Function Approximation Methods-IV

CS771: Introduction to Machine Learning

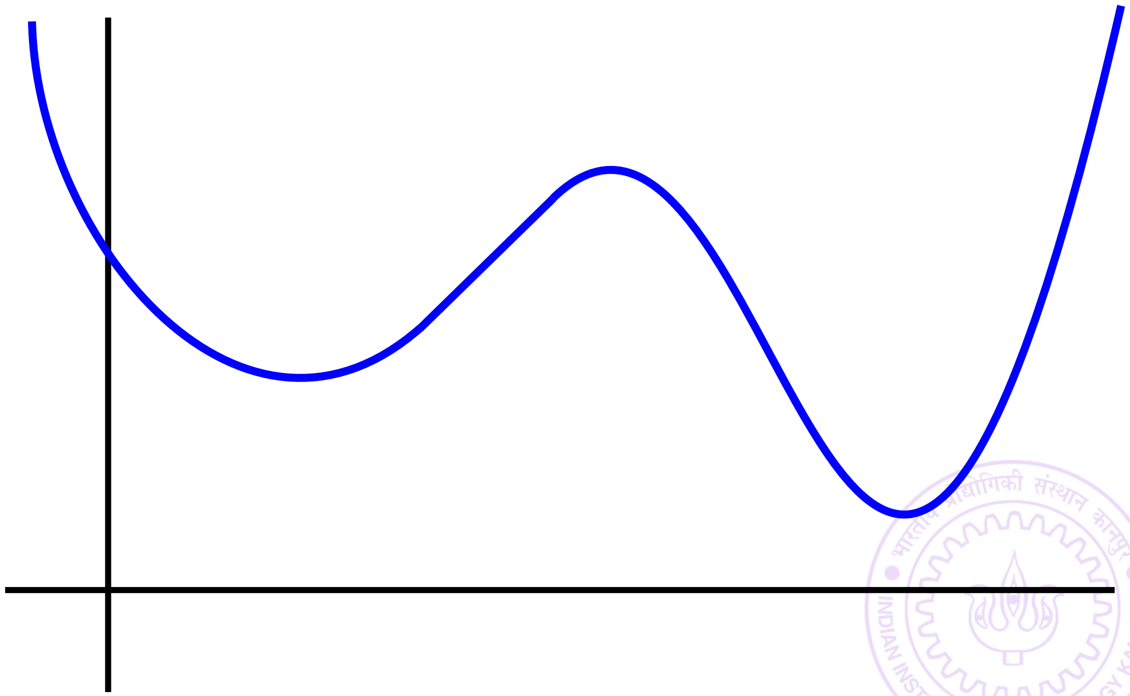
Purushottam Kar

Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w}) + r(\mathbf{w})$$

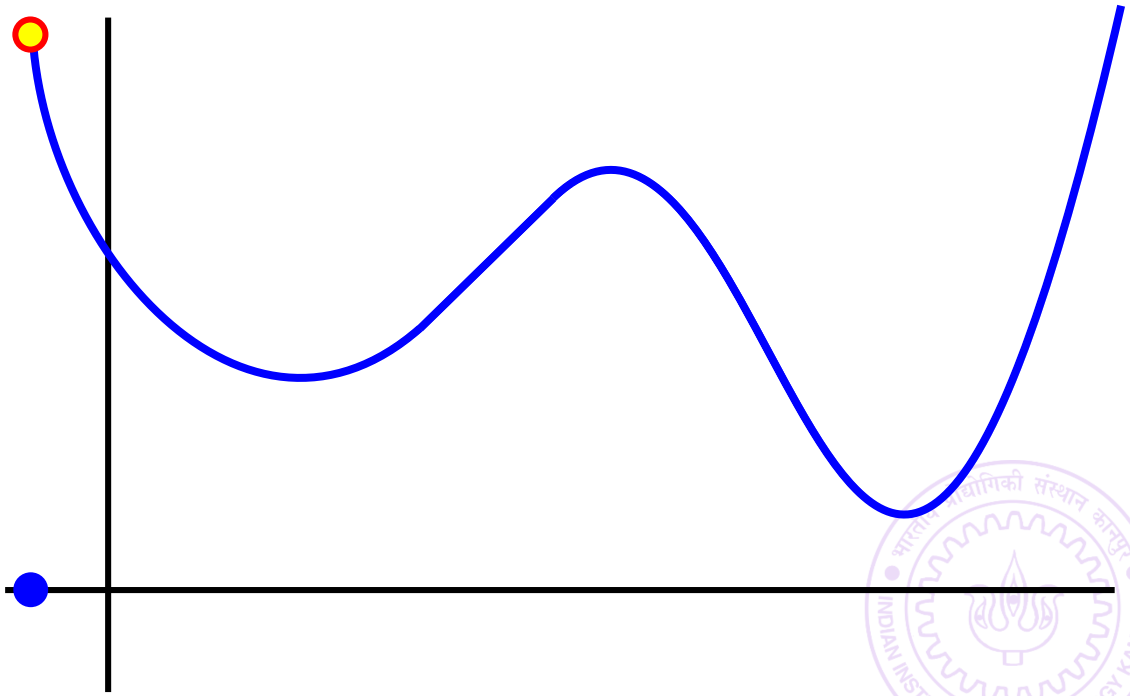
Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$



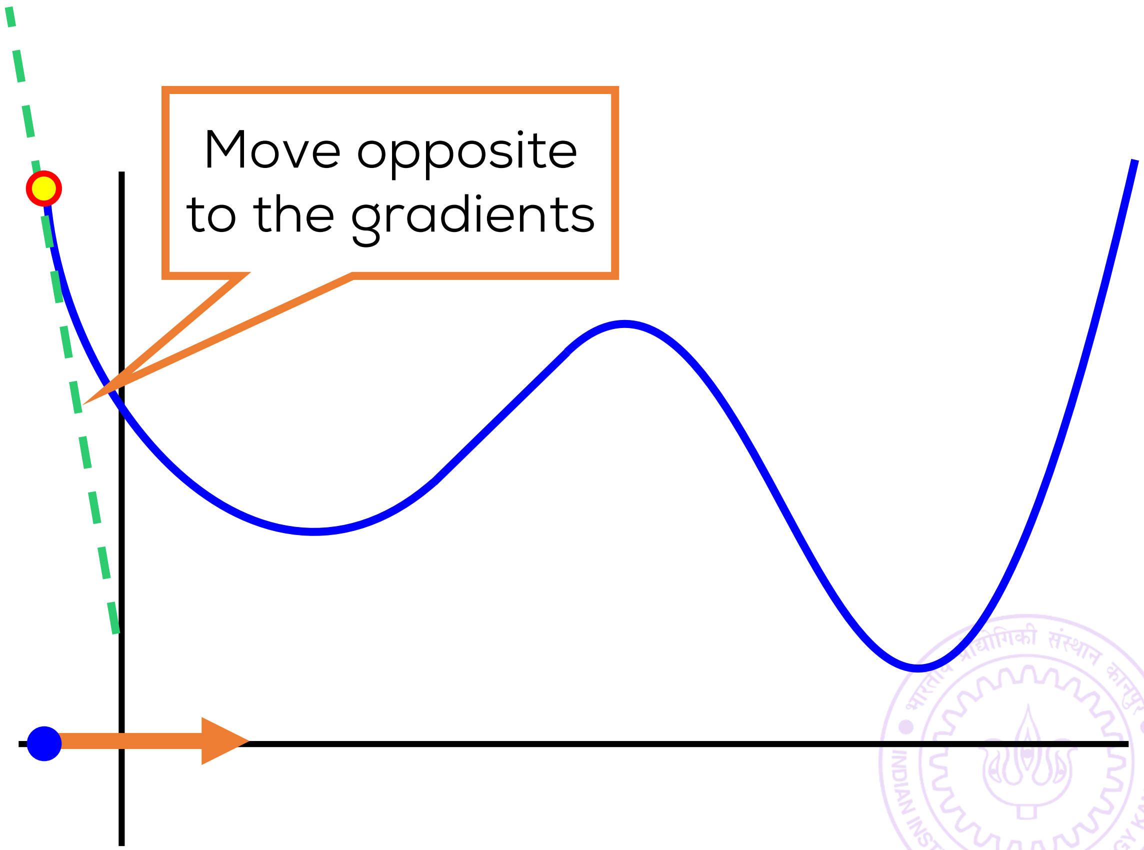
Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w}) + r(\mathbf{w})$$



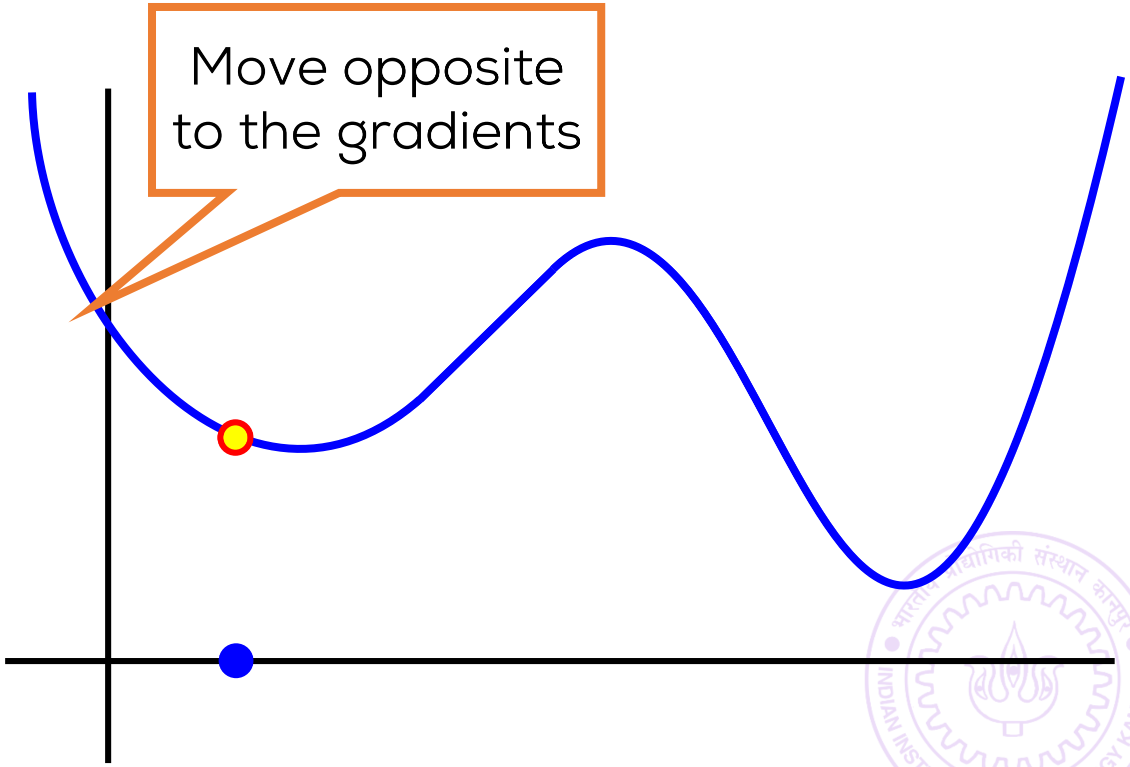
Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$



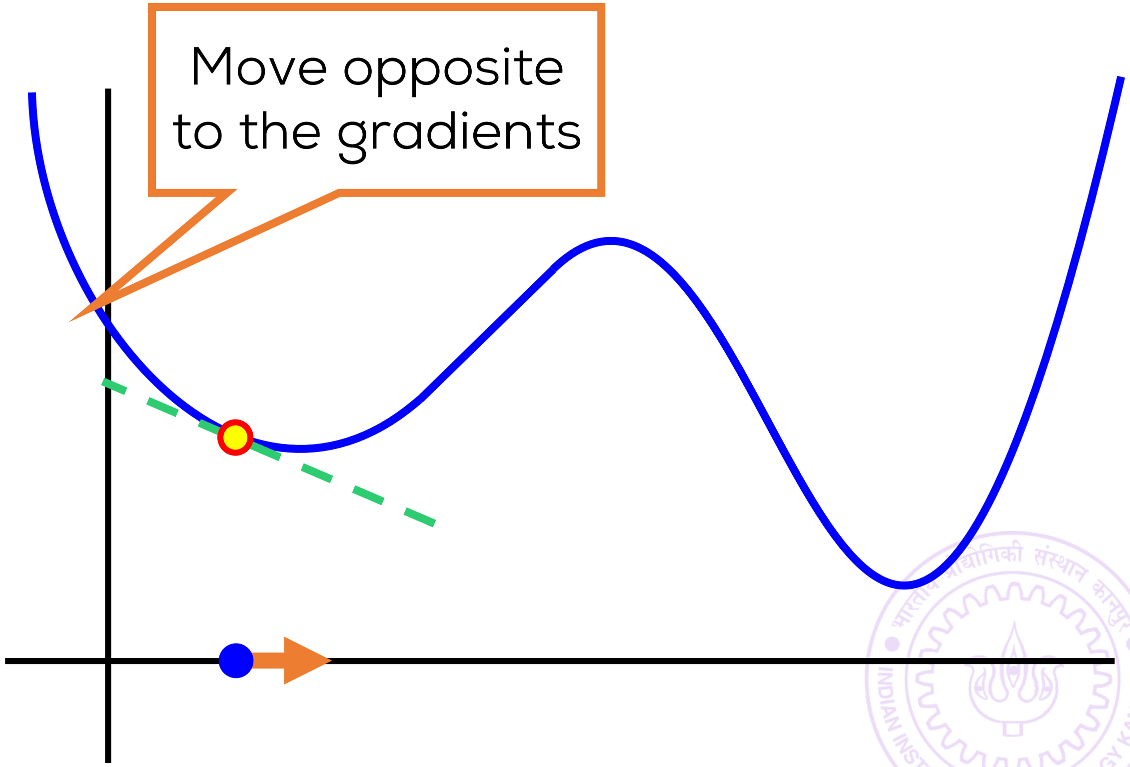
Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \underset{\mathbf{w} \in \mathbb{R}^d}{\arg\min}\; f(\mathbf{w}) + r(\mathbf{w})$$



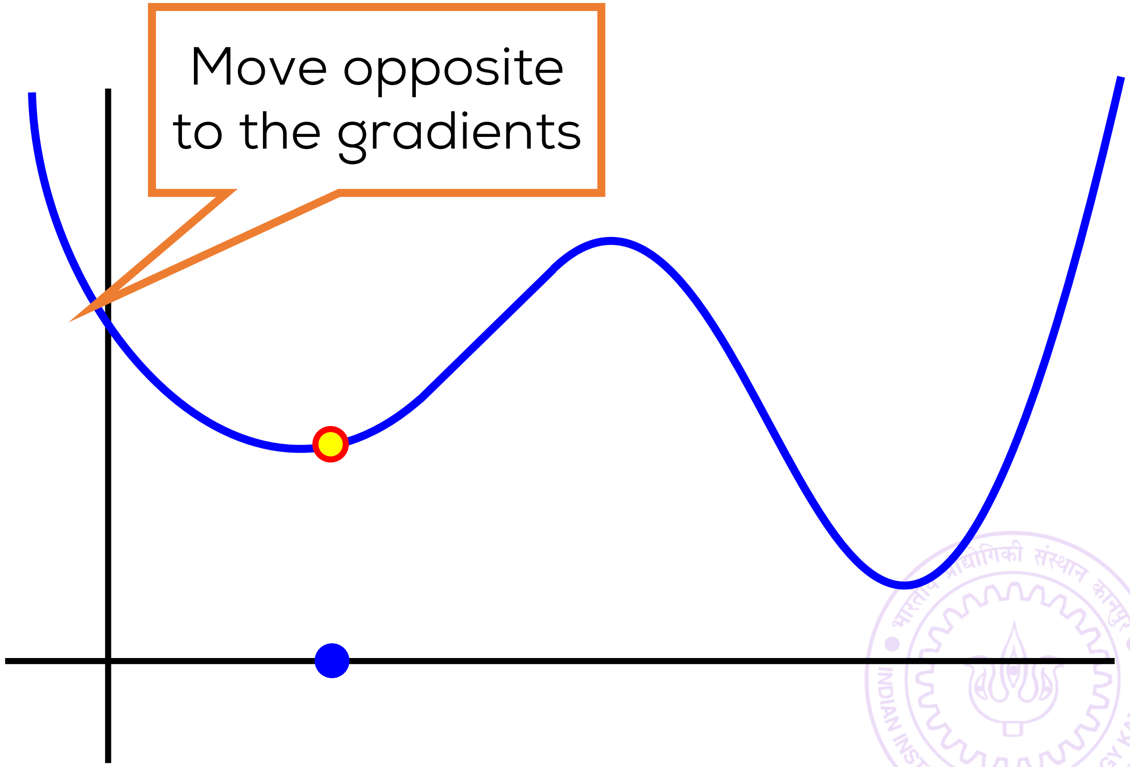
Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$



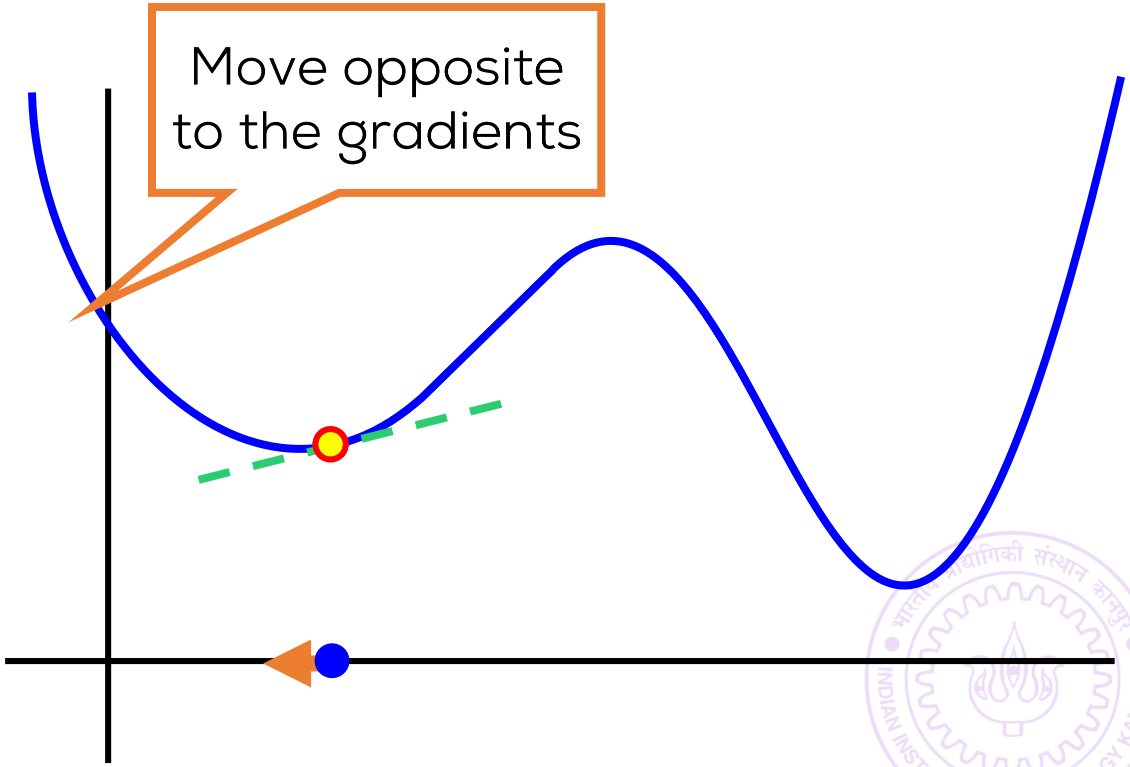
Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$



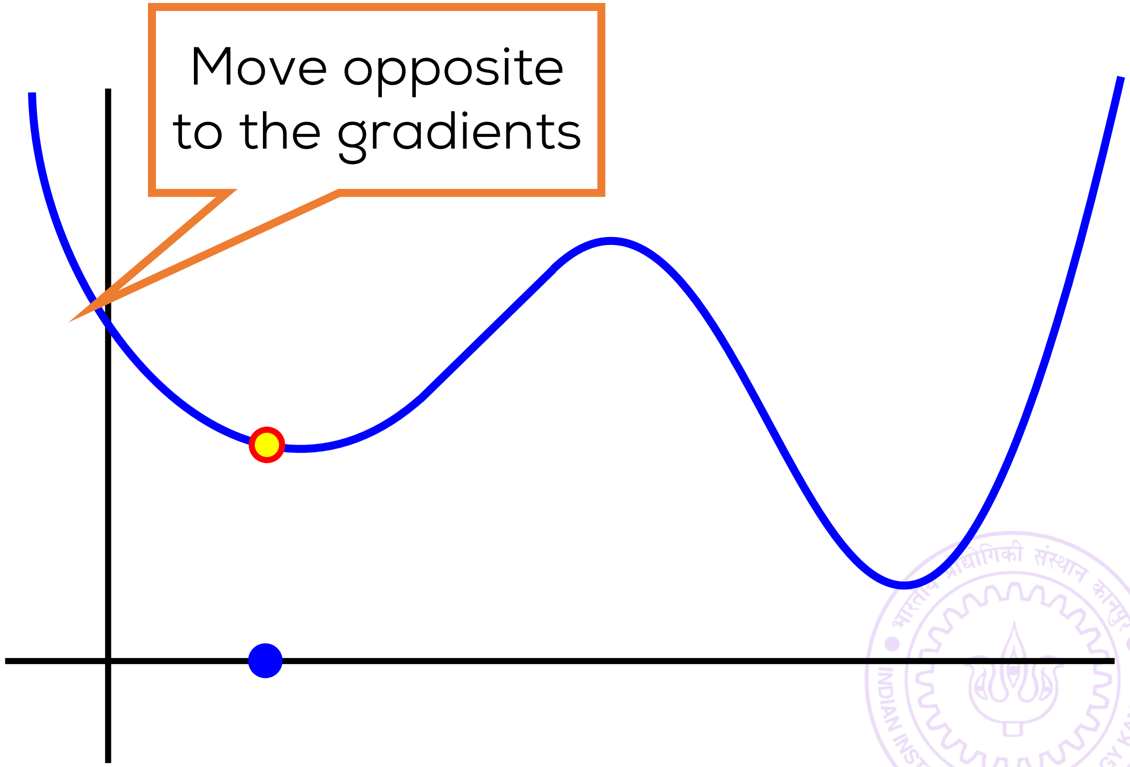
Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$



Stochastic Gradient Descent

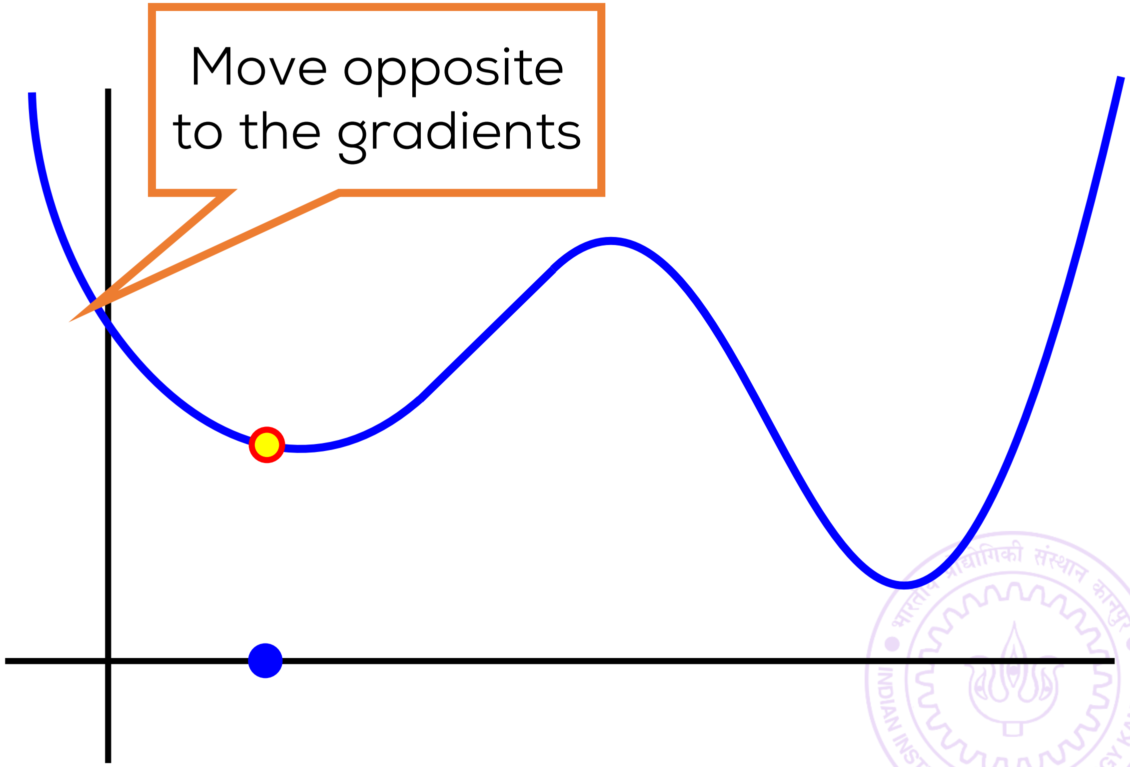
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \; f(\mathbf{w}) + r(\mathbf{w})$$



Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \; f(\mathbf{w}) + r(\mathbf{w})$$

$$\mathbf{g}^t = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$



Stochastic Gradient Descent

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w}) + r(\mathbf{w})$$

$$\mathbf{g}^t = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

Stochastic Gradient Descent

$$f(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

Stochastic Gradient Descent

$O(nd)$ time

$$\nabla f(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

Stochastic Gradient Descent

$O(nd)$ time

$$\nabla f(\mathbf{w}) \approx \quad \nabla f_i(\mathbf{w}) \approx \quad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \qquad \nabla f_i(\mathbf{w}) \approx \qquad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \qquad \nabla f_i(\mathbf{w}) \approx \qquad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \quad \nabla f_i(\mathbf{w}) \approx \quad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

STOCHASTIC GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Select a unif. random point $I_t \in [n]$
3. Let $\mathbf{g}^t \leftarrow \nabla f_{I_t}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \quad \nabla f_i(\mathbf{w}) \approx \quad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

$$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

STOCHASTIC GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Select a unif. random point $I_t \in [n]$
3. Let $\mathbf{g}^t \leftarrow \nabla f_{I_t}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \quad \nabla f_i(\mathbf{w}) \approx \quad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$

Only $O(d)$ time per iter!

STOCHASTIC GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Select a unif. random point $I_t \in [n]$
3. Let $\mathbf{g}^t \leftarrow \nabla f_{I_t}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \quad \nabla f_i(\mathbf{w}) \approx \quad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$

Only $O(d)$ time per iter!

Step length
$\eta_t = C, \frac{C}{t}, \frac{C}{\sqrt{t}}$

STOCHASTIC GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Select a unif. random point $I_t \in [n]$
3. Let $\mathbf{g}^t \leftarrow \nabla f_{I_t}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \quad \nabla f_i(\mathbf{w}) \approx \quad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$

Only $O(d)$ time per iter!

Step length $\eta_t = C, \frac{C}{t}, \frac{C}{\sqrt{t}}$

w.h.p. ϵ-optimal solution in $O\left(\frac{1}{\epsilon^2}\right)$ iterations

STOCHASTIC GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Select a unif. random point $I_t \in [n]$
3. Let $\mathbf{g}^t \leftarrow \nabla f_{I_t}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \quad \nabla f_i(\mathbf{w}) \approx \quad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$

Only $O(d)$ time per iter!

STOCHASTIC GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Select a unif. random point $I_t \in [n]$
3. Let $\mathbf{g}^t \leftarrow \nabla f_{I_t}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Step length $\eta_t = C, \frac{C}{t}, \frac{C}{\sqrt{t}}$

w.h.p. ϵ-optimal solution in $O\left(\frac{1}{\epsilon^2}\right)$ iterations

$f(\mathbf{w}^t) + r(\mathbf{w}^t) \leq f(\mathbf{w}^*) + r(\mathbf{w}^*) + \epsilon$

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \qquad \nabla f_i(\mathbf{w}) \approx \qquad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

Only $O(d)$ time per iter!

ONLINE GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Select a unif. random point $I_t \in [n]$
3. Let $\mathbf{g}^t \leftarrow \nabla f_{I_t}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Step length $\eta_t = C, \frac{C}{t}, \frac{C}{\sqrt{t}}$

w.h.p. ϵ-optimal solution in $O\left(\frac{1}{\epsilon^2}\right)$ iterations

$f(\mathbf{w}^t) + r(\mathbf{w}^t) \leq f(\mathbf{w}^*) + r(\mathbf{w}^*) + \epsilon$

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \quad \nabla f_i(\mathbf{w}) \approx \quad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

ONLINE GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. Let $\mathbf{g}^t \leftarrow \nabla f_{I_t}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Only $O(d)$ time per iter!

Step length $\eta_t = C, \frac{C}{t}, \frac{C}{\sqrt{t}}$

w.h.p. ϵ-optimal solution in $O\left(\frac{1}{\epsilon^2}\right)$ iterations

$f(\mathbf{w}^t) + r(\mathbf{w}^t) \leq f(\mathbf{w}^*) + r(\mathbf{w}^*) + \epsilon$

Stochastic Gradient Descent

$O(d)$ time!!

$$\nabla f(\mathbf{w}) \approx \quad \nabla f_i(\mathbf{w}) \approx \quad \nabla \ell(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

$$\mathbf{g}^t = \nabla f_i(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$$

ONLINE GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. Let $\mathbf{g}^t \leftarrow \nabla f(\mathbf{w}^t, \mathbf{z}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

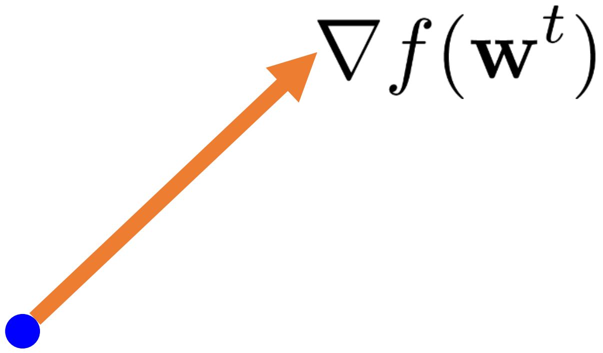
Only $O(d)$ time per iter!

Step length $\eta_t = C, \frac{C}{t}, \frac{C}{\sqrt{t}}$

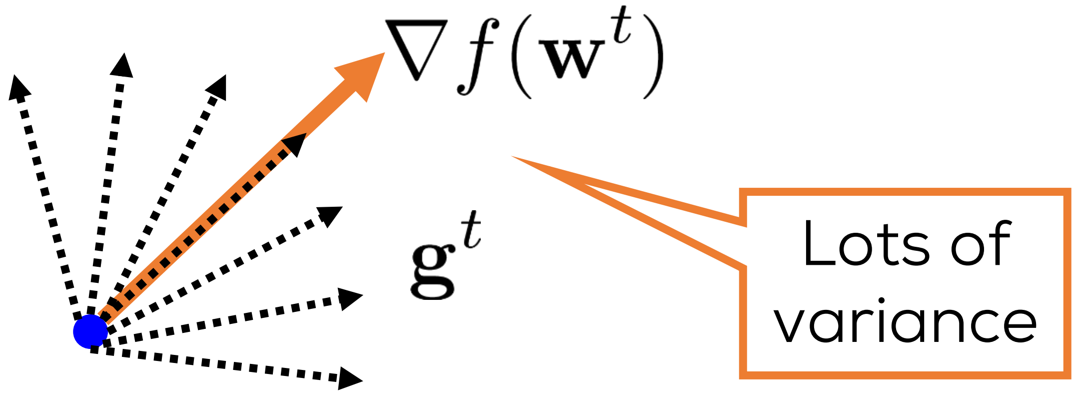
w.h.p. ϵ-optimal solution in $O\left(\frac{1}{\epsilon^2}\right)$ iterations

$f(\mathbf{w}^t) + r(\mathbf{w}^t) \leq f(\mathbf{w}^*) + r(\mathbf{w}^*) + \epsilon$

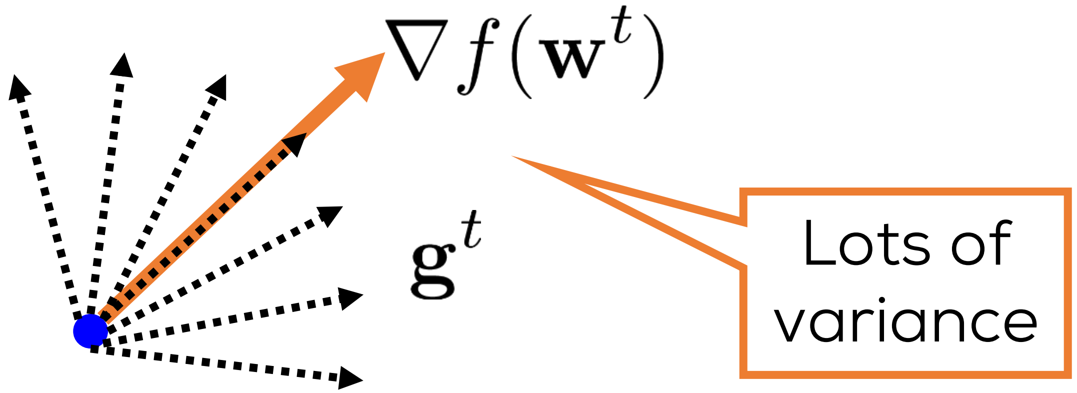
Mini-batch Gradient Descent



Mini-batch Gradient Descent



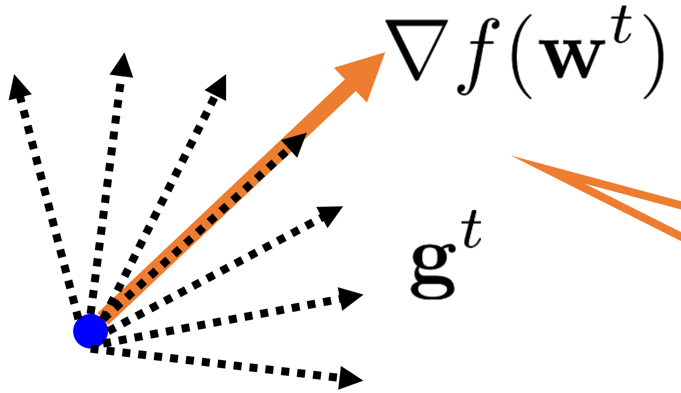
Mini-batch Gradient Descent



MINI-BATCH SGD

1. Initialize $\mathbf{w}^0$
2. Select B unif. random points $\{I_{t,1}, \ldots, I_{t,B}\} \in [n]$
3. Let $\mathbf{g}^t \leftarrow \frac{1}{B}\sum_{i=1}^{n} \nabla f_{I_{t,i}}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Mini-batch Gradient Descent



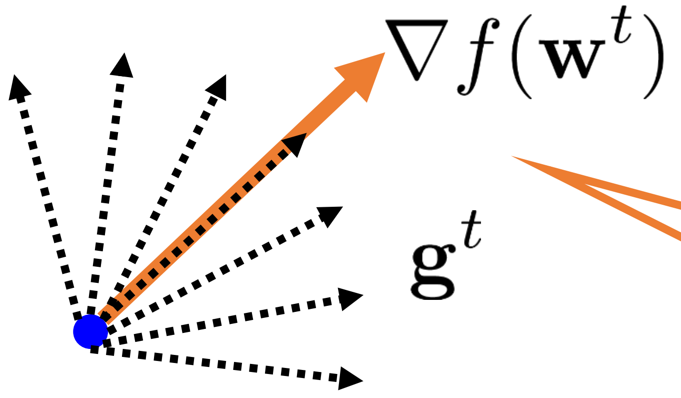
Lots of variance

MINI-BATCH SGD

1. Initialize $\mathbf{w}^0$
2. Select B unif. random points $\{I_{t,1}, \ldots, I_{t,B}\} \in [n]$
3. Let $\mathbf{g}^t \leftarrow \frac{1}{\mathrm{B}} \sum_{i=1}^{n} \nabla f_{I_{t,i}}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Without replacement?

Mini-batch Gradient Descent



Lots of variance

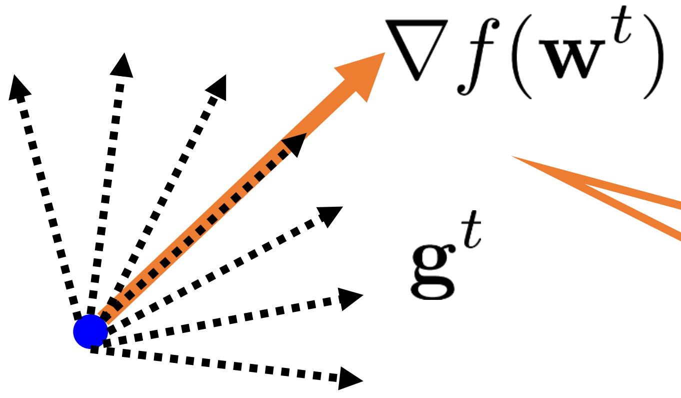
MINI-BATCH SGD

1. Initialize $\mathbf{w}^0$
2. Select B unif. random points $\{I_{t,1}, \ldots, I_{t,B}\} \in [n]$
3. Let $\mathbf{g}^t \leftarrow \frac{1}{\mathrm{B}} \sum_{i=1}^{n} \nabla f_{I_{t,i}}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

Without replacement?

$O(B \cdot d)$ time per iter!

Mini-batch Gradient Descent



Lots of variance

MINI-BATCH SGD

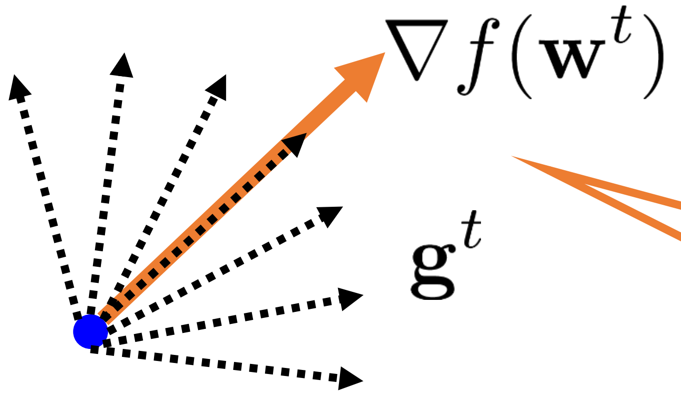
1. Initialize $\mathbf{w}^0$
2. Select B unif. random points $\{I_{t,1}, \ldots, I_{t,B}\} \in [n]$
3. Let $\mathbf{g}^t \leftarrow \frac{1}{\mathrm{B}} \sum_{i=1}^{n} \nabla f_{I_{t,i}}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$

Without replacement?

$O(B \cdot d)$ time per iter!

Mini-batch Gradient Descent



Lots of variance

MINI-BATCH SGD

1. Initialize $\mathbf{w}^0$
2. Select B unif. random points $\{I_{t,1}, \ldots, I_{t,B}\} \in [n]$
3. Let $\mathbf{g}^t \leftarrow \frac{1}{\mathrm{B}} \sum_{i=1}^{n} \nabla f_{I_{t,i}}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

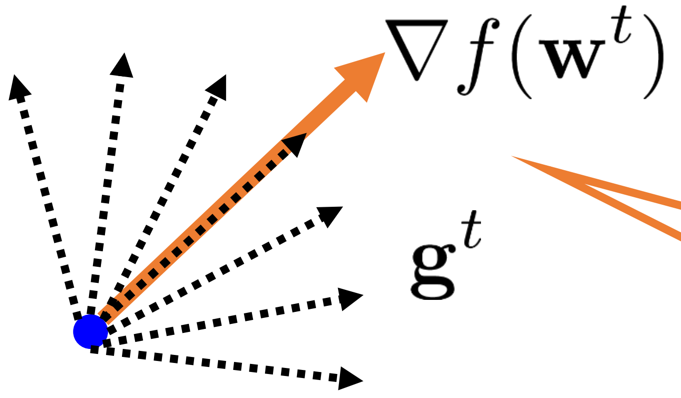
$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$

Without replacement?

Usually $B \ll n$

$O(B \cdot d)$ time per iter!

Mini-batch Gradient Descent



Lots of variance

MINI-BATCH SGD

1. Initialize $\mathbf{w}^0$
2. Select B unif. random points $\{I_{t,1}, \ldots, I_{t,B}\} \in [n]$
3. Let $\mathbf{g}^t \leftarrow \frac{1}{\mathrm{B}} \sum_{i=1}^{n} \nabla f_{I_{t,i}}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

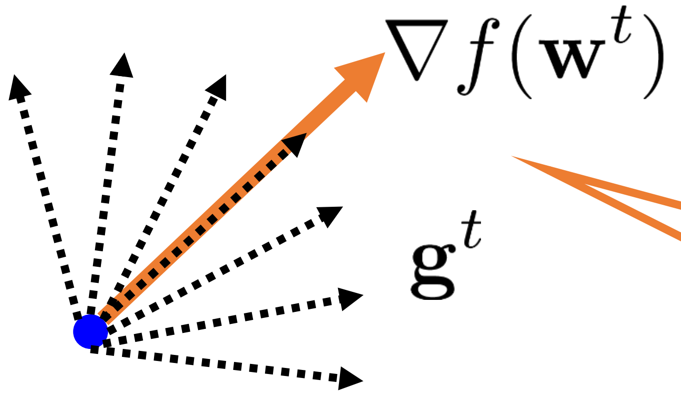
$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$

Without replacement?

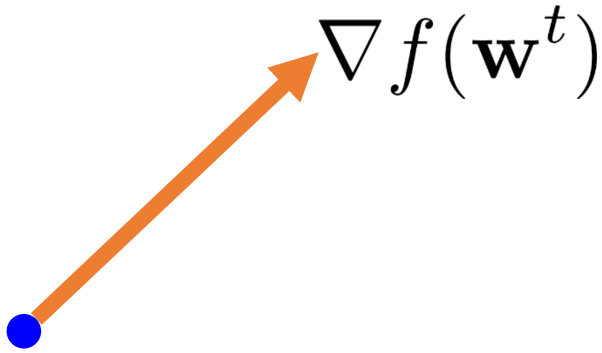
Usually $B \ll n$

$O(B \cdot d)$ time per iter!

Mini-batch Gradient Descent



Lots of variance



MINI-BATCH SGD

1. Initialize $\mathbf{w}^0$
2. Select B unif. random points $\{I_{t,1}, \ldots, I_{t,B}\} \in [n]$
3. Let $\mathbf{g}^t \leftarrow \frac{1}{B}\sum_{i=1}^{n} \nabla f_{I_{t,i}}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

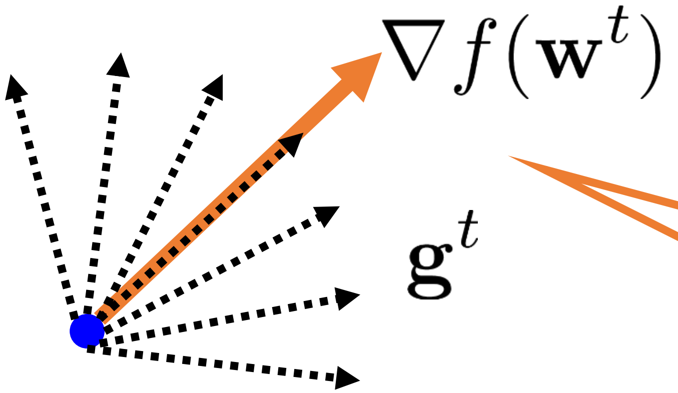
$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$

Without replacement?

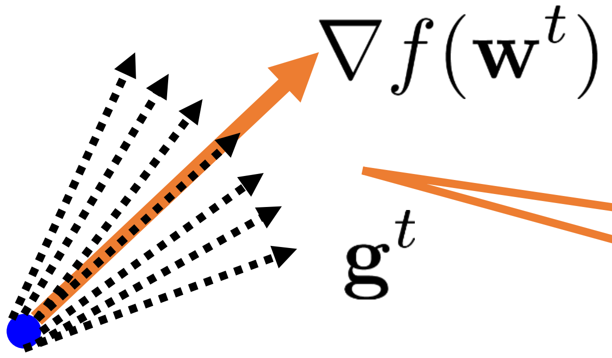
Usually $B \ll n$

$O(B \cdot d)$ time per iter!

Mini-batch Gradient Descent



Lots of variance



No variance if all points chosen i.e with $B = n$

Less variance with mini-batch

MINI-BATCH SGD

1. Initialize $\mathbf{w}^0$
2. Select B unif. random points $\{I_{t,1}, \dots, I_{t,B}\} \in [n]$
3. Let $\mathbf{g}^t \leftarrow \frac{1}{\mathrm{B}} \sum_{i=1}^{n} \nabla f_{I_{t,i}}(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat until convergence

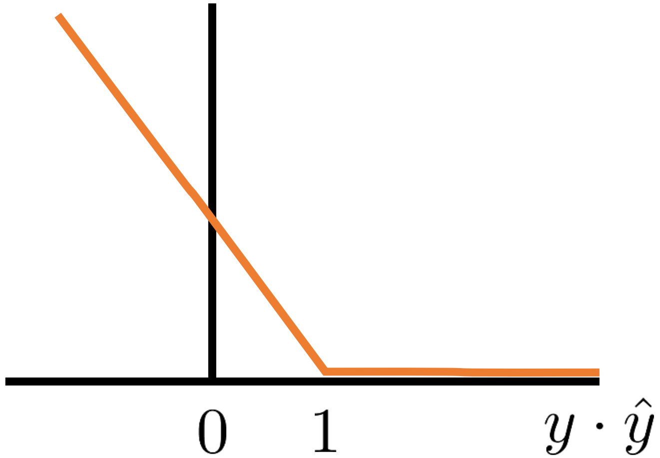
$\mathbb{E}[\mathbf{g}^t | \mathbf{w}^t] = \nabla f(\mathbf{w}^t) + \nabla r(\mathbf{w}^t)$

Without replacement?

Usually $B \ll n$

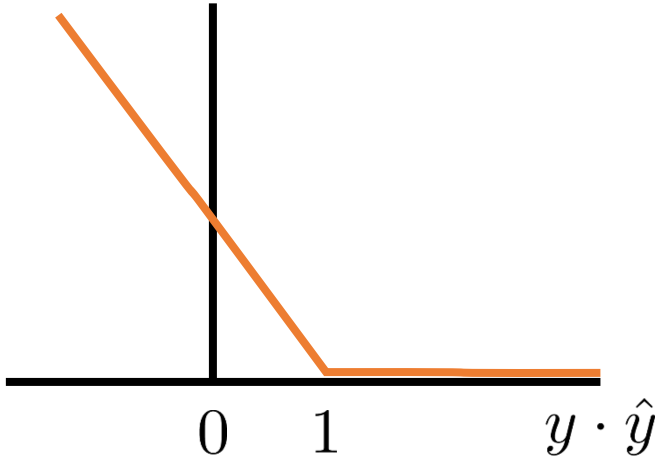
$O(B \cdot d)$ time per iter!

App: The Perceptron Rule via OGD



Hinge loss

App: The Perceptron Rule via OGD

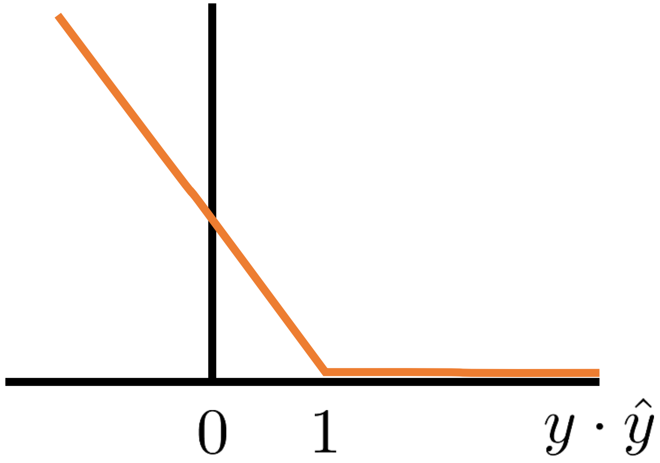


Hinge loss

ONLINE GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. Let $\mathbf{g}^t \leftarrow \nabla f(\mathbf{w}^t, \mathbf{z}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat

App: The Perceptron Rule via OGD



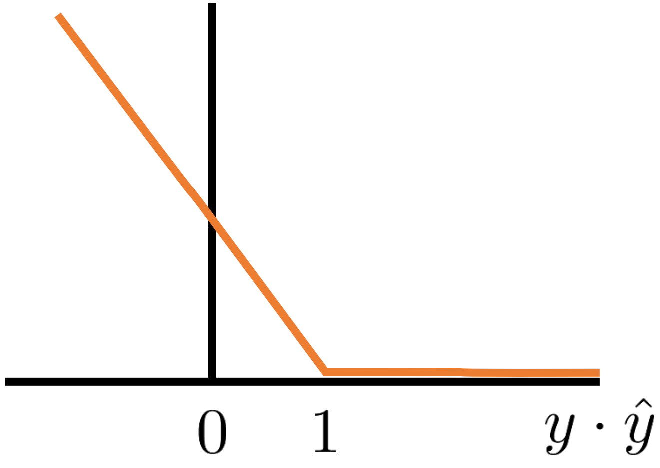
Hinge loss

ONLINE GRADIENT DESCE

$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^{\mathrm{t}} = (y^t, \mathbf{x}^t)$
3. Let $\mathbf{g}^t \leftarrow \nabla f(\mathbf{w}^t, \mathbf{z}^t) + \nabla r(\mathbf{w}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^{\mathrm{t}}$
5. Repeat

App: The Perceptron Rule via OGD



Hinge loss

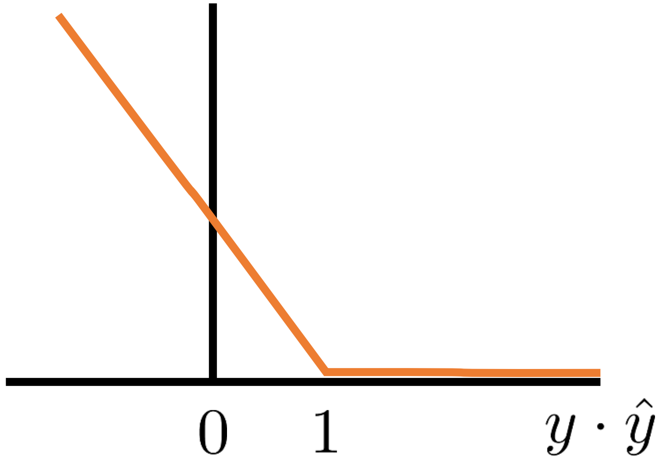
ONLINE GRADIENT DESCE

$y^t \in \{-1, +1\}$

Forget reg. for now

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. Let $\mathbf{g}^t \leftarrow \nabla f(\mathbf{w}^t, \mathbf{z}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat

App: The Perceptron Rule via OGD



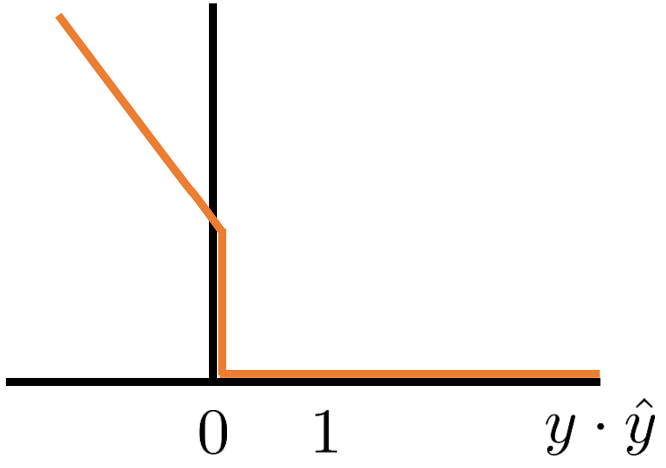
Hinge loss

ONLINE GRADIENT DESCE

$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^{\mathrm{t}} = (y^t, \mathbf{x}^t)$
3. Let $\mathbf{g}^t \leftarrow \nabla f(\mathbf{w}^t, \mathbf{z}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^{\mathrm{t}}$
5. Repeat

App: The Perceptron Rule via OGD



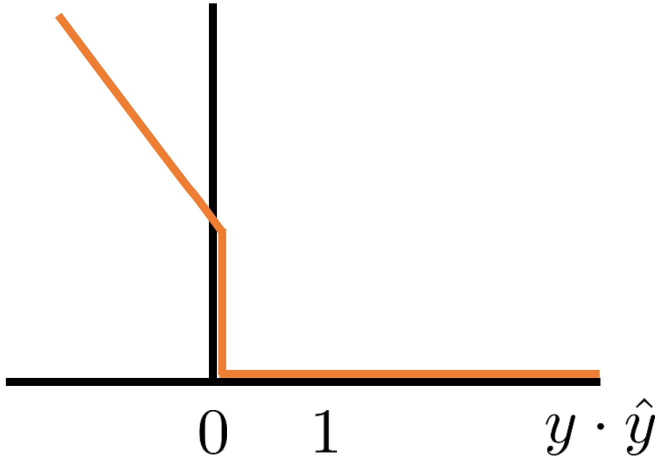
Hinge loss

ONLINE GRADIENT DESCE

$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^{\mathrm{t}} = (y^t, \mathbf{x}^t)$
3. Let $\mathbf{g}^t \leftarrow \nabla f(\mathbf{w}^t, \mathbf{z}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^{\mathrm{t}}$
5. Repeat

App: The Perceptron Rule via OGD



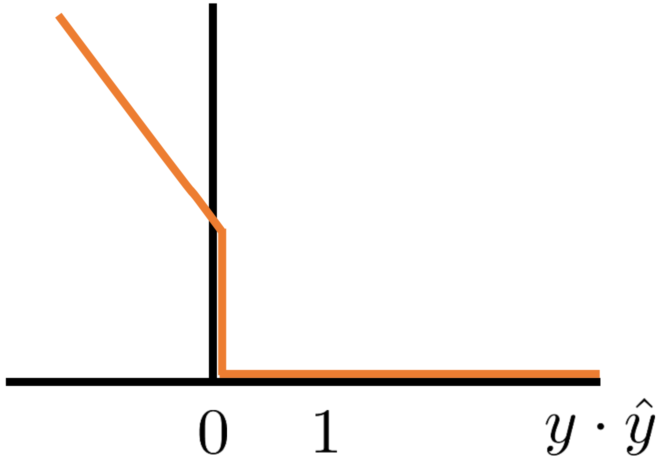
Truncated Hinge loss

ONLINE GRADIENT DESCENT

$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. Let $\mathbf{g}^t \leftarrow \nabla f(\mathbf{w}^t, \mathbf{z}^t)$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat

App: The Perceptron Rule via OGD



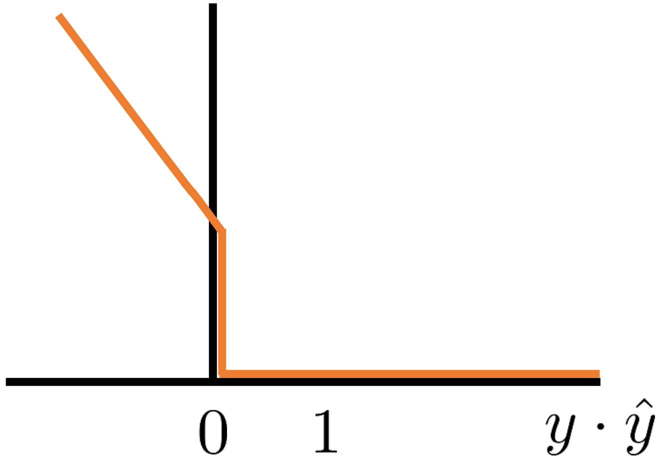
Truncated Hinge loss

ONLINE GRADIENT DESCENT

$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
5. Repeat

App: The Perceptron Rule via OGD



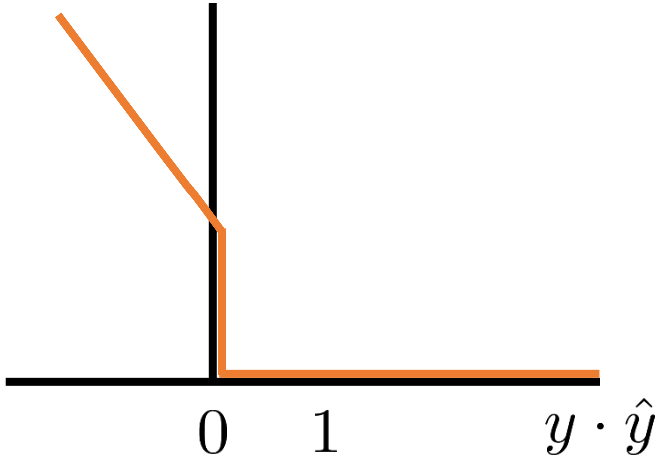
Truncated Hinge loss

ONLINE GRADIENT DESCE

$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^{\mathrm{t}} = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t \cdot y^t \cdot \mathbf{x}^{\mathrm{t}}$
5. Repeat

App: The Perceptron Rule via OGD



Truncated Hinge loss

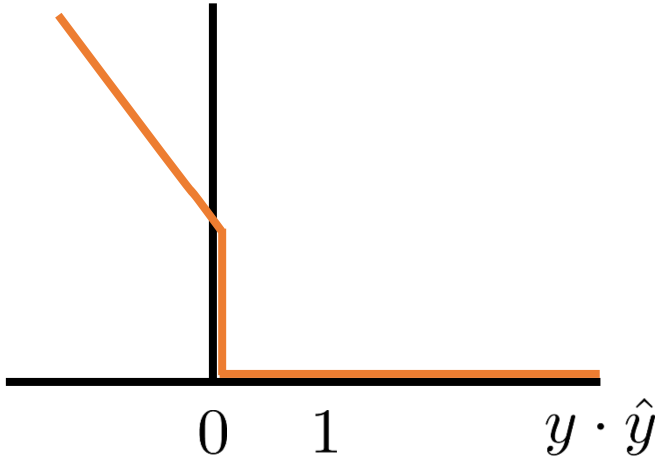
ONLINE GRADIENT DESCE

$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t \cdot y^t \cdot \mathbf{x}^t$
5. Repeat

Make updates only when making a mistake

App: The Perceptron Rule via OGD



Truncated Hinge loss

ONLINE GRADIENT DESCE

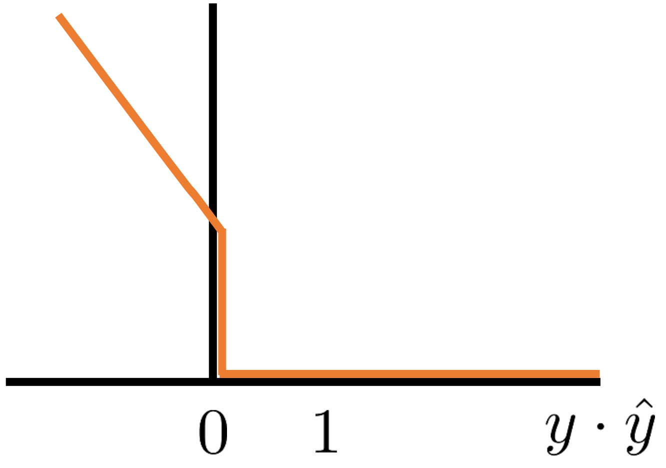
$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t \cdot y^t \cdot \mathbf{x}^t$
5. Repeat

Make updates only when making a mistake

Push $\mathbf{w}^t$ "away"/"toward" $\mathbf{x}^t$ depending on mistake

App: The Perceptron Rule via OGD



Truncated Hinge loss

PERCEPTRON

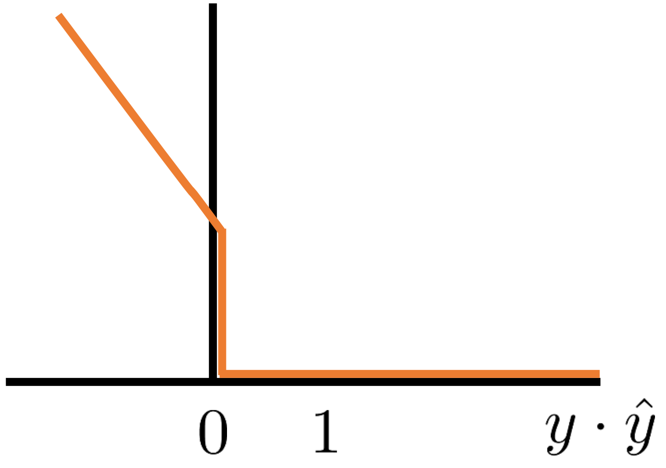
$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t \cdot y^t \cdot \mathbf{x}^t$
5. Repeat

Make updates only when making a mistake

Push $\mathbf{w}^t$ "away"/"toward" $\mathbf{x}^t$ depending on mistake

App: The Perceptron Rule via OGD



Truncated Hinge loss

PERCEPTRON

$y^t \in \{-1, +1\}$

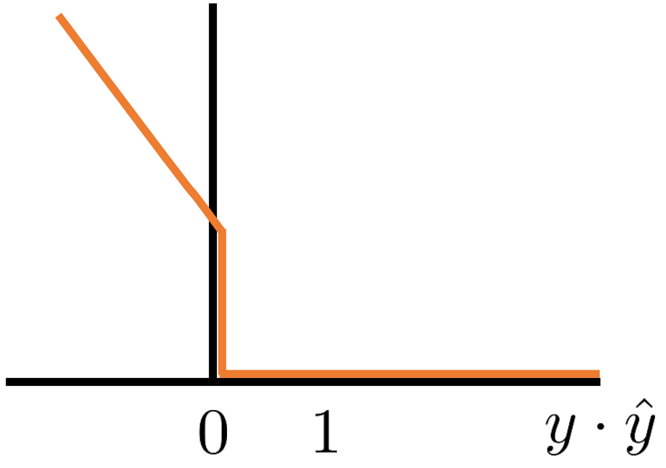
1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t \cdot y^t \cdot \mathbf{x}^t$
5. Repeat

Make updates only when making a mistake

Push $\mathbf{w}^t$ "away"/"toward" $\mathbf{x}^t$ depending on mistake

Perceptron Logistic Regression??

App: The Perceptron Rule via OGD



Truncated Hinge loss

PERCEPTRON

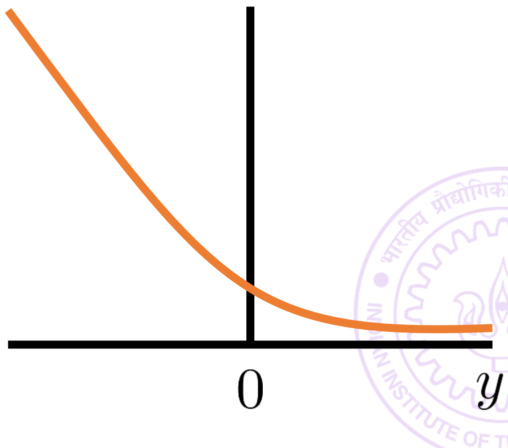
$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t \cdot y^t \cdot \mathbf{x}^t$
5. Repeat

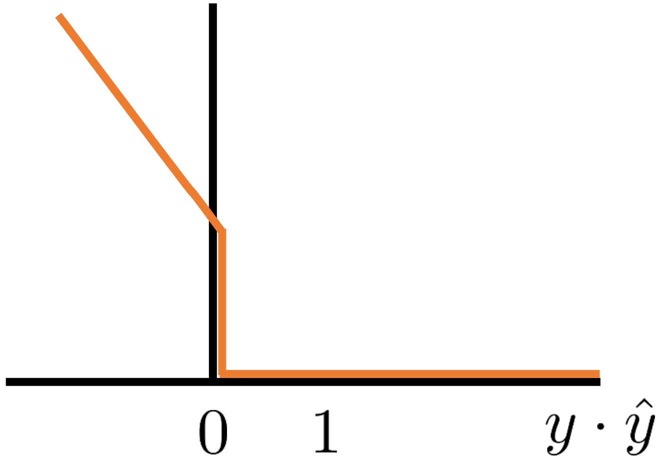
Make updates only when making a mistake

Push $\mathbf{w}^t$ "away"/"toward" $\mathbf{x}^t$ depending on mistake

Perceptron Logistic Regression??



App: The Perceptron Rule via OGD



Truncated Hinge loss

PERCEPTRON

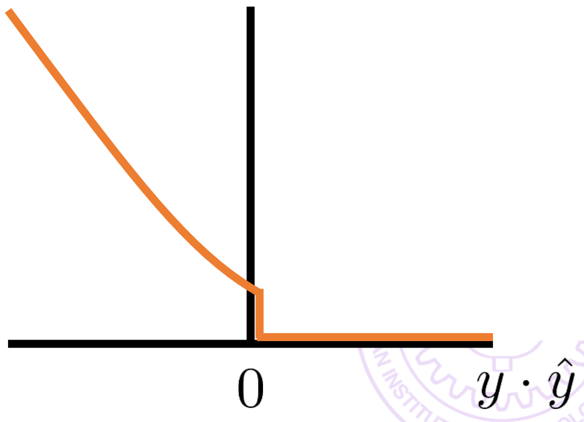
$y^t \in \{-1, +1\}$

1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t \cdot y^t \cdot \mathbf{x}^t$
5. Repeat

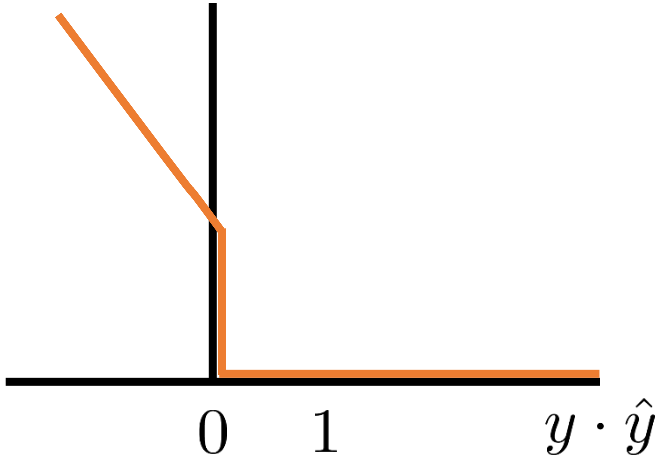
Make updates only when making a mistake

Push $\mathbf{w}^t$ "away"/"toward" $\mathbf{x}^t$ depending on mistake

Perceptron Logistic Regression??



App: The Perceptron Rule via OGD



Truncated Hinge loss

PERCEPTRON

$y^t \in \{-1, +1\}$

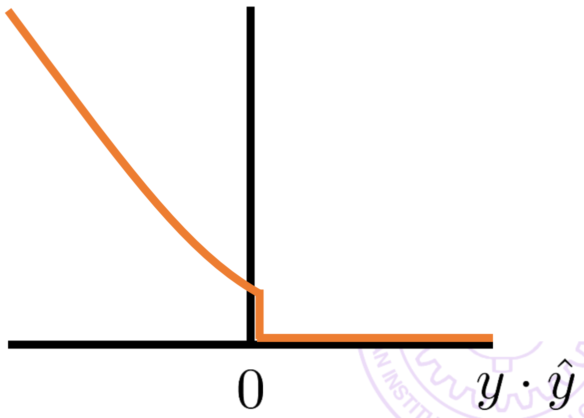
1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t \cdot y^t \cdot \mathbf{x}^t$
5. Repeat

Make updates only when making a mistake

Push $\mathbf{w}^t$ "away"/"toward" $\mathbf{x}^t$ depending on mistake

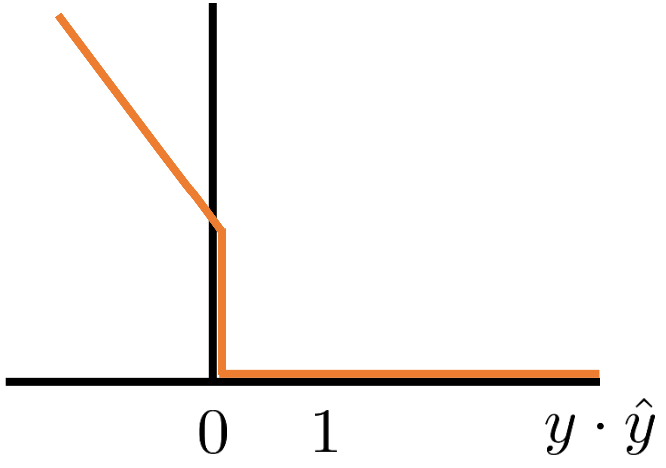
Perceptron Logistic Regression??

Perceptron Crammer Singer for multi-classification?



App: The Perceptron Rule via OGD

Can do stochastic perceptron too



Truncated Hinge loss

PERCEPTRON

$y^t \in \{-1, +1\}$

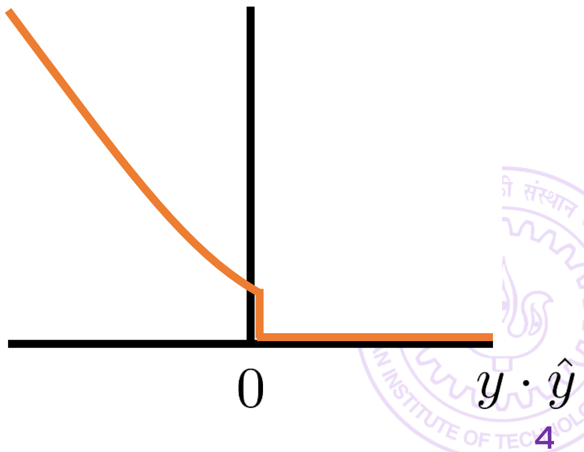
1. Initialize $\mathbf{w}^0$
2. Receive a data point $\mathbf{z}^t = (y^t, \mathbf{x}^t)$
3. If $y^t \langle \mathbf{w}^t, \mathbf{x}^t \rangle < 0$
4. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t \cdot y^t \cdot \mathbf{x}^t$
5. Repeat

Make updates only when making a mistake

Push $\mathbf{w}^t$ "away"/"toward" $\mathbf{x}^t$ depending on mistake

Perceptron Logistic Regression??

Perceptron Crammer Singer for multi-classification?



Coordinate Descent

Coordinate Descent

- Similar to gradient descent except update one variable at a time
- Notation: $\nabla_i f(\mathbf{w}) = [\nabla f(\mathbf{w})]_\mathrm{i}$ i.e. the i-th directional derivative

Coordinate Descent

- Similar to gradient descent except update one variable at a time
- Notation: $\nabla_i f(\mathbf{w}) = [\nabla f(\mathbf{w})]_{\mathrm{i}}$ i.e. the i-th directional derivative

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \; f(\mathbf{w}) + r(\mathbf{w})$$

Coordinate Descent

- Similar to gradient descent except update one variable at a time
- Notation: $\nabla_i f(\mathbf{w}) = [\nabla f(\mathbf{w})]_{\mathrm{i}}$ i.e. the i-th directional derivative

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$

COORDINATE DESCENT

1. Initialize $\mathbf{w}^0 \in \mathbb{R}^d$
2. Select a coordinate $j_t \in [d]$
3. Let $\mathbf{g}_{j_t}^t \leftarrow \nabla_{\mathrm{j_t}} f(\mathbf{w}^t) + \nabla_{\mathrm{j_t}} r(\mathbf{w}^t) \in \mathbb{R}$
4. Update $\mathbf{w}_{j_t}^{t+1} \leftarrow \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$
5. Preserve other coord. $\mathbf{w}_k^{t+1} \leftarrow \mathbf{w}_k^t,\ k \neq j_t$
6. Repeat until convergence

Coordinate Descent

- Similar to gradient descent except update one variable at a time
- Notation: $\nabla_i f(\mathbf{w}) = [\nabla f(\mathbf{w})]_\mathrm{i}$ i.e. the i-th directional derivative

$$\hat{\mathbf{w}} = \underset{\mathbf{w} \in \mathbb{R}^d}{\arg\min}\ f(\mathbf{w}) + r(\mathbf{w})$$

Random (Stochastic CD), Cyclic $1,2,\ldots,d,1,2,3\ldots d,\ldots$

COORDINATE DESCENT

1. Initialize $\mathbf{w}^0 \in \mathbb{R}^d$
2. Select a coordinate $j_t \in [d]$
3. Let $\mathbf{g}_{j_t}^t \leftarrow \nabla_{\mathrm{j_t}} f(\mathbf{w}^t) + \nabla_{\mathrm{j_t}} r(\mathbf{w}^t) \in \mathbb{R}$
4. Update $\mathbf{w}_{j_t}^{t+1} \leftarrow \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$
5. Preserve other coord. $\mathbf{w}_k^{t+1} \leftarrow \mathbf{w}_k^t,\ k \neq j_t$
6. Repeat until convergence

Coordinate Descent

- Similar to gradient descent except update one variable at a time
- Notation: $\nabla_i f(\mathbf{w}) = [\nabla f(\mathbf{w})]_\mathrm{i}$ i.e. the i-th directional derivative

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \; f(\mathbf{w}) + r(\mathbf{w})$$

COORDINATE DESCENT

1. Initialize $\mathbf{w}^0 \in \mathbb{R}^d$
2. Select a coordinate $j_t \in [d]$
3. Let $\mathbf{g}^t_{j_t} \leftarrow \nabla_{\mathrm{j_t}} f(\mathbf{w}^t) + \nabla_{\mathrm{j_t}} r(\mathbf{w}^t) \in \mathbb{R}$
4. Update $\mathbf{w}^{t+1}_{j_t} \leftarrow \mathbf{w}^t_{j_t} - \eta_t \cdot \mathbf{g}^t_{j_t}$
5. Preserve other coord. $\mathbf{w}^{t+1}_k \leftarrow \mathbf{w}^t_k,\ k \neq j_t$
6. Repeat until convergence

Random (Stochastic CD), Cyclic $1,2,\ldots,d,1,2,3\ldots d,\ldots$

Only $O(n)$ time per iter!

Coordinate Descent

- Similar to gradient descent except update one variable at a time
- Notation: $\nabla_i f(\mathbf{w}) = [\nabla f(\mathbf{w})]_\mathrm{i}$ i.e. the i-th directional derivative

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$

Useful when $d \gg n$

Random (Stochastic CD), Cyclic $1,2,\ldots,d,1,2,3\ldots d,\ldots$

Only $O(n)$ time per iter!

COORDINATE DESCENT

1. Initialize $\mathbf{w}^0 \in \mathbb{R}^d$
2. Select a coordinate $j_t \in [d]$
3. Let $\mathbf{g}^t_{j_t} \leftarrow \nabla_{\mathrm{j_t}} f(\mathbf{w}^t) + \nabla_{\mathrm{j_t}} r(\mathbf{w}^t) \in \mathbb{R}$
4. Update $\mathbf{w}^{t+1}_{j_t} \leftarrow \mathbf{w}^t_{j_t} - \eta_t \cdot \mathbf{g}^t_{j_t}$
5. Preserve other coord. $\mathbf{w}^{t+1}_k \leftarrow \mathbf{w}^t_k,\ k \neq j_t$
6. Repeat until convergence

Coordinate Descent

- Similar to gradient descent except update one variable at a time
- Notation: $\nabla_i f(\mathbf{w}) = [\nabla f(\mathbf{w})]_\mathrm{i}$ i.e. the i-th directional derivative

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w}) + r(\mathbf{w})$$

Useful when $d \gg n$

Random (Stochastic CD), Cyclic $1,2,\dots,d,1,2,3\dots d,\dots$

COORDINATE DESCENT

1. Initialize $\mathbf{w}^0 \in \mathbb{R}^d$
2. Select a coordinate $j_t \in [d]$
3. Let $\mathbf{g}^t_{j_t} \leftarrow \nabla_{\mathrm{j_t}} f(\mathbf{w}^t) + \nabla_{\mathrm{j_t}} r(\mathbf{w}^t) \in \mathbb{R}$
4. Update $\mathbf{w}^{t+1}_{j_t} \leftarrow \mathbf{w}^t_{j_t} - \eta_t \cdot \mathbf{g}^t_{j_t}$
5. Preserve other coord. $\mathbf{w}^{t+1}_k \leftarrow \mathbf{w}^t_k,\, k \neq j_t$
6. Repeat until convergence

Only $O(n)$ time per iter!

Can work with a block of coordinates instead

Coordinate Descent

- Similar to gradient descent except update one variable at a time
- Notation: $\nabla_i f(\mathbf{w}) = [\nabla f(\mathbf{w})]_{\mathrm{i}}$ i.e. the i-th directional derivative

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$

Useful when $d \gg n$

COORDINATE DESCENT

1. Initialize $\mathbf{w}^0 \in \mathbb{R}^d$
2. Select a coordinate $j_t \in [d]$
3. Let $\mathbf{g}^t_{j_t} \leftarrow \nabla_{\mathrm{j_t}} f(\mathbf{w}^t) + \nabla_{\mathrm{j_t}} r(\mathbf{w}^t) \in \mathbb{R}$
4. Update $\mathbf{w}^{t+1}_{j_t} \leftarrow \mathbf{w}^t_{j_t} - \eta_t \cdot \mathbf{g}^t_{j_t}$
5. Preserve other coord. $\mathbf{w}^{t+1}_k \leftarrow \mathbf{w}^t_k,\ k \neq j_t$
6. Repeat until convergence

Random (Stochastic CD), Cyclic $1,2,\ldots,d,1,2,3\ldots d,\ldots$

Only $O(n)$ time per iter!

Block Coord. Descent

Can work with a block of coordinates instead

Coordinate Descent

- Similar to gradient descent except update one variable at a time
- Notation: $\nabla_i f(\mathbf{w}) = [\nabla f(\mathbf{w})]_{\mathrm{i}}$ i.e. the i-th directional derivative

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$

Useful when $d \gg n$

Somewhat like mini-batch SGD

Block Coord. Descent

Can work with a block of coordinates instead

Random (Stochastic CD), Cyclic $1,2,\ldots,d,1,2,3\ldots d,\ldots$

Only $O(n)$ time per iter!

COORDINATE DESCENT

1. Initialize $\mathbf{w}^0 \in \mathbb{R}^d$
2. Select a coordinate $j_t \in [d]$
3. Let $\mathbf{g}^t_{j_t} \leftarrow \nabla_{\mathrm{j_t}} f(\mathbf{w}^t) + \nabla_{\mathrm{j_t}} r(\mathbf{w}^t) \in \mathbb{R}$
4. Update $\mathbf{w}^{t+1}_{j_t} \leftarrow \mathbf{w}^t_{j_t} - \eta_t \cdot \mathbf{g}^t_{j_t}$
5. Preserve other coord. $\mathbf{w}^{t+1}_k \leftarrow \mathbf{w}^t_k, k \neq j_t$
6. Repeat until convergence

App: Linear Regression via CD

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \left\langle \mathbf{w}, \mathbf{x}^i \right\rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

App: Linear Regression via CD

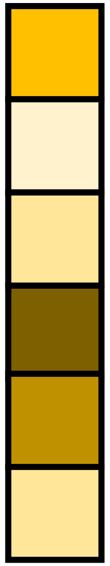
$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}^t = 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}^i + 2\lambda \cdot \mathbf{w}^t$$

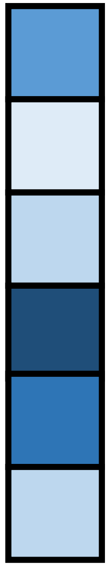
App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

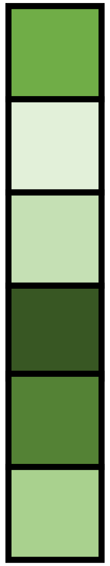
$$\mathbf{g}^t = 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}^i + 2\lambda \cdot \mathbf{w}^t$$



$$= 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot$$



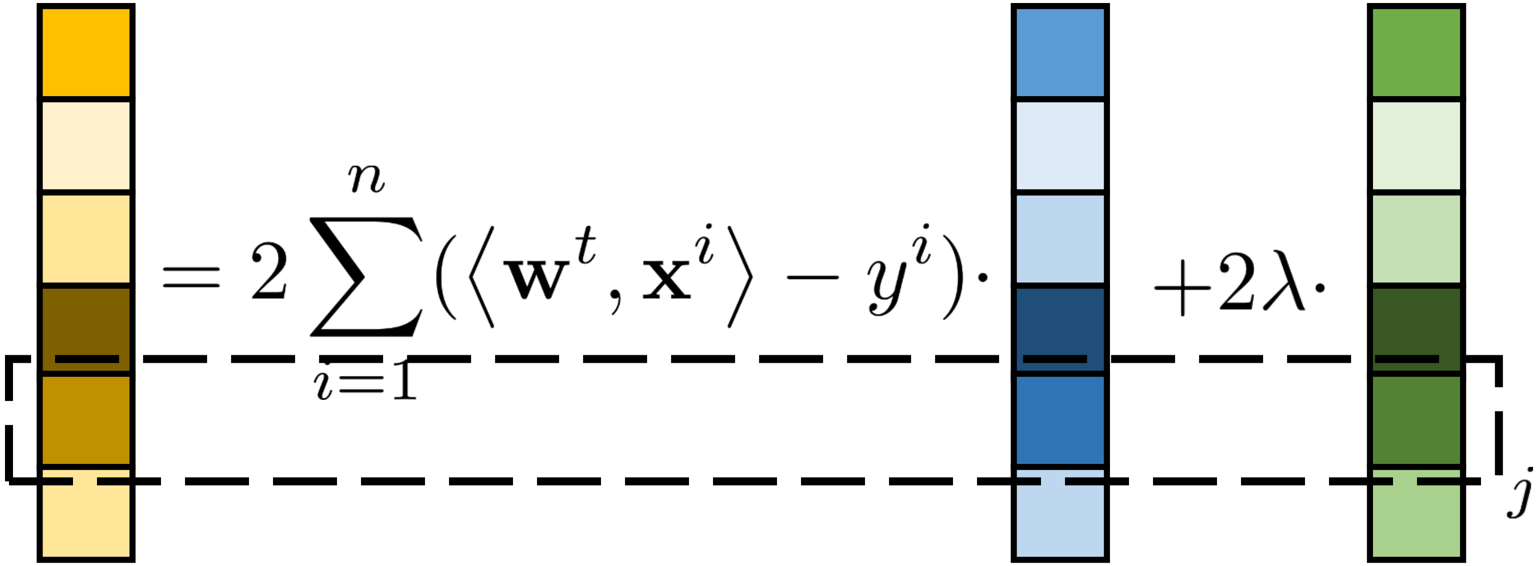
$$+2\lambda \cdot$$



App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

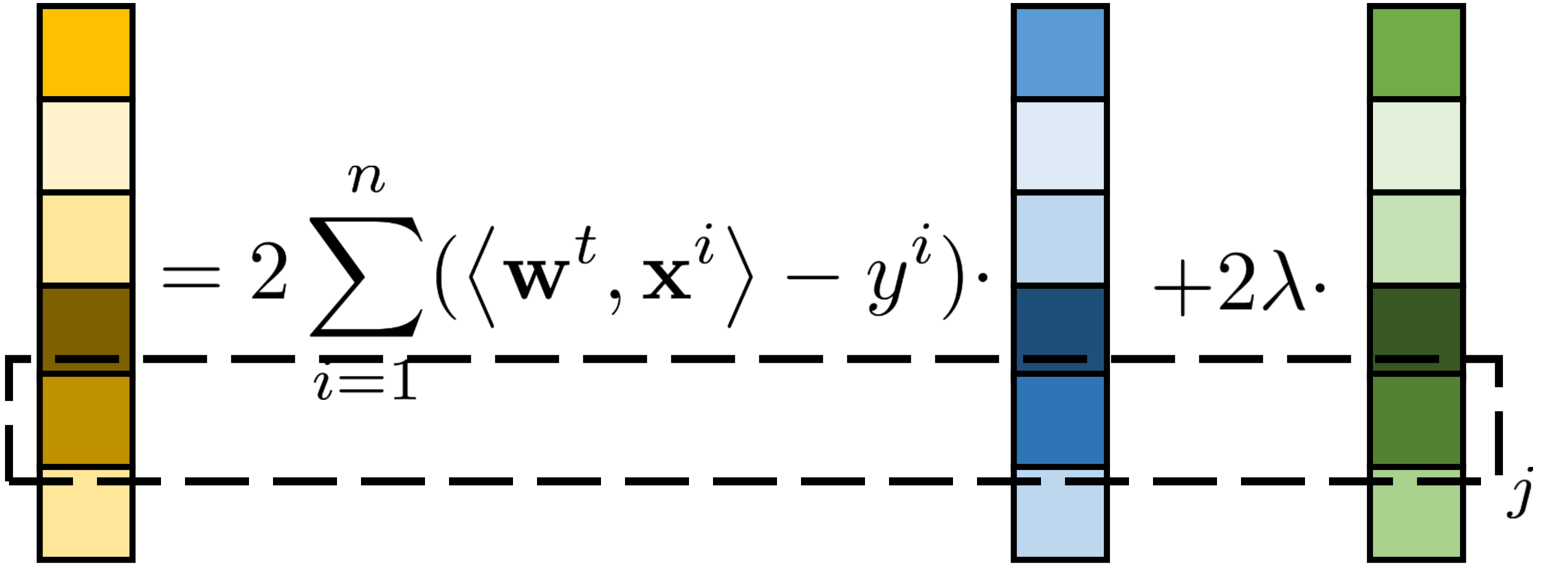
$$\mathbf{g}^t = 2\sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}^i + 2\lambda \cdot \mathbf{w}^t$$



App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

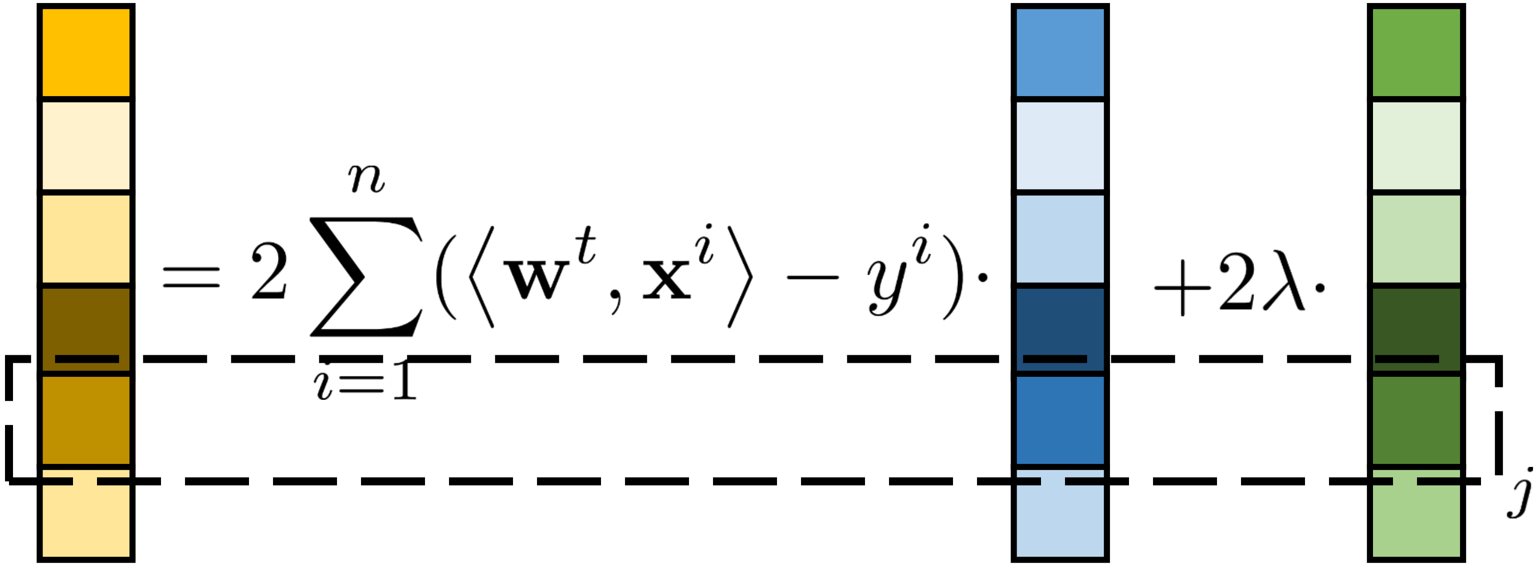
$$\mathbf{g}_j^t = 2\sum_{i=1}^{n} (\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}^i + 2\lambda \cdot \mathbf{w}^t$$



App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}^t$$

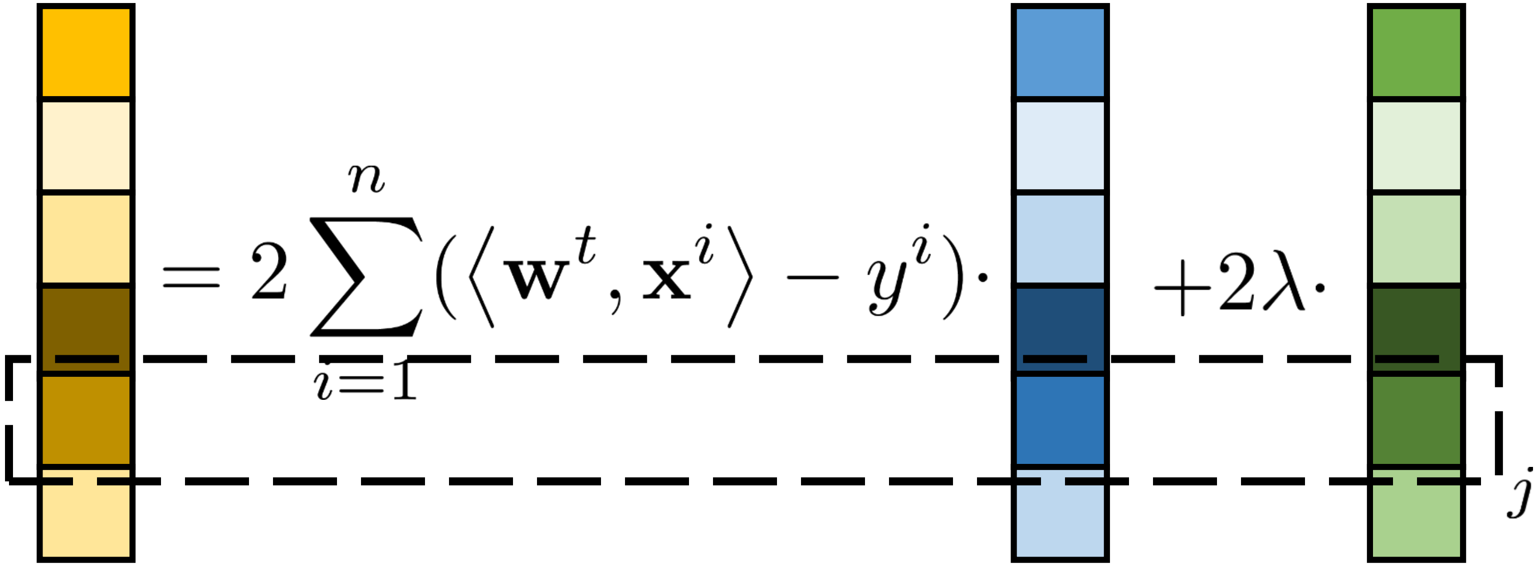


$$= 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \quad +2\lambda \cdot$$

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2 \sum_{i=1}^{n} (\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

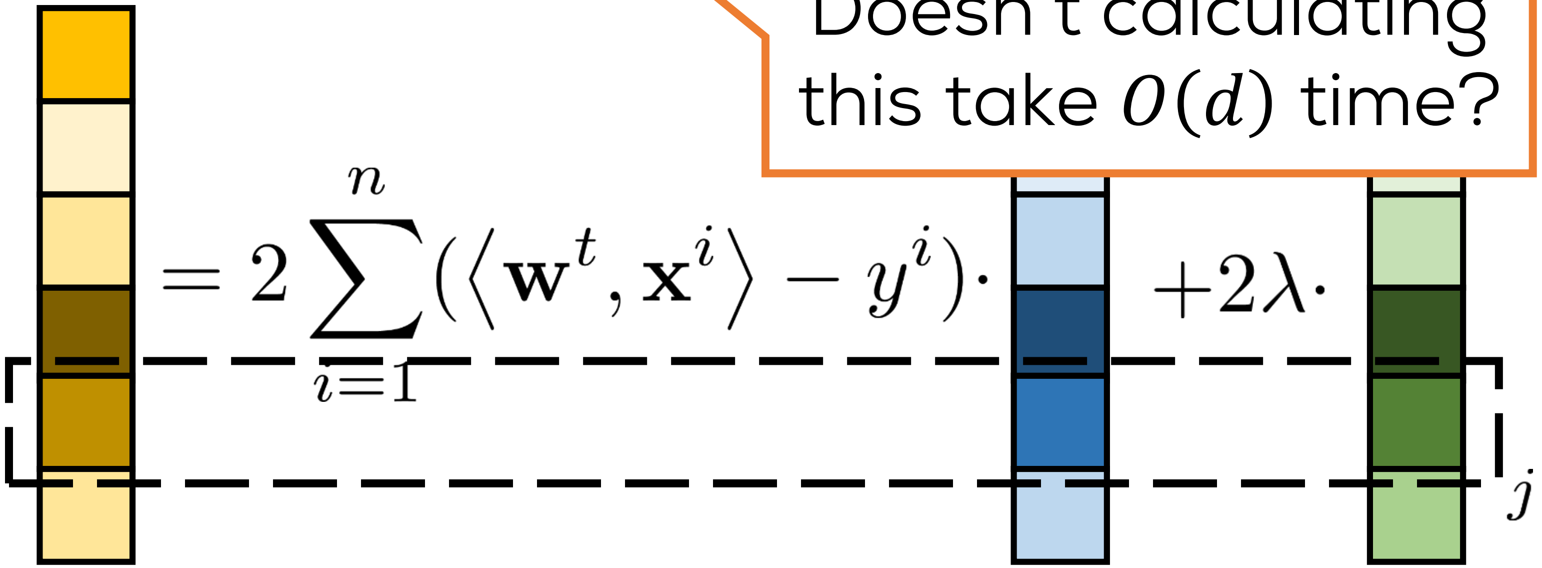


App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2 \sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

Doesn't calculating this take $O(d)$ time?



App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2 \sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

Doesn't calculating this take $O(d)$ time?

$$= 2 \sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \qquad + 2\lambda \cdot$$

j

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n} (\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

Doesn't calculating this take $O(d)$ time?

Only if you are not careful

$$= 2\sum_{i=1}^{n} (\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \quad + 2\lambda \cdot$$

j

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2 \sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0}$$

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2 \sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0}$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0$$

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0}$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0$$

$O(n)$ time

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0$$

$O(n)$ time

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$\mathcal{O}(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle$$

$\mathcal{O}(n)$ time

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2 \sum_{i=1}^{n} (\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle$$

$O(n)$ time

Available

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle$$

$O(n)$ time

Available

Assumed

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

$\mathbf{w}^0 = \mathbf{0}$ $\qquad$ $\mathbf{w}^t$ $\qquad$ $\mathbf{w}_{j_t}^{t+1} = \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$

$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0$ $\qquad$ $\langle \mathbf{w}^t, \mathbf{x}^i \rangle$

$O(n)$ time

Available

Assumed

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2 \sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t \qquad \mathbf{w}_{j_t}^{t+1} = \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle \qquad \langle \mathbf{w}^{t+1}, \mathbf{x}^i \rangle = \langle \mathbf{w}^t, \mathbf{x}^i \rangle - \eta_t \cdot \mathbf{g}_{j_t}^t \cdot \mathbf{x}_{j_t}^i$$

$O(n)$ time

Available

Assumed

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$\mathcal{O}(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t \qquad \mathbf{w}_{j_t}^{t+1} = \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle \qquad \langle \mathbf{w}^{t+1}, \mathbf{x}^i \rangle = \langle \mathbf{w}^t, \mathbf{x}^i \rangle - \eta_t \cdot \mathbf{g}_{j_t}^t \cdot \mathbf{x}_{j_t}^i$$

$\mathcal{O}(n)$ time

Available

Already available

Assumed

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2 \sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$\mathcal{O}(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t \qquad \mathbf{w}_{j_t}^{t+1} = \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle \qquad \langle \mathbf{w}^{t+1}, \mathbf{x}^i \rangle = \langle \mathbf{w}^t, \mathbf{x}^i \rangle - \eta_t \cdot \mathbf{g}_{j_t}^t \cdot \mathbf{x}_{j_t}^i$$

$\mathcal{O}(n)$ time

Available

Assumed

Already available

$\mathcal{O}(1)$ time per data point!

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n} (\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$\mathcal{O}(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t \qquad \mathbf{w}_{j_t}^{t+1} = \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle \qquad \langle \mathbf{w}^{t+1}, \mathbf{x}^i \rangle = \langle \mathbf{w}^t, \mathbf{x}^i \rangle - \eta_t \cdot \mathbf{g}_{j_t}^t \cdot \mathbf{x}_{j_t}^i$$

$\mathcal{O}(n)$ time

Available

Assumed

Already available

$\mathcal{O}(1)$ time per data point!

$\mathcal{O}(n)$ time in total

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$g_j^t = 2\sum_{i=1}^{n}(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(nd)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t \qquad \mathbf{w}_{j_t}^{t+1} = \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle \qquad \langle \mathbf{w}^{t+1}, \mathbf{x}^i \rangle = \langle \mathbf{w}^t, \mathbf{x}^i \rangle - \eta_t \cdot \mathbf{g}_{j_t}^t \cdot \mathbf{x}_{j_t}^i$$

$O(n)$ time

Available

Assumed

Already available

$O(1)$ time per data point!

Induction complete!

$O(n)$ time in total

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n} \left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(n)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t \qquad \mathbf{w}_{j_t}^{t+1} = \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle \qquad \langle \mathbf{w}^{t+1}, \mathbf{x}^i \rangle = \langle \mathbf{w}^t, \mathbf{x}^i \rangle - \eta_t \cdot \mathbf{g}_{j_t}^t \cdot \mathbf{x}_{j_t}^i$$

$O(n)$ time

Available

Assumed

Already available

$O(1)$ time per data point!

Induction complete!

$O(n)$ time in total

App: Linear Regression via CD

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2 + \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{g}_j^t = 2\sum_{i=1}^{n}\left(\langle \mathbf{w}^t, \mathbf{x}^i \rangle - y^i\right) \cdot \mathbf{x}_j^i + 2\lambda \cdot \mathbf{w}_j^t$$

$O(n)$ time in total

$$\mathbf{w}^0 = \mathbf{0} \qquad \mathbf{w}^t \qquad \mathbf{w}_{j_t}^{t+1} = \mathbf{w}_{j_t}^t - \eta_t \cdot \mathbf{g}_{j_t}^t$$

$$\langle \mathbf{w}^0, \mathbf{x}^i \rangle = 0 \qquad \langle \mathbf{w}^t, \mathbf{x}^i \rangle \qquad \langle \mathbf{w}^{t+1}, \mathbf{x}^i \rangle = \langle \mathbf{w}^t, \mathbf{x}^i \rangle - \eta_t \cdot \mathbf{g}_{j_t}^t \cdot \mathbf{x}_{j_t}^i$$

$O(n)$ time

Available

Assumed

Already available

$O(1)$ time per data point!

Induction complete!

$O(n)$ time in total

Handling Constraints

The Tale of Two Techniques

The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2$$

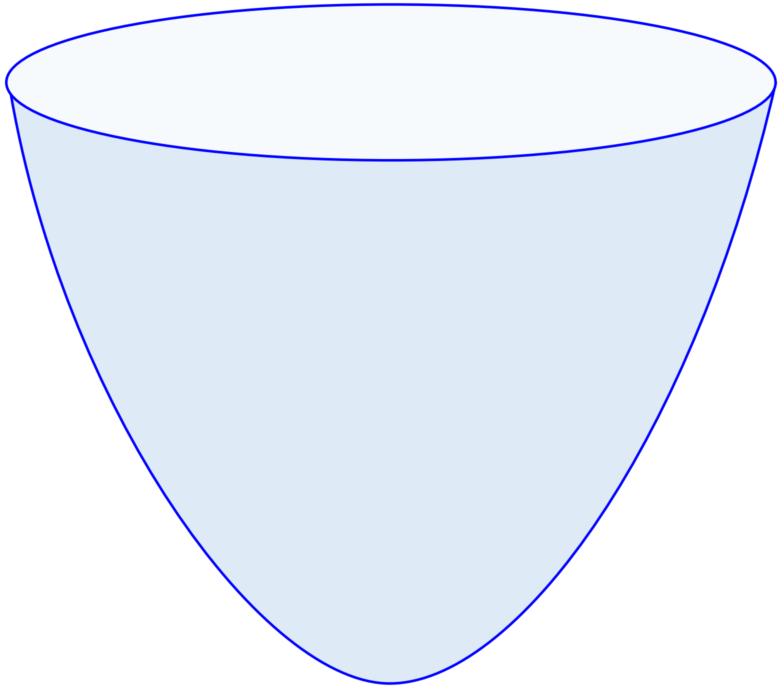
$$\text{s.t. } \|\mathbf{w}\|_2 \leq r$$

The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

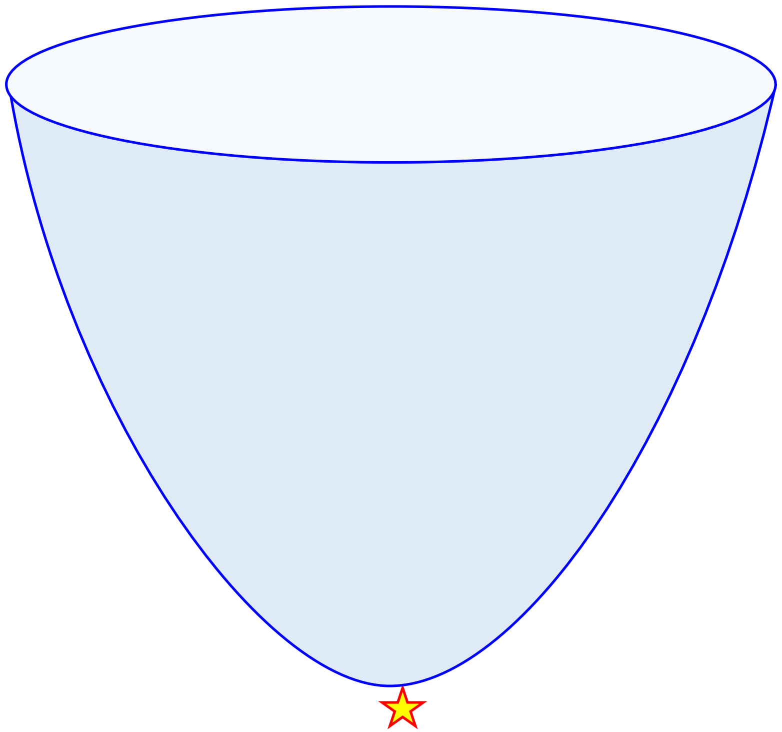
The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$



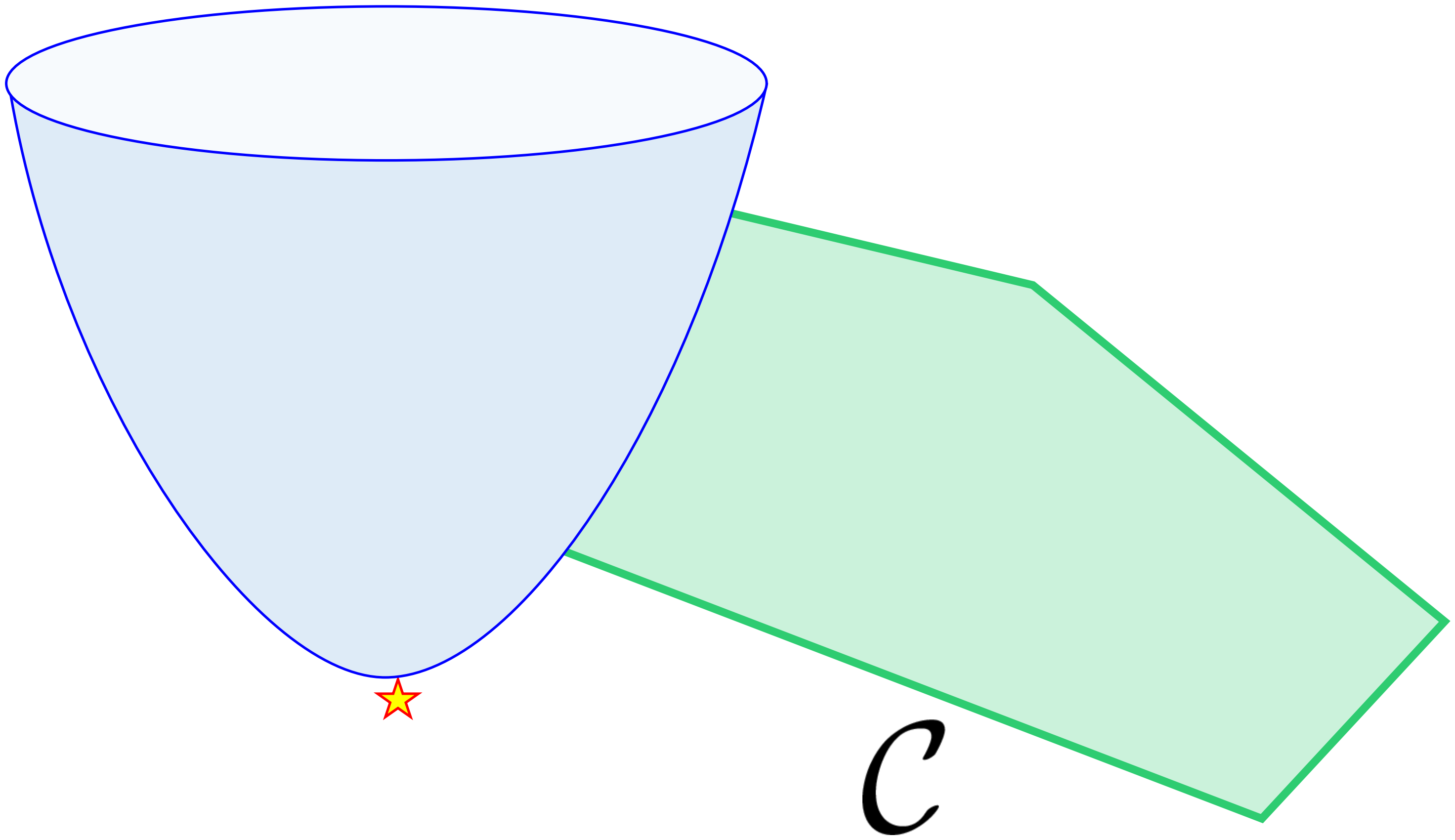
The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$



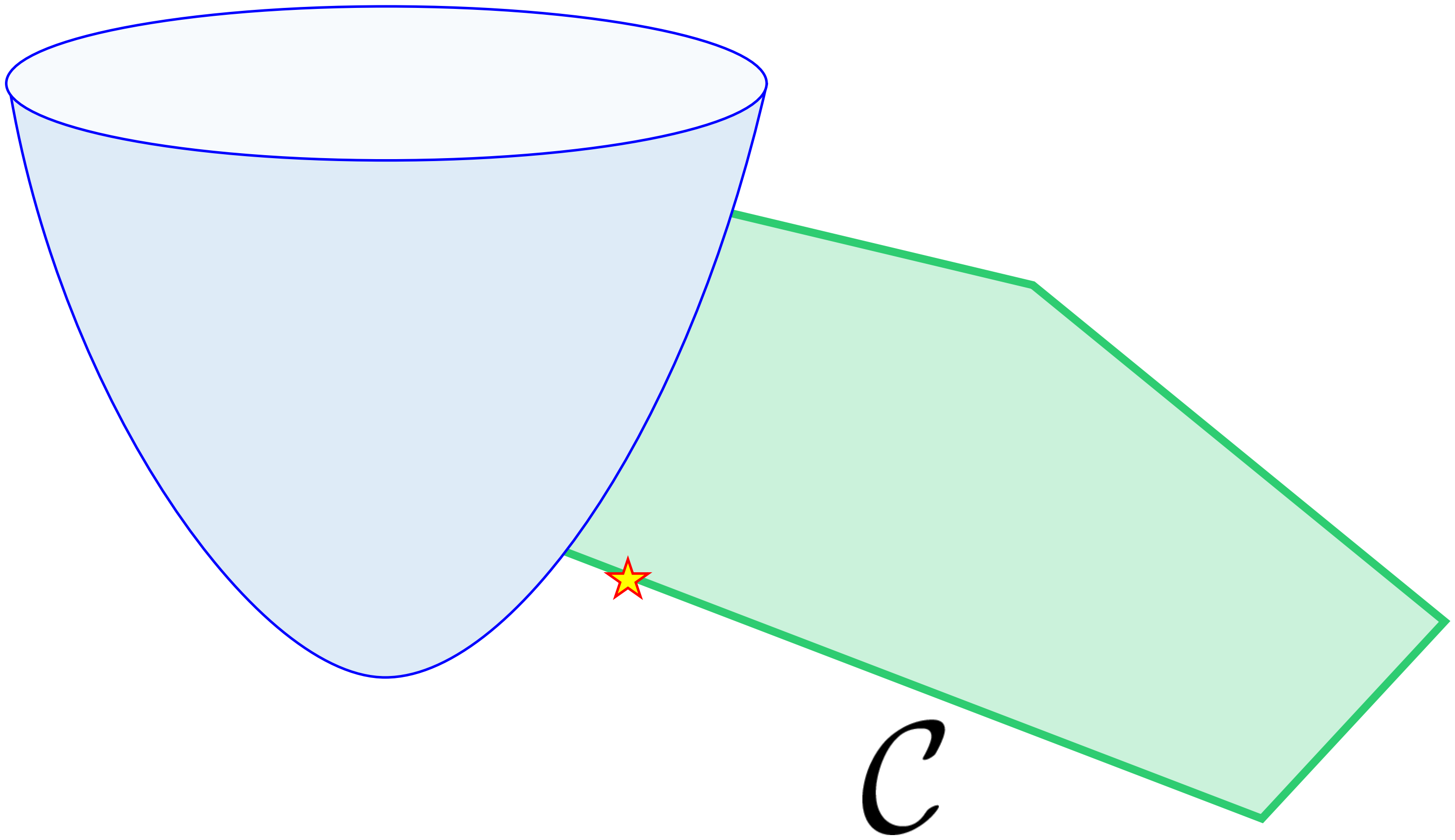
The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$



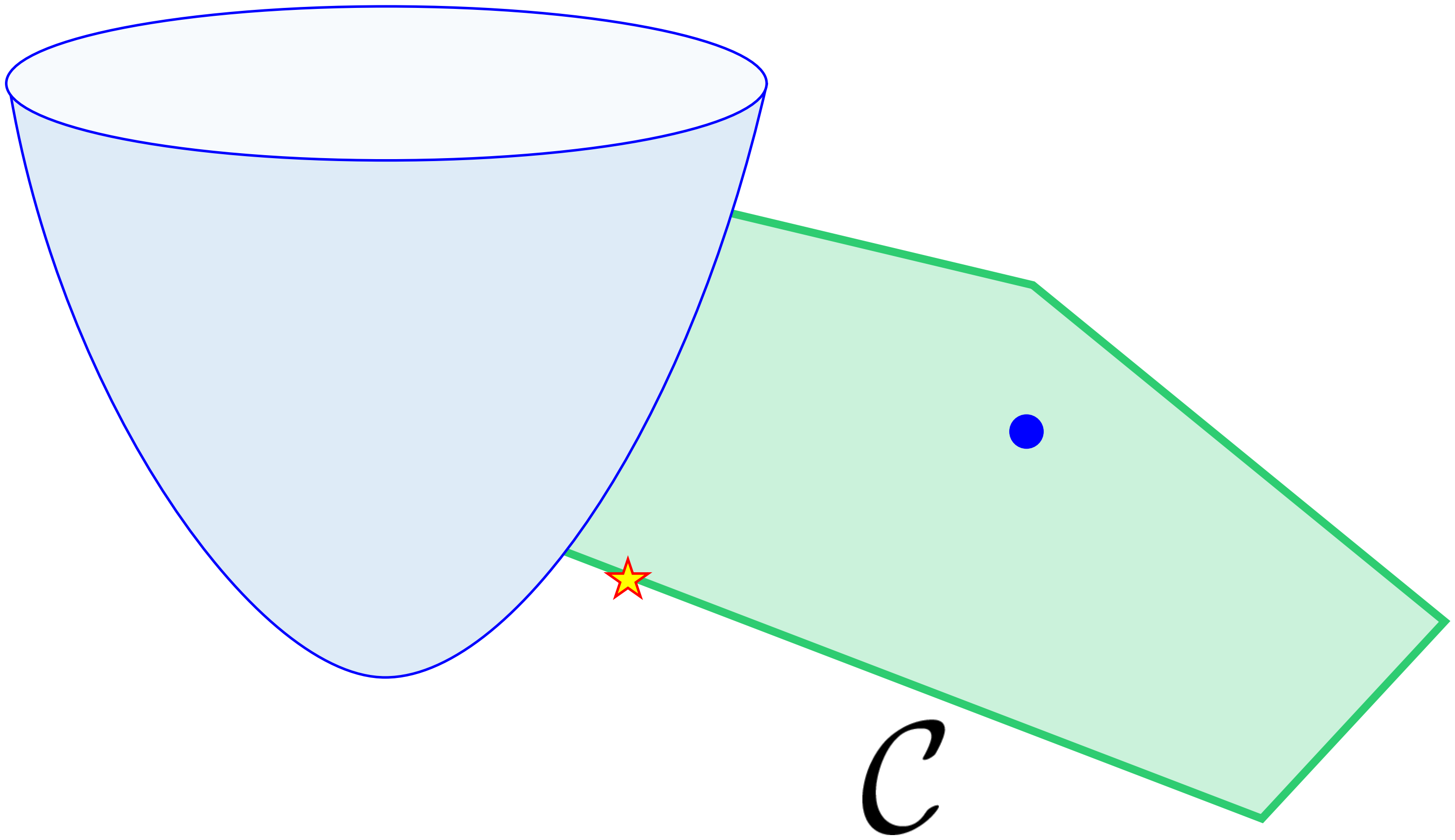
The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$



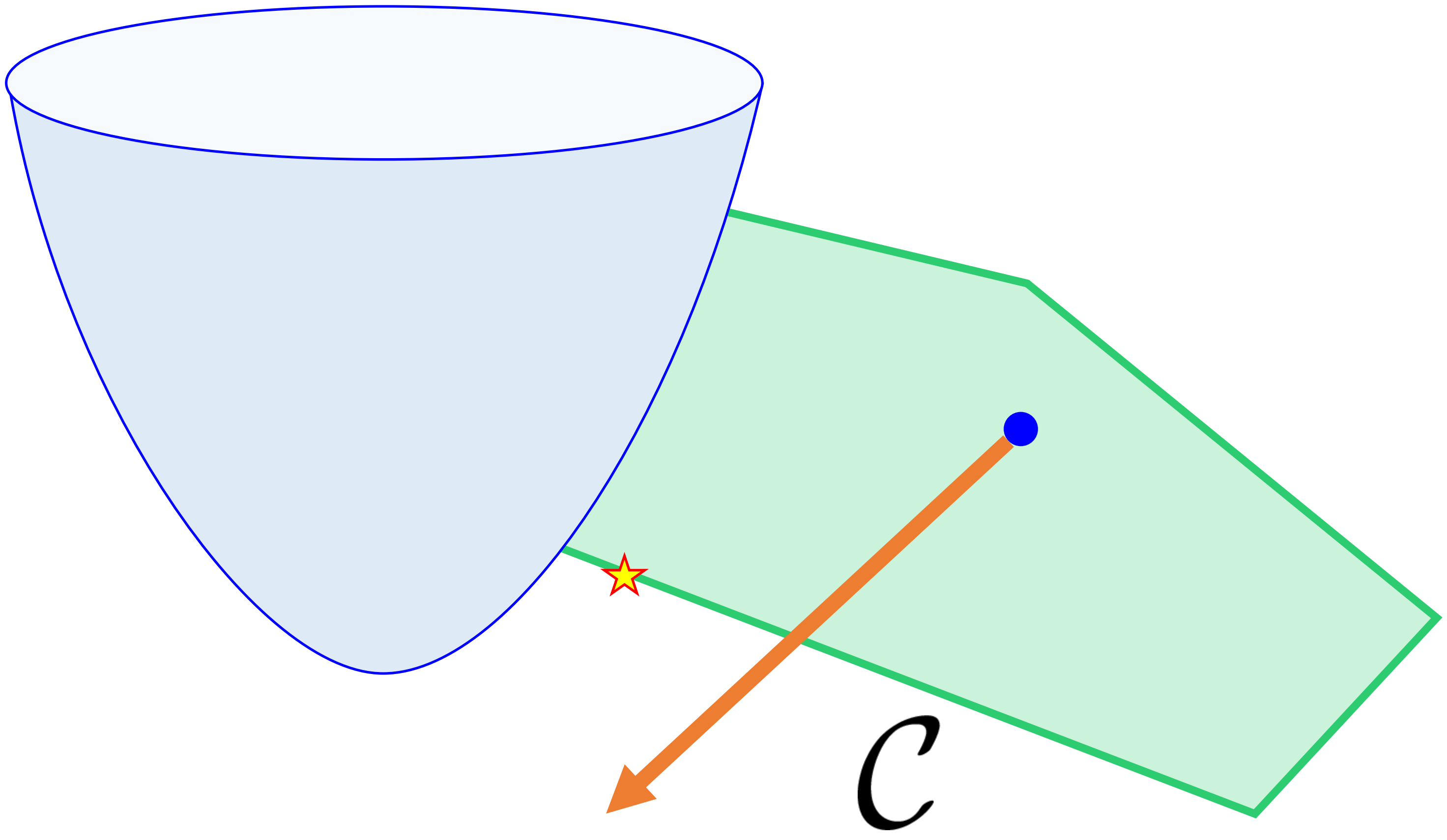
The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$



The Tale of Two Techniques

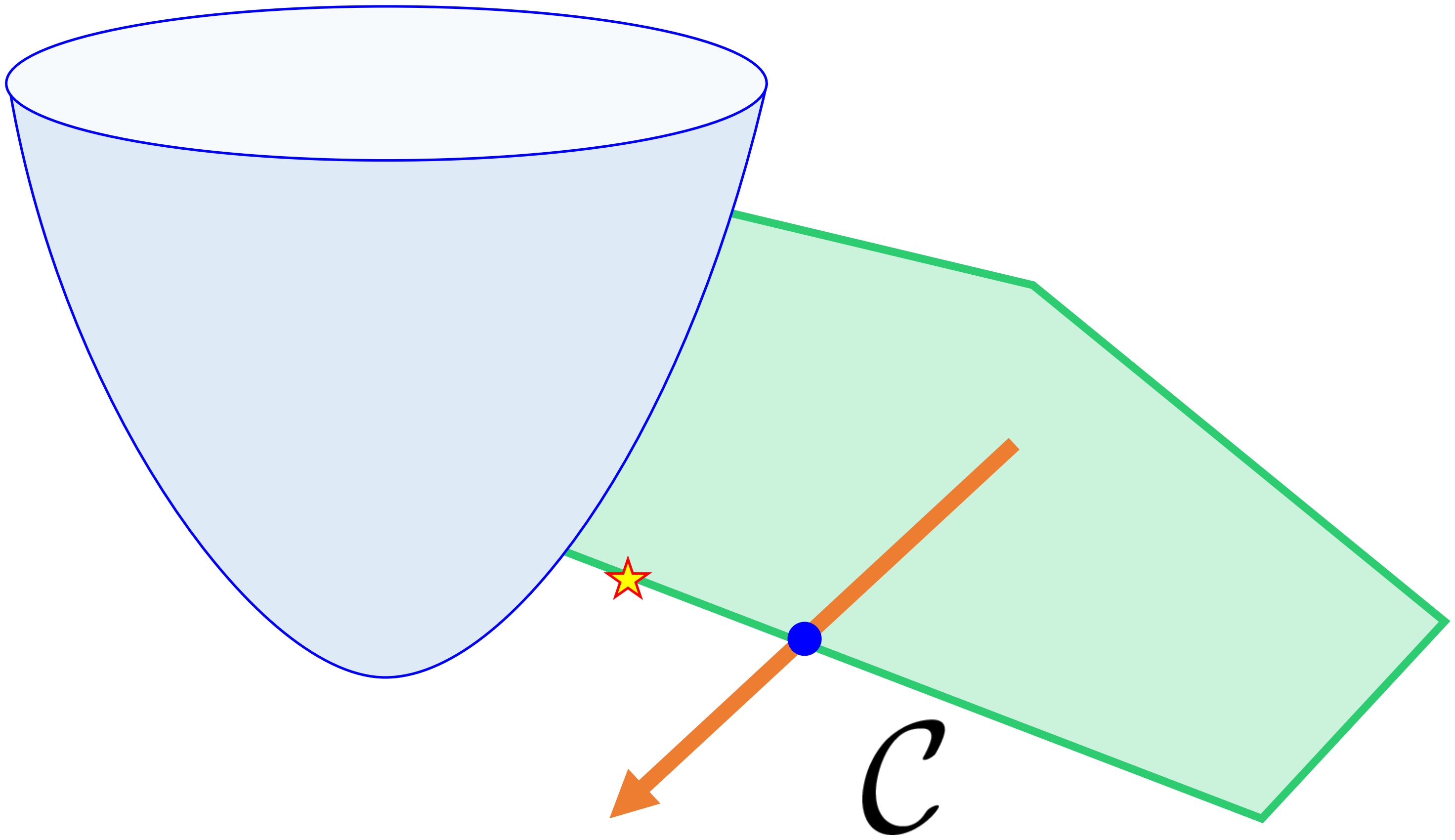
$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$



The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Interior Point Method

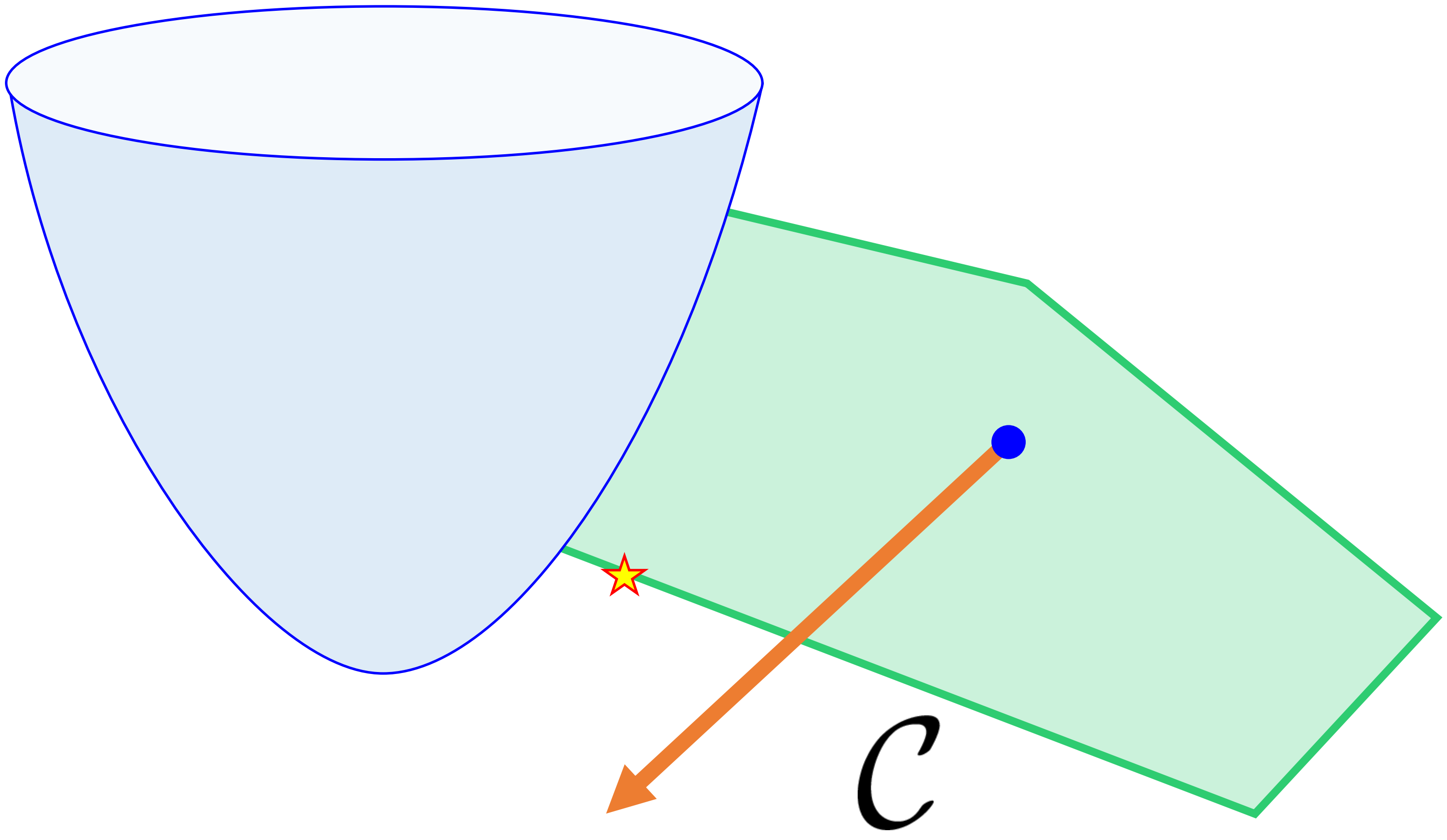


The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$

$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Interior Point Method

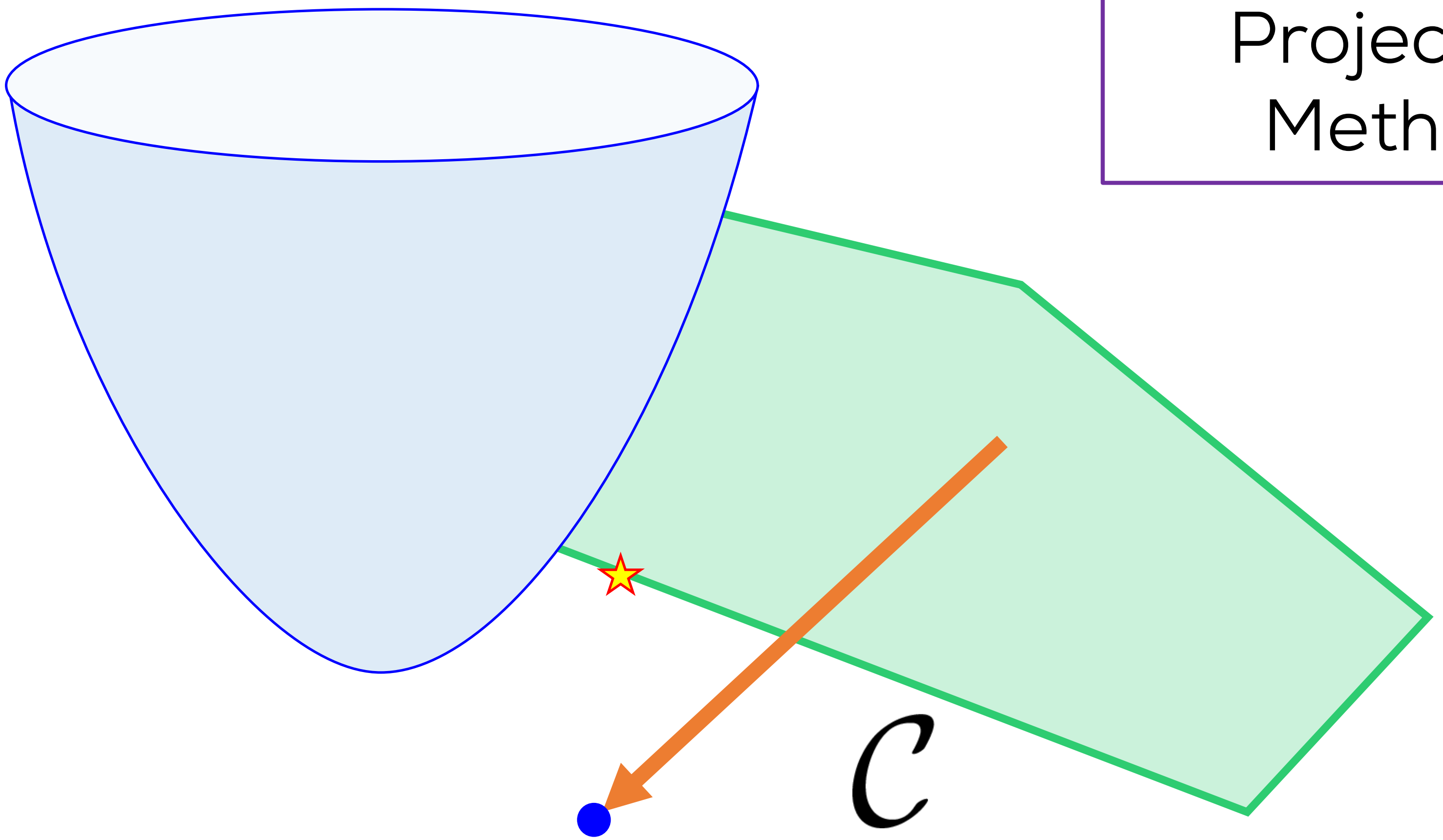


The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Interior Point Method

Projected Method

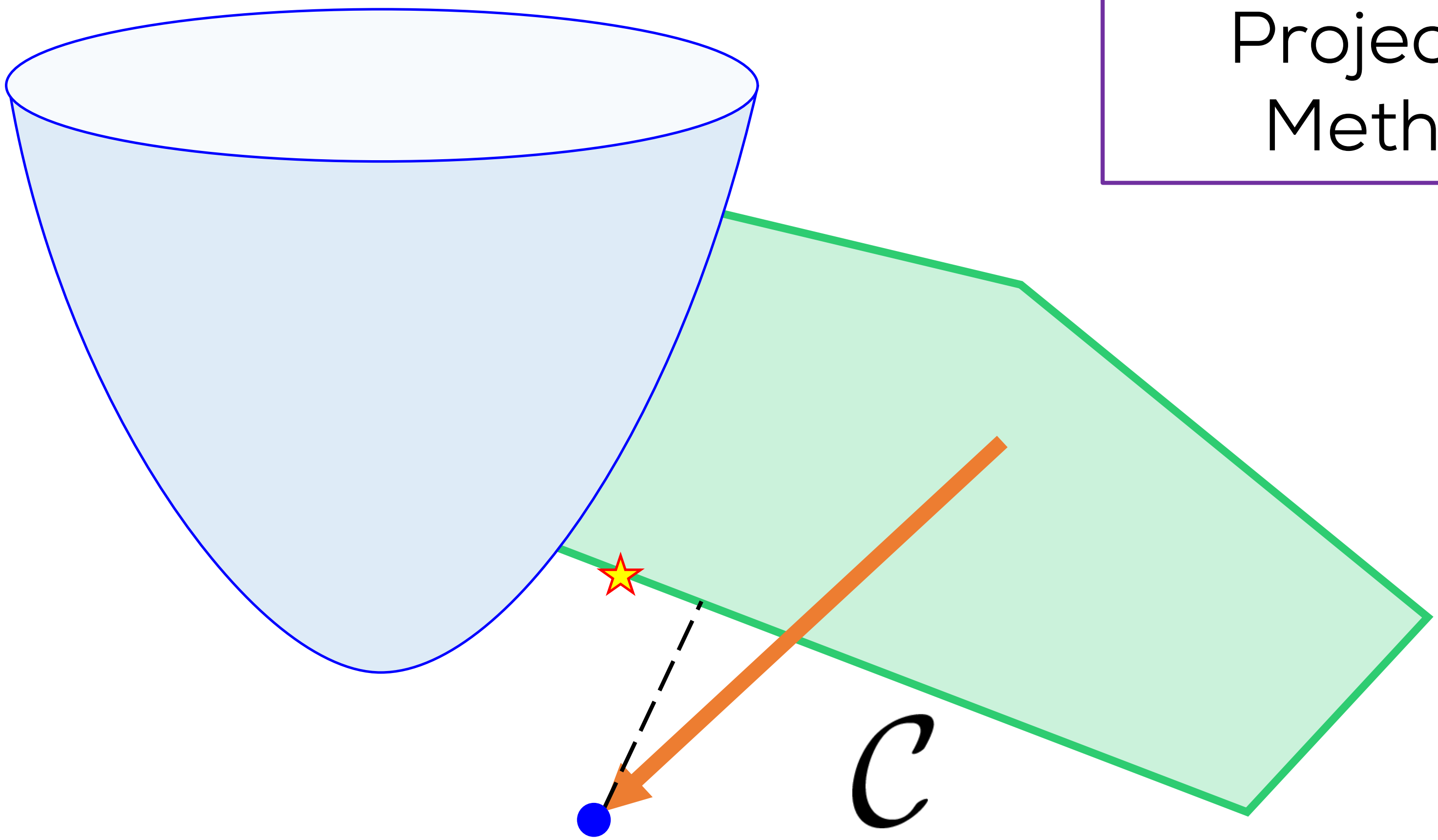


The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Interior Point Method

Projected Method

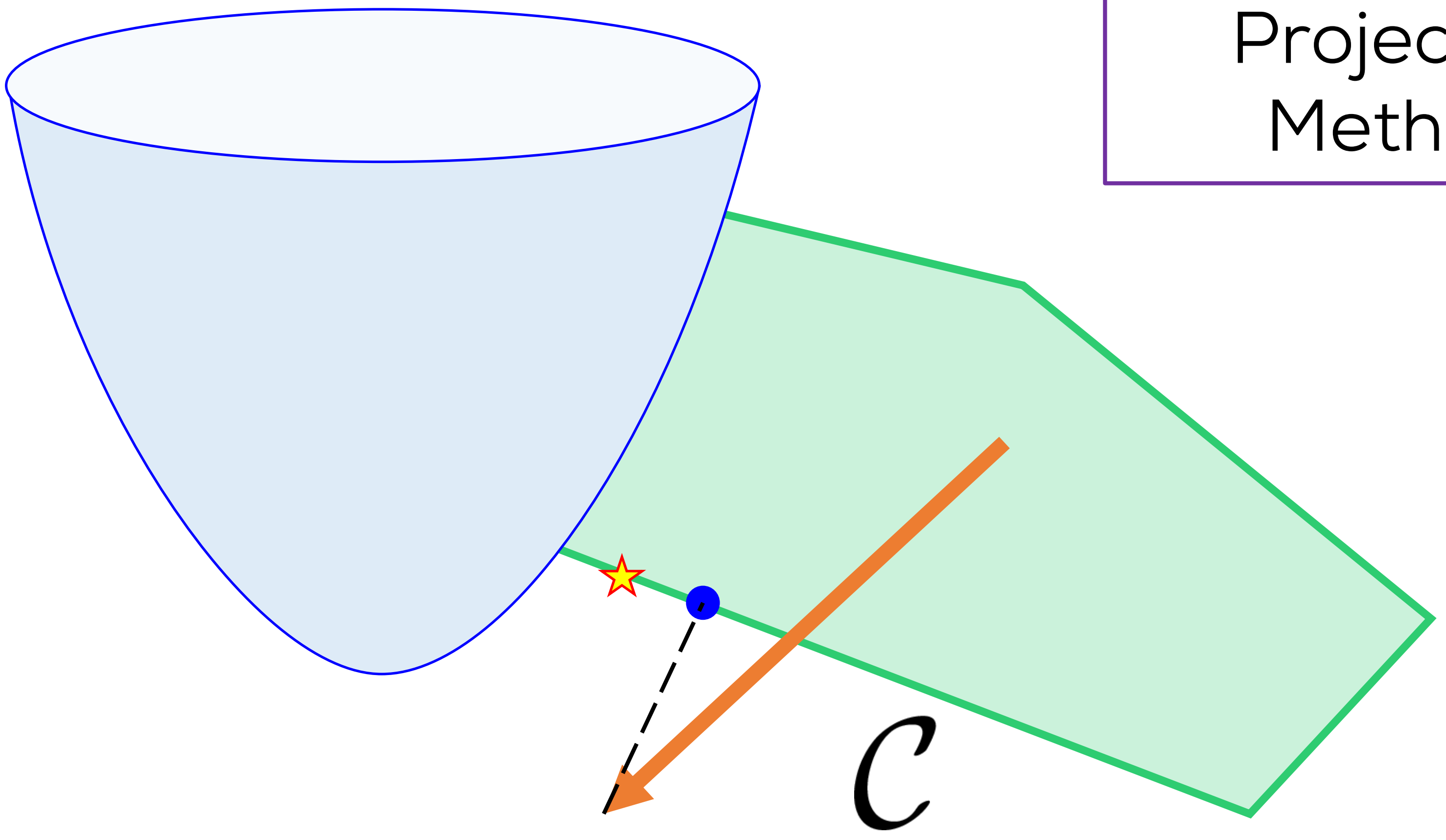


The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$

$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Interior Point Method



Projected Method

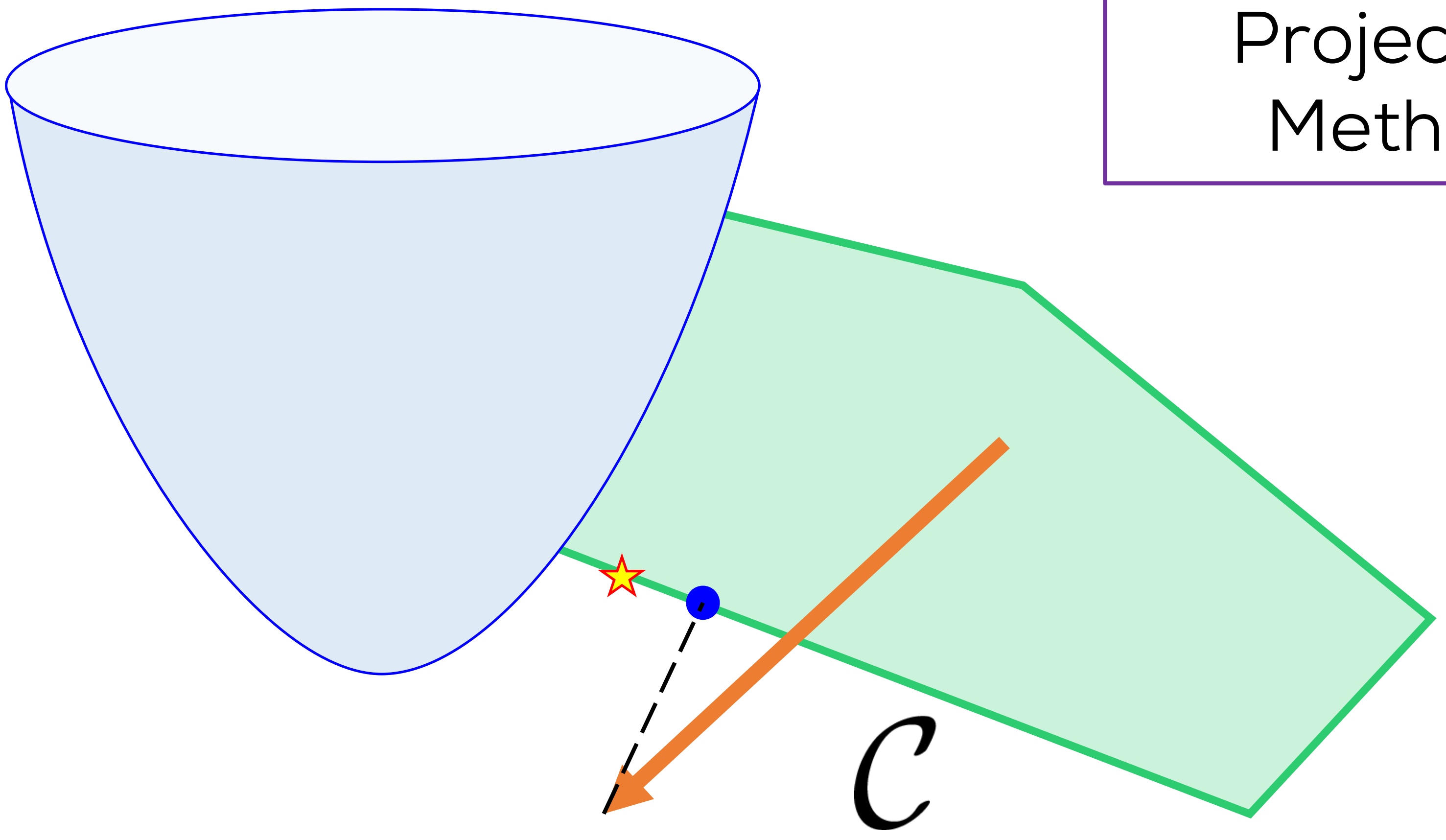
The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Interior Point Method

Powerful but can be slow, difficult to implement

Projected Method

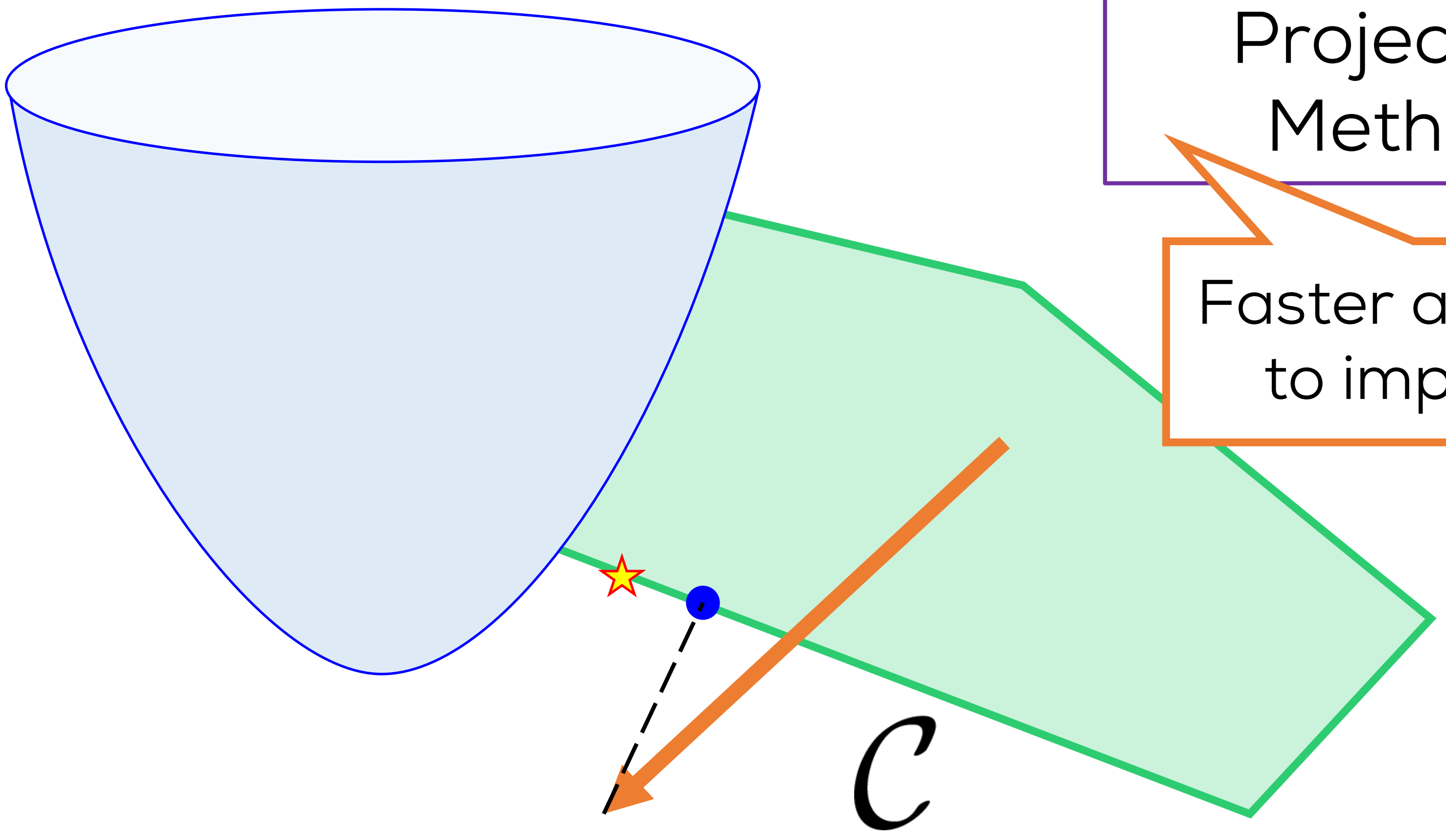


The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Interior Point Method

Powerful but can be slow, difficult to implement



Projected Method

Faster and easier to implement

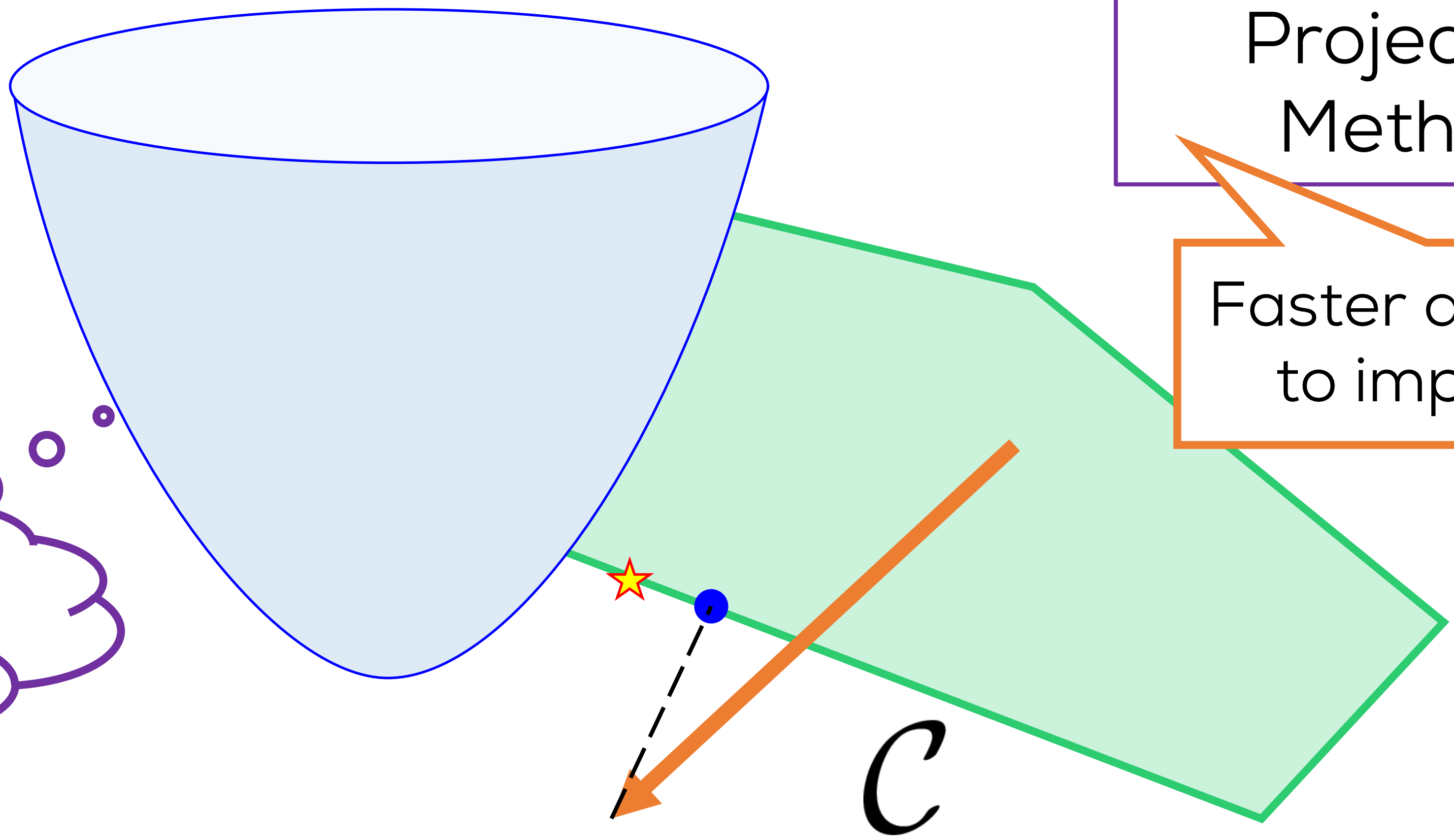
The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Interior Point Method

Powerful but can be slow, difficult to implement

What if the set is non-convex?



Projected Method

Faster and easier to implement

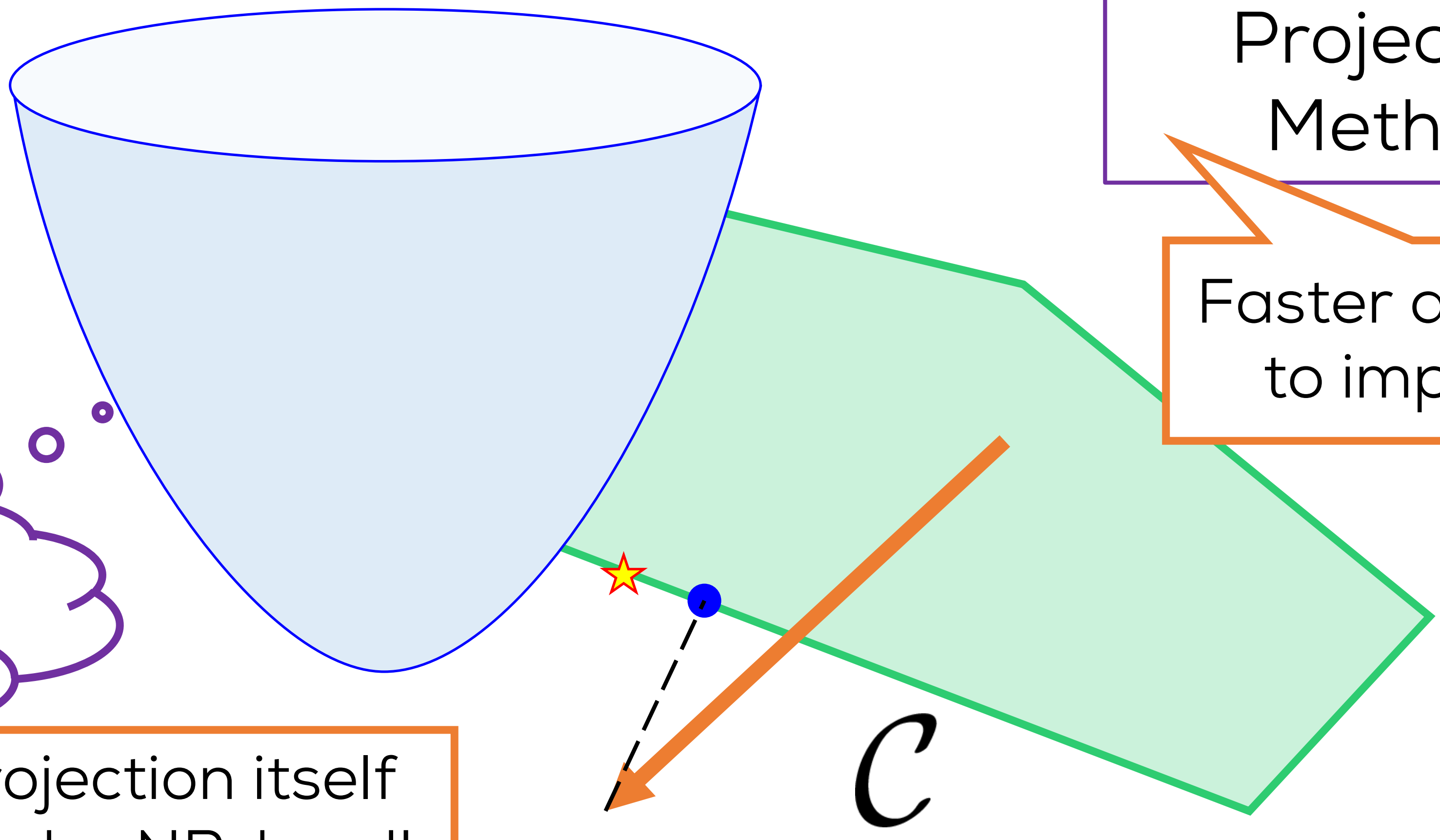
The Tale of Two Techniques

$$\hat{\mathbf{w}} = \arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Interior Point Method

Powerful but can be slow, difficult to implement

What if the set is non-convex?



Projected Method

Faster and easier to implement

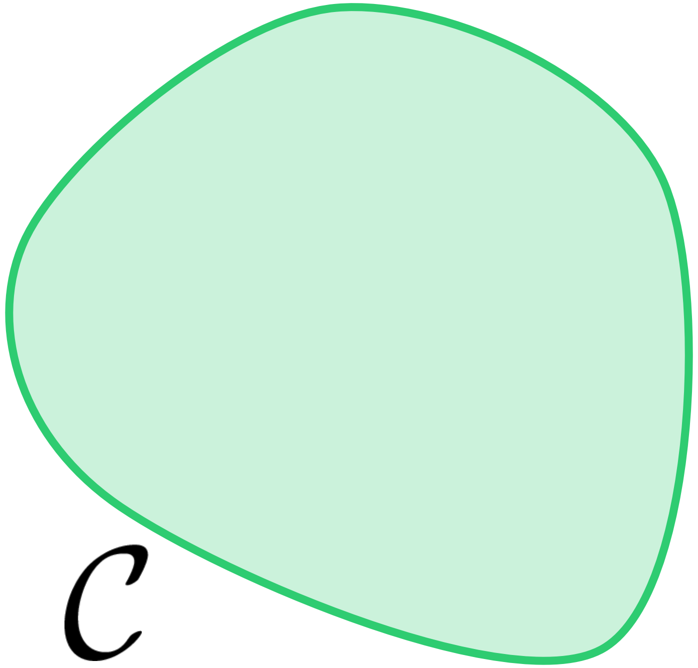
Projection itself can be NP-hard!

Projected Gradient Descent

Projected Gradient Descent

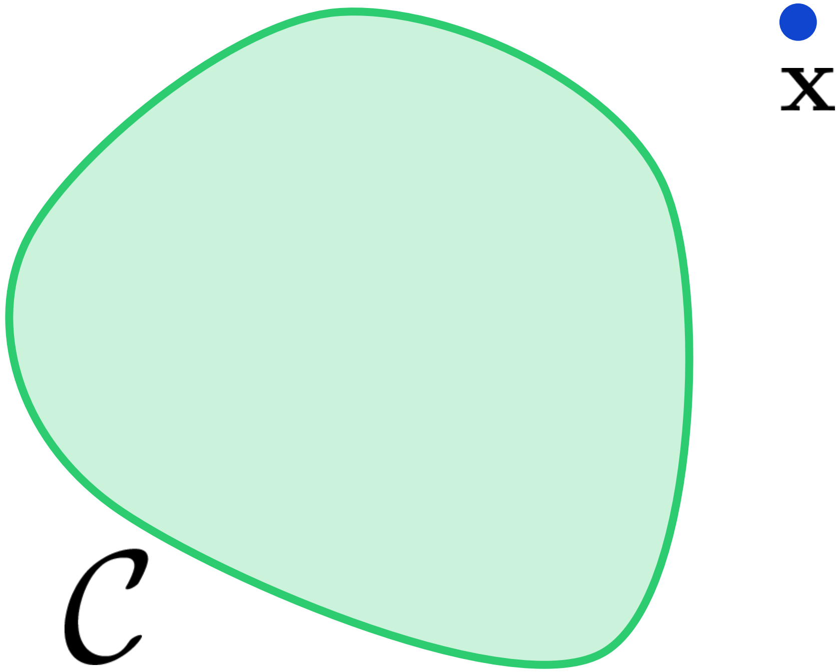
$$\arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Projected Gradient Descent



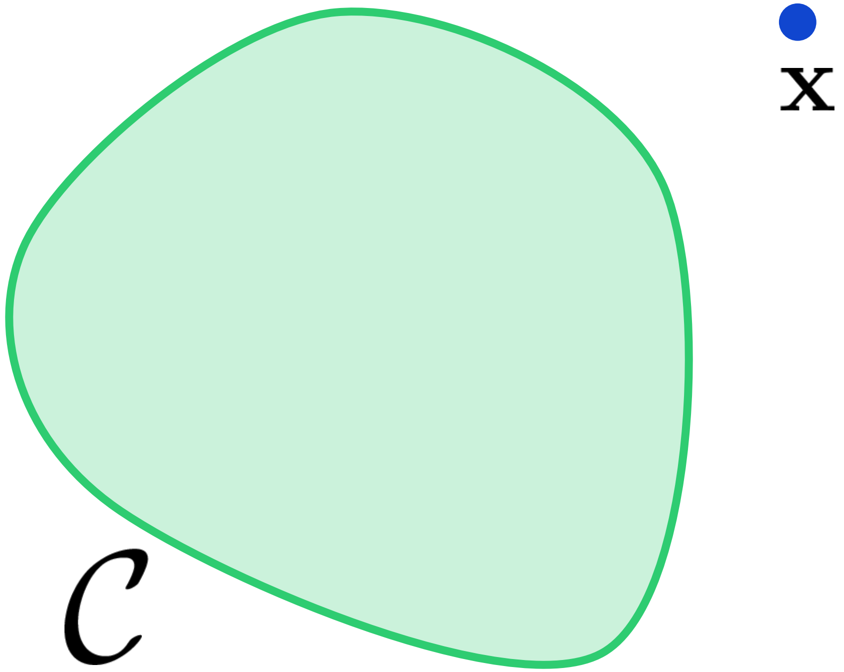
$$\arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Projected Gradient Descent



$$\arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

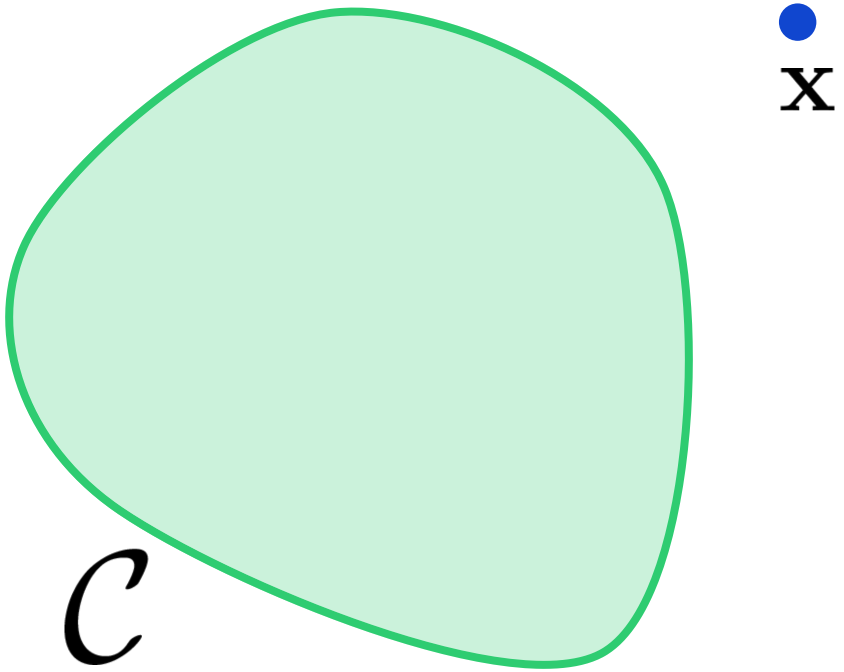
Projected Gradient Descent



$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{C}} \|\mathbf{z} - \mathbf{x}\|$$

$$\arg\min \; f(\mathbf{w}) \quad \text{s.t. } \mathbf{w} \in \mathcal{C}$$

Projected Gradient Descent

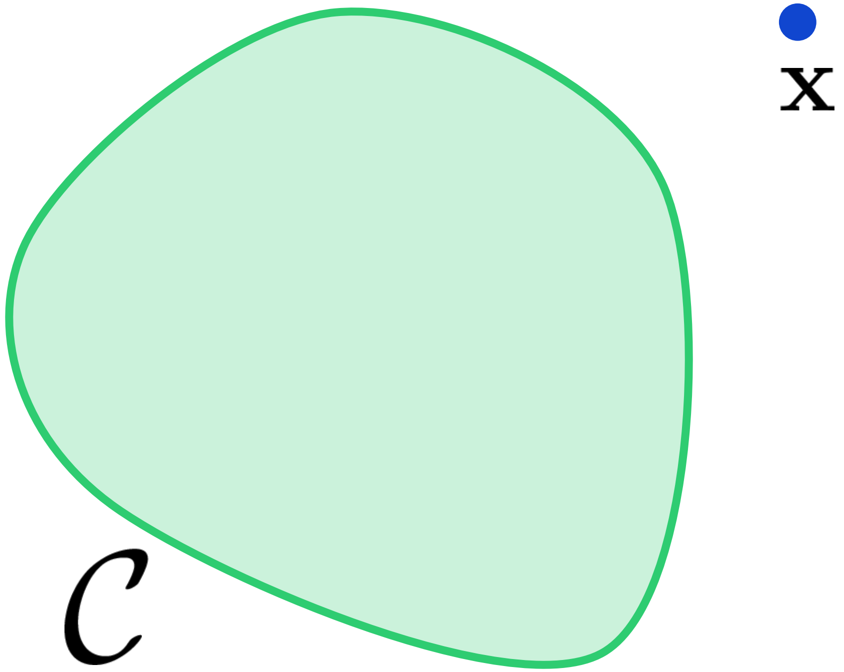


$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{C}} \|\mathbf{z}-\mathbf{x}\|_2$$

Can use other norms too!

$$\arg\min f(\mathbf{w}) \quad \text{s.t. } \mathbf{w} \in \mathcal{C}$$

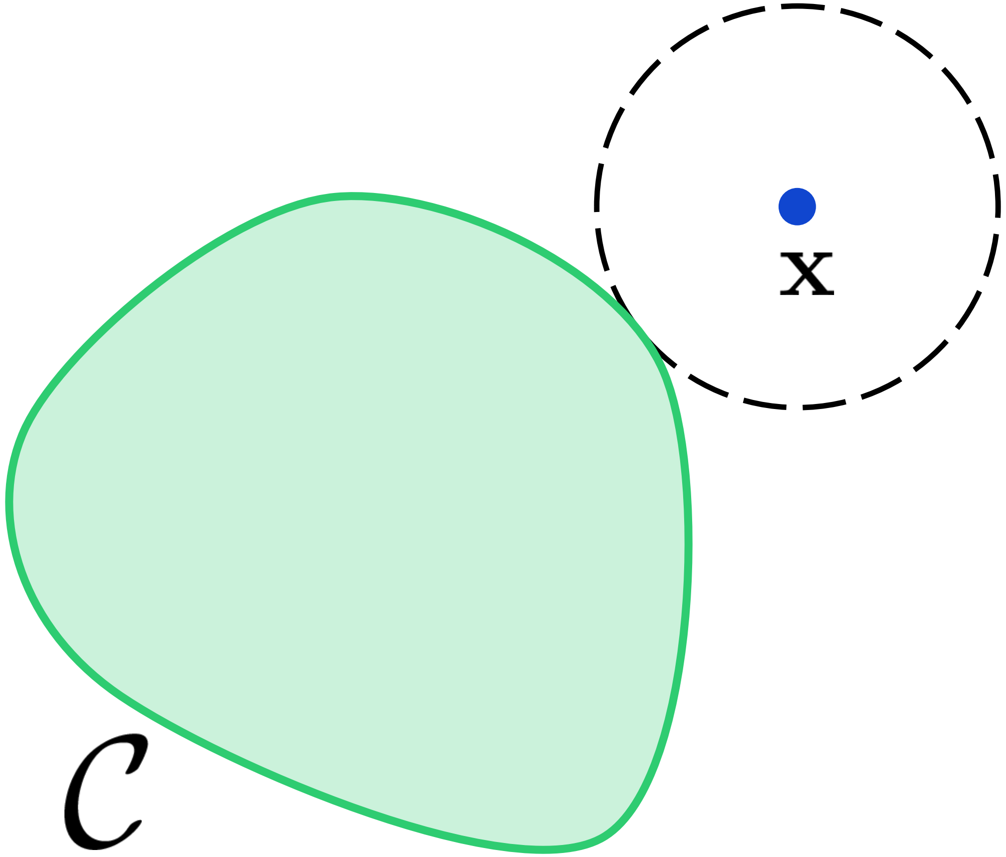
Projected Gradient Descent



$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z} \in \mathcal{C}} \|\mathbf{z} - \mathbf{x}\|_2$$

$$\arg\min f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

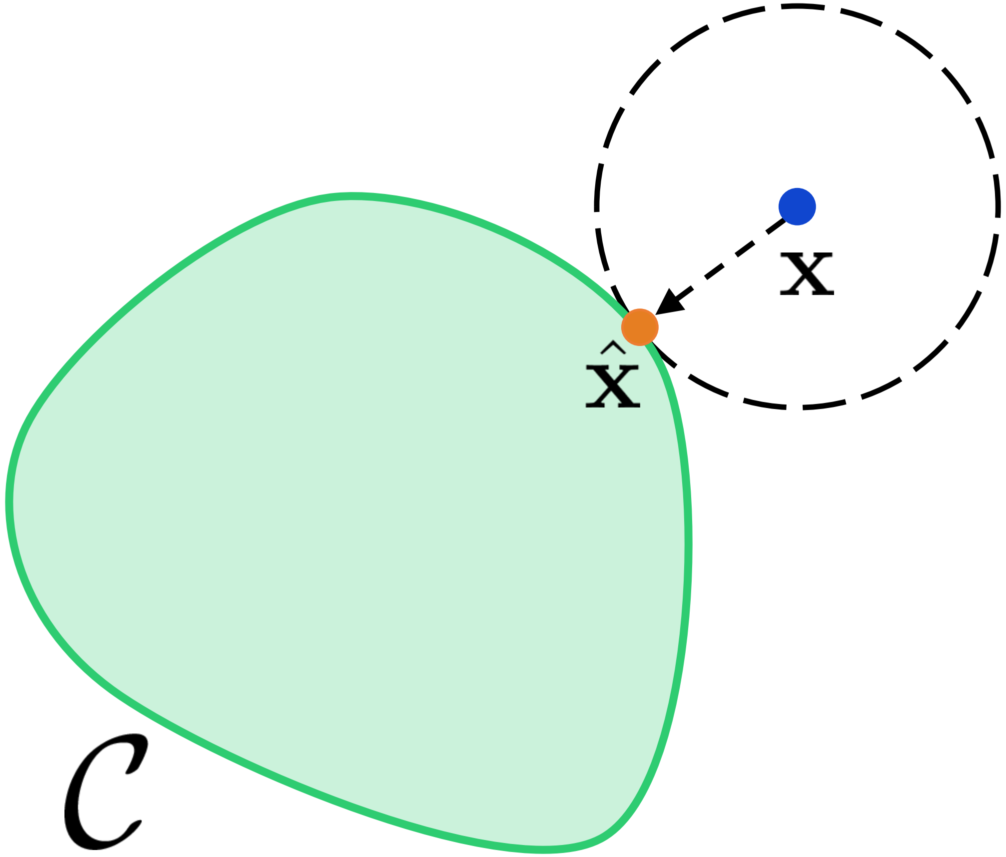
Projected Gradient Descent



$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z} \in \mathcal{C}} \ \|\mathbf{z} - \mathbf{x}\|_2$$

$$\arg\min \ f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

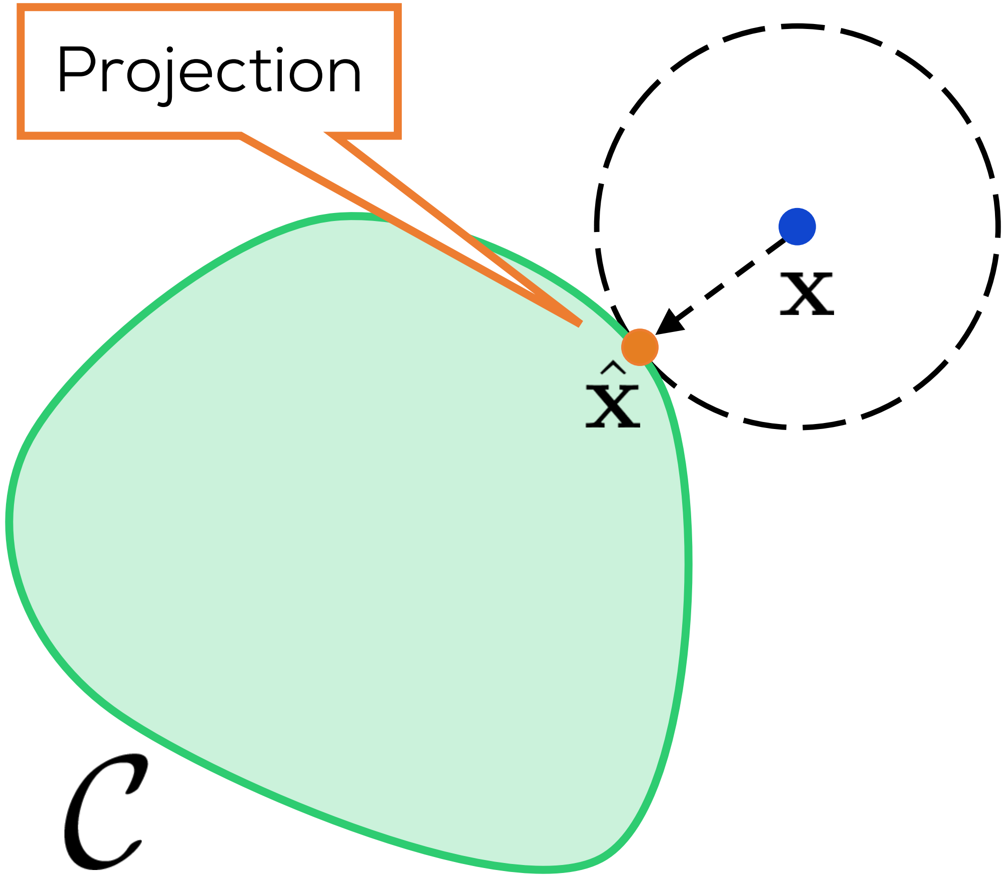
Projected Gradient Descent



$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{C}} \ \|\mathbf{z}-\mathbf{x}\|_2$$

$$\arg\min \ f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w}\in\mathcal{C}$$

Projected Gradient Descent

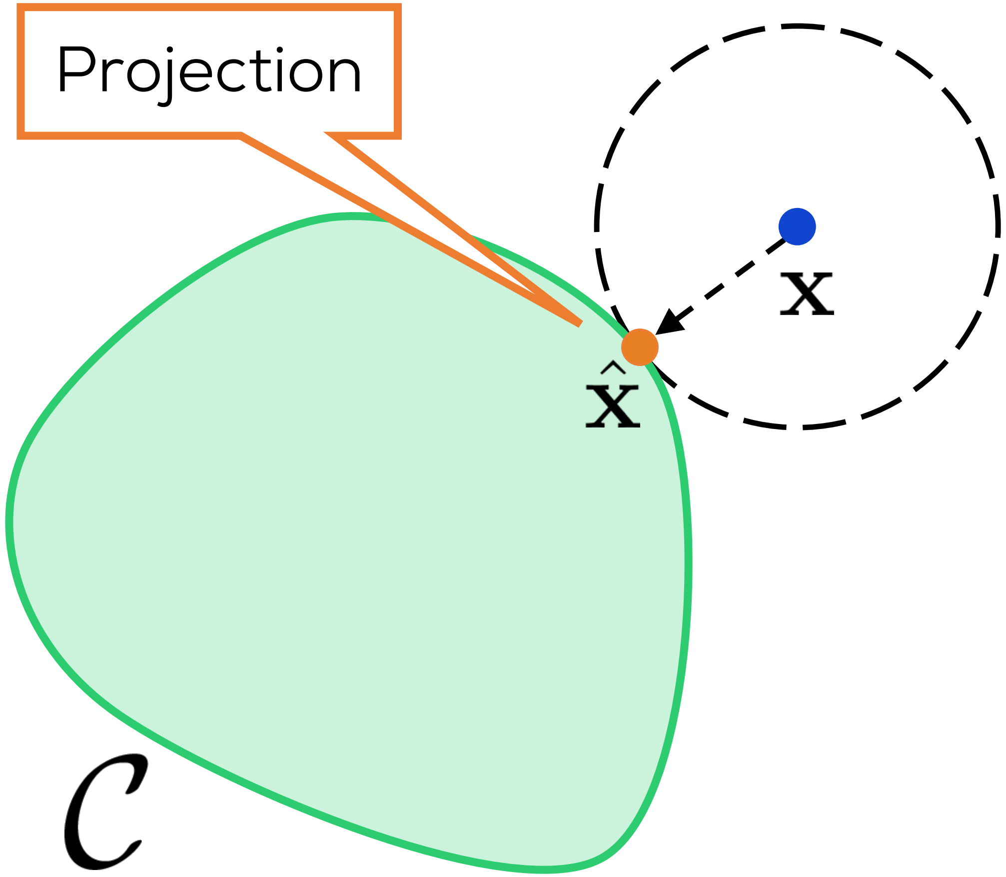


$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{C}} \|\mathbf{z} - \mathbf{x}\|_2$$

$$\arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Projected Gradient Descent

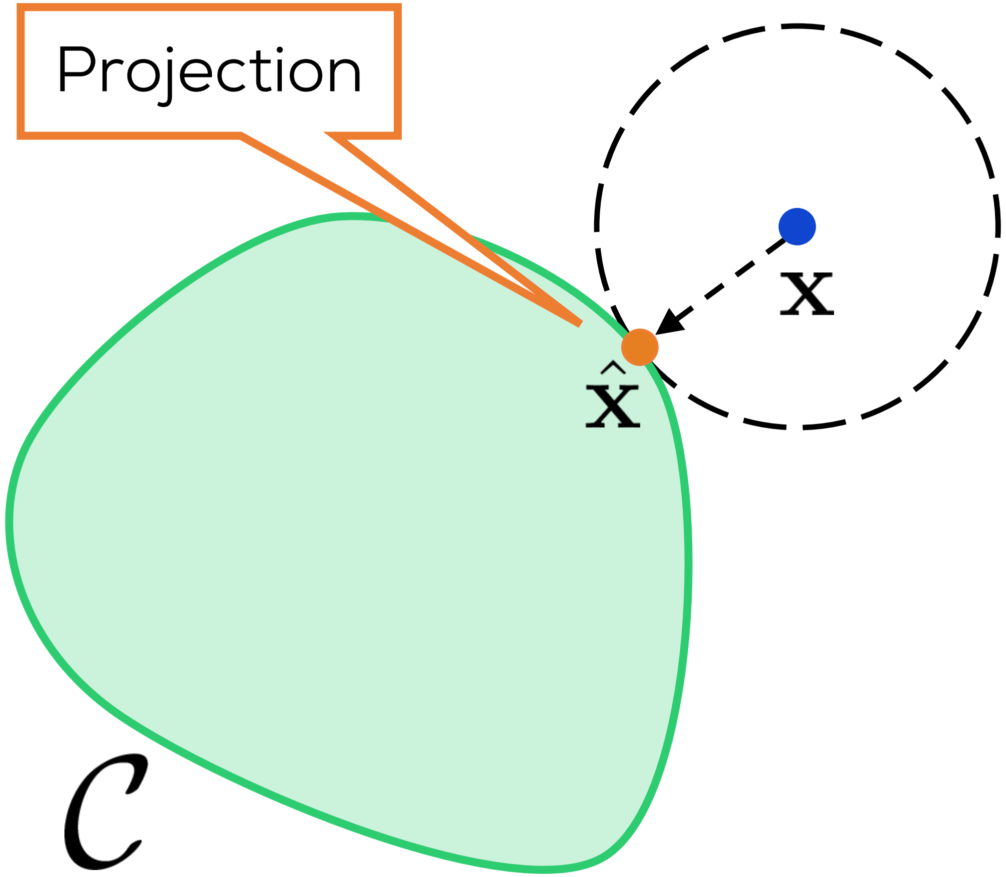
Doesn't change the problem



$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{C}} \ \|\mathbf{z} - \mathbf{x}\|_2^2$$

$$\arg\min \ f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

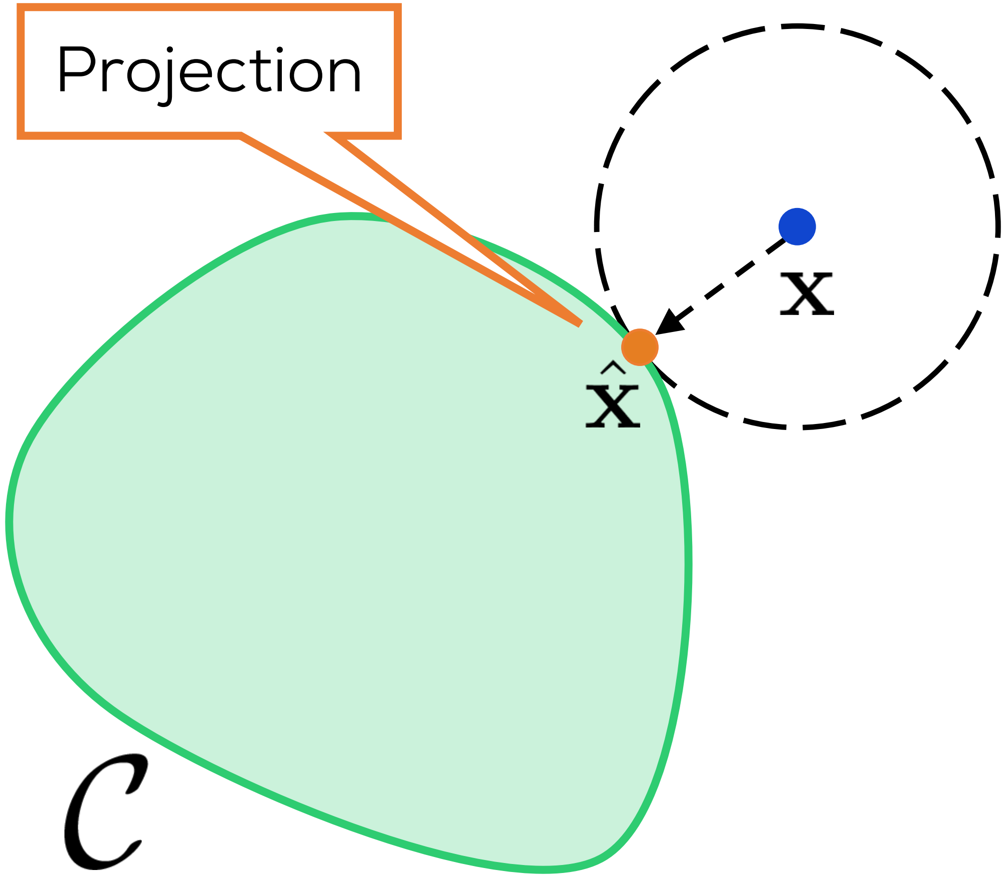
Projected Gradient Descent



$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{C}} \|\mathbf{z} - \mathbf{x}\|_2^2$$

$$\arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Projected Gradient Descent

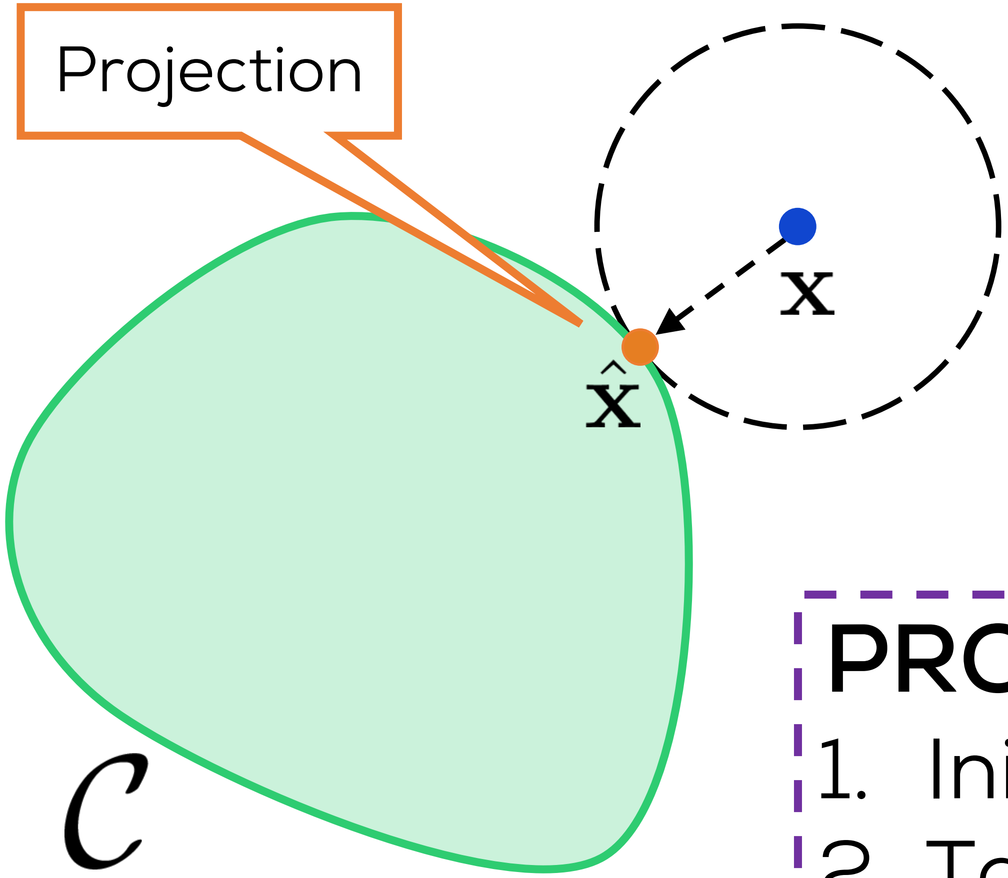


Projection is just another optimization problem!

$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{C}} \|\mathbf{z} - \mathbf{x}\|_2^2$$

$$\arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Projected Gradient Descent



Projection is just another optimization problem!

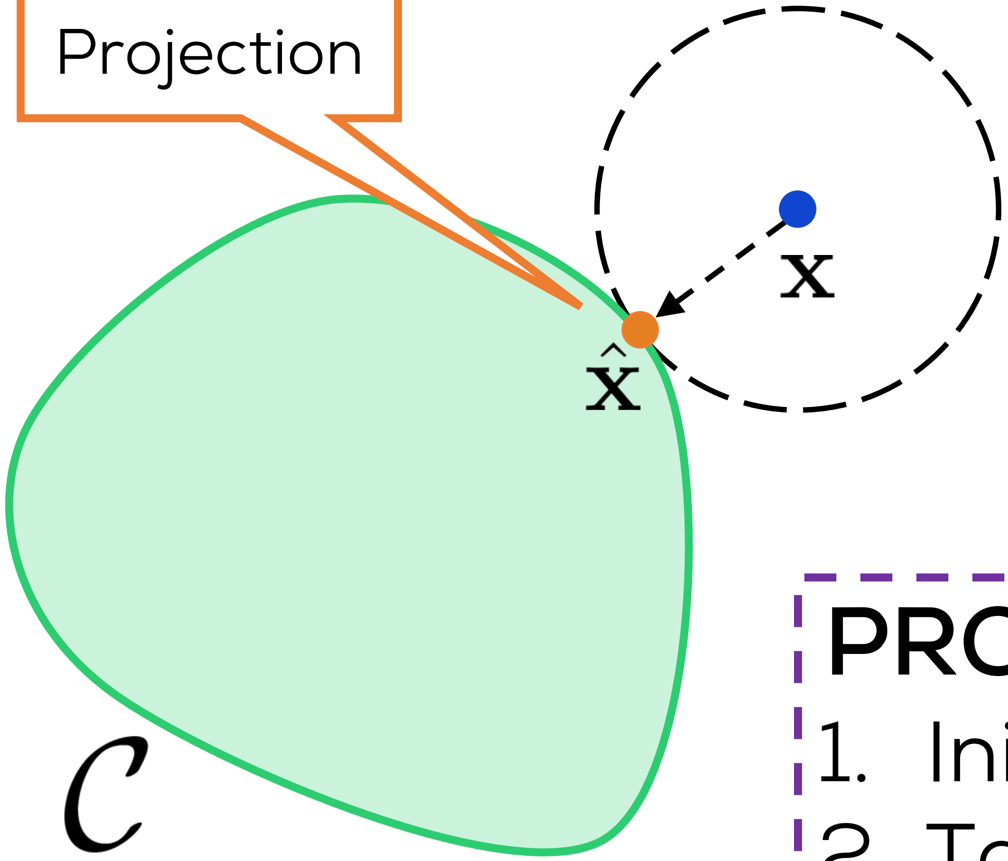
$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{C}} \|\mathbf{z} - \mathbf{x}\|_2^2$$

$$\arg\min \; f(\mathbf{w}) \quad \text{s.t. } \mathbf{w} \in \mathcal{C}$$

PROJECTED GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Take gradient step $\mathbf{u}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
3. Satisfy constraint $\mathbf{w}^{t+1} \leftarrow \Pi_{\mathcal{C}}(\mathbf{u}^{t+1})$
4. Repeat until convergence

Projected Gradient Descent



$$\arg\min \; f(\mathbf{w})$$
$$\text{s.t. } \mathbf{w} \in \mathcal{C}$$

Projection is just another optimization problem!

$$\Pi_{\mathcal{C}}(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{C}} \; \|\mathbf{z} - \mathbf{x}\|_2^2$$

Projected Stochastic Gradient Descent??

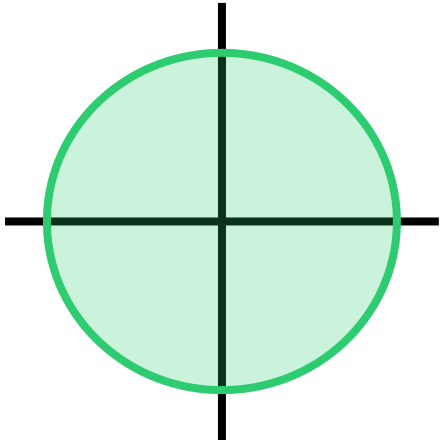
PROJECTED GRADIENT DESCENT

1. Initialize $\mathbf{w}^0$
2. Take gradient step $\mathbf{u}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
3. Satisfy constraint $\mathbf{w}^{t+1} \leftarrow \Pi_{\mathcal{C}}(\mathbf{u}^{t+1})$
4. Repeat until convergence

A few useful projections

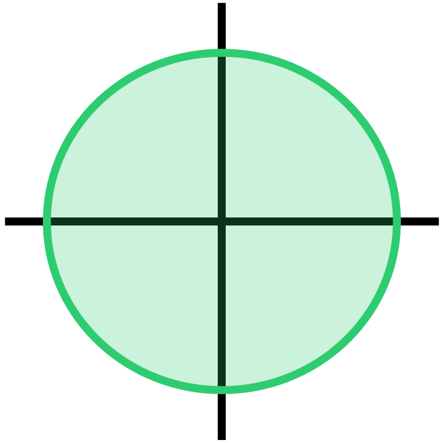
A few useful projections

$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$$

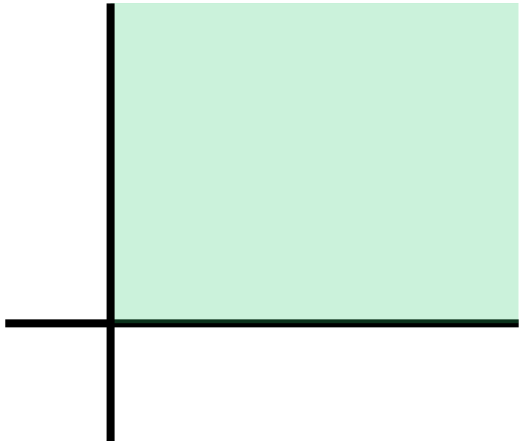


A few useful projections

$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$$

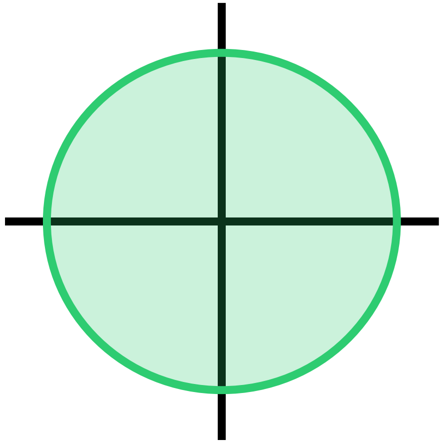


$$\mathcal{C} = \{\mathbf{x} : \mathbf{x}_i \geq 0\}$$

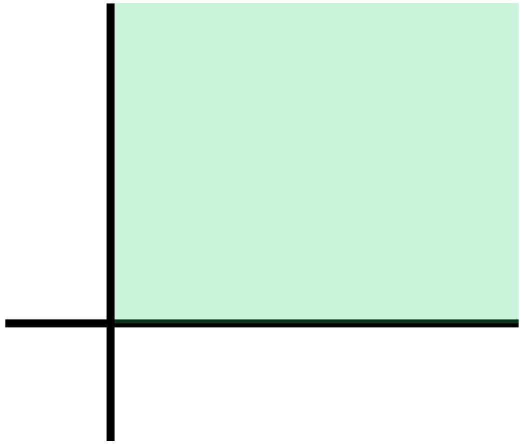


A few useful projections

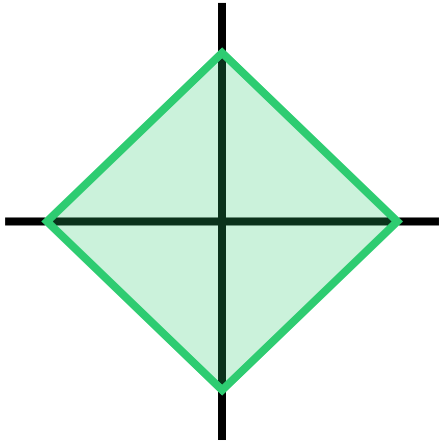
$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$$



$$\mathcal{C} = \{\mathbf{x} : \mathbf{x}_i \geq 0\}$$

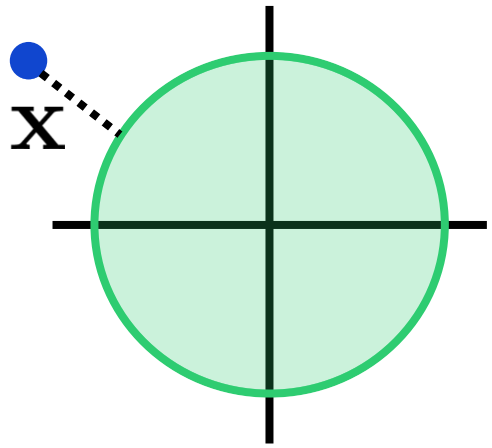


$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq 1\}$$

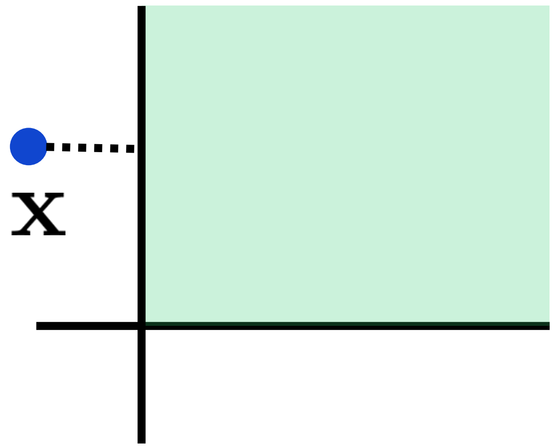


A few useful projections

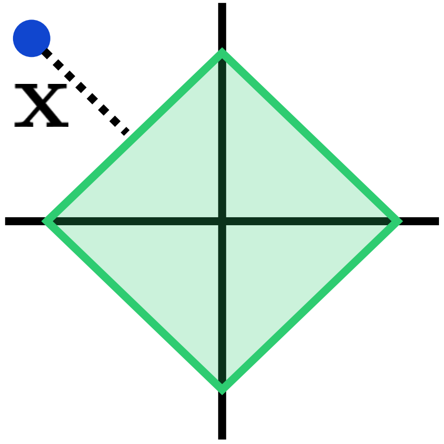
$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$$



$$\mathcal{C} = \{\mathbf{x} : \mathbf{x}_i \geq 0\}$$



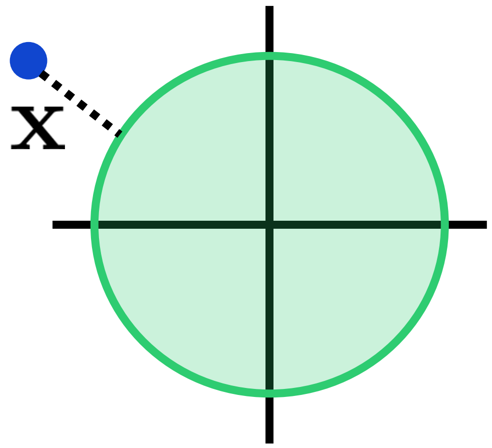
$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq 1\}$$



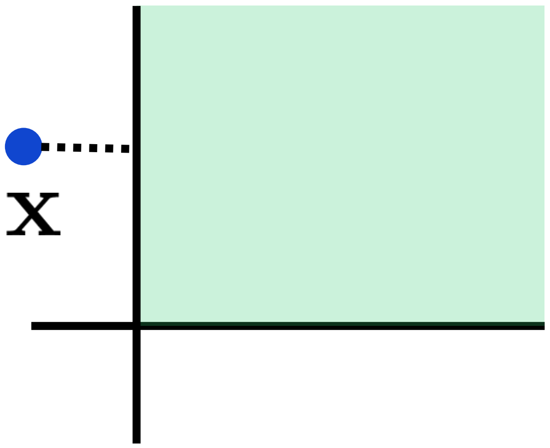
A few useful projections

$$\hat{\mathbf{x}} = \Pi_{\mathcal{C}}(\mathbf{x})$$

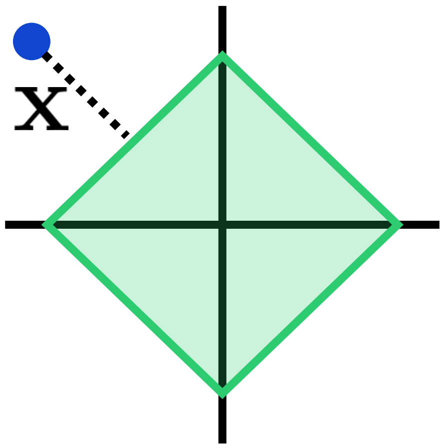
$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$$



$$\mathcal{C} = \{\mathbf{x} : \mathbf{x}_i \geq 0\}$$



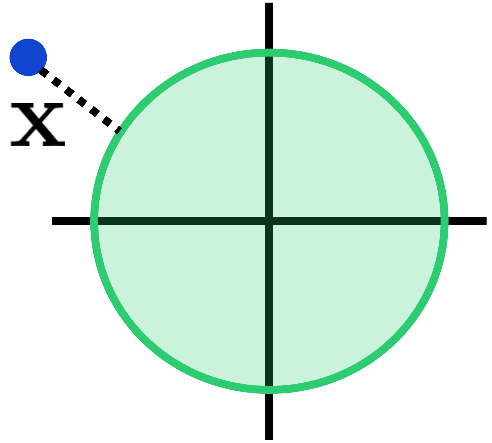
$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq 1\}$$



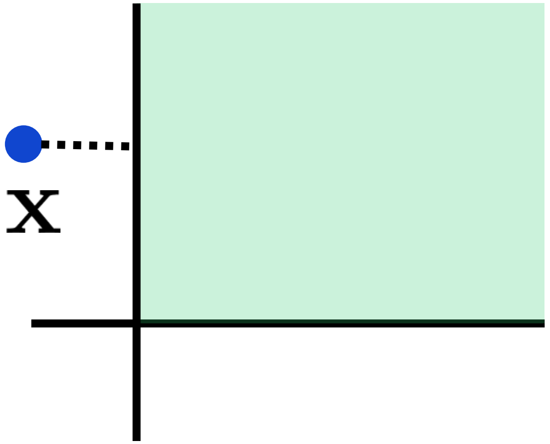
A few useful projections

$$\hat{\mathbf{x}} = \Pi_{\mathcal{C}}(\mathbf{x})$$

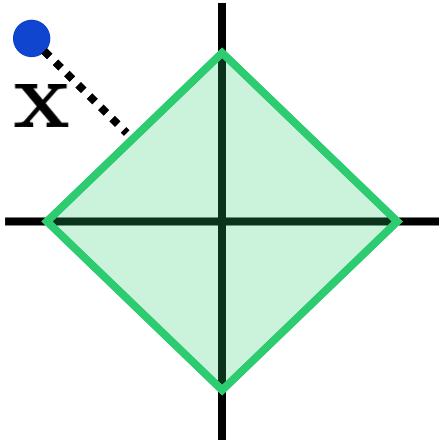
$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$$



$$\mathcal{C} = \{\mathbf{x} : \mathbf{x}_i \geq 0\}$$



$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq 1\}$$

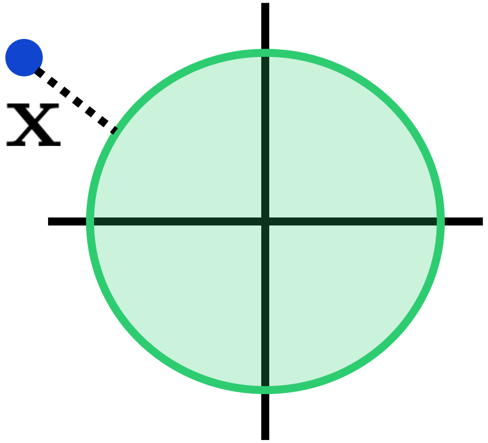


$$\hat{\mathbf{x}} = \begin{cases} \mathbf{x} & \text{if } \|\mathbf{x}\|_2 \leq 1 \\ \frac{\mathbf{x}}{\|\mathbf{x}\|_2} & \text{if } \|\mathbf{x}\|_2 > 1 \end{cases}$$

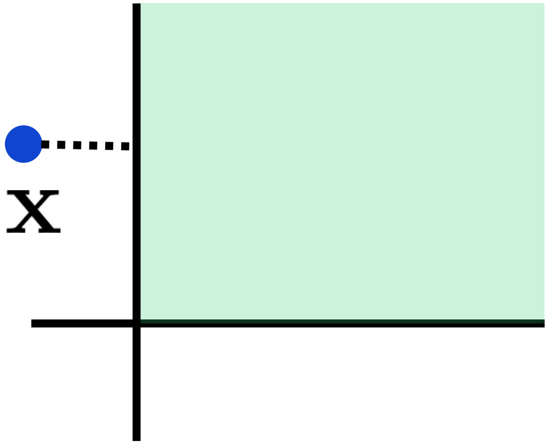
A few useful projections

$$\hat{\mathbf{x}} = \Pi_{\mathcal{C}}(\mathbf{x})$$

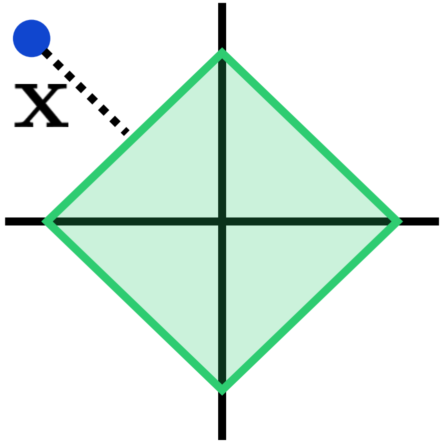
$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$$



$$\mathcal{C} = \{\mathbf{x} : \mathbf{x}_i \geq 0\}$$



$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq 1\}$$



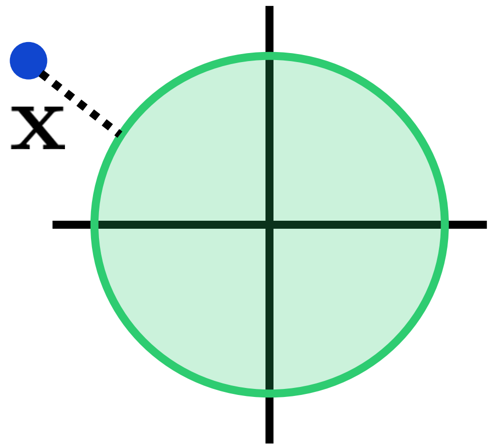
$$\hat{\mathbf{x}} = \begin{cases} \mathbf{x} & \text{if } \|\mathbf{x}\|_2 \leq 1 \\ \frac{\mathbf{x}}{\|\mathbf{x}\|_2} & \text{if } \|\mathbf{x}\|_2 > 1 \end{cases}$$

What if $\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq r\}$?

A few useful projections

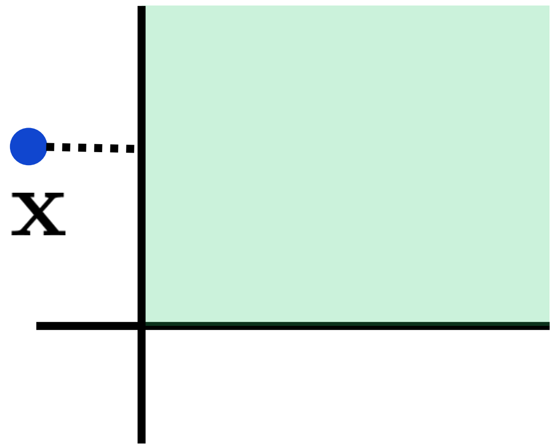
$$\hat{\mathbf{x}} = \Pi_{\mathcal{C}}(\mathbf{x})$$

$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$$



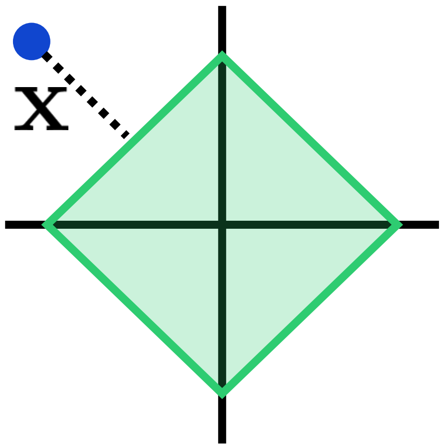
$$\hat{\mathbf{x}} = \begin{cases} \mathbf{x} & \text{if } \|\mathbf{x}\|_2 \leq 1 \\ \frac{\mathbf{x}}{\|\mathbf{x}\|_2} & \text{if } \|\mathbf{x}\|_2 > 1 \end{cases}$$

$$\mathcal{C} = \{\mathbf{x} : \mathbf{x}_i \geq 0\}$$



$$\hat{\mathbf{x}}_i = \begin{cases} \mathbf{x}_i & \text{if } \mathbf{x}_i \geq 0 \\ 0 & \text{if } \mathbf{x}_i < 0 \end{cases}$$

$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq 1\}$$

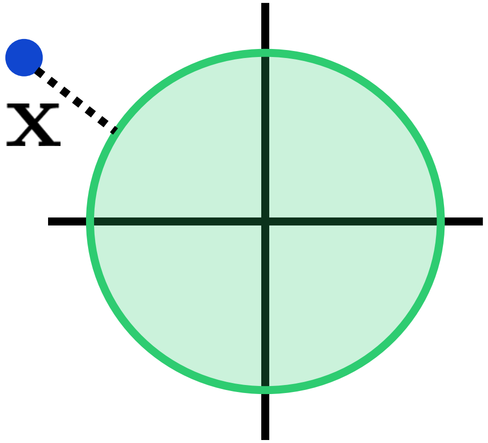


What if $\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq r\}$?

A few useful projections

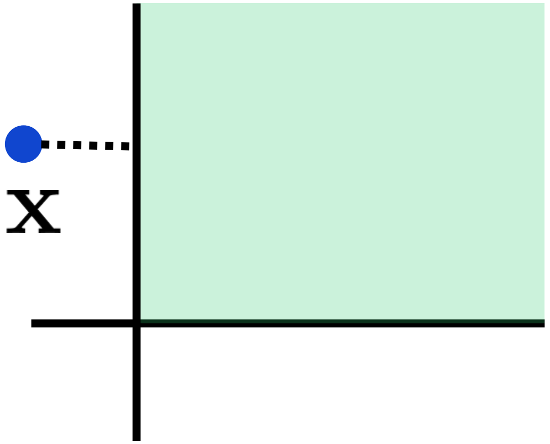
$$\hat{\mathbf{x}} = \Pi_{\mathcal{C}}(\mathbf{x})$$

$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$$



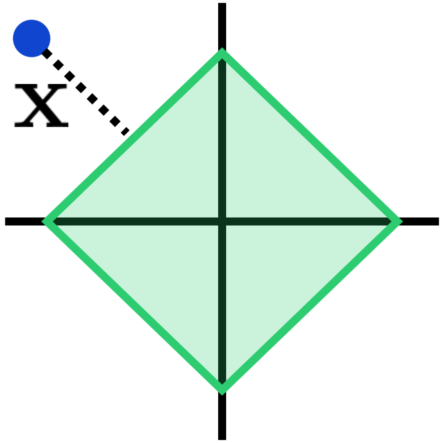
$$\hat{\mathbf{x}} = \begin{cases} \mathbf{x} & \text{if } \|\mathbf{x}\|_2 \leq 1 \\ \frac{\mathbf{x}}{\|\mathbf{x}\|_2} & \text{if } \|\mathbf{x}\|_2 > 1 \end{cases}$$

$$\mathcal{C} = \{\mathbf{x} : \mathbf{x}_i \geq 0\}$$



$$\hat{\mathbf{x}}_i = \begin{cases} \mathbf{x}_i & \text{if } \mathbf{x}_i \geq 0 \\ 0 & \text{if } \mathbf{x}_i < 0 \end{cases}$$

$$\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq 1\}$$



Find out!

What if $\mathcal{C} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq r\}$?

Some disclaimers

- GD can be slow in convergence
  - Step size tuning can affect performance quite a bit – Armijo, Adagrad
  - Momentum-based methods $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t - \gamma \cdot (\mathbf{w}^t - \mathbf{w}^{t-1})$
  - Second order methods – expensive but super fast!

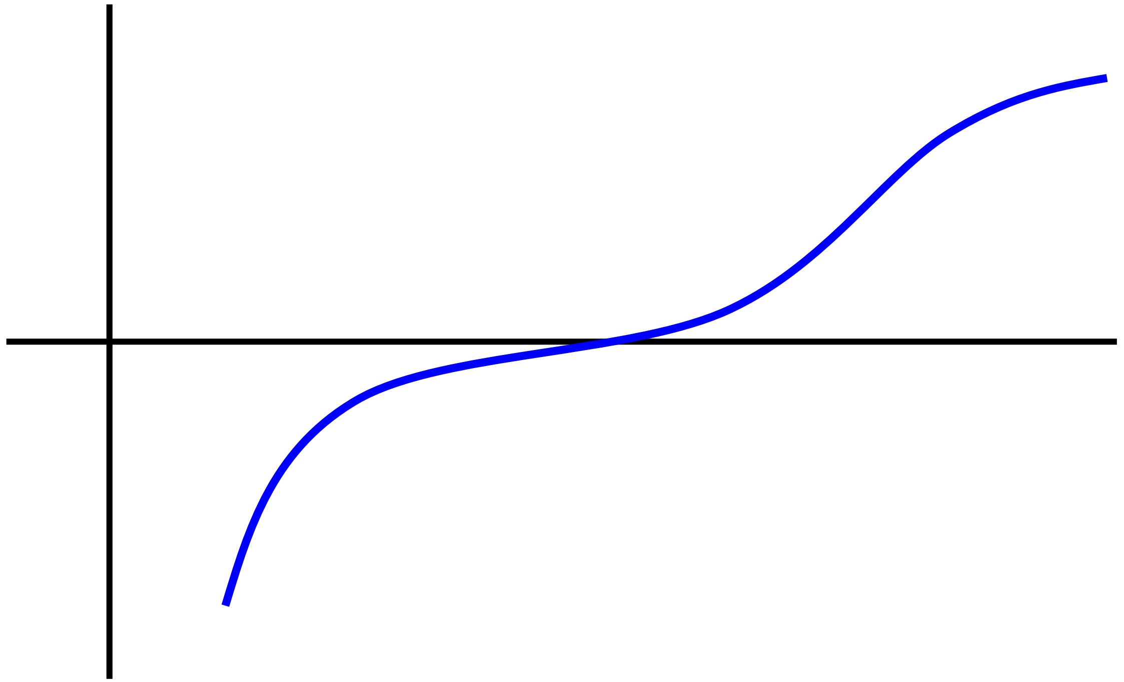
Momentum weight

- Exist superior methods for specific problems
  - RR (L_2 regularized ls regression) – conjugate gradient method
  - Lasso (L_1 regularized ls regression) – proximal gradient method
  - Logistic regression – trust region Newton method
- Details, convergence analysis beyond scope of this course
- CS774: Optimization Techniques
- Next: Newton method, duality, coordinate descent

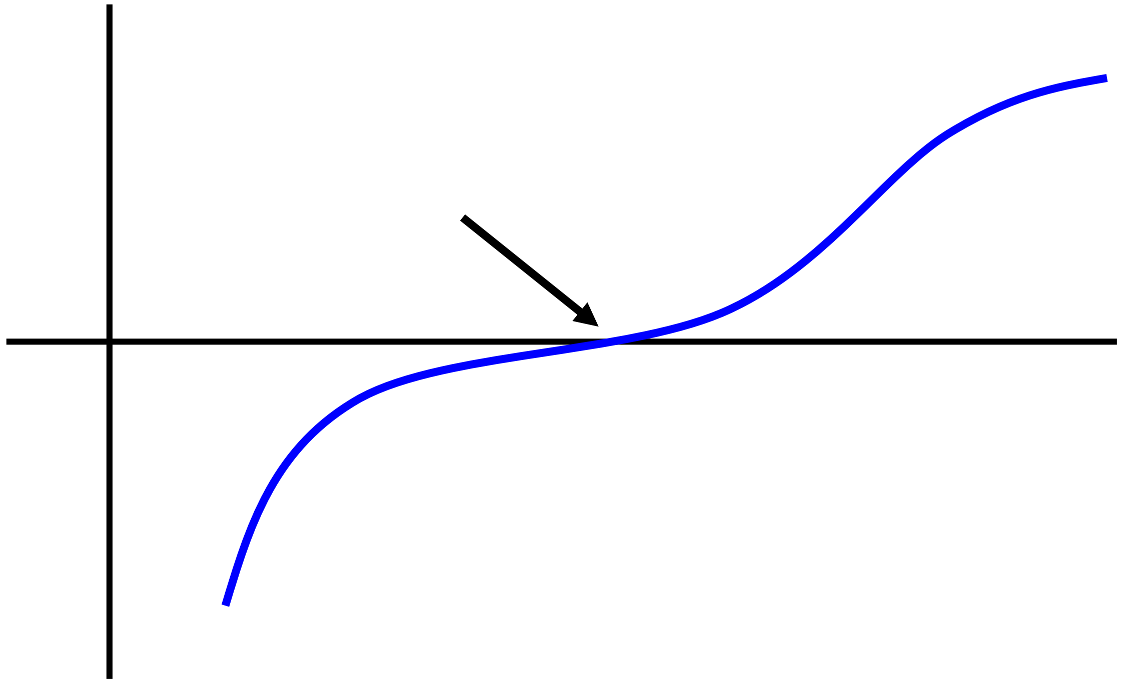
Newton Method

Descent with second order derivatives

Newton-Raphson Method for Root Finding

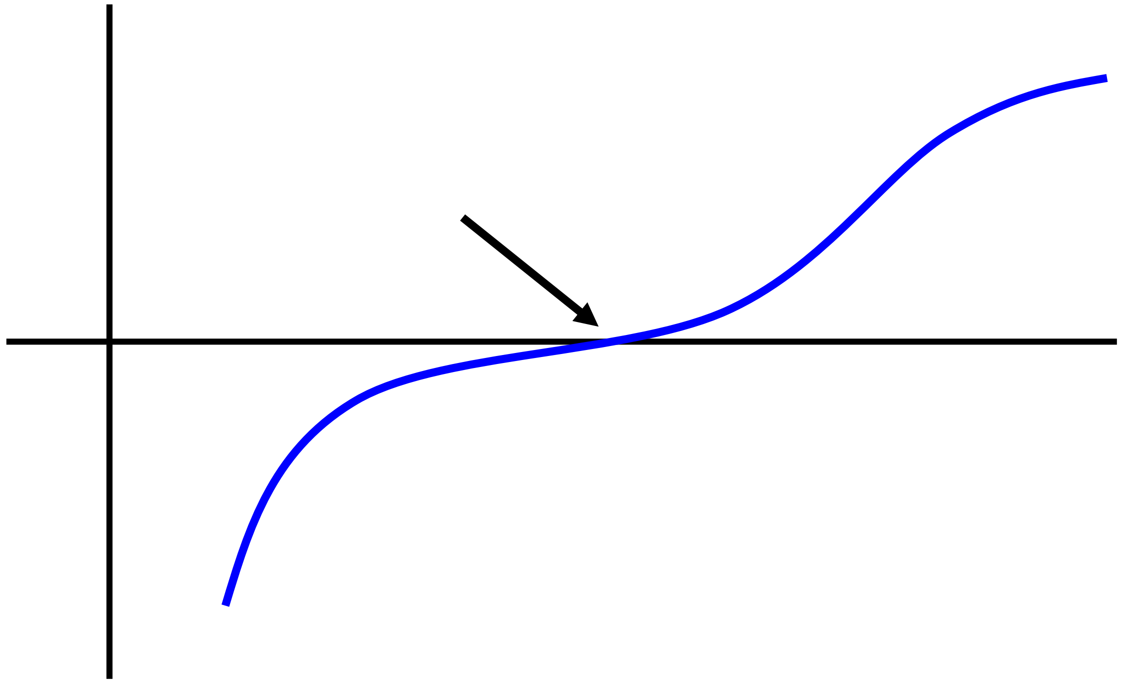


Newton-Raphson Method for Root Finding



$$x : f(x) = 0$$

Newton-Raphson Method for Root Finding

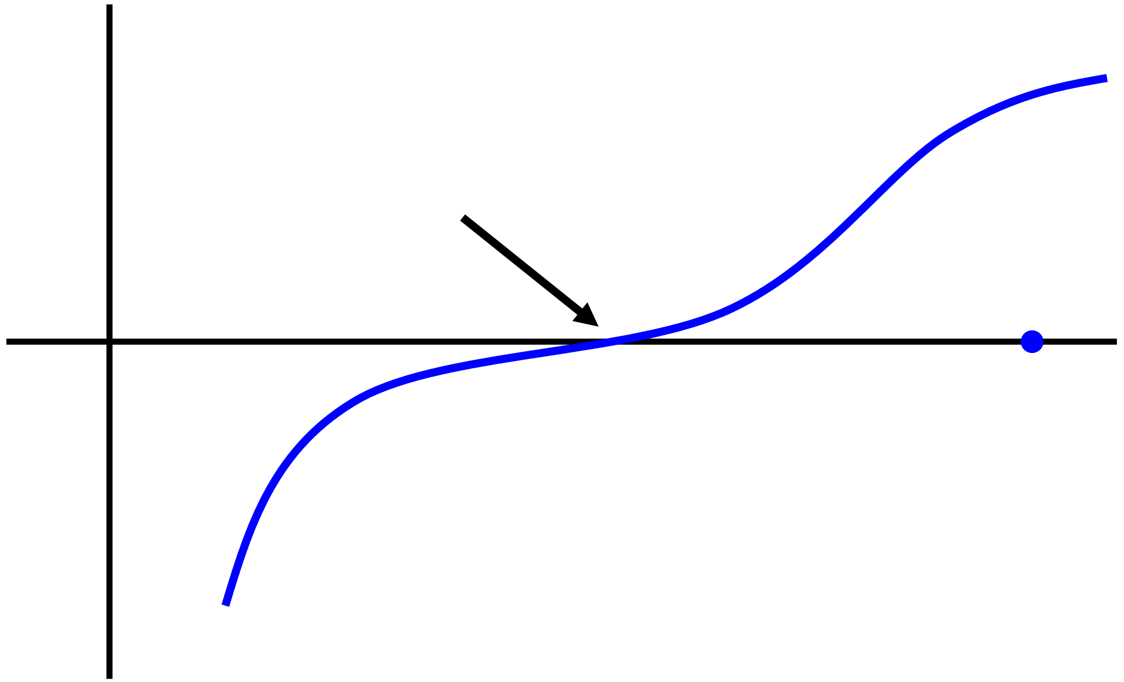


$$x : f(x) = 0$$

Finding roots of linear functions is easy

$f(x) = ax + b$, $x_0 = \frac{-b}{a}$

Newton-Raphson Method for Root Finding

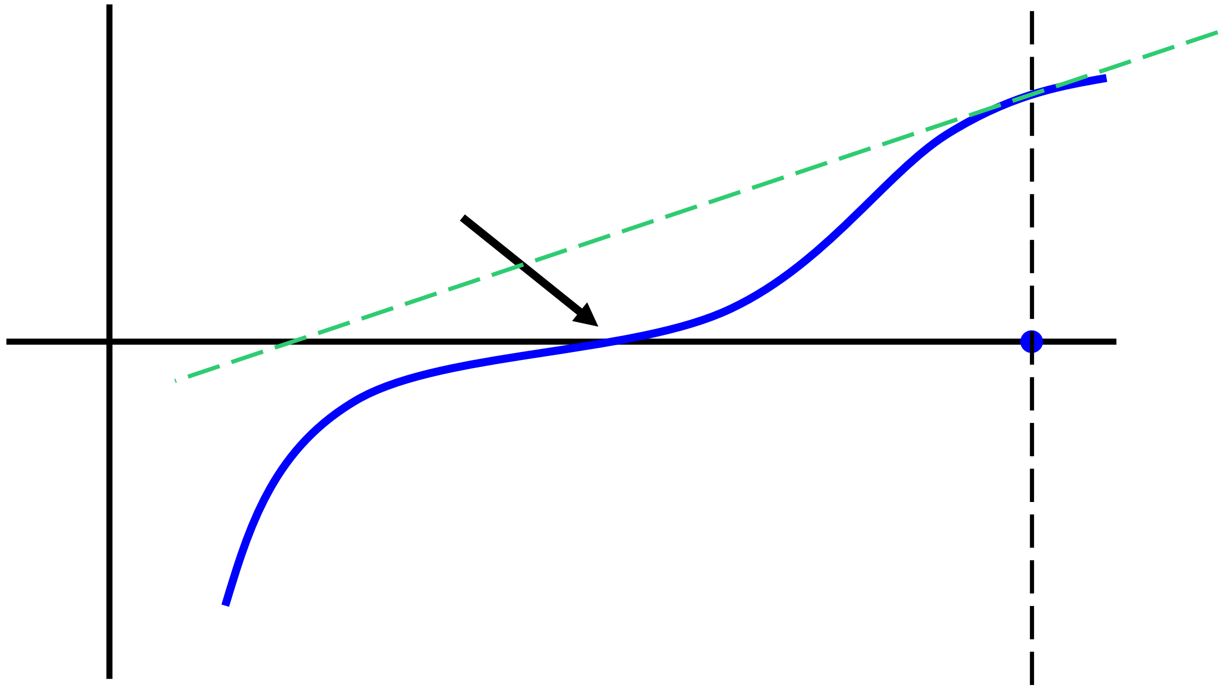


$$x : f(x) = 0$$

Finding roots of linear functions is easy

$f(x) = ax + b$, $x_0 = \frac{-b}{a}$

Newton-Raphson Method for Root Finding

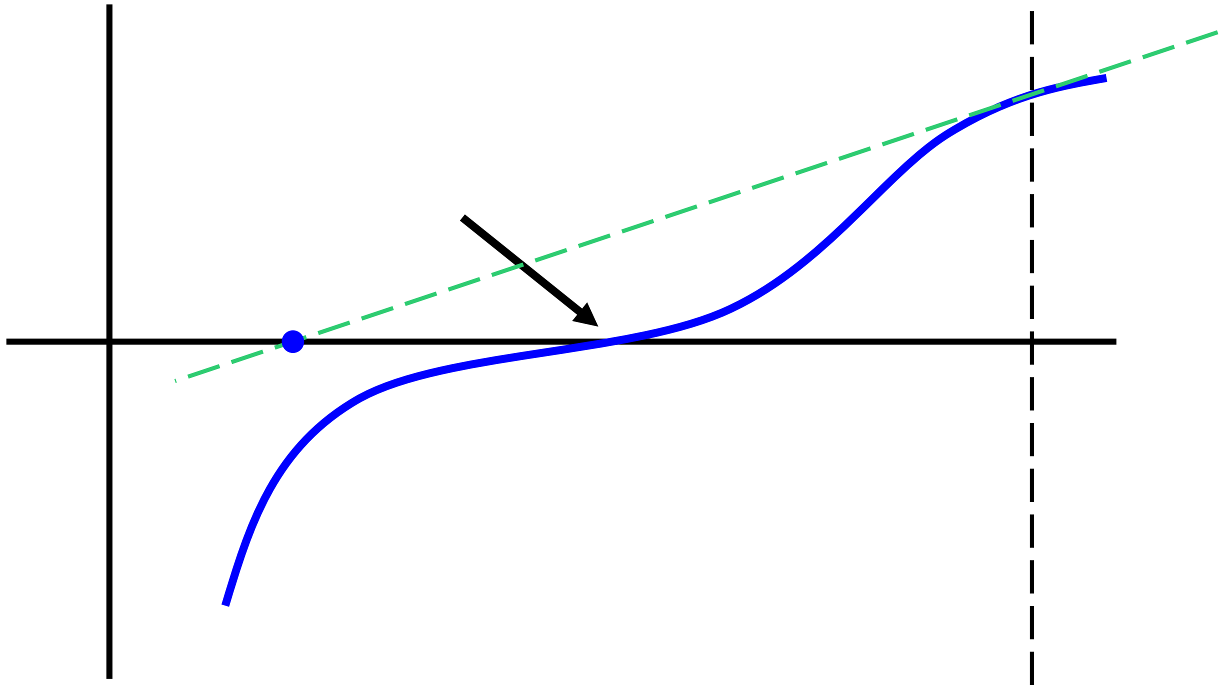


$$x : f(x) = 0$$

Finding roots of linear functions is easy

$f(x) = ax + b$, $x_0 = \frac{-b}{a}$

Newton-Raphson Method for Root Finding

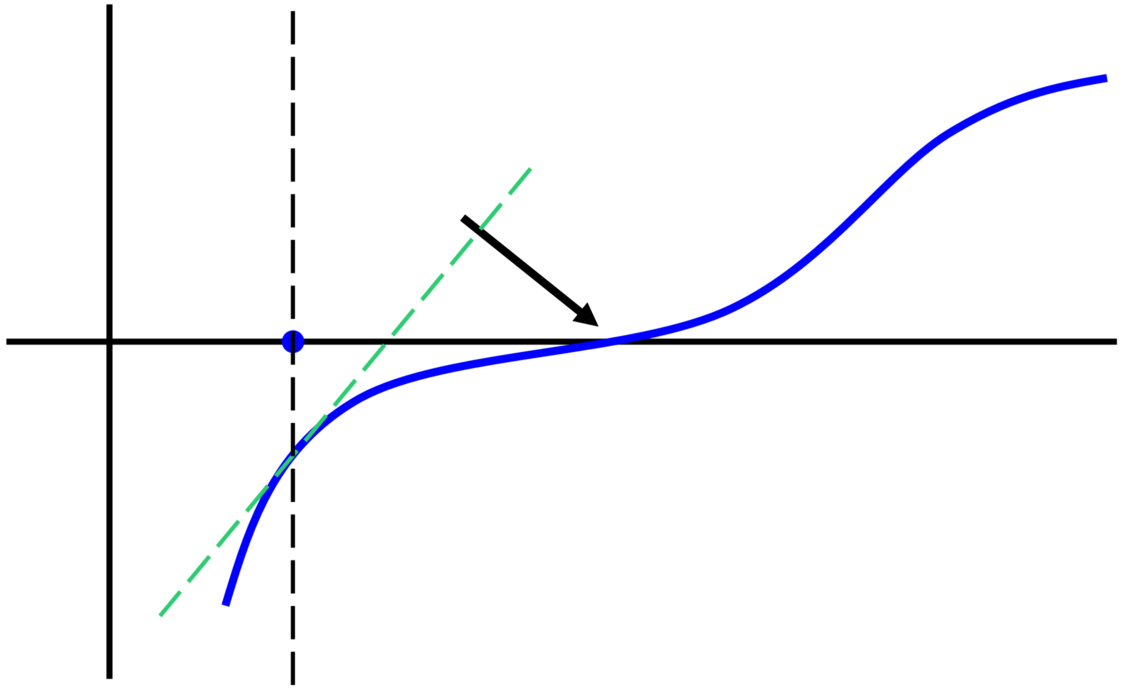


$$x : f(x) = 0$$

Finding roots of linear functions is easy

$f(x) = ax + b$, $x_0 = \frac{-b}{a}$

Newton-Raphson Method for Root Finding

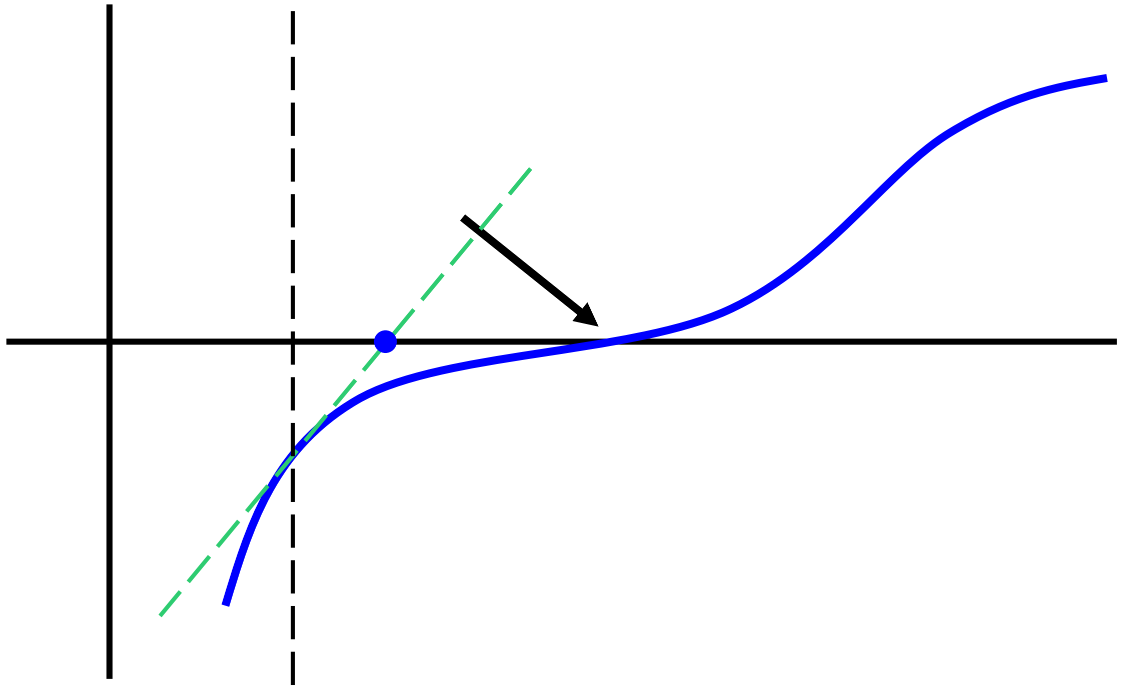


$$x : f(x) = 0$$

Finding roots of linear functions is easy

$f(x) = ax + b,\ x_0 = \frac{-b}{a}$

Newton-Raphson Method for Root Finding

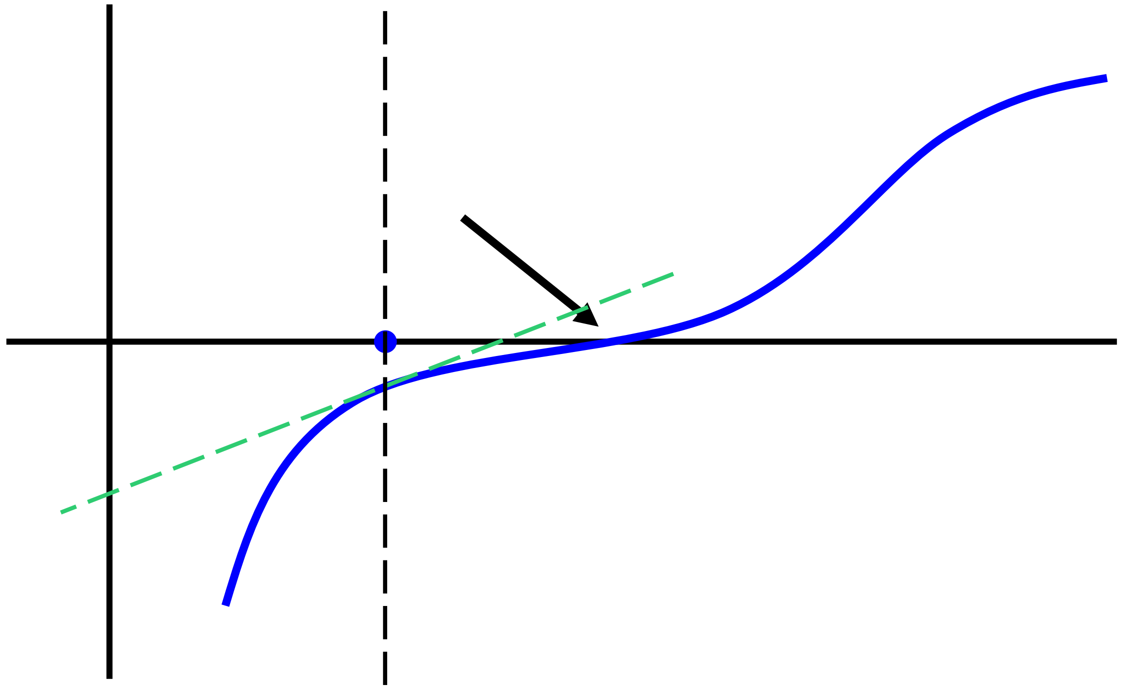


$$x : f(x) = 0$$

Finding roots of linear functions is easy

$f(x) = ax + b$, $x_0 = \frac{-b}{a}$

Newton-Raphson Method for Root Finding

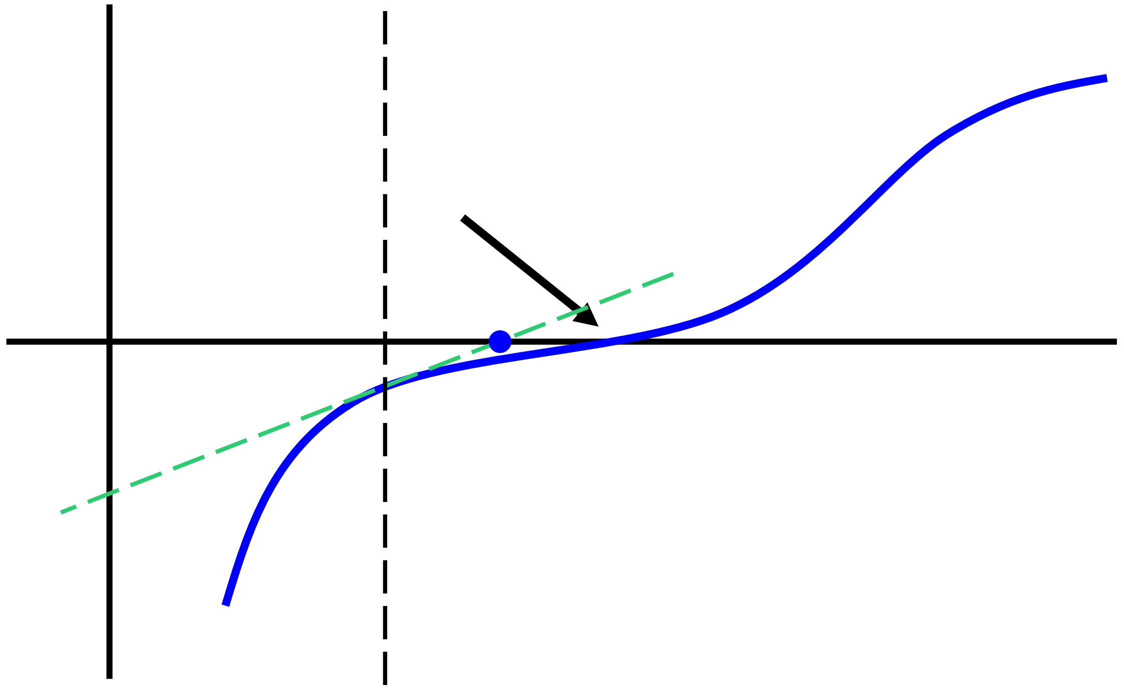


$$x : f(x) = 0$$

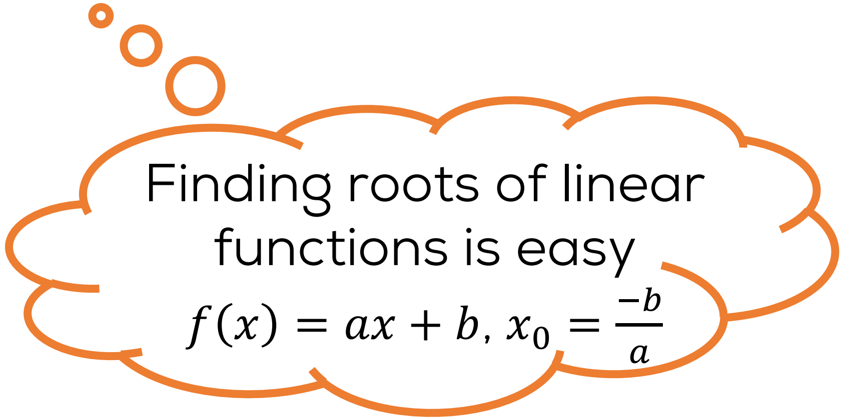
Finding roots of linear functions is easy

$f(x) = ax + b,\ x_0 = \frac{-b}{a}$

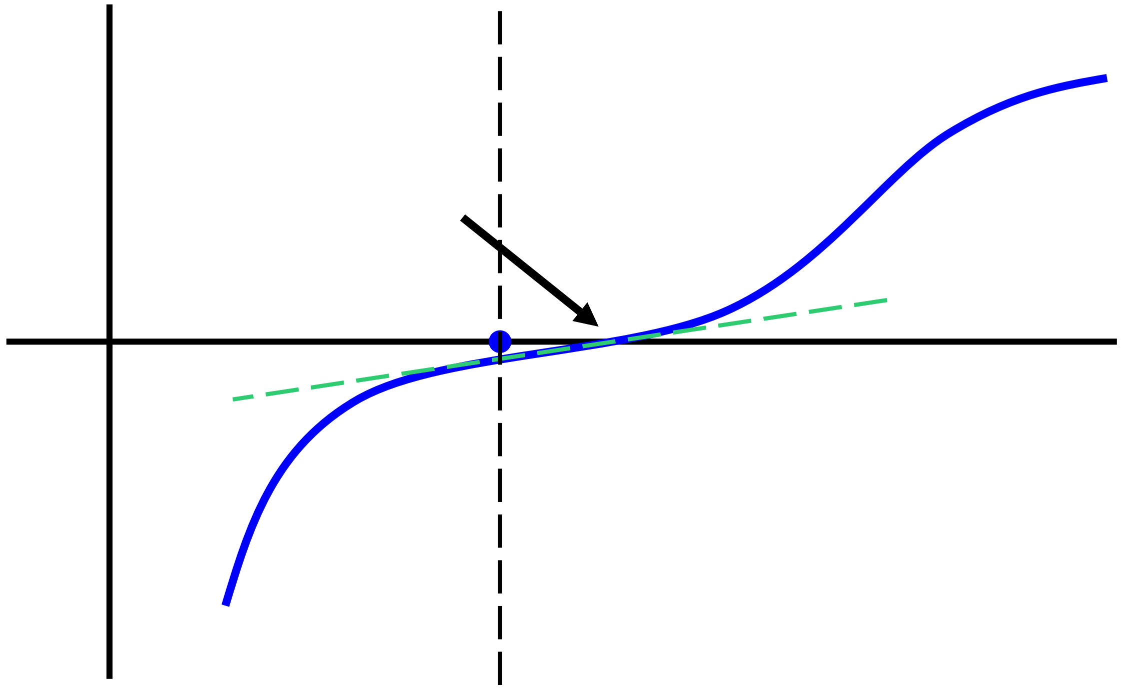
Newton-Raphson Method for Root Finding



$$x : f(x) = 0$$



Newton-Raphson Method for Root Finding

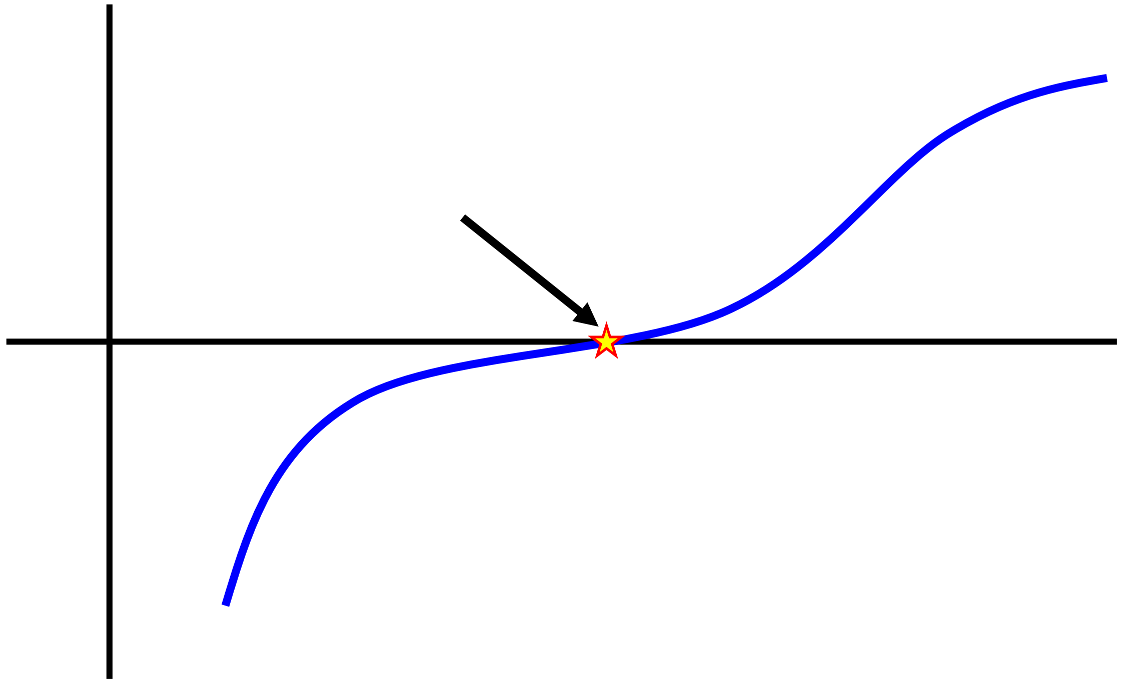


$$x : f(x) = 0$$

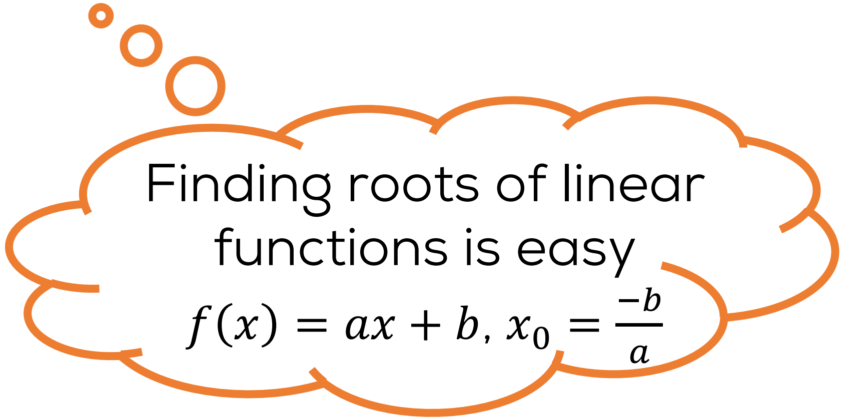
Finding roots of linear functions is easy

$f(x) = ax + b$, $x_0 = \frac{-b}{a}$

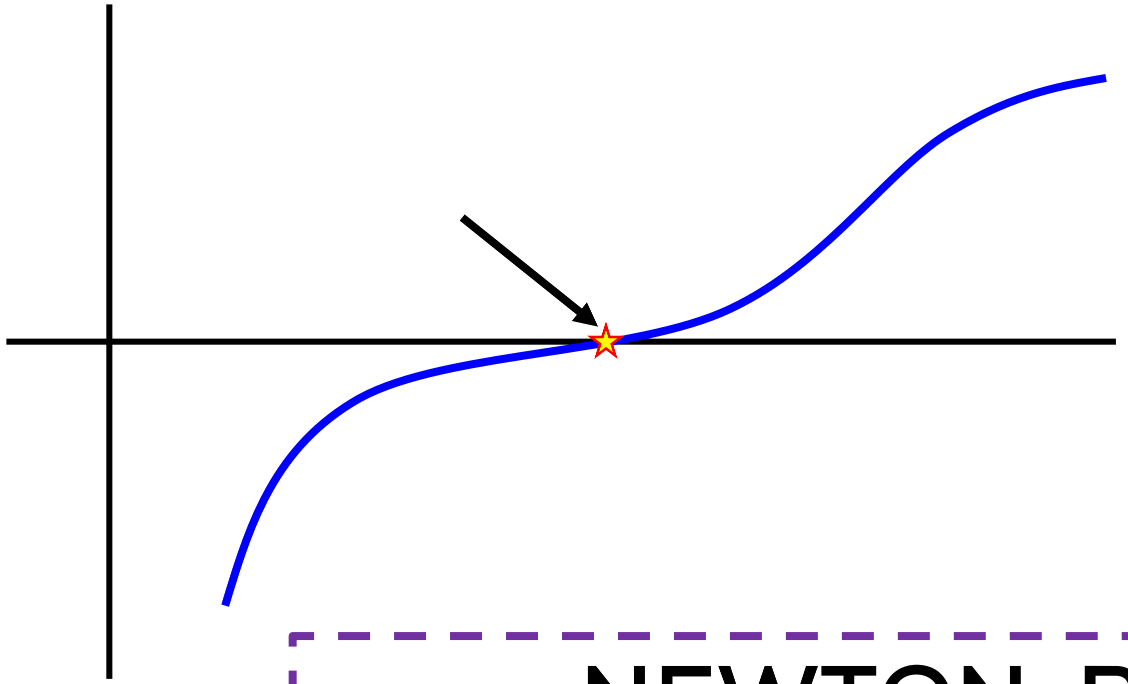
Newton-Raphson Method for Root Finding



$$x : f(x) = 0$$



Newton-Raphson Method for Root Finding



$$x : f(x) = 0$$

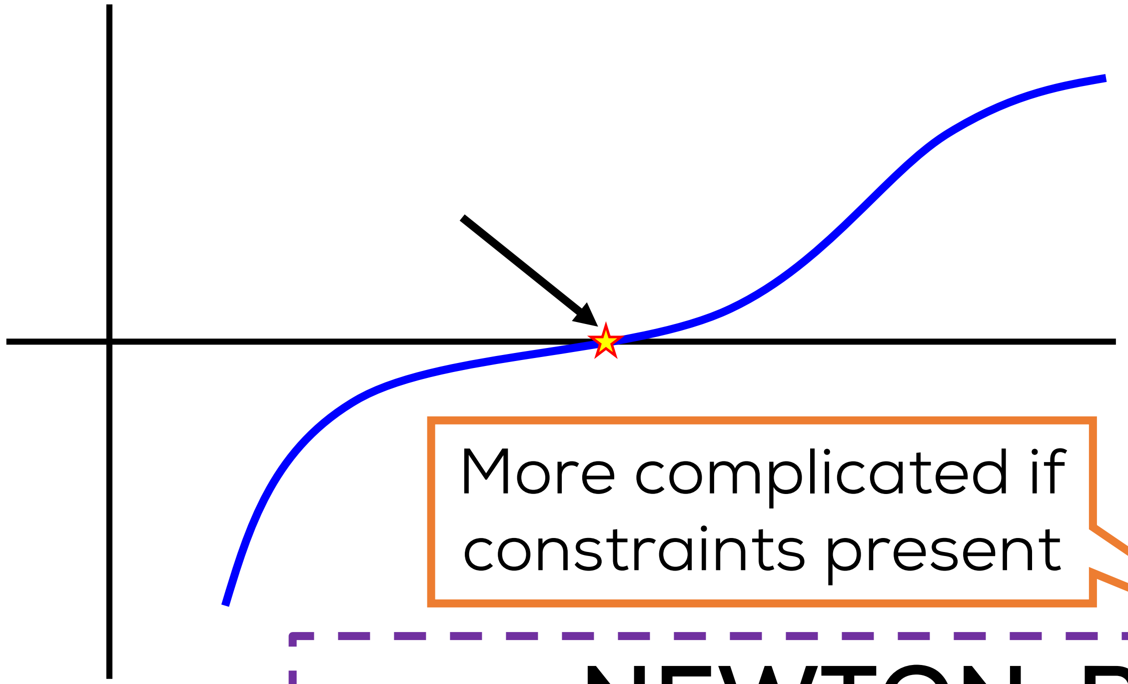
Finding roots of linear functions is easy

$f(x) = ax + b,\ x_0 = \frac{-b}{a}$

NEWTON-RAPHSON METHOD

1. Initialize x^0
2. Approximate $f(\cdot)$ by $\tilde{f}_t(x) = f(x^t) + \langle f'(x^t), x - x^t \rangle$
3. Update $x^{t+1} \leftarrow \mathrm{ROOT}(\tilde{f}_t) = x^t - \frac{f(x^t)}{f'(x^t)}$
4. Repeat until convergence

Newton-Raphson Method for Root Finding



More complicated if constraints present

$$x : f(x) = 0$$

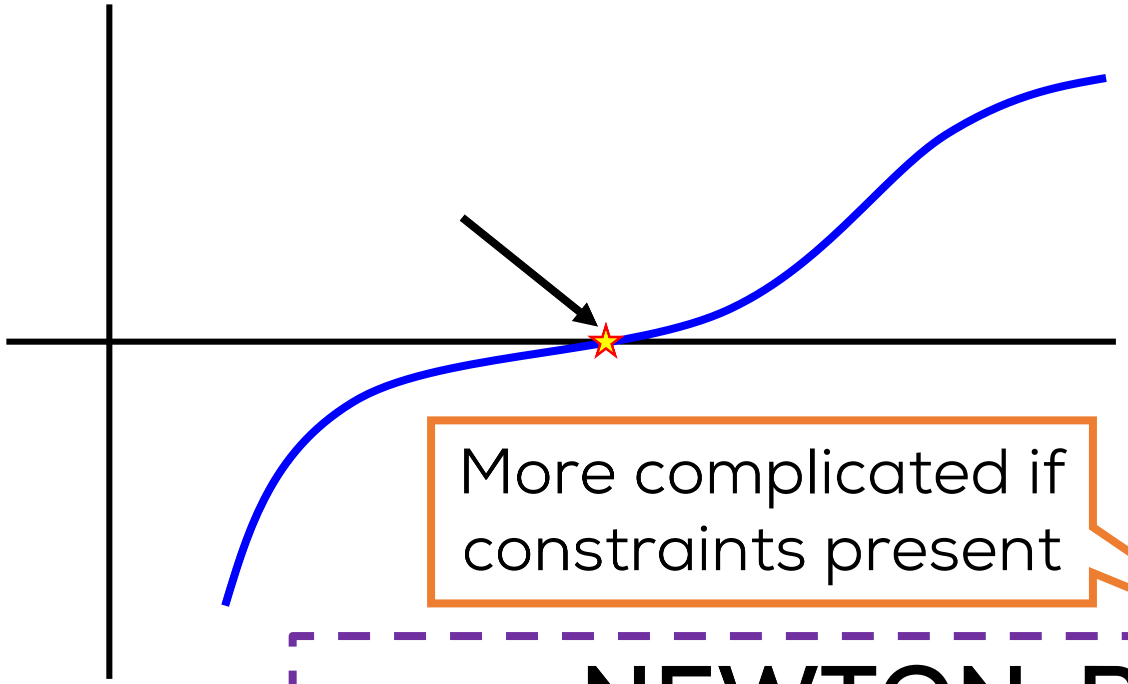
Finding roots of linear functions is easy

$f(x) = ax + b,\ x_0 = \frac{-b}{a}$

NEWTON-RAPHSON METHOD

1. Initialize x^0
2. Approximate $f(\cdot)$ by $\tilde{f}_t(x) = f(x^t) + \langle f'(x^t), x - x^t \rangle$
3. Update $x^{t+1} \leftarrow \mathrm{ROOT}\left(\tilde{f}_t\right) = x^t - \dfrac{f(x^t)}{f'(x^t)}$
4. Repeat until convergence

Newton-Raphson Method for Root Finding



$$x : f(x) = 0$$

Finding roots of linear functions is easy

$f(x) = ax + b,\ x_0 = \frac{-b}{a}$

More complicated if constraints present

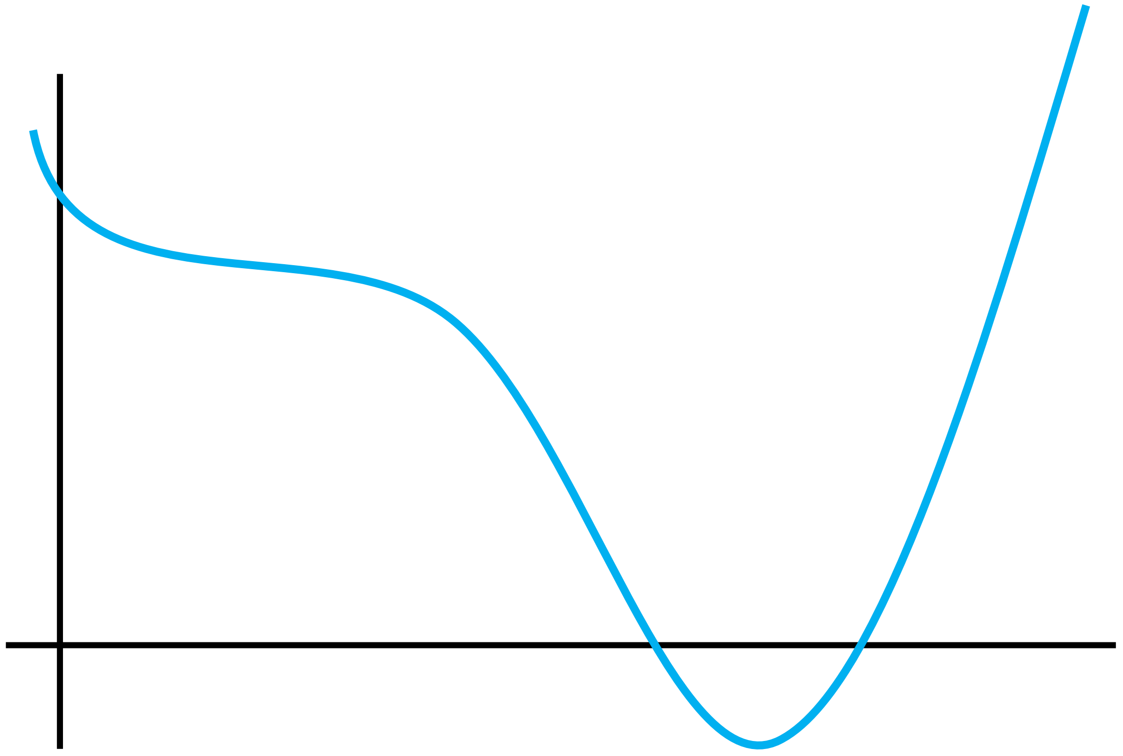
No guarantees in general

NEWTON-RAPHSON METHOD

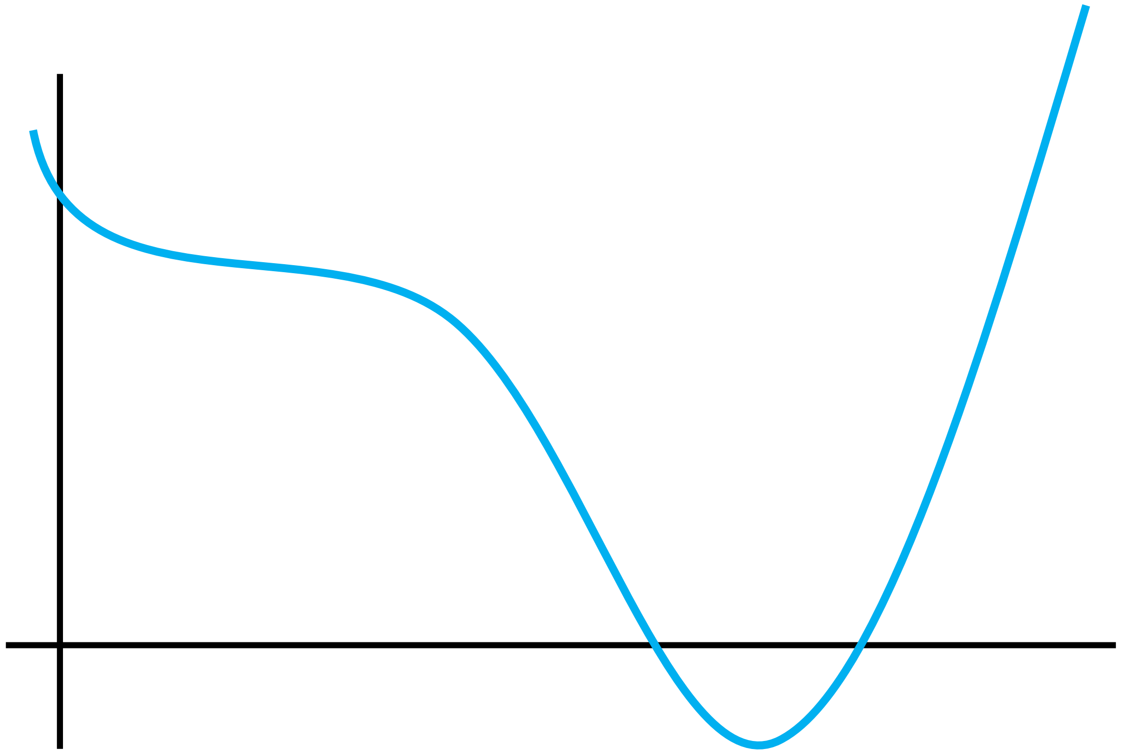
1. Initialize x^0
2. Approximate $f(\cdot)$ by $\tilde{f}_t(x) = f(x^t) + \langle f'(x^t), x - x^t \rangle$
3. Update $x^{t+1} \leftarrow \mathrm{ROOT}(\tilde{f}_t) = x^t - \frac{f(x^t)}{f'(x^t)}$
4. Repeat until convergence

Newton Method

Newton Method

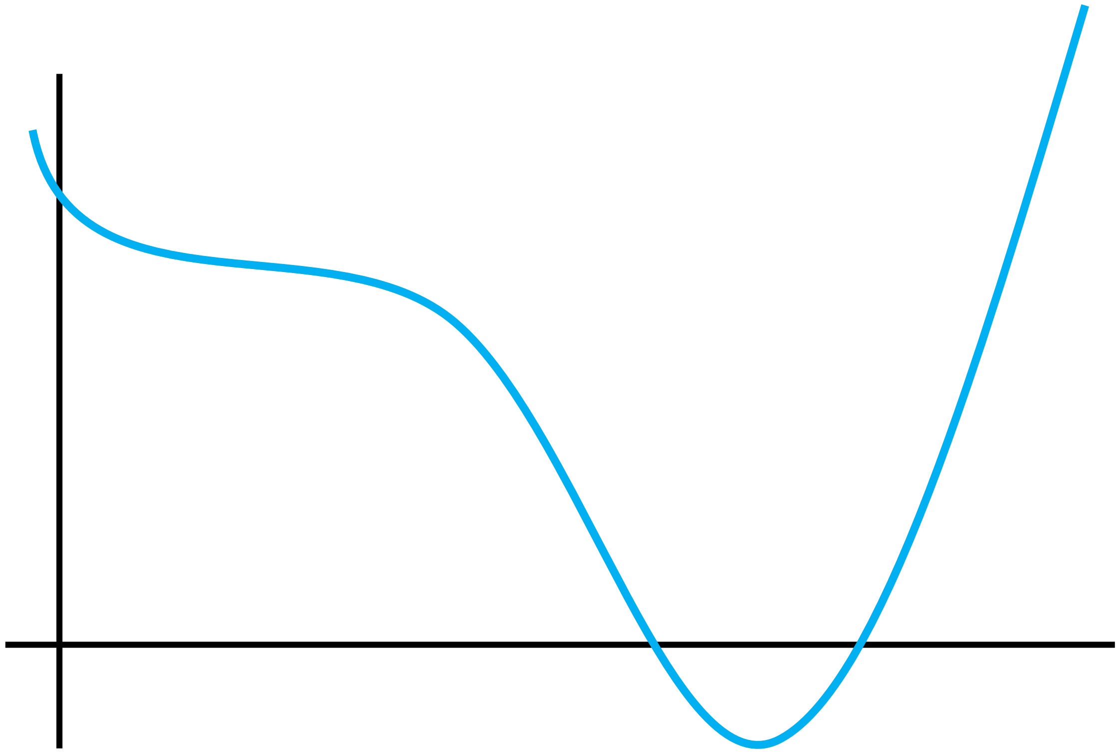


Newton Method



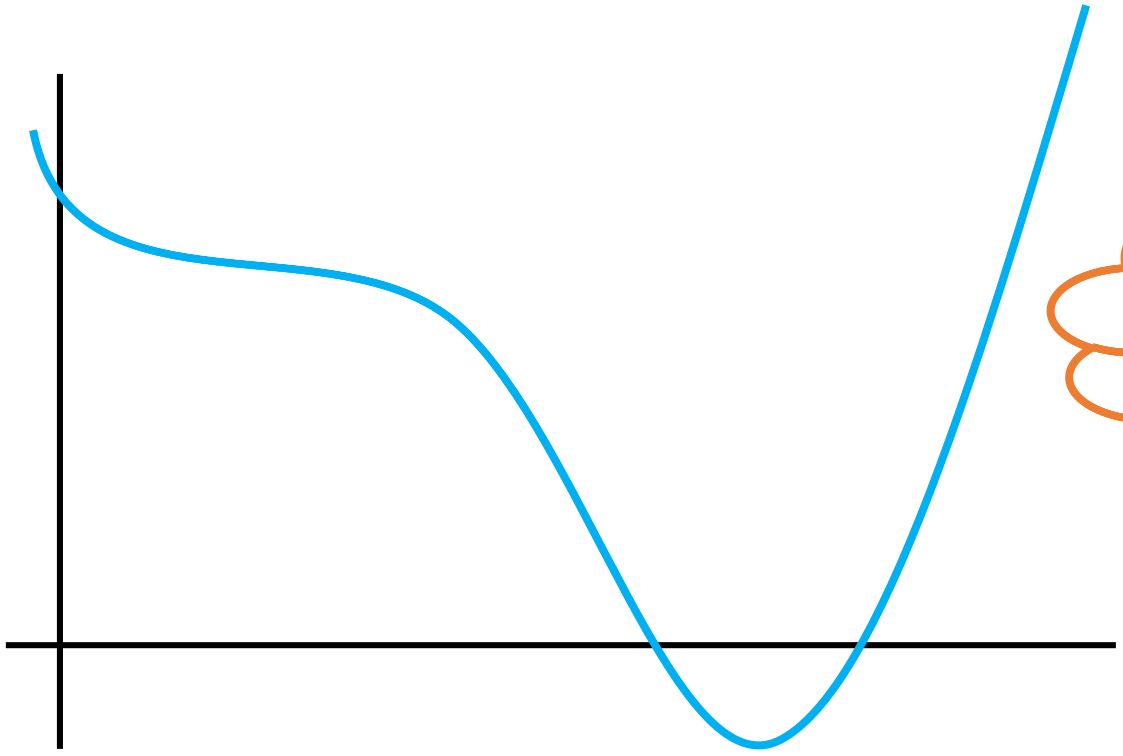
$$\mathbf{x} \in \arg\min f(\mathbf{x})$$

Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

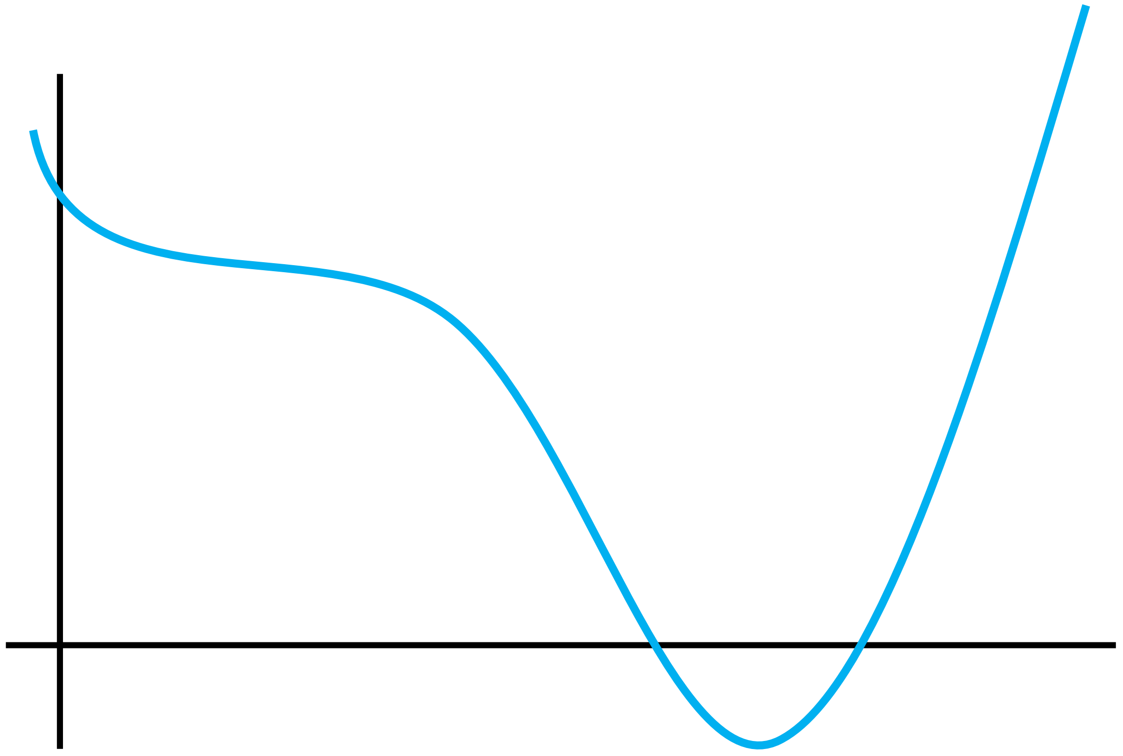
Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

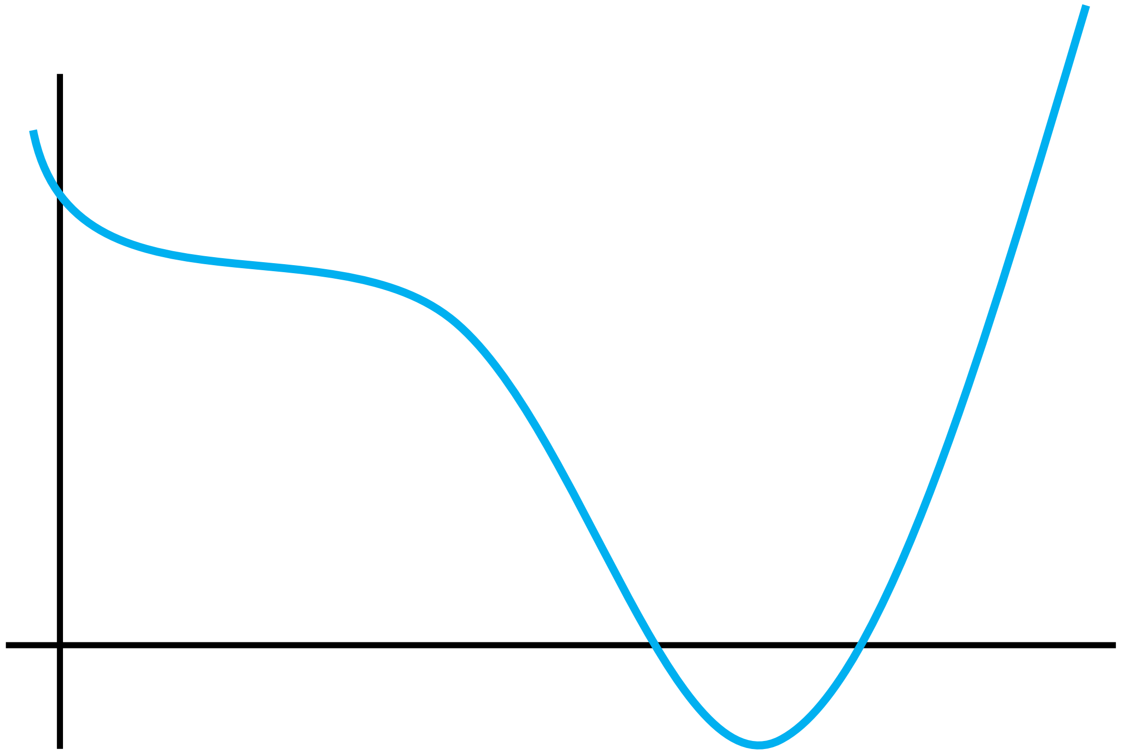
Instead of finding minima, search for stable points instead i.e. $\nabla f(\mathbf{x}) = \mathbf{0}$

Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

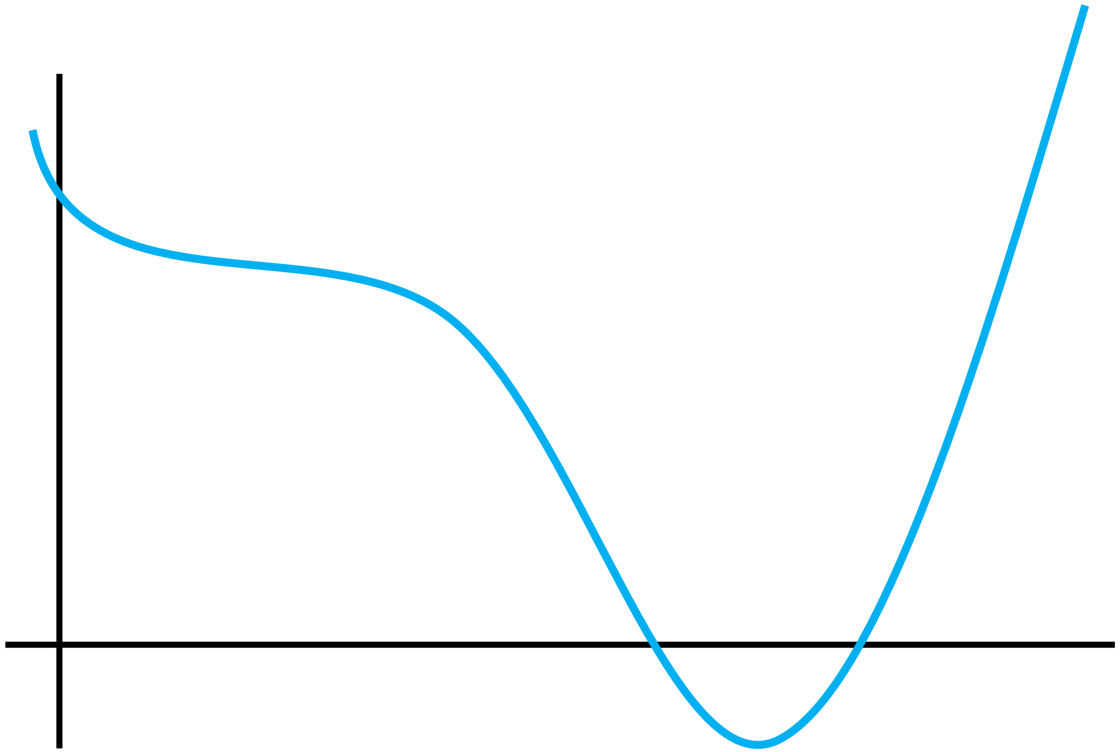
Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

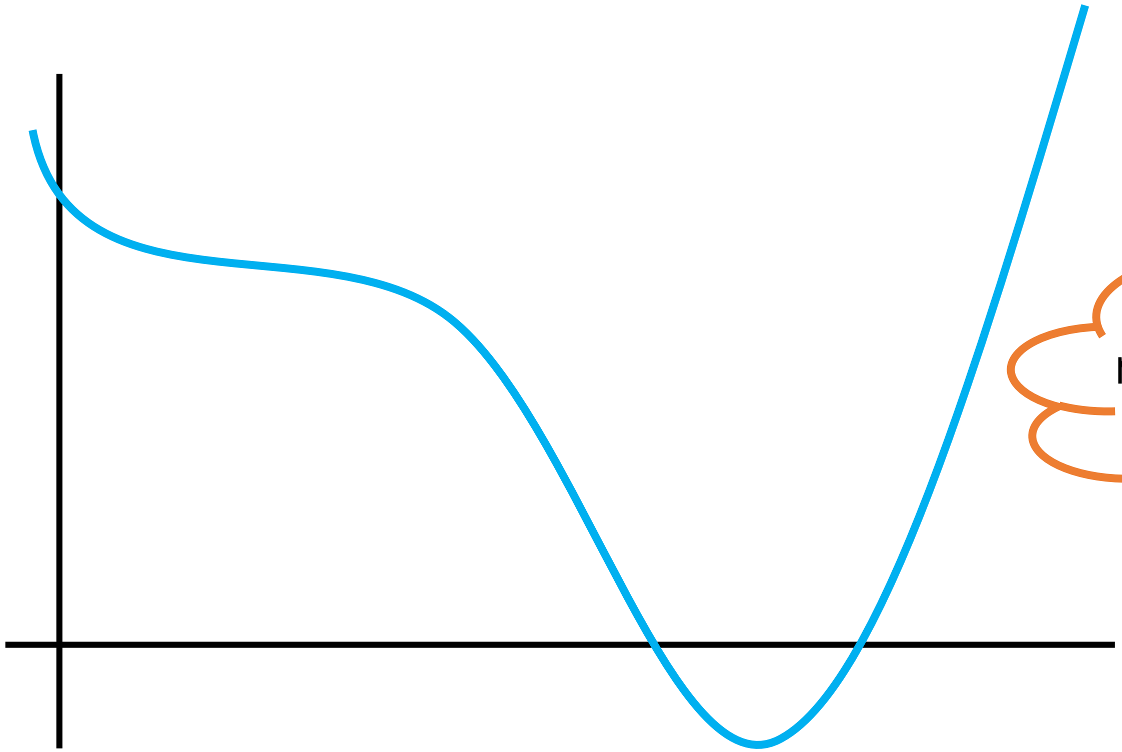
What is the gradient of $\nabla f(\cdot)$?

Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

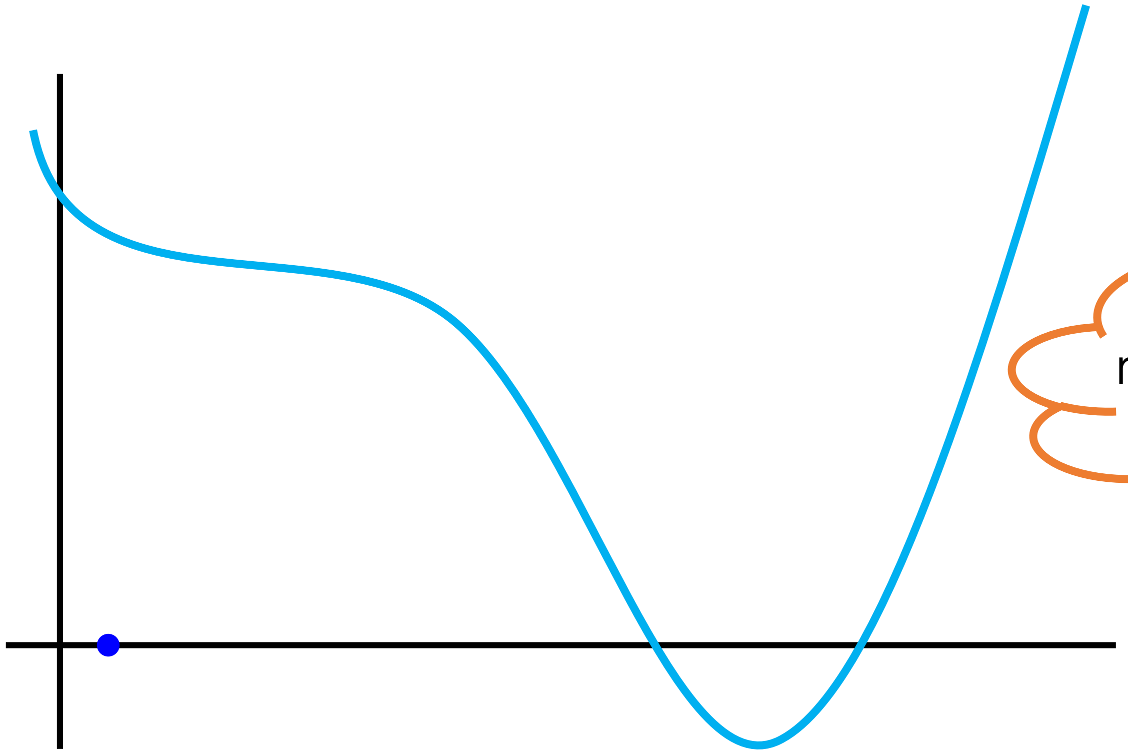
Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy
$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

Newton Method

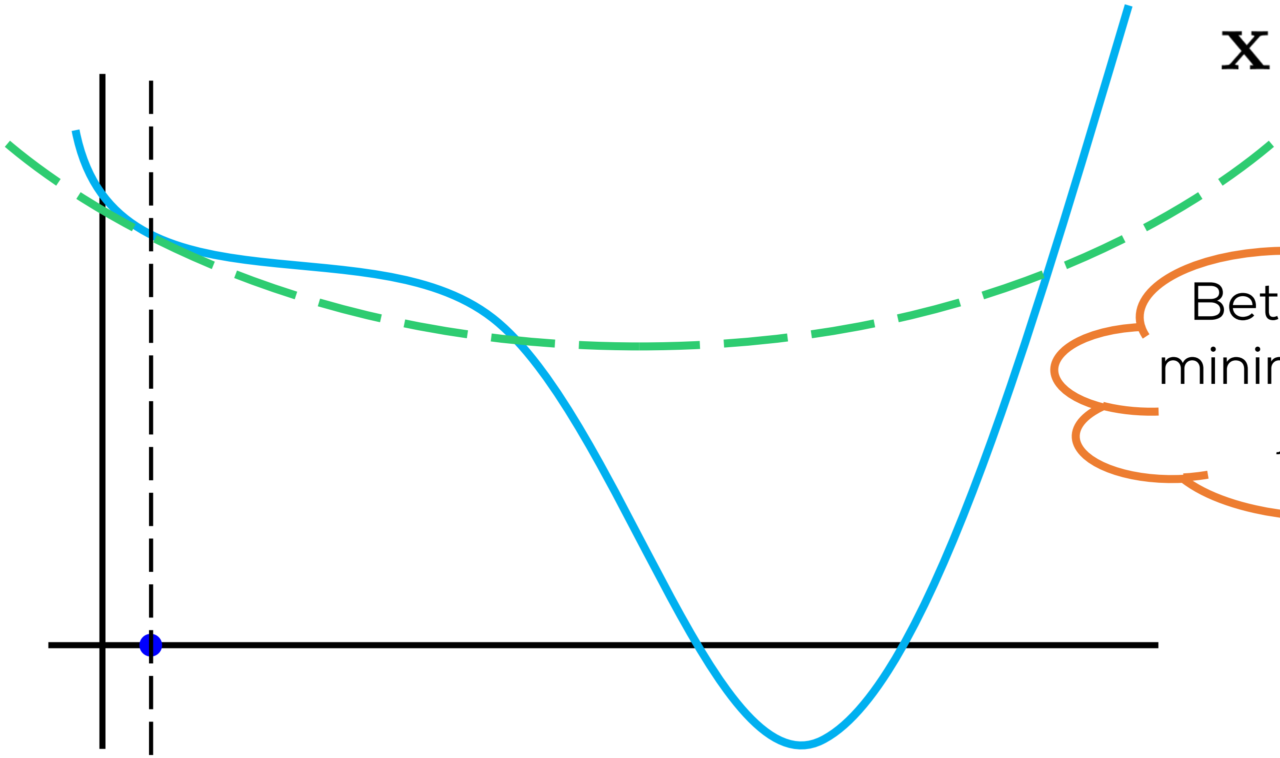


$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy

$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

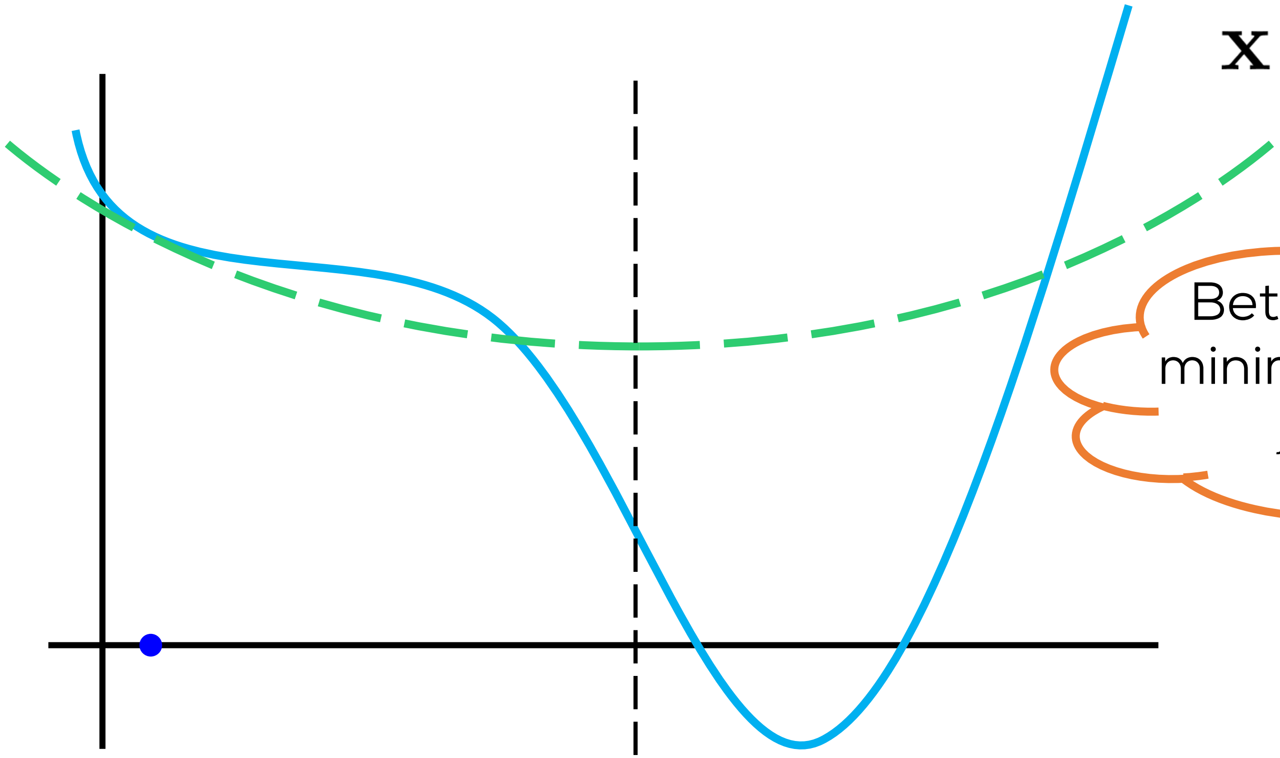
Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy
$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

Newton Method

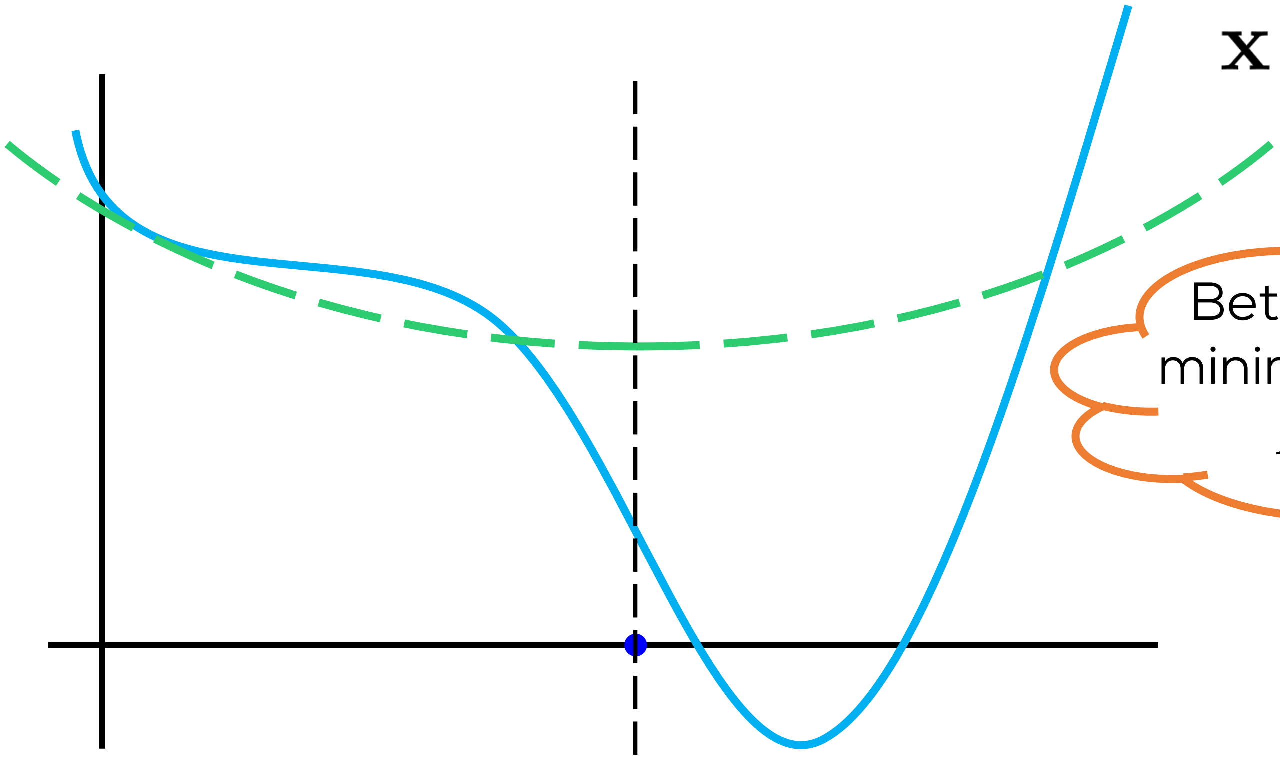


$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy

$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

Newton Method

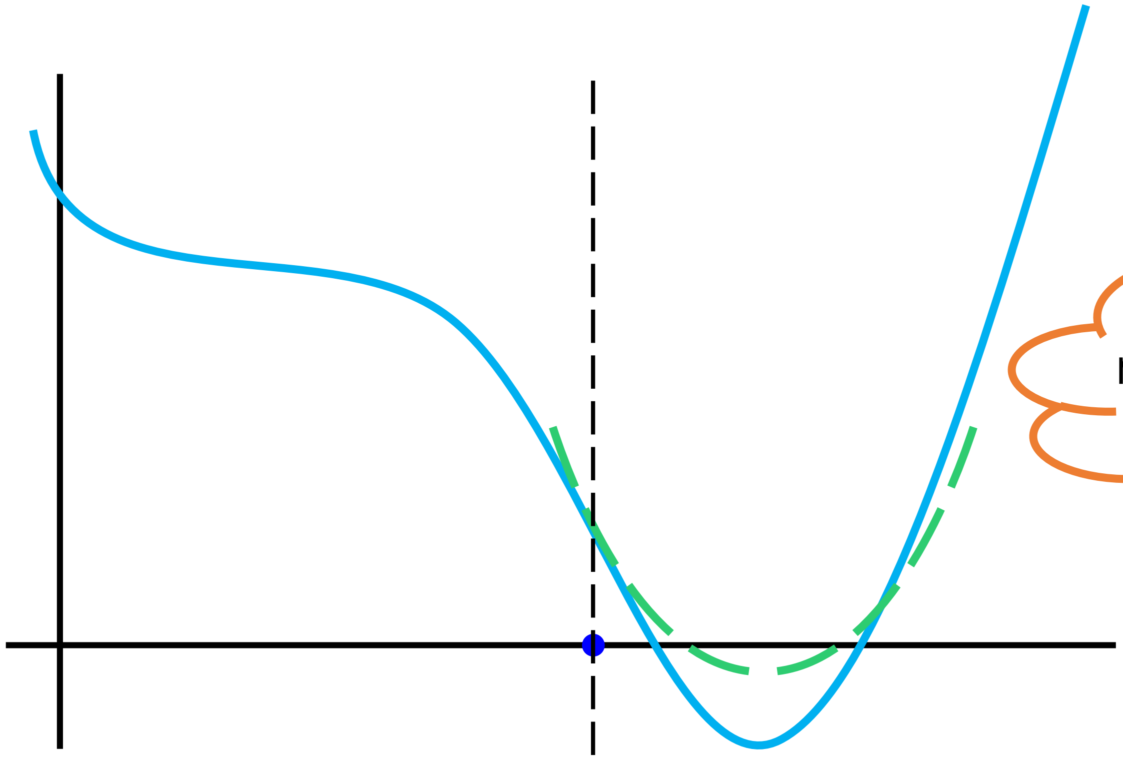


$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$$

Newton Method

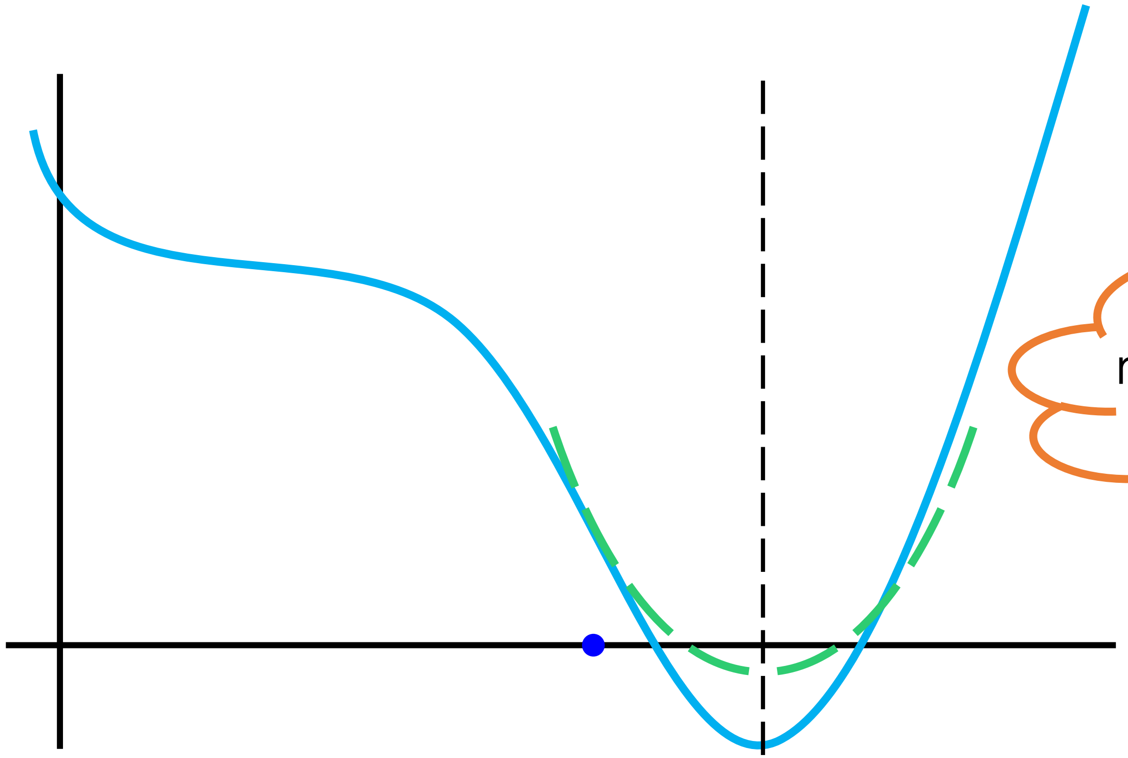


$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy

$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

Newton Method

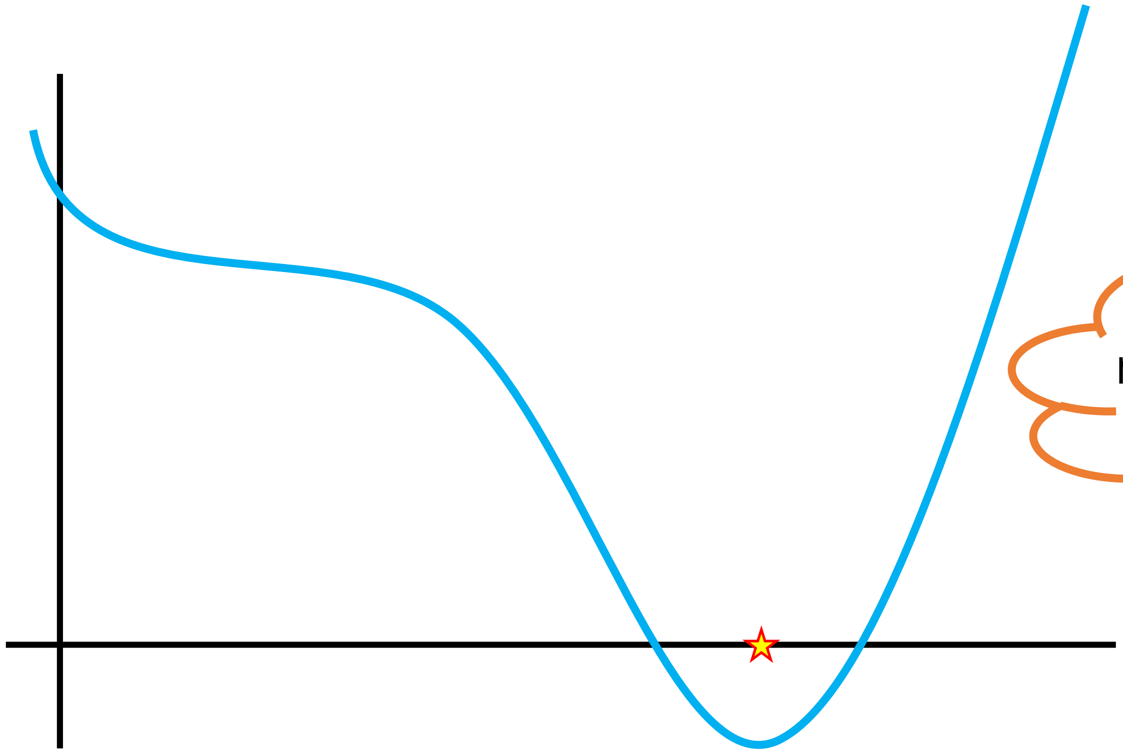


$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$$

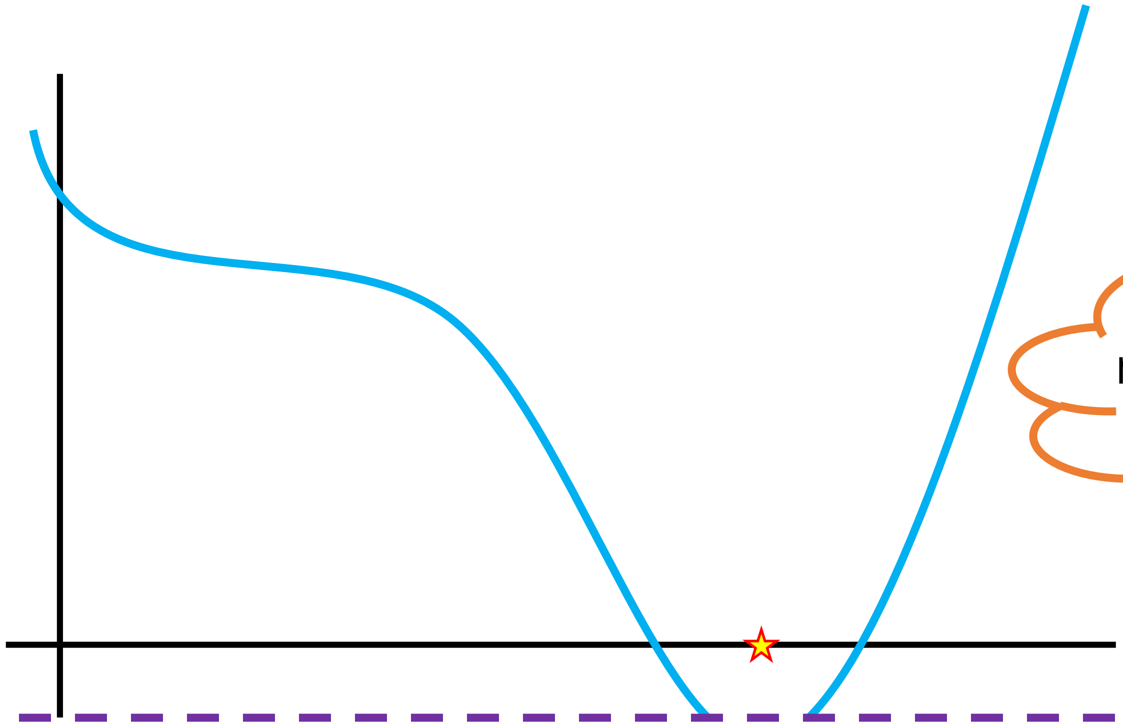
Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy
$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

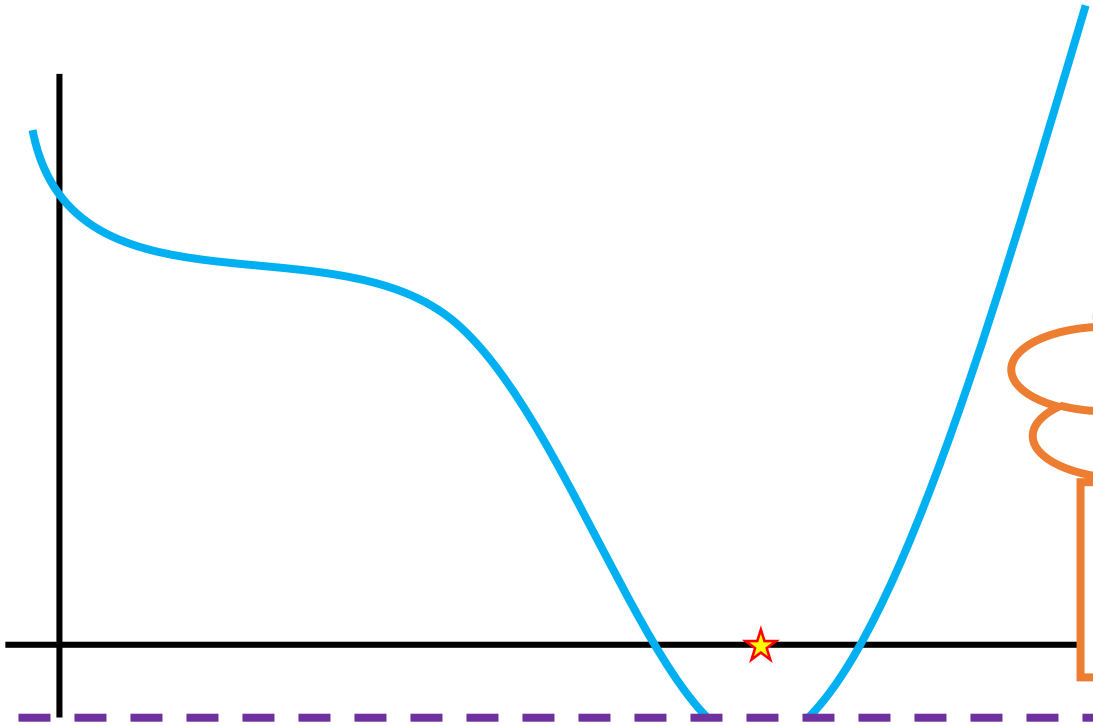
Better motivation: Finding minima of quadratic fn. easy

$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

NEWTON METHOD

1. Initialize $\mathbf{x}^0$
2. Approx. $f(\cdot)$ by $\tilde{f}_t(\mathbf{x}) = f(x^t) + \langle \nabla f(\mathbf{x}^t), \mathbf{x} - \mathbf{x}^t \rangle + \frac{1}{2}\langle \mathbf{x} - \mathbf{x}^t, \nabla^2 f(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t)\rangle$
3. Update $\mathbf{x}^{t+1} \leftarrow \arg\min \tilde{f}_t(\mathbf{x}) = \mathbf{x}^t - \left(\nabla^2 f(\mathbf{x}^t)\right)^{-1} \nabla f(\mathbf{x}^t)$
4. Repeat until convergence

Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

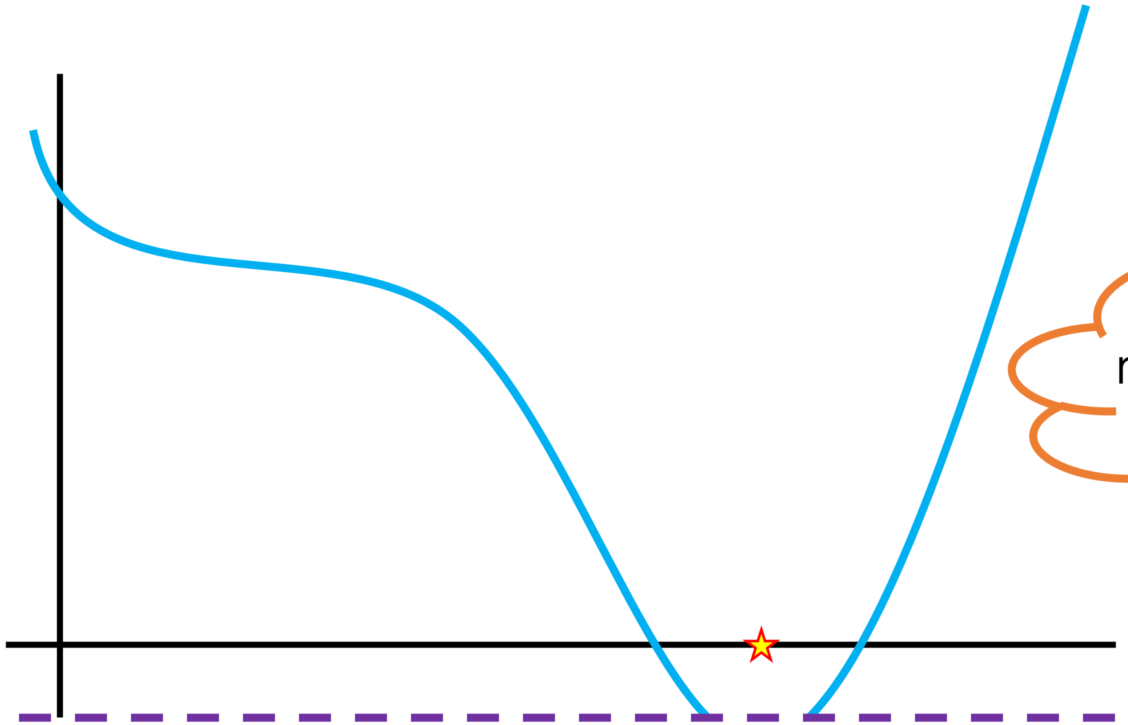
Better motivation: Finding minima of quadratic fn. easy $f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

$f(\cdot)$ and $\tilde{f}_t(\cdot)$ have the same gradient and Hessian at $\mathbf{x}^t$

NEWTON METHOD

1. Initialize $\mathbf{x}^0$
2. Approx. $f(\cdot)$ by $\tilde{f}_t(\mathbf{x}) = f(x^t) + \langle \nabla f(\mathbf{x}^t), \mathbf{x} - \mathbf{x}^t \rangle + \frac{1}{2}\langle \mathbf{x} - \mathbf{x}^t, \nabla^2 f(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t) \rangle$
3. Update $\mathbf{x}^{t+1} \leftarrow \arg\min \tilde{f}_t(\mathbf{x}) = \mathbf{x}^t - \left(\nabla^2 f(\mathbf{x}^t)\right)^{-1} \nabla f(\mathbf{x}^t)$
4. Repeat until convergence

Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

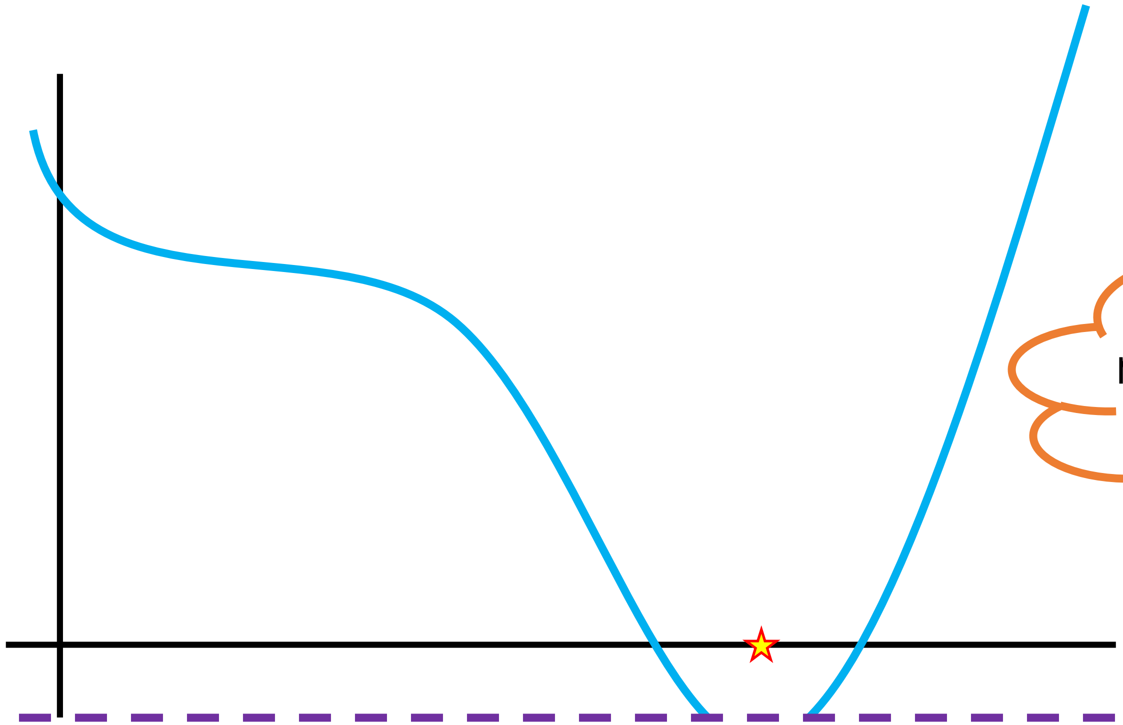
Better motivation: Finding minima of quadratic fn. easy
$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

Ridiculously fast for convex problems

NEWTON METHOD

1. Initialize $\mathbf{x}^0$
2. Approx. $f(\cdot)$ by $\tilde{f}_t(\mathbf{x}) = f(x^t) + \langle \nabla f(\mathbf{x}^t), \mathbf{x} - \mathbf{x}^t \rangle + \frac{1}{2} \langle \mathbf{x} - \mathbf{x}^t, \nabla^2 f(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t) \rangle$
3. Update $\mathbf{x}^{t+1} \leftarrow \arg\min \tilde{f}_t(\mathbf{x}) = \mathbf{x}^t - \left(\nabla^2 f(\mathbf{x}^t)\right)^{-1} \nabla f(\mathbf{x}^t)$
4. Repeat until convergence

Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy
$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

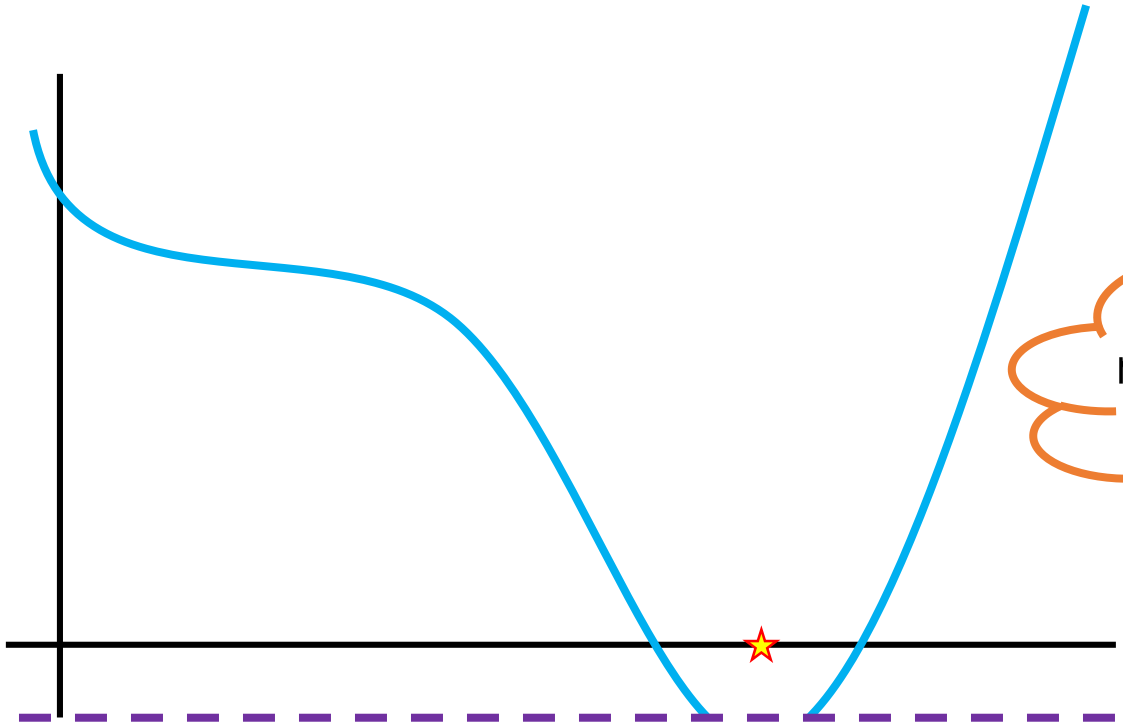
$O\left(\log\log\frac{1}{\epsilon}\right)$ steps

Ridiculously fast for convex problems

NEWTON METHOD

1. Initialize $\mathbf{x}^0$
2. Approx. $f(\cdot)$ by $\tilde{f}_t(\mathbf{x}) = f(x^t) + \langle \nabla f(\mathbf{x}^t), \mathbf{x} - \mathbf{x}^t \rangle + \frac{1}{2}\langle \mathbf{x} - \mathbf{x}^t, \nabla^2 f(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t)\rangle$
3. Update $\mathbf{x}^{t+1} \leftarrow \arg\min \tilde{f}_t(\mathbf{x}) = \mathbf{x}^t - \left(\nabla^2 f(\mathbf{x}^t)\right)^{-1} \nabla f(\mathbf{x}^t)$
4. Repeat until convergence

Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy
$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

$O\left(\log\log\frac{1}{\epsilon}\right)$ steps

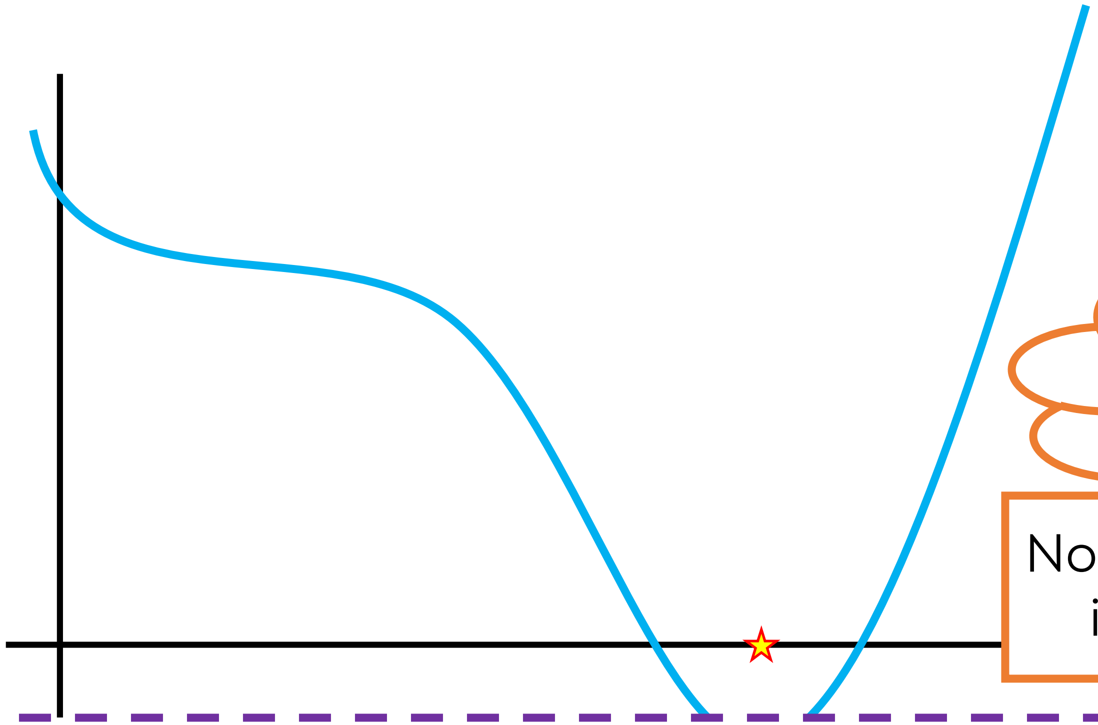
Steps more costly $O(d^2)$

Ridiculously fast for convex problems

NEWTON METHOD

1. Initialize $\mathbf{x}^0$
2. Approx. $f(\cdot)$ by $\tilde{f}_t(\mathbf{x}) = f(x^t) + \langle \nabla f(\mathbf{x}^t), \mathbf{x} - \mathbf{x}^t \rangle + \frac{1}{2}\langle \mathbf{x} - \mathbf{x}^t, \nabla^2 f(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t) \rangle$
3. Update $\mathbf{x}^{t+1} \leftarrow \arg\min \tilde{f}_t(\mathbf{x}) = \mathbf{x}^t - \left(\nabla^2 f(\mathbf{x}^t)\right)^{-1} \nabla f(\mathbf{x}^t)$
4. Repeat until convergence

Newton Method



$$\mathbf{x} \in \arg\min f(\mathbf{x}) \subseteq \mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}$$

Better motivation: Finding minima of quadratic fn. easy $f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \langle \mathbf{b}, \mathbf{x} \rangle + c$

$\mathcal{O}\left(\log\log\frac{1}{\epsilon}\right)$ steps

No guarantees in general

Steps more costly $\mathcal{O}(d^2)$

Ridiculously fast for convex problems

NEWTON METHOD

1. Initialize $\mathbf{x}^0$
2. Approx. $f(\cdot)$ by $\tilde{f}_t(\mathbf{x}) = f(x^t) + \langle \nabla f(\mathbf{x}^t), \mathbf{x} - \mathbf{x}^t \rangle + \frac{1}{2}\langle \mathbf{x} - \mathbf{x}^t, \nabla^2 f(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t) \rangle$
3. Update $\mathbf{x}^{t+1} \leftarrow \arg\min \tilde{f}_t(\mathbf{x}) = \mathbf{x}^t - \left(\nabla^2 f(\mathbf{x}^t)\right)^{-1} \nabla f(\mathbf{x}^t)$
4. Repeat until convergence

Duality

How to take the problem you want to solve
… and convert it to a problem you can solve

Fenchel, Lagrange, Wolfe, Pontryagin

Duality

How to take the problem you want to solve
… and convert it to a problem you can solve

Fenchel, Lagrange, Wolfe, Pontryagin

Duality

How to take the problem you want to solve
… and convert it to a problem you can solve

The Lagrangian

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w})$$

$$\text{s.t.} \;\; g(\mathbf{w}) \le 0$$

$$\hat{\mathbf{w}} = \arg\min \sum_{i=1}^{n} \left(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right)^2$$

$$s.t. \; \|\mathbf{w}\|_2 \le r$$

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

Want $g(\mathbf{w}) \geq 0$?
$-g(\mathbf{w}) \leq 0$

The Lagrangian

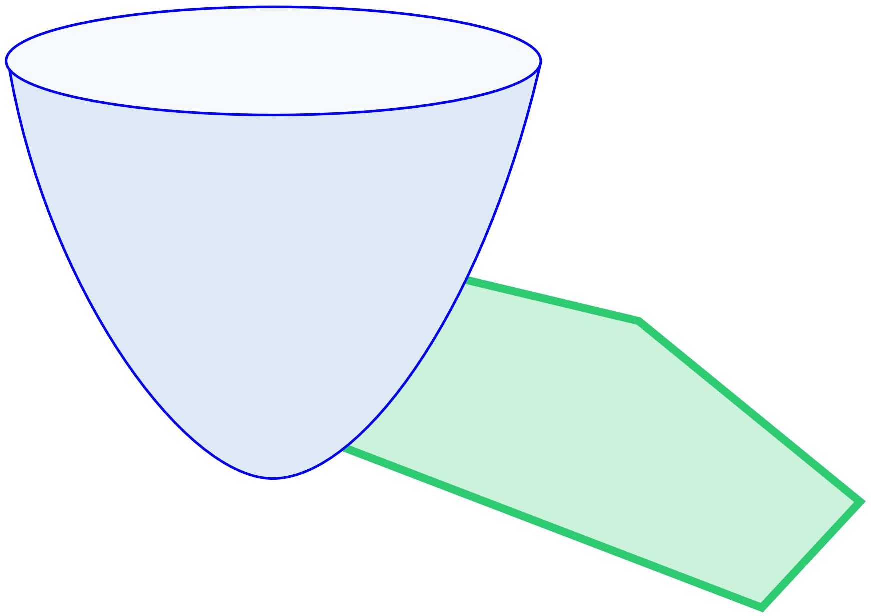
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

The Lagrangian

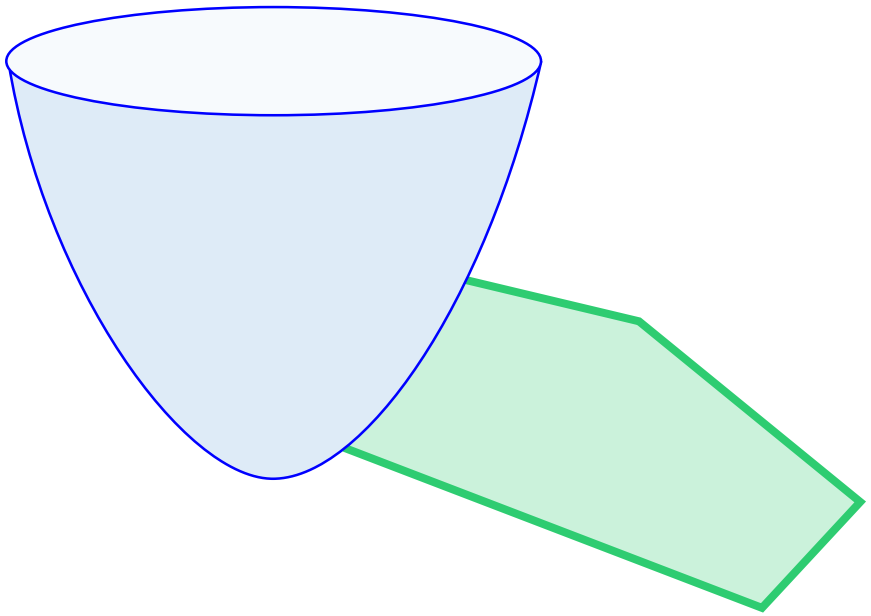
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



The Lagrangian

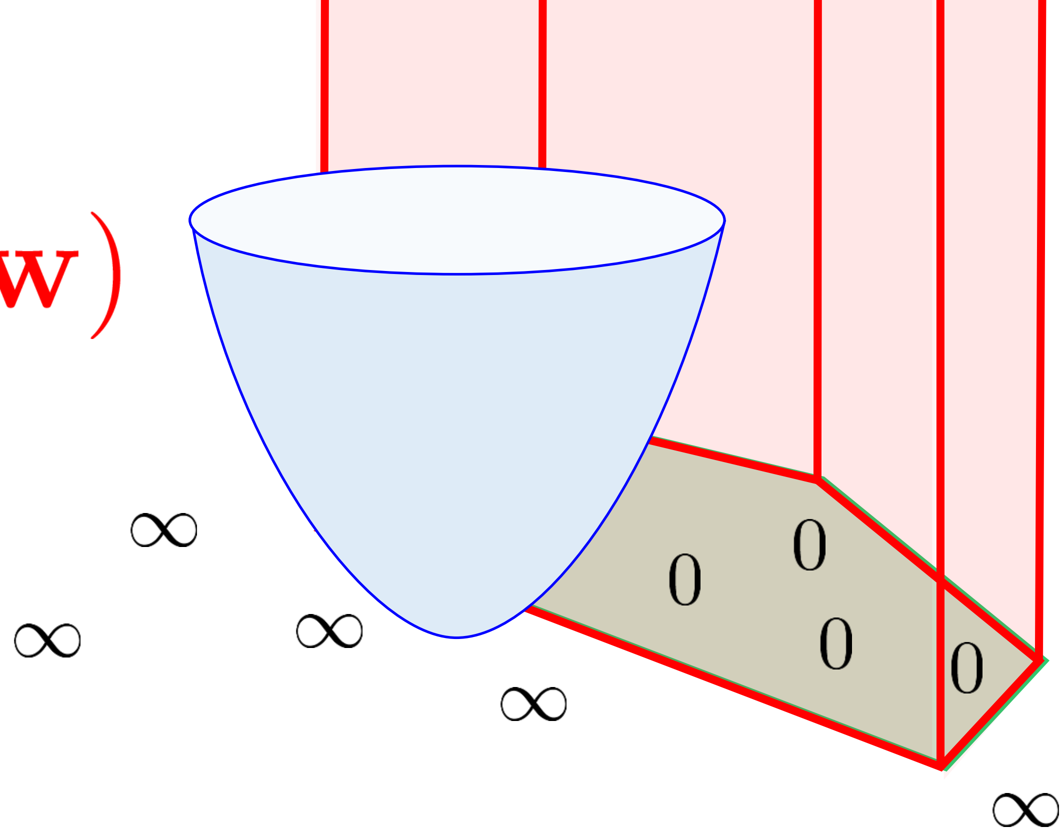
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$



The Lagrangian

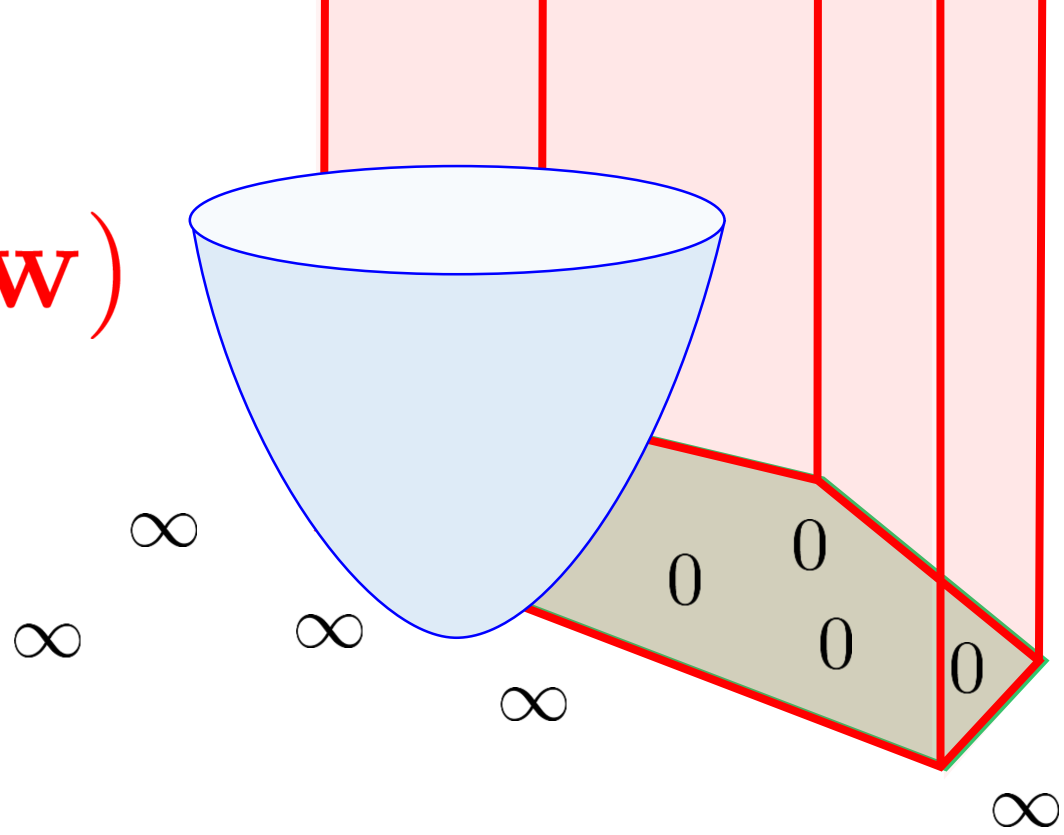
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) + r(\mathbf{w})$$

Barrier Function



The Lagrangian

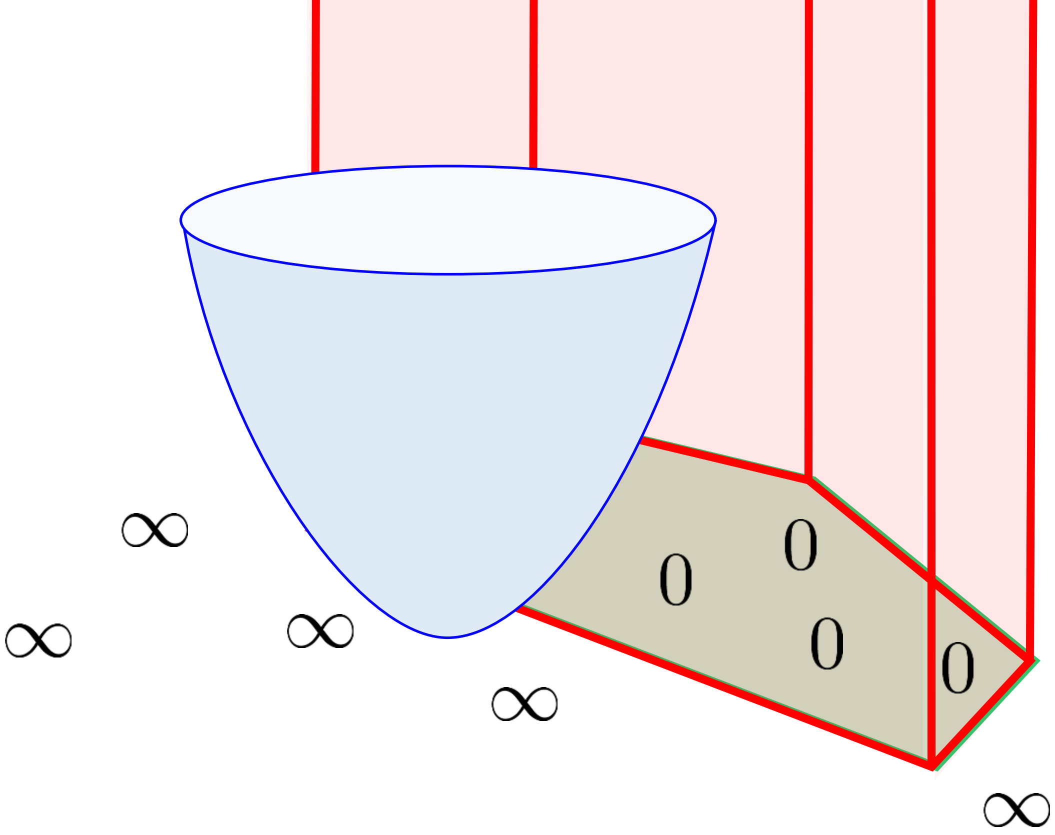
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w}) + r(\mathbf{w})$$



The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

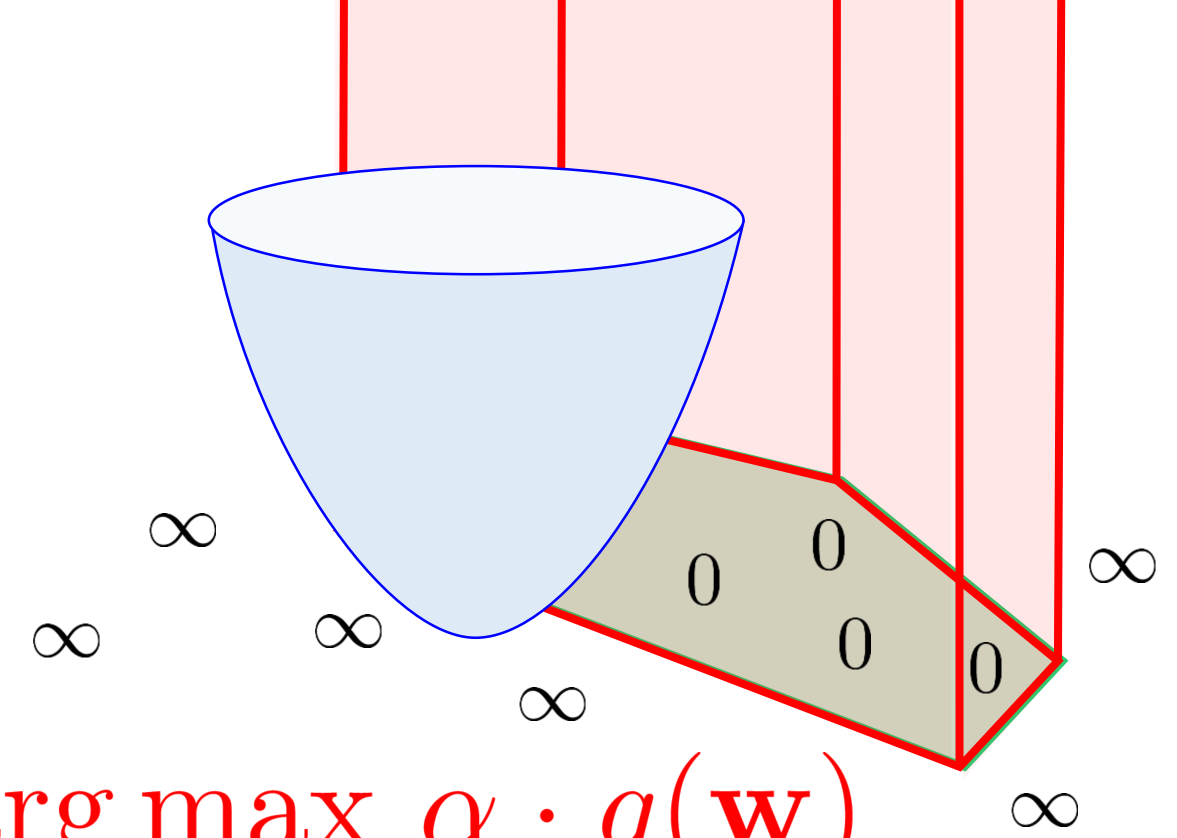
$$\text{s.t. } g(\mathbf{w}) \leq 0$$



The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

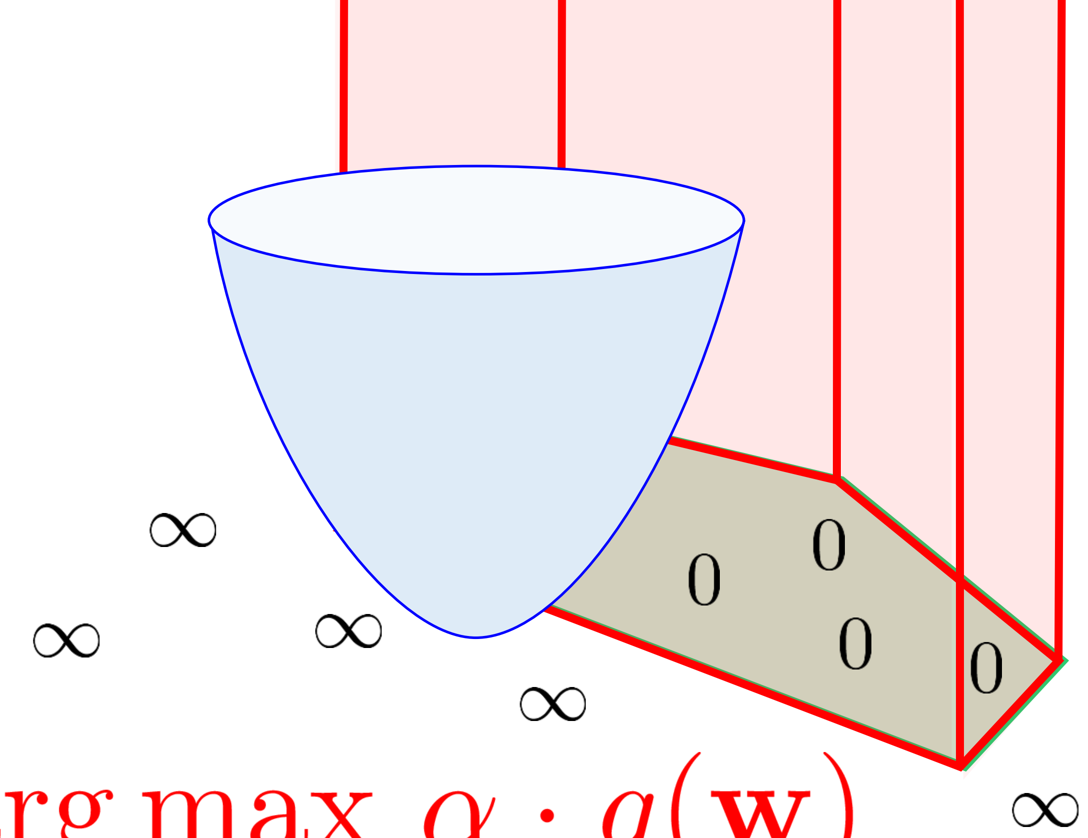


$$\arg\max_{\alpha \geq 0} \alpha \cdot g(\mathbf{w})$$

The Lagrangian

$$\hat{\mathbf{w}} = \underset{\mathbf{w} \in \mathbb{R}^d}{\arg\min}\; f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



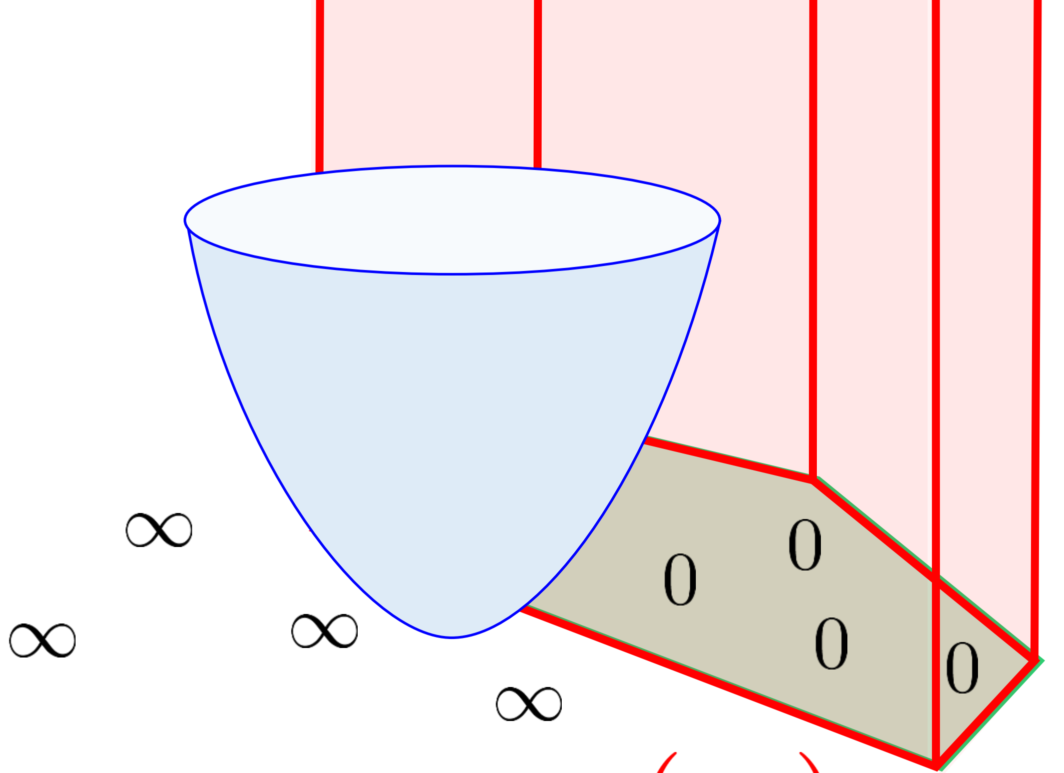
$$\underset{\alpha \geq 0}{\arg\max}\; \alpha \cdot g(\mathbf{w})$$

∞ if $g(\mathbf{w}) > 0$

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



$$\arg\max_{\alpha\geq 0} \alpha \cdot g(\mathbf{w})$$

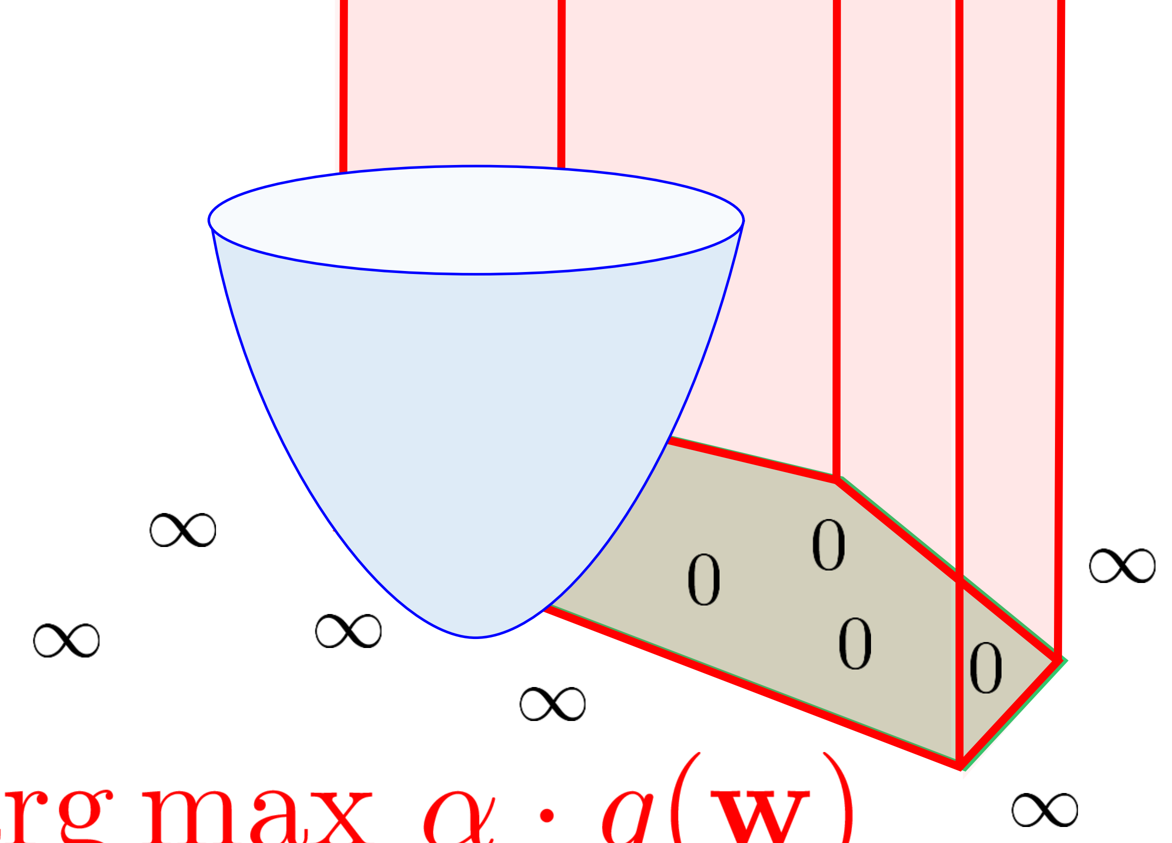
∞ if $g(\mathbf{w}) > 0$

0 if $g(\mathbf{w}) \leq 0$

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



$$f(\mathbf{w}) + \arg\max_{\alpha\geq0} \; \alpha \cdot g(\mathbf{w})$$

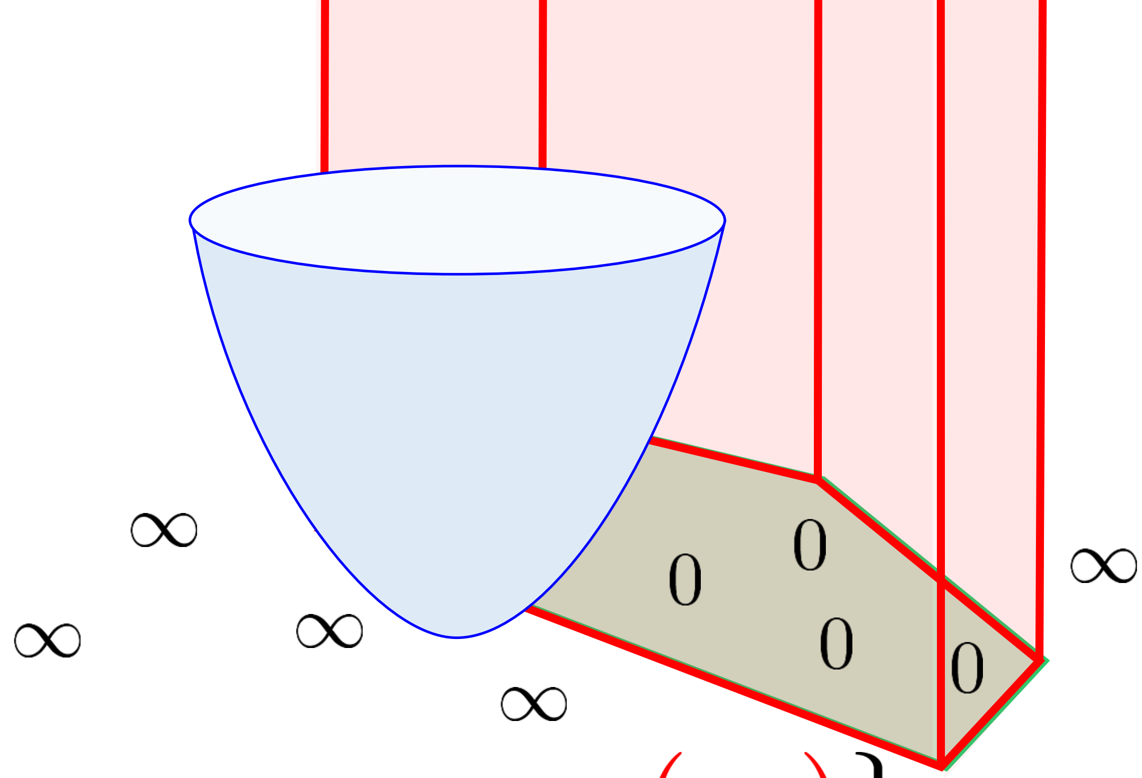
∞ if $g(\mathbf{w}) > 0$

0 if $g(\mathbf{w}) \leq 0$

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



$$\arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{ f(\mathbf{w}) + \arg\max_{\alpha\geq 0} \alpha \cdot g(\mathbf{w}) \right\}$$

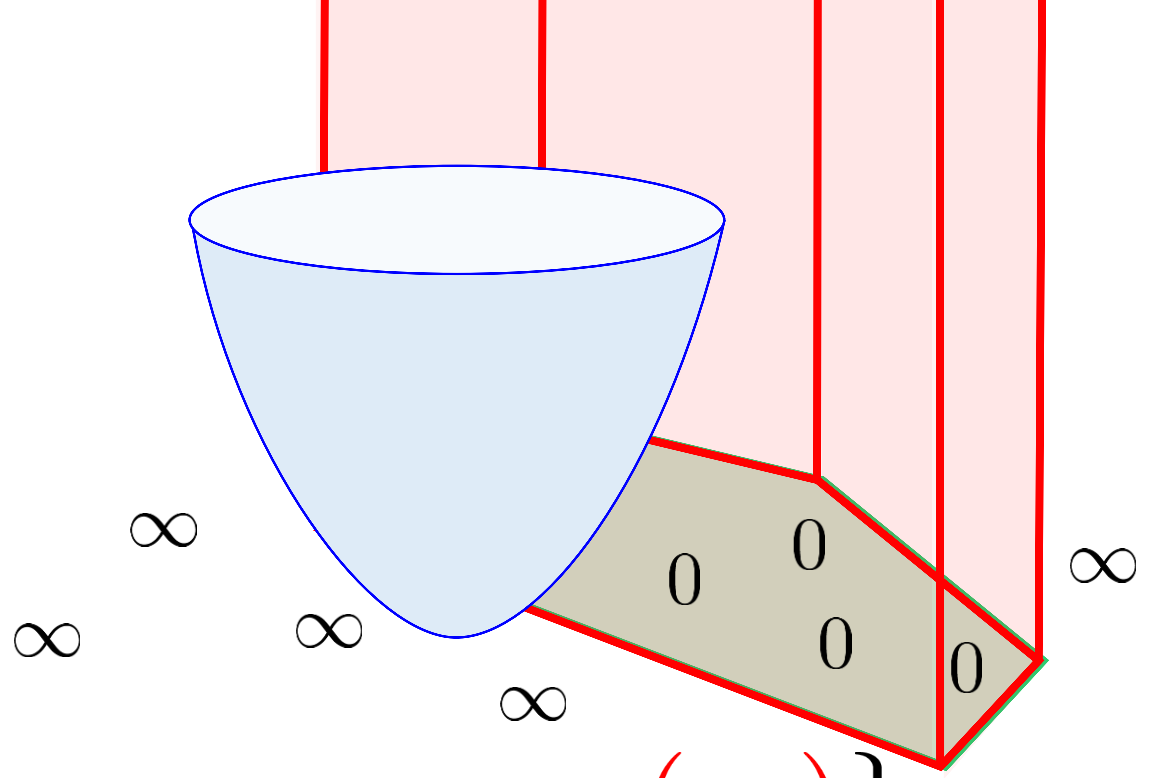
∞ if $g(\mathbf{w}) > 0$

0 if $g(\mathbf{w}) \leq 0$

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

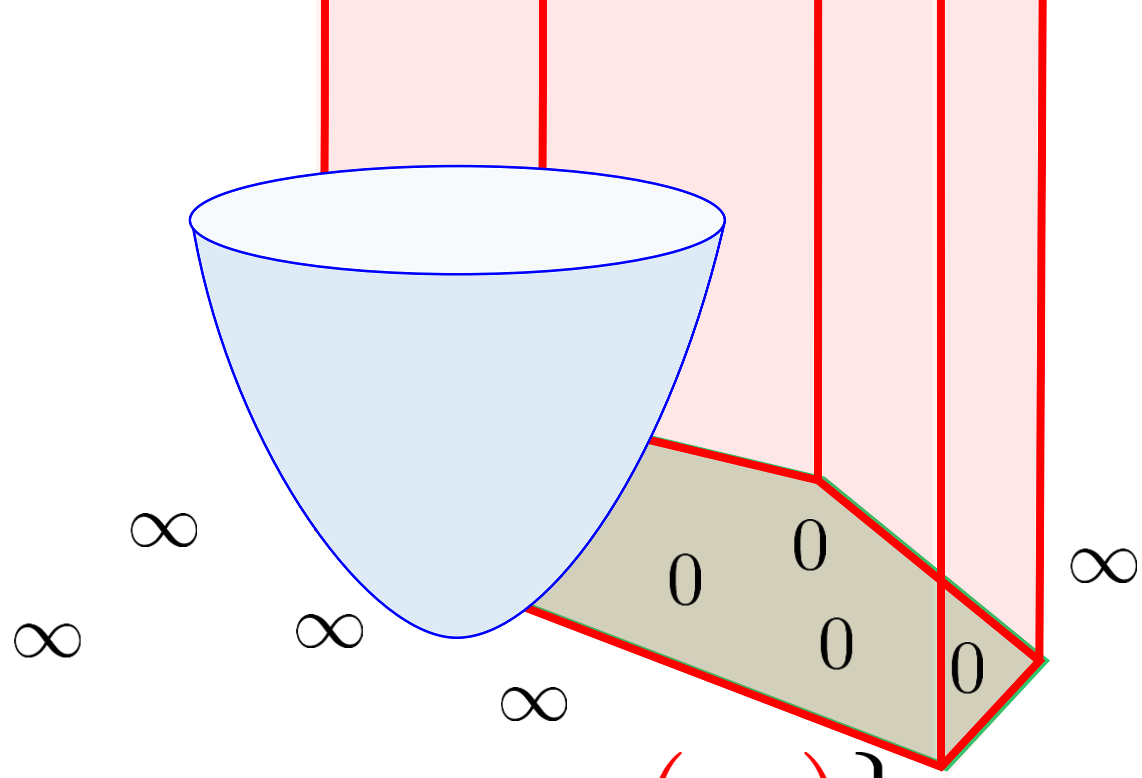


$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{ f(\mathbf{w}) + \arg\max_{\alpha\geq 0} \alpha\cdot g(\mathbf{w}) \right\}$$

∞ if $g(\mathbf{w}) > 0$

0 if $g(\mathbf{w}) \leq 0$

The Lagrangian



$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{ f(\mathbf{w}) + \arg\max_{\alpha\geq 0} \alpha\cdot g(\mathbf{w}) \right\} \quad \infty$$

∞ if $g(\mathbf{w}) > 0$

0 if $g(\mathbf{w}) \leq 0$

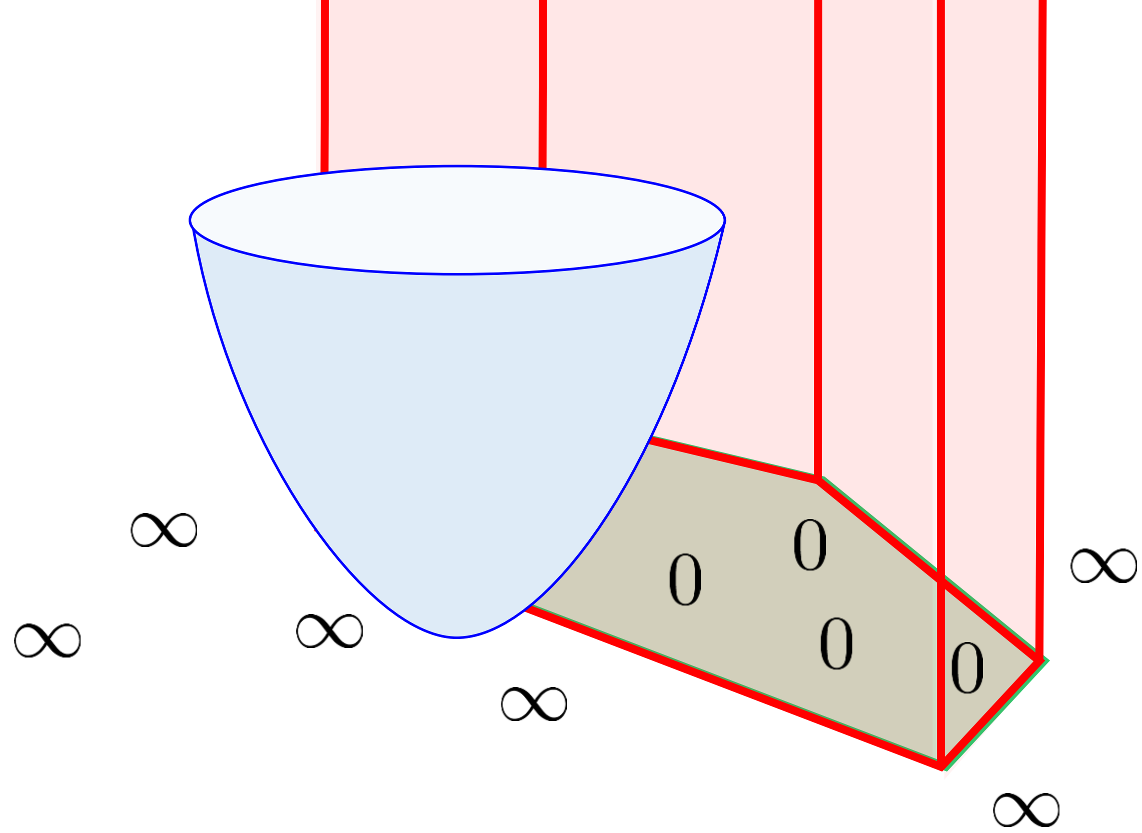
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{ \arg\max_{\alpha\geq 0} \left\{ f(\mathbf{w}) + \alpha\cdot g(\mathbf{w}) \right\} \right\}$$

Exercise

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

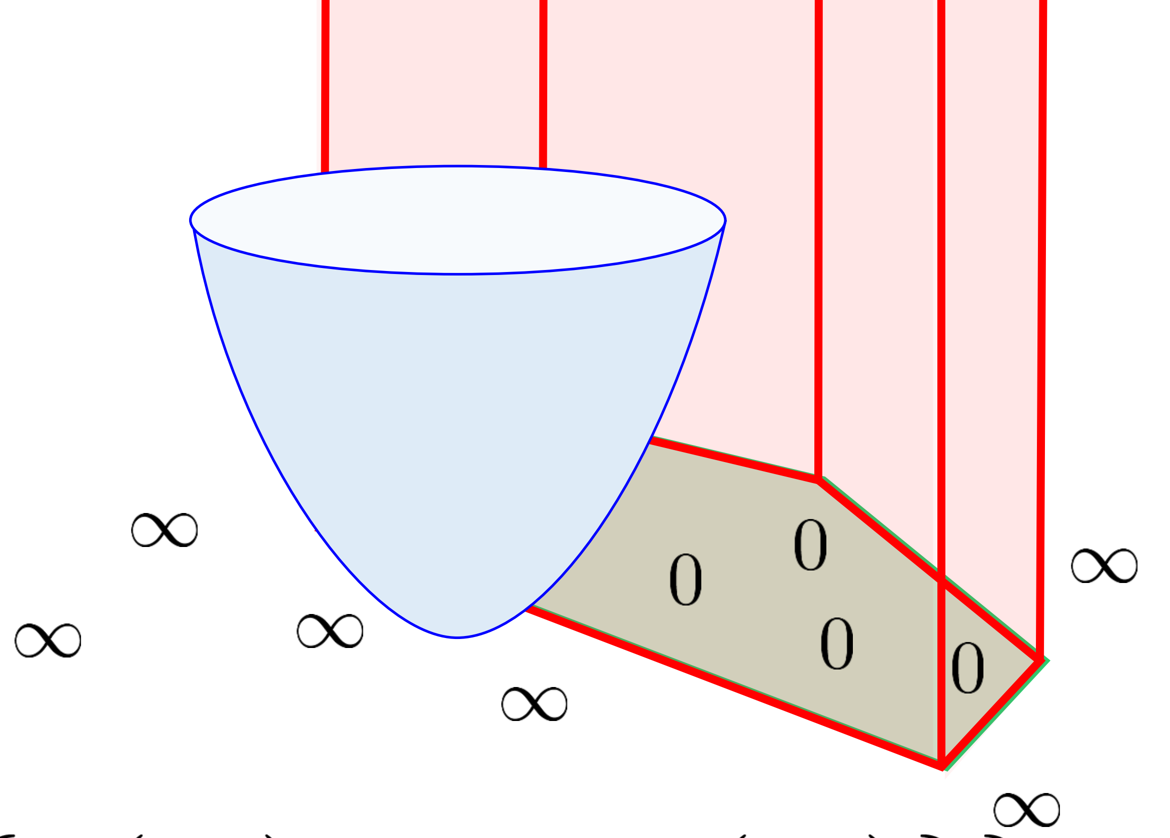


$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\arg\max_{\alpha\geq 0} \left\{f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})\right\}\right\}$$

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



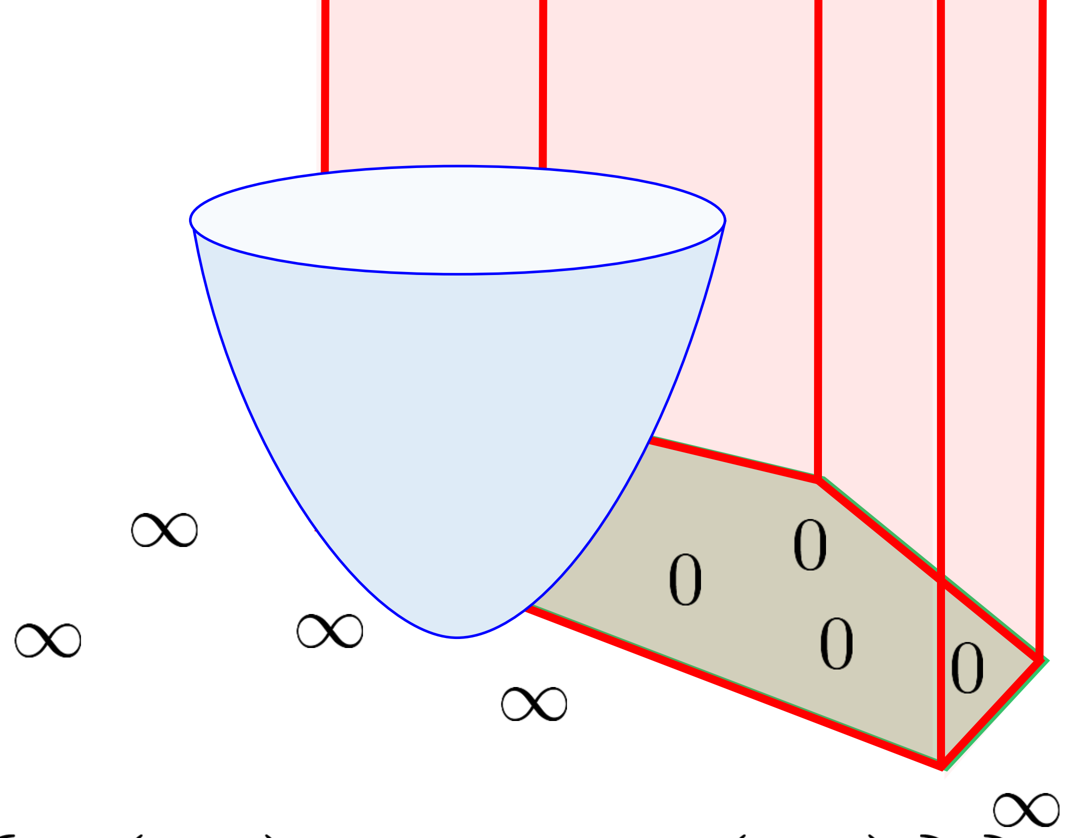
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\arg\max_{\alpha\geq 0} \left\{f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})\right\}\right\}$$

$$\mathcal{L}(\mathbf{w},\alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\arg\max_{\alpha\geq 0} \left\{f(\mathbf{w}) + \alpha\cdot g(\mathbf{w})\right\}\right\}$$

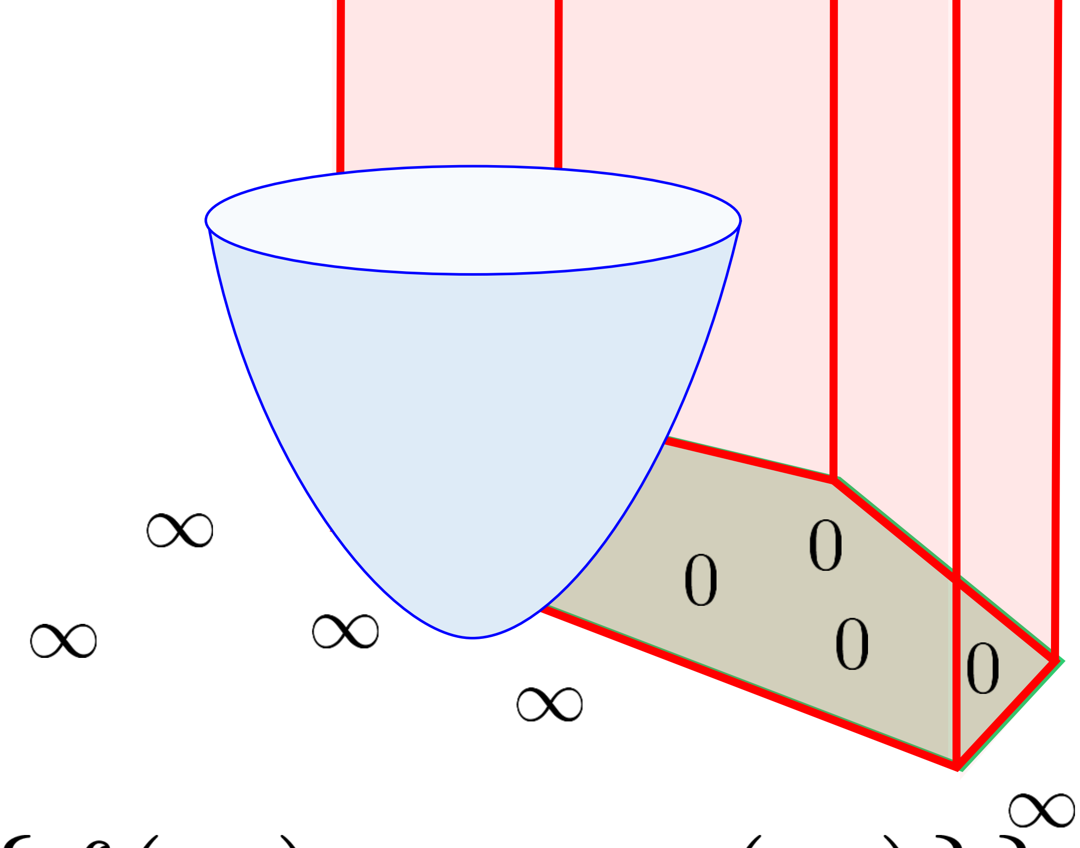
$$\mathcal{L}(\mathbf{w},\alpha) = f(\mathbf{w}) + \alpha\cdot g(\mathbf{w})$$

Lagrangian

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\arg\max_{\alpha\geq 0} \{f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})\}\right\}$$

Lagrange multiplier

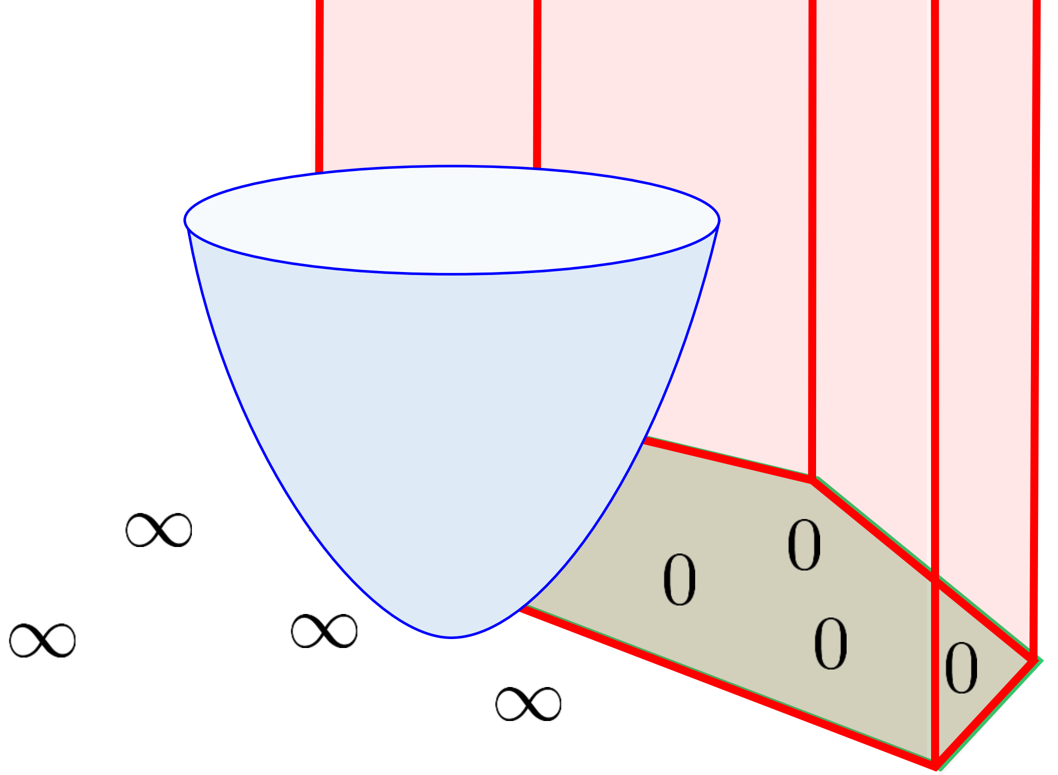
$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

Lagrangian

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \{\arg\max_{\alpha\geq 0} \{ \quad \mathcal{L}(\mathbf{w},\alpha) \quad \}\}$$

∞

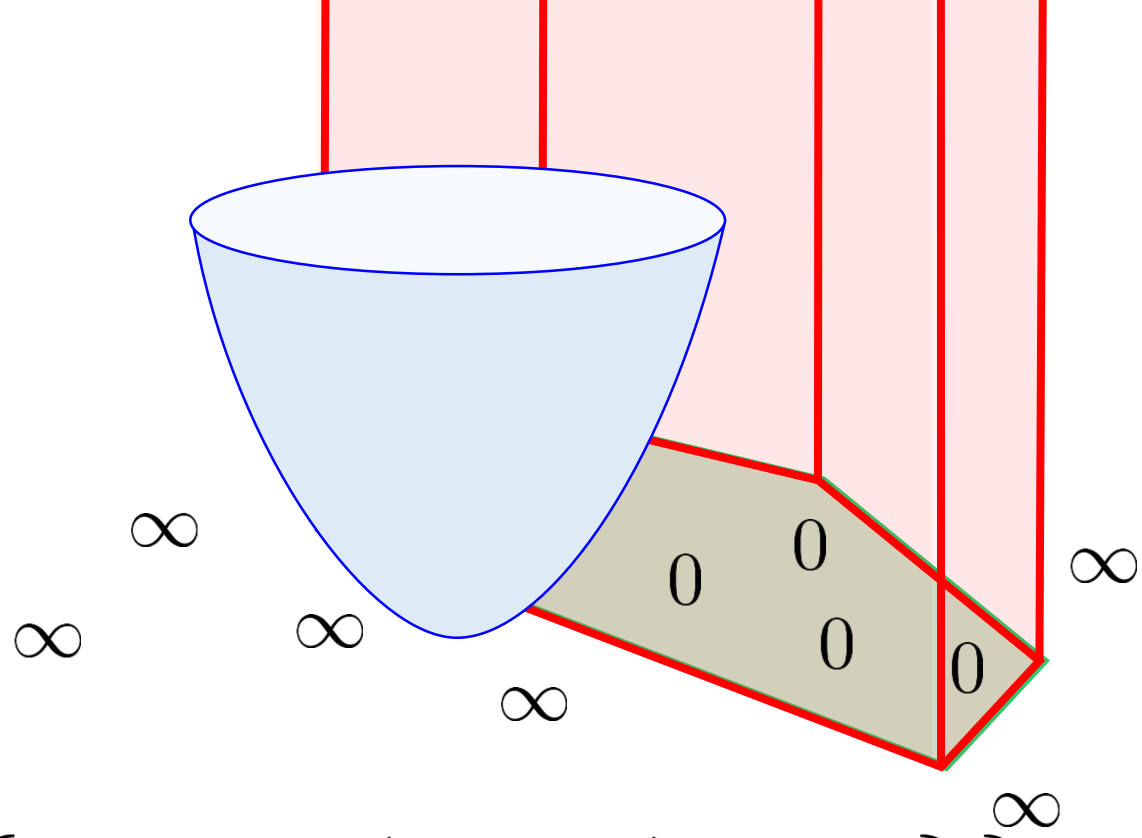
Lagrange multiplier

$$\mathcal{L}(\mathbf{w},\alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

Lagrangian

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) \quad \text{s.t. } g(\mathbf{w}) \le 0$$



$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{ \arg\max_{\alpha\ge 0} \left\{ \mathcal{L}(\mathbf{w},\alpha) \right\} \right\}$$

Lagrange multiplier

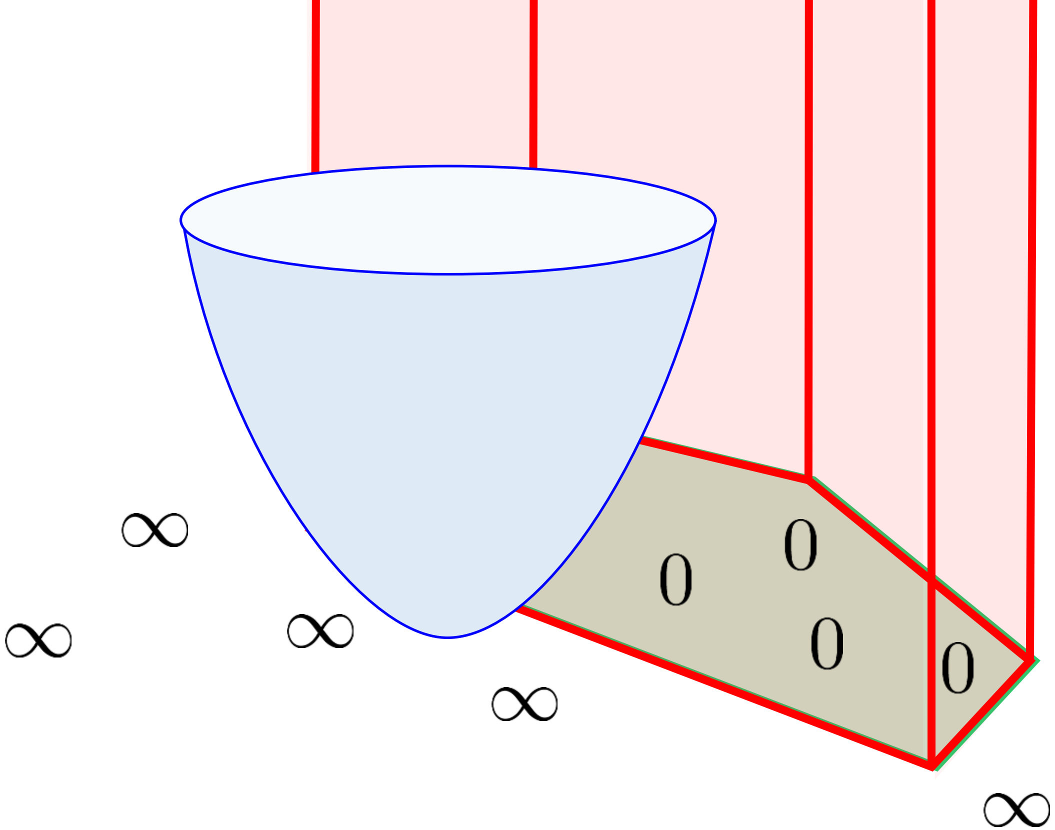
$$\mathcal{L}(\mathbf{w},\alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

Lagrangian

The Lagrangian

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



$$\hat{\mathbf{w}}_P = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{ \arg\max_{\alpha\geq 0} \left\{ \mathcal{L}(\mathbf{w},\alpha) \right\} \right\}$$

Lagrange multiplier

Primal problem

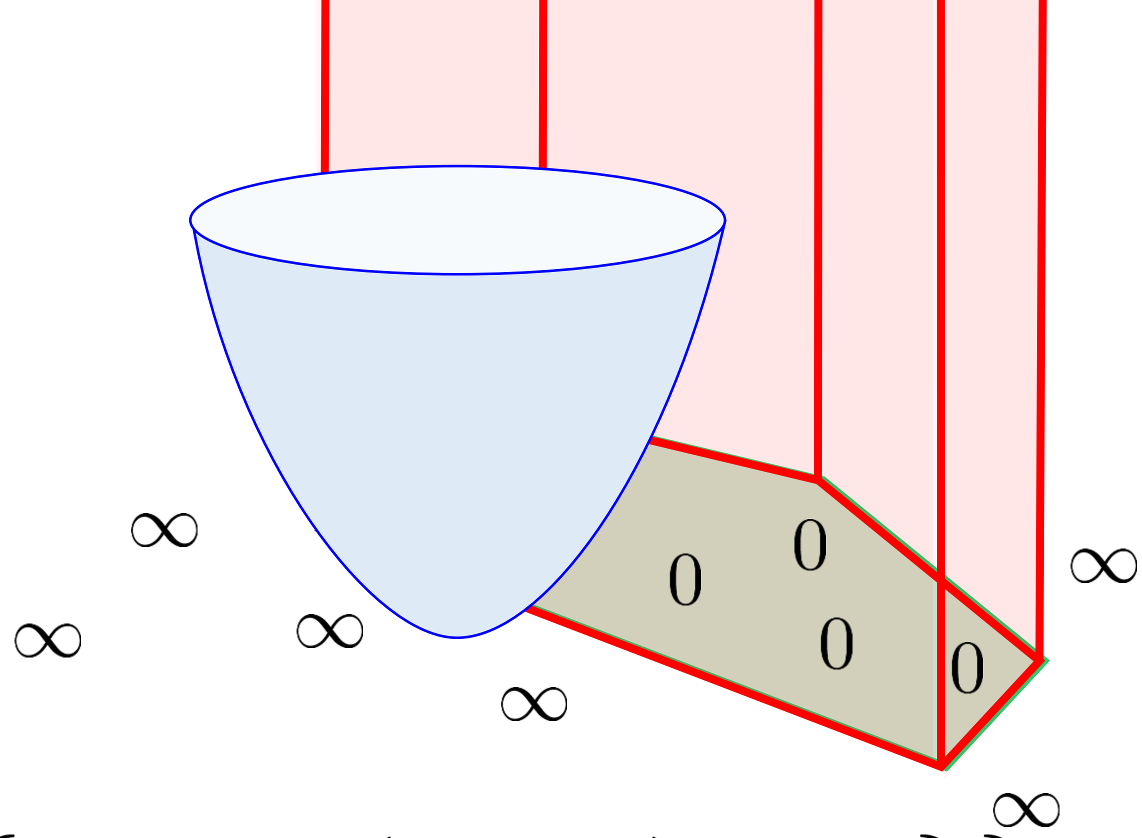
$$\mathcal{L}(\mathbf{w},\alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

Lagrangian

The Lagrangian

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$



$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\arg\max_{\alpha\geq 0} \left\{ \mathcal{L}(\mathbf{w},\alpha) \right\}\right\}$$

Lagrange multiplier

Primal problem

$$\mathcal{L}(\mathbf{w},\alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

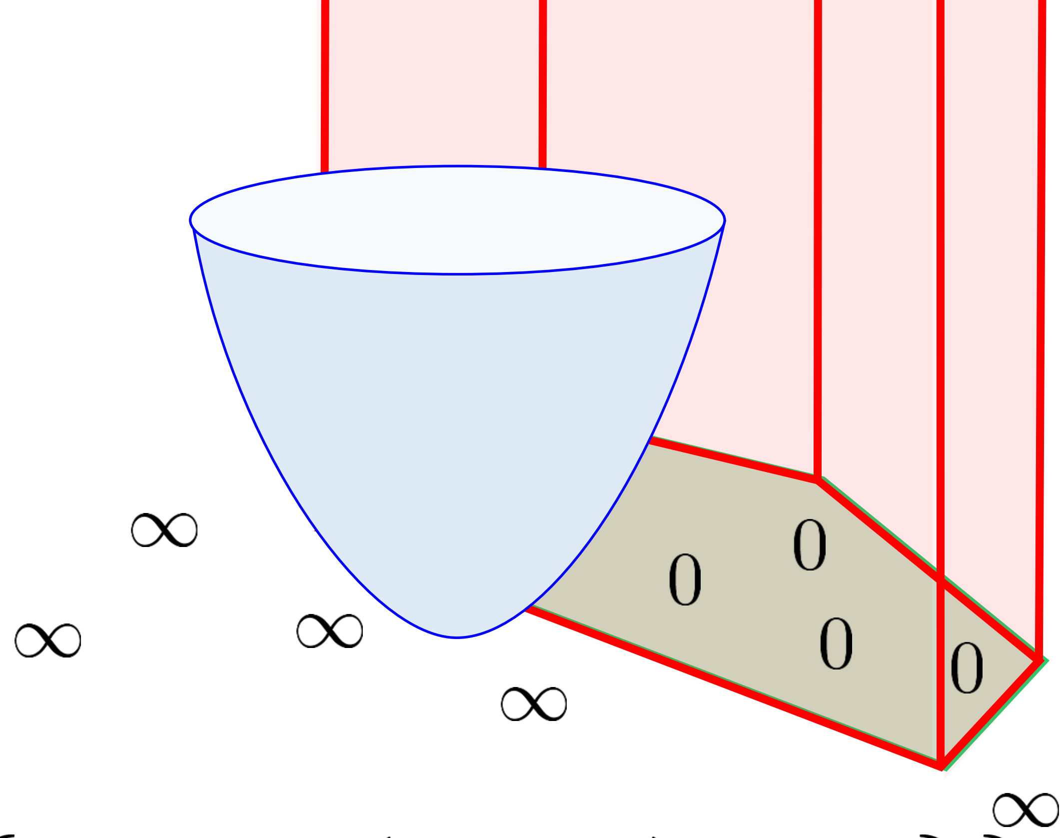
Lagrangian

The Lagrangian

$$\hat{\mathbf{w}}_P = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

Primal problem



$$\hat{\mathbf{w}}_P = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \{\arg\max_{\alpha\geq 0} \{ \quad \mathcal{L}(\mathbf{w},\alpha) \quad \}\}$$

Lagrange multiplier

Primal problem

$$\mathcal{L}(\mathbf{w},\alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

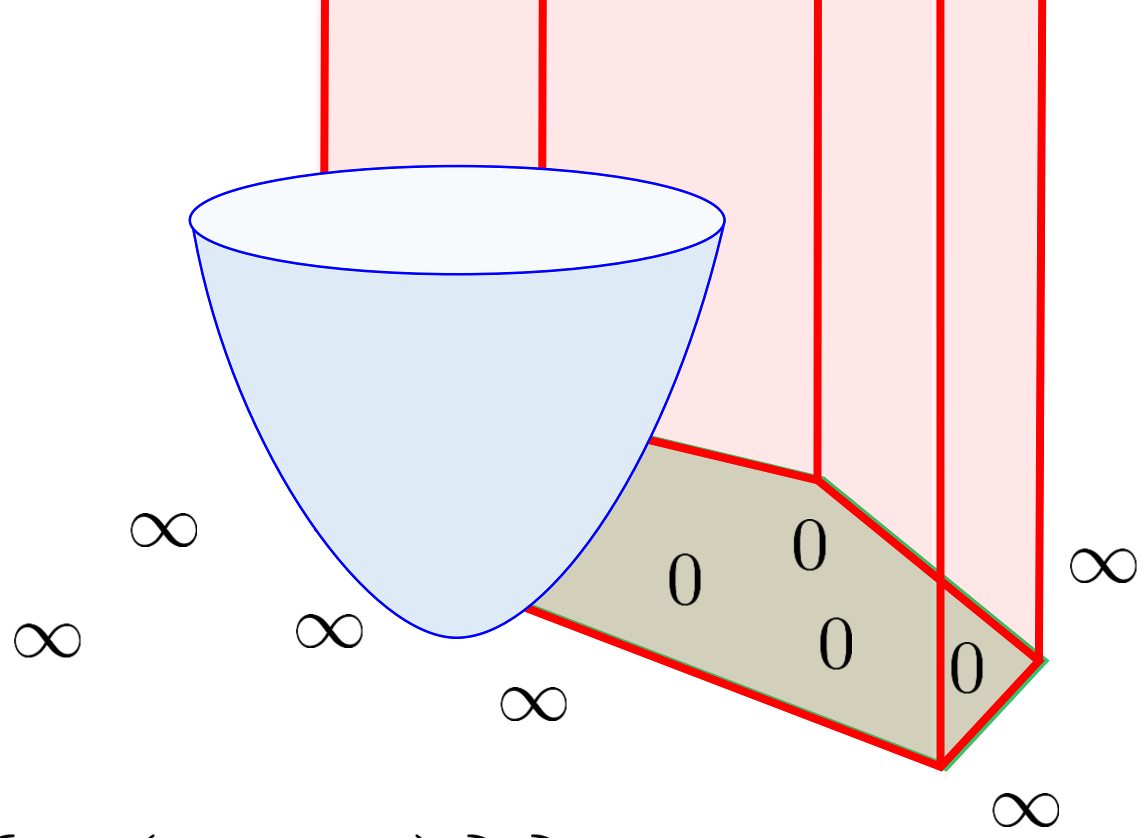
Lagrangian

The Lagrangian

$$\hat{\mathbf{w}}_P = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

Primal problem



$$\hat{\mathbf{w}}_P = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\arg\max_{\alpha\geq 0} \{\mathcal{L}(\mathbf{w},\alpha)\}\right\}$$

Lagrange multiplier

Primal problem

$$\mathcal{L}(\mathbf{w},\alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

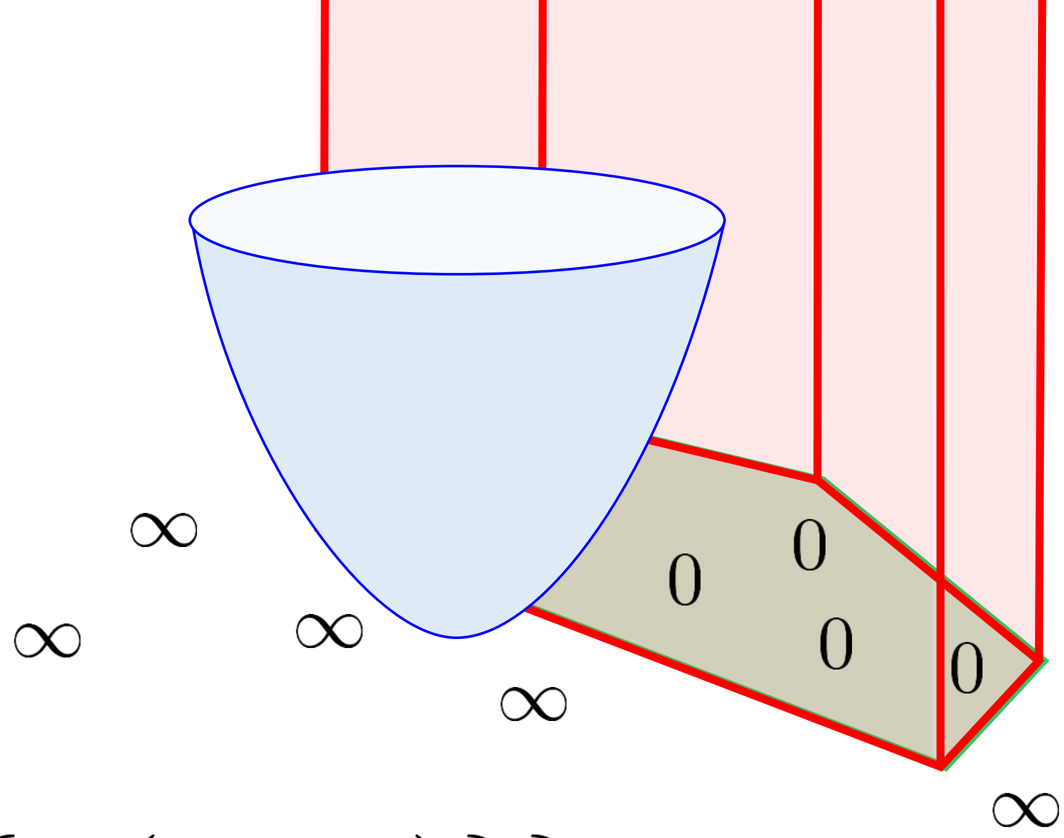
Lagrangian

The Lagrangian

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

Primal problem



$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\arg\max_{\alpha\geq 0} \left\{\mathcal{L}(\mathbf{w},\alpha)\right\}\right\}$$

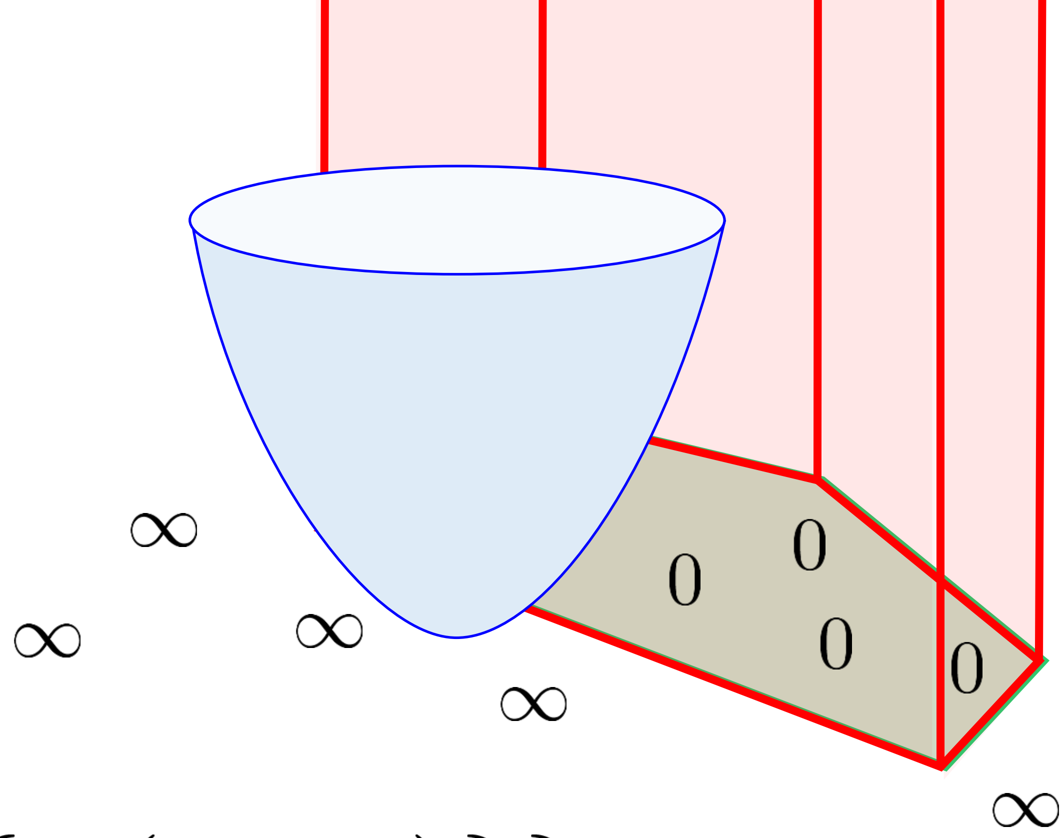
Primal problem

The Lagrangian

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

Primal problem



$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\arg\max_{\alpha\geq 0} \left\{\mathcal{L}(\mathbf{w},\alpha)\right\}\right\}$$

Primal problem

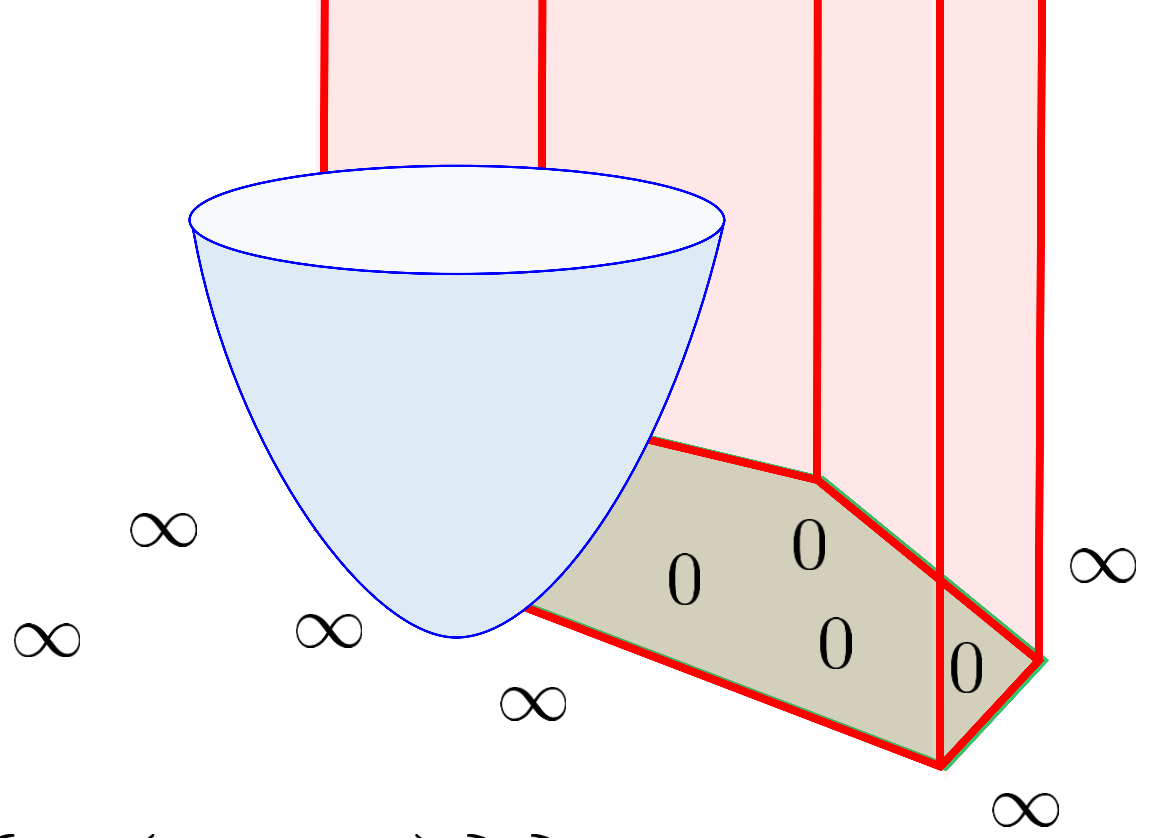
$$\hat{\mathbf{w}}_{\mathrm{D}} = \arg\max_{\alpha\geq 0} \left\{\arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\mathcal{L}(\mathbf{w},\alpha)\right\}\right\}$$

The Lagrangian

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \; f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

Primal problem



$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{ \arg\max_{\alpha\geq 0} \left\{ \mathcal{L}(\mathbf{w},\alpha) \right\} \right\}$$

Primal problem

$$\hat{\mathbf{w}}_{\mathrm{D}} = \arg\max_{\alpha\geq 0} \left\{ \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{ \mathcal{L}(\mathbf{w},\alpha) \right\} \right\}$$

Dual problem

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_{\mathrm{P}} = \underset{\mathbf{w}\in\mathbb{R}^d}{\arg\min}\; f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_P = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

$$\hat{\mathbf{w}}_D = \arg\max_{\alpha \geq 0} \left\{ \arg\min_{\mathbf{w} \in \mathbb{R}^d} \left\{ \mathcal{L}(\mathbf{w}, \alpha) \right\} \right\}$$

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\hat{\mathbf{w}}_{\mathrm{P}} \overset{?}{=} \hat{\mathbf{w}}_{\mathrm{D}}$$

$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

$$\hat{\mathbf{w}}_{\mathrm{D}} = \arg\max_{\alpha \geq 0} \left\{ \arg\min_{\mathbf{w} \in \mathbb{R}^d} \left\{ \mathcal{L}(\mathbf{w}, \alpha) \right\} \right\}$$

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\hat{\mathbf{w}}_{\mathrm{P}} \overset{?}{=} \hat{\mathbf{w}}_{\mathrm{D}}$$

$\hat{\mathbf{w}}_p = \hat{\mathbf{w}}_d$ for nice problems*

$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

$$\hat{\mathbf{w}}_{\mathrm{D}} = \arg\max_{\alpha \geq 0} \left\{ \arg\min_{\mathbf{w} \in \mathbb{R}^d} \left\{ \mathcal{L}(\mathbf{w}, \alpha) \right\} \right\}$$

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_\mathrm{P} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\hat{\mathbf{w}}_\mathrm{P} \overset{?}{=} \hat{\mathbf{w}}_\mathrm{D}$$

$\hat{\mathbf{w}}_p = \hat{\mathbf{w}}_d$ for nice problems*

$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

E.g. if $f(\cdot)$ is convex and $\{\mathbf{w} : g(\mathbf{w}) \leq 0\}$ is convex

$$\hat{\mathbf{w}}_\mathrm{D} = \arg\max_{\alpha\geq 0} \{\arg\min_{\mathbf{w}\in\mathbb{R}^d} \{\mathcal{L}(\mathbf{w}, \alpha)\}\}$$

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\hat{\mathbf{w}}_{\mathrm{P}} \stackrel{?}{=} \hat{\mathbf{w}}_{\mathrm{D}}$$

$\hat{\mathbf{w}}_p = \hat{\mathbf{w}}_d$ for nice problems*

E.g. if $f(\cdot)$ is convex and $\{\mathbf{w} : g(\mathbf{w}) \leq 0\}$ is convex

$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

$$\hat{\mathbf{w}}_{\mathrm{D}} = \arg\max_{\alpha\geq 0} \left\{\arg\min_{\mathbf{w}\in\mathbb{R}^d} \{\mathcal{L}(\mathbf{w}, \alpha)\}\right\}$$

$$\alpha_{\mathrm{D}} \cdot g(\hat{\mathbf{w}}_{\mathrm{D}}) = 0$$

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

$$\hat{\mathbf{w}}_{\mathrm{D}} = \arg\max_{\alpha\geq 0} \left\{\arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\mathcal{L}(\mathbf{w}, \alpha)\right\}\right\}$$

$$\alpha_{\mathrm{D}} \cdot g(\hat{\mathbf{w}}_{\mathrm{D}}) = 0$$

$$\hat{\mathbf{w}}_{\mathrm{P}} \stackrel{?}{=} \hat{\mathbf{w}}_{\mathrm{D}}$$

$\hat{\mathbf{w}}_p = \hat{\mathbf{w}}_d$ for nice problems*

E.g. if $f(\cdot)$ is convex and $\{\mathbf{w} : g(\mathbf{w}) \leq 0\}$ is convex

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\hat{\mathbf{w}}_{\mathrm{P}} \overset{?}{=} \hat{\mathbf{w}}_{\mathrm{D}}$$

$\hat{\mathbf{w}}_p = \hat{\mathbf{w}}_d$ for nice problems*

E.g. if $f(\cdot)$ is convex and $\{\mathbf{w} : g(\mathbf{w}) \leq 0\}$ is convex

$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

$$\hat{\mathbf{w}}_{\mathrm{D}} = \arg\max_{\alpha\geq 0} \left\{\arg\min_{\mathbf{w}\in\mathbb{R}^d} \left\{\mathcal{L}(\mathbf{w}, \alpha)\right\}\right\}$$

The maximizer

$$\alpha_{\mathrm{D}} \cdot g(\hat{\mathbf{w}}_{\mathrm{D}}) = 0$$

The Lagrange Dual Problem

$$\hat{\mathbf{w}}_{\mathrm{P}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w})$$

$$\text{s.t. } g(\mathbf{w}) \leq 0$$

$$\hat{\mathbf{w}}_{\mathrm{P}} \stackrel{?}{=} \hat{\mathbf{w}}_{\mathrm{D}}$$

$\hat{\mathbf{w}}_p = \hat{\mathbf{w}}_d$ for nice problems*

E.g. if $f(\cdot)$ is convex and $\{\mathbf{w} : g(\mathbf{w}) \leq 0\}$ is convex

$$\mathcal{L}(\mathbf{w}, \alpha) = f(\mathbf{w}) + \alpha \cdot g(\mathbf{w})$$

$$\hat{\mathbf{w}}_{\mathrm{D}} = \arg\max_{\alpha \geq 0} \left\{ \arg\min_{\mathbf{w} \in \mathbb{R}^d} \left\{ \mathcal{L}(\mathbf{w}, \alpha) \right\} \right\}$$

The maximizer

$$\alpha_{\mathrm{D}} \cdot g(\hat{\mathbf{w}}_{\mathrm{D}}) = 0$$

Complimentary slackness KKT Condition

Please give your Feedback

http://tinyurl.com/ml17-18afb