) + \nabla f(w^t)^T d + \frac{1}{2\alpha}\|d\|^2 \\ &> f(w^t) + \nabla f(w^t)^T d + \frac{L}{2}\|d\|^2 \\ &\geq f(w^t + d) \end{aligned}$$
  - So, $f(w^t + d^*) < g(d^*) \leq g(0) = f(w^t)$
  - In formal proof, need to show $f(w^t + d^*)$ is sufficiently smaller than $f(w^t)$

Optimization

Gradient descent convergence rate

- Suppose f is convex and differentiable and its gradient is lipshcitz continuous, then if we run gradient for t iterations with a fixed step $\alpha \leq \frac{1}{L}$, it will yield a solution that satisfies:
  - $f(w^t) - f(w*) \leq \frac{\|w^0 - w*\|_2^2}{2\alpha t}$
- Proof

Optimization

Convergence

- Let L be a Lipchitz constant ($\nabla^2 f(x) \preceq LI$ for all x)
- Theorem: gradient descent converges if $\alpha < \dfrac{1}{L}$
- In practice, we do not know L …
  - Need to tune step size when running gradient descent

Optimization

Applying to logistic regression

gradient descent for logistic regression

- Initialize the weights $\boldsymbol{w}_0$
- For $t = 1, 2, \cdots$
  - Compute the gradient

$$\nabla f(\boldsymbol{w}) = -\frac{1}{N}\sum_{n=1}^{N}\frac{y_n \boldsymbol{x}_n}{1 + e^{y_n \boldsymbol{w}^T \boldsymbol{x}_n}}$$

  - Update the weights: $\boldsymbol{w} \leftarrow \boldsymbol{w} - \eta \nabla f(\boldsymbol{w})$
- Return the final weights $\boldsymbol{w}$

Optimization

Applying to logistic regression

- When to stop?
  - Fixed number of iterations, or
  - Stop when $||\nabla f(w)|| < \epsilon$

gradient descent for logistic regression

- Initialize the weights $\boldsymbol{w}_0$
- For $t = 1, 2, \cdots$
  - Compute the gradient

$$\nabla f(\boldsymbol{w}) = -\frac{1}{N}\sum_{n=1}^{N} \frac{y_n \boldsymbol{x}_n}{1 + e^{y_n \boldsymbol{w}^T \boldsymbol{x}_n}}$$

  - Update the weights: $\boldsymbol{w} \leftarrow \boldsymbol{w} - \eta \nabla f(\boldsymbol{w})$
- Return the final weights $\boldsymbol{w}$

Optimization

Line search

- In practice, we do not know L …
  - Need to tune step size when running gradient descent
- Line Search: Select step size automatically (for gradient descent)

Optimization

Line search

- The back-tracking line search:
  - Start from some large α_0
  - Try $\alpha = \alpha_0, \alpha_0/2, \alpha_0/4, \ldots$
    - Stop when α satisfies some sufficient decrease condition

Optimization

Line search

- The back-tracking line search:
  - Start from some large α_0
  - Try $\alpha = \alpha_0, \alpha_0/2, \alpha_0/4, \ldots$
    - Stop when α satisfies some sufficient decrease condition
  - A simple condition: $f(w + \alpha d) < f(w)$

Optimization

Line search

- The back-tracking line search:
  - Start from some large α_0
  - Try $\alpha = \alpha_0, \alpha_0/2, \alpha_0/4, \ldots$
    - Stop when α satisfies some sufficient decrease condition
  - A simple condition: $f(w + \alpha d) < f(w)$
    - Often works in practice but doesn't work in theory

Optimization

Line search

- The back-tracking line search:
  - Start from some large α_0
  - Try $\alpha = \alpha_0, \alpha_0/2, \alpha_0/4, \ldots$
    - Stop when α satisfies some sufficient decrease condition
  - A simple condition: $f(w + \alpha d) < f(w)$
    - Often works in practice but doesn't work in theory
  - A (provable) sufficient decrease condition $f(w + \alpha d) \leq f(w) + c_1 \alpha \nabla f(w)^T d$ (armijo condition)
  - $\nabla f(w + \alpha d)^T d \geq c_2 \nabla f(w)^T d$ (curvature)
    - + armijo = wolfe condition
    - For constant $c_1, c_2 \in (0,1)$

Optimization

Line search

gradient descent with backtracking line search

- Initialize the weights $\boldsymbol{w}_0$
- For $t = 1, 2, \cdots$
  - Compute the gradient

    $$\boldsymbol{d} = -\nabla f(\boldsymbol{w})$$

  - For $\alpha = \alpha_0, \alpha_0/2, \alpha_0/4, \cdots$

    Break if $f(\boldsymbol{w} + \alpha \boldsymbol{d}) \leq f(\boldsymbol{w}) + \sigma \alpha \nabla f(\boldsymbol{w})^T \boldsymbol{d}$
  - Update $\boldsymbol{w} \leftarrow \boldsymbol{w} + \alpha \boldsymbol{d}$
- Return the final solution $\boldsymbol{w}$