

Function Approximation Methods-II

CS771: Introduction to Machine Learning
Purushottam Kar



Recap

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

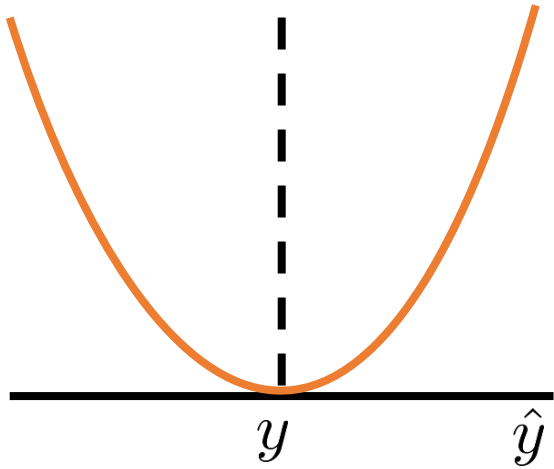
$$\text{s.t. } \mathbf{w}_i \geq 0$$

Constraint

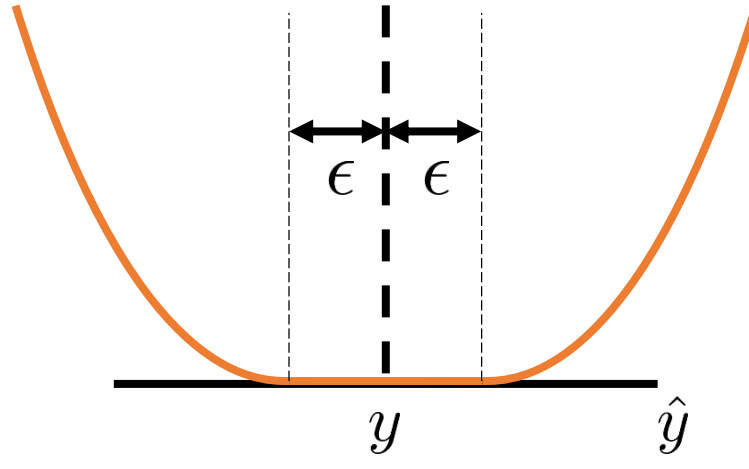
Regularizer

Loss Function

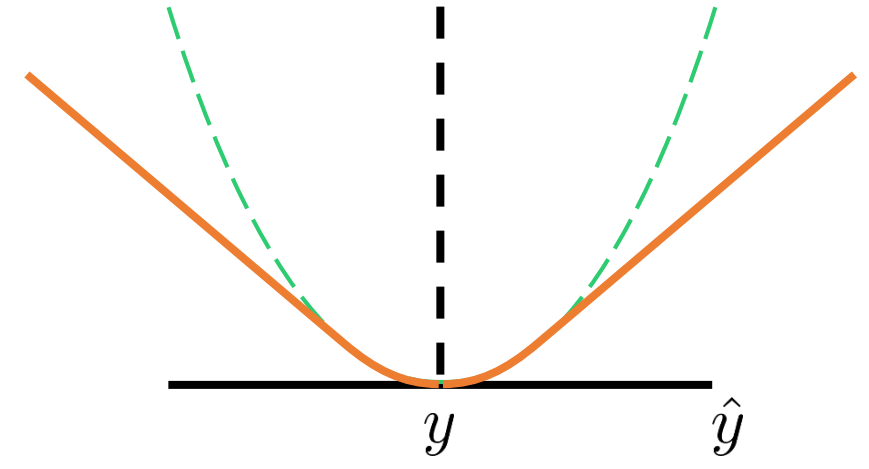
Recap



Square loss



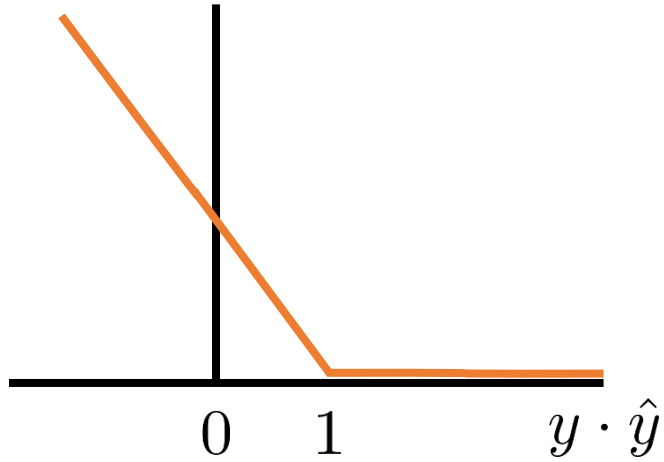
Vapnik's ϵ -insensitive loss



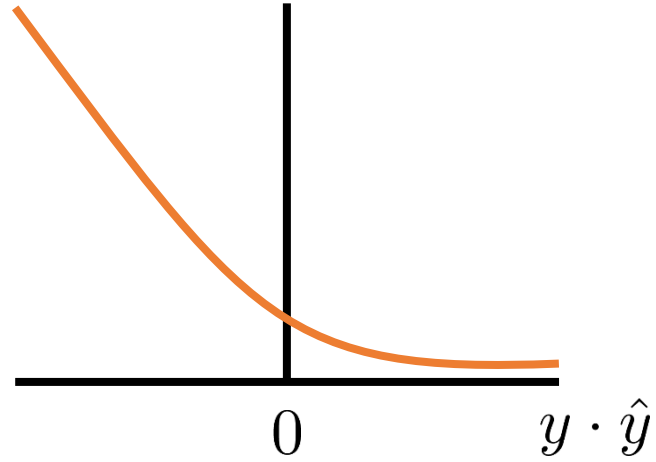
Huber loss

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

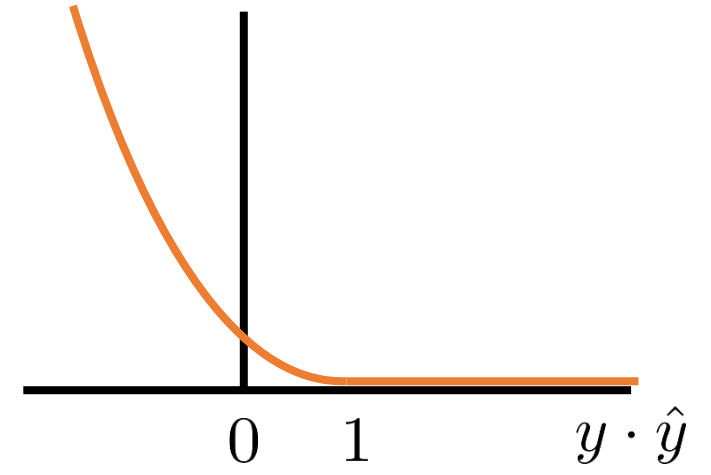
Recap



Hinge loss



Logistic loss



Squared
Hinge loss

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

Recap: Looking into the Hinge Loss

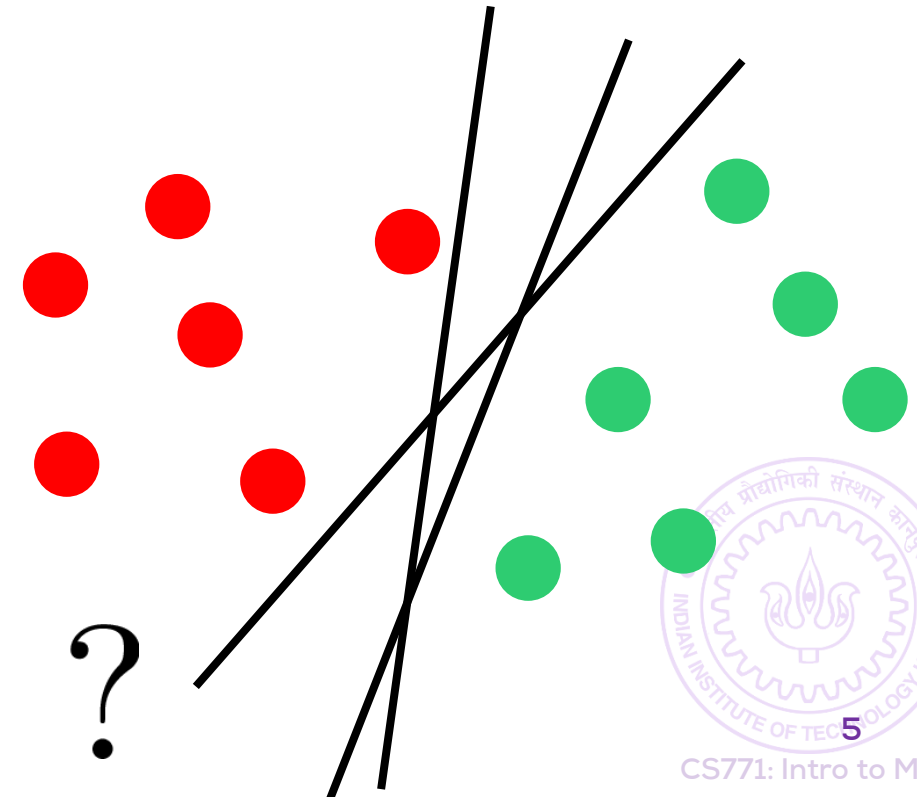
Binary Classification

$$\hat{y}^i = \text{sign}(\langle \mathbf{w}, \mathbf{x}^i \rangle)$$

Can use non-linear functions too!

Want sign of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be correct

Want magnitude of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be large too!



Recap: Looking into the Hinge Loss

Binary Classification

$$\hat{y}^i = \text{sign}(\langle \mathbf{w}, \mathbf{x}^i \rangle)$$

Want sign of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be correct

Want magnitude of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be large too!

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} \frac{\lambda}{2} \|\mathbf{w}\|_2^2$$

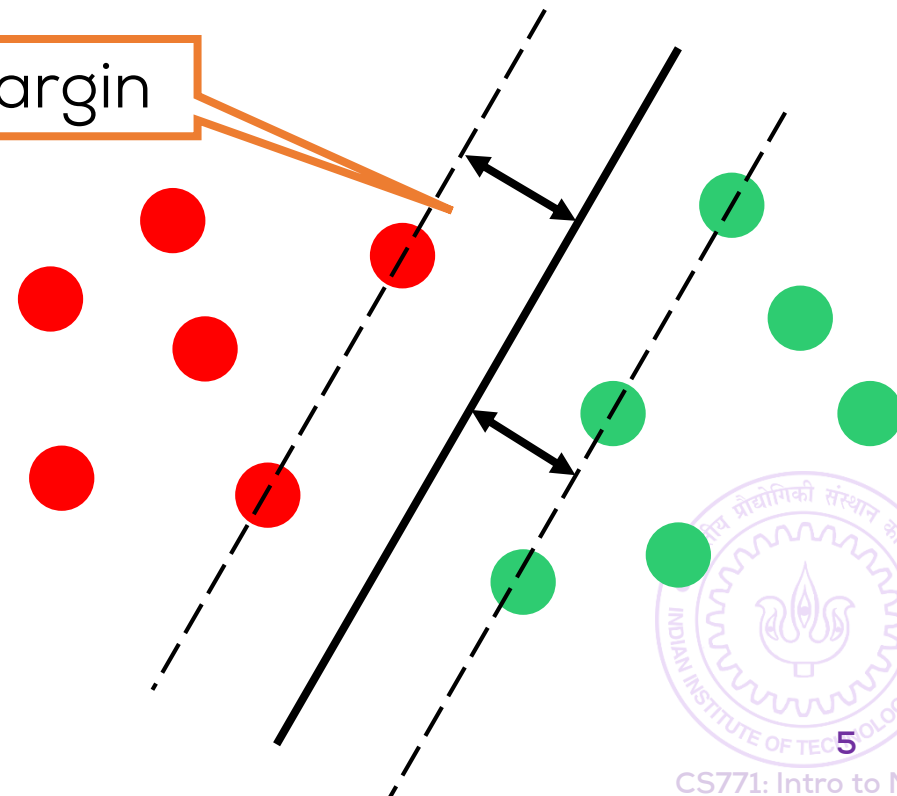
s.t. $y^i \langle \mathbf{w}, \mathbf{x}^i \rangle \geq 1$

Regularization parameter

Margin

Why 1?

Margin



Recap: Looking into the Hinge Loss

Binary Classification

$$\hat{y}^i = \text{sign}(\langle \mathbf{w}, \mathbf{x}^i \rangle)$$

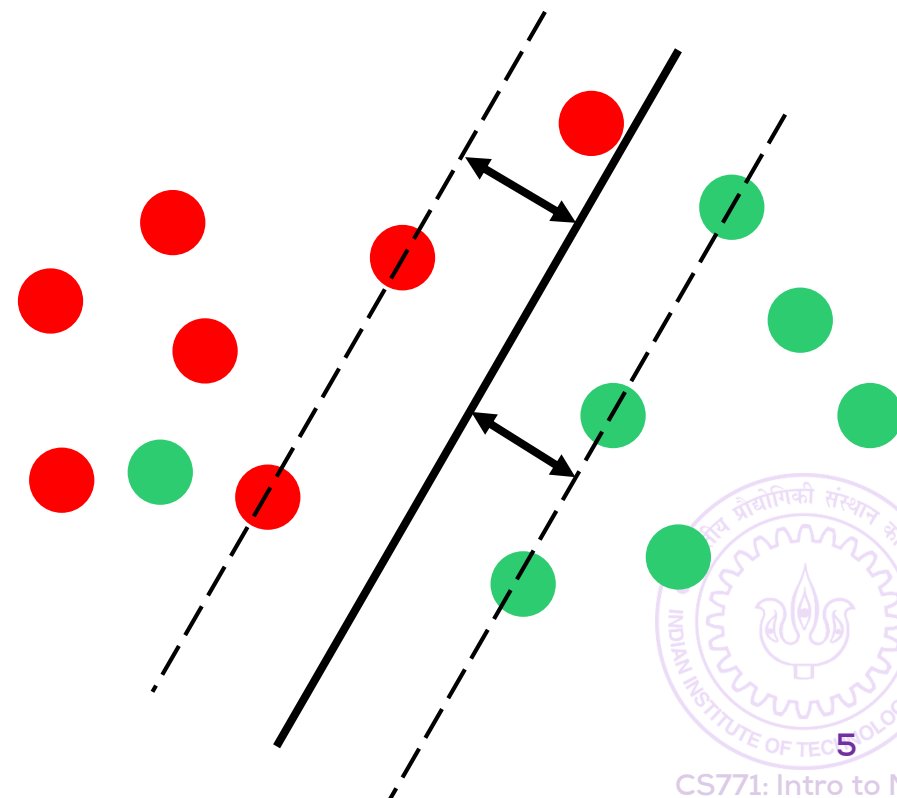
Want sign of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be correct

Slack variable

Want magnitude of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be large too!

$$\begin{aligned} \hat{\mathbf{w}} = \arg \min_{\mathbf{w}, \{\xi_i\}} & \frac{\lambda}{2} \|\mathbf{w}\|_2^2 + \sum_{i=1}^n \xi_i \\ \text{s.t. } & y^i \langle \mathbf{w}, \mathbf{x}^i \rangle \geq 1 - \xi_i \\ & \xi_i \geq 0 \end{aligned}$$

More params to optimize



Recap: Looking into the Hinge Loss

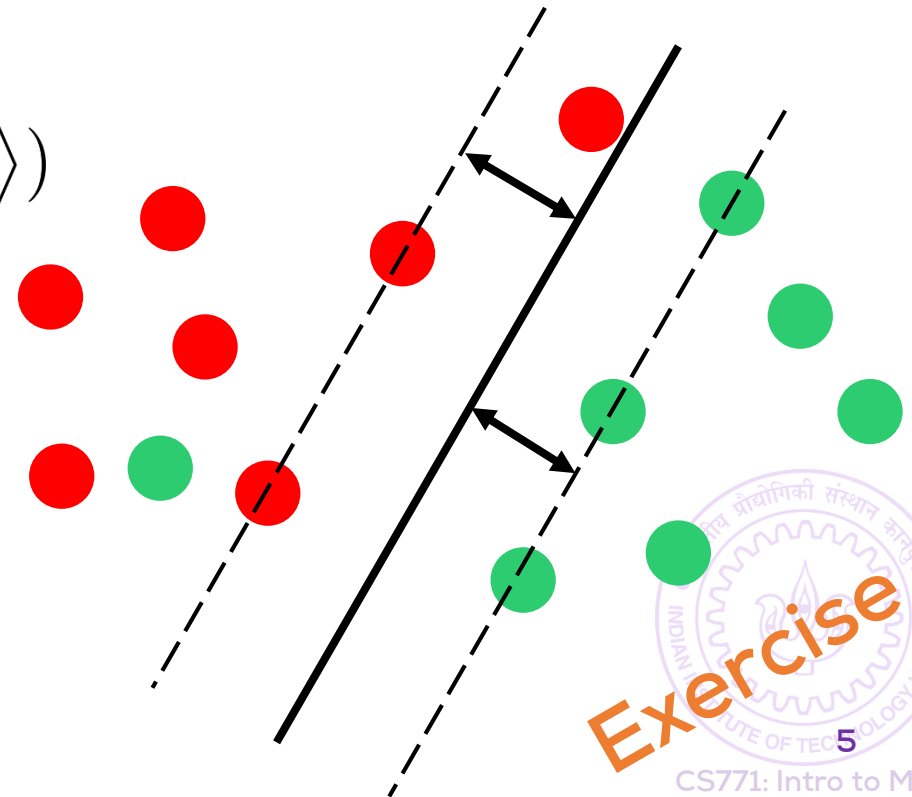
Binary Classification

$$\hat{y}^i = \text{sign}(\langle \mathbf{w}, \mathbf{x}^i \rangle)$$

Want sign of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be correct

Want magnitude of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be large too!

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} \frac{\lambda}{2} \|\mathbf{w}\|_2^2 + \sum_{i=1}^n \ell_{\text{hinge}}(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$



Recap: Looking into the Hinge Loss

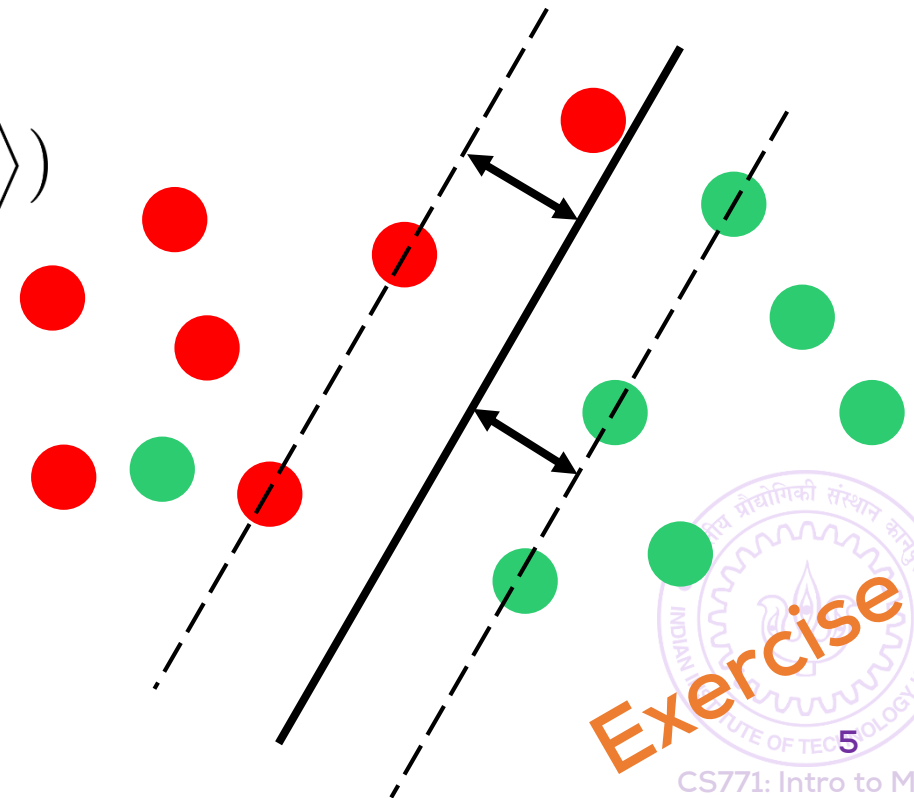
Binary Classification

$$\hat{y}^i = \text{sign}(\langle \mathbf{w}, \mathbf{x}^i \rangle)$$

Want sign of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be correct

Want magnitude of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be large too!

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} \quad \|\mathbf{w}\|_2^2 + C \cdot \sum_{i=1}^n \ell_{\text{hinge}}(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$



Recap: Looking into the Hinge Loss

Binary Classification

$$\hat{y}^i = \text{sign}(\langle \mathbf{w}, \mathbf{x}^i \rangle)$$

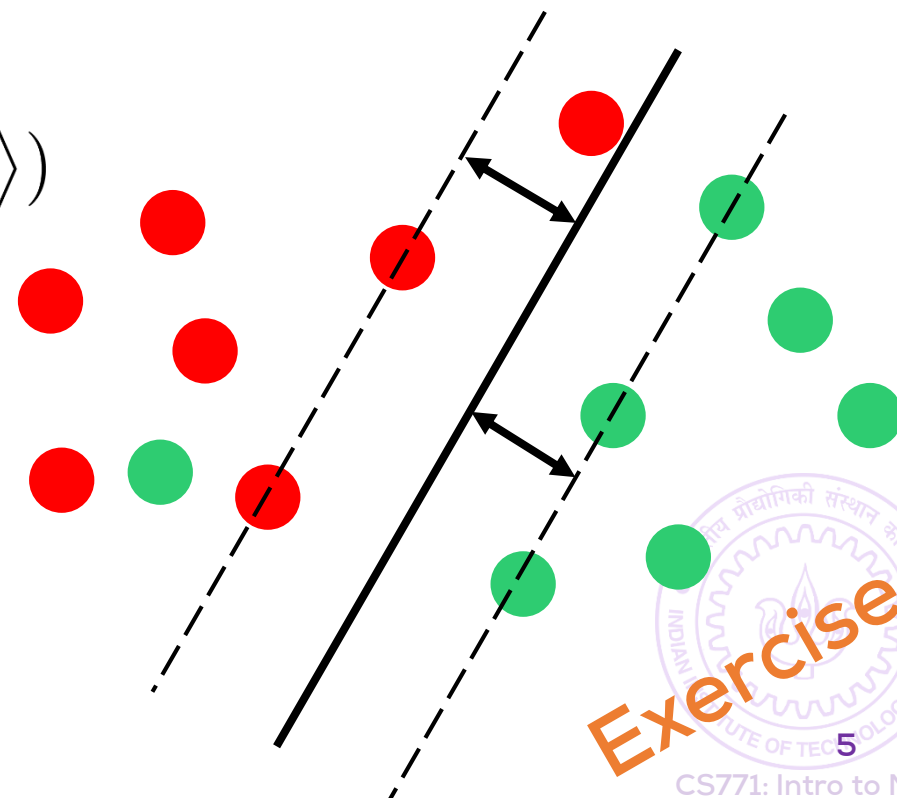
Want sign of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be correct

Want magnitude of $\langle \mathbf{w}, \mathbf{x}^i \rangle$ to be large too!

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} \|\mathbf{w}\|_2^2 + C \cdot \sum_{i=1}^n \ell_{\text{hinge}}(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$

Large Margin Classifier

SVM



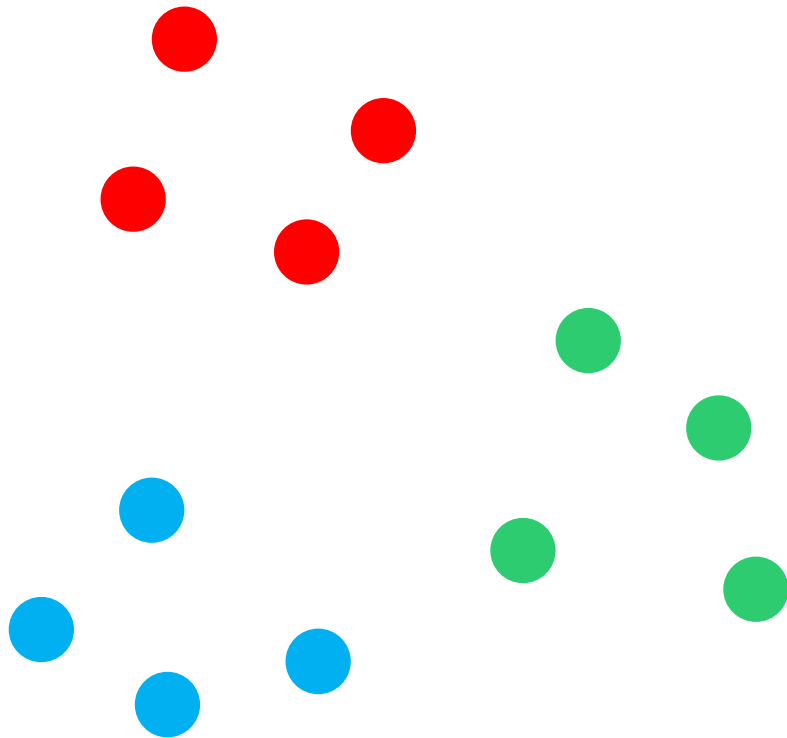
Loss Functions for Structured ML problems

Multi-class and Multi-label classification

Multi-classification Loss Functions

One-vs-All (OVA)

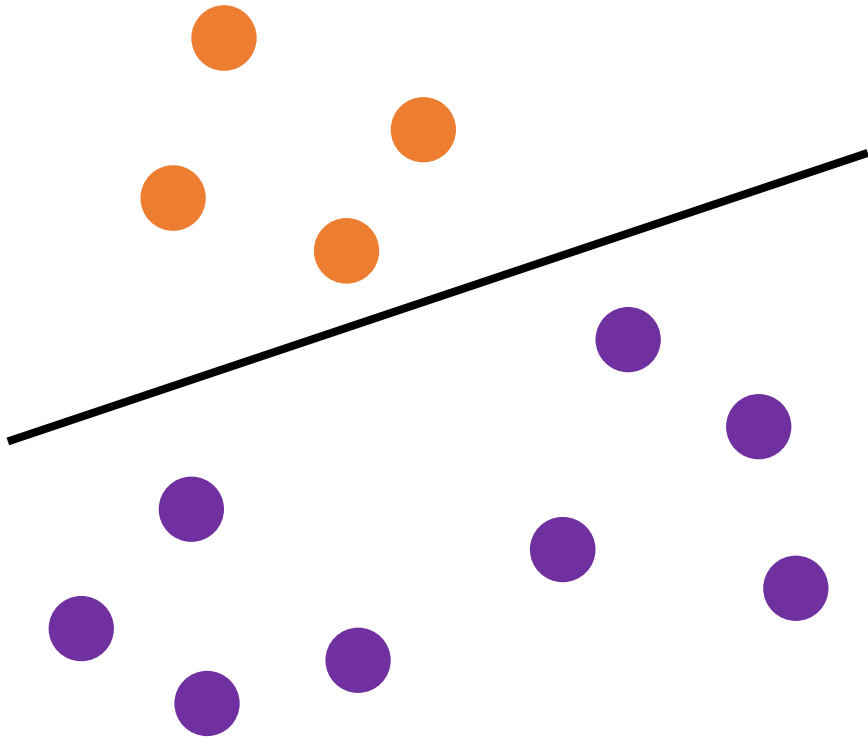
$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$



Multi-classification Loss Functions

One-vs-All (OVA)

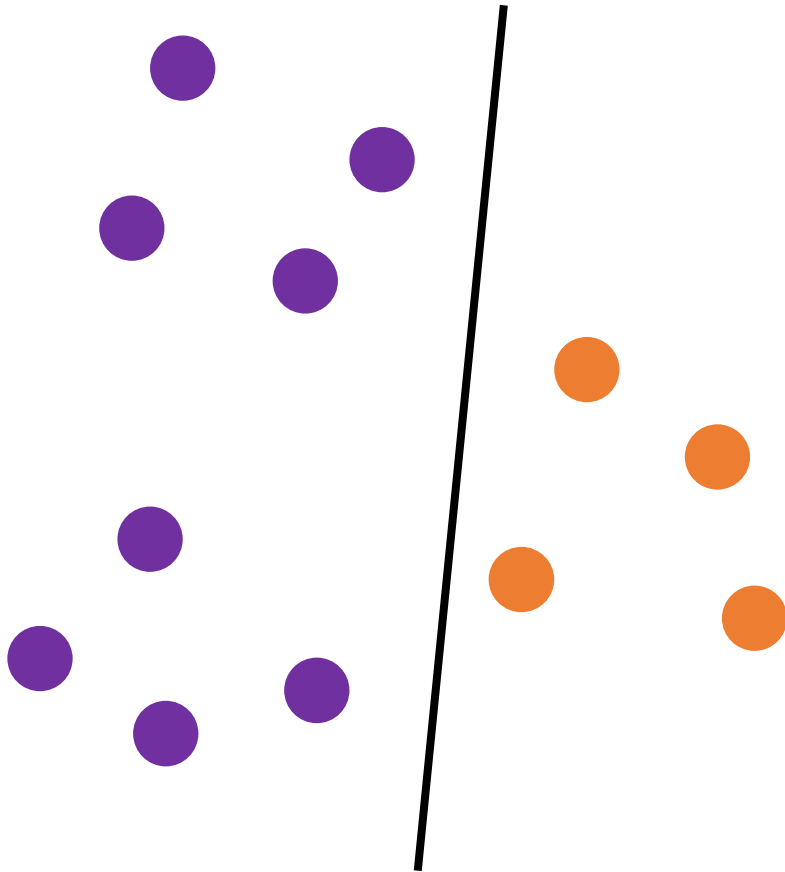
$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$



Multi-classification Loss Functions

One-vs-All (OVA)

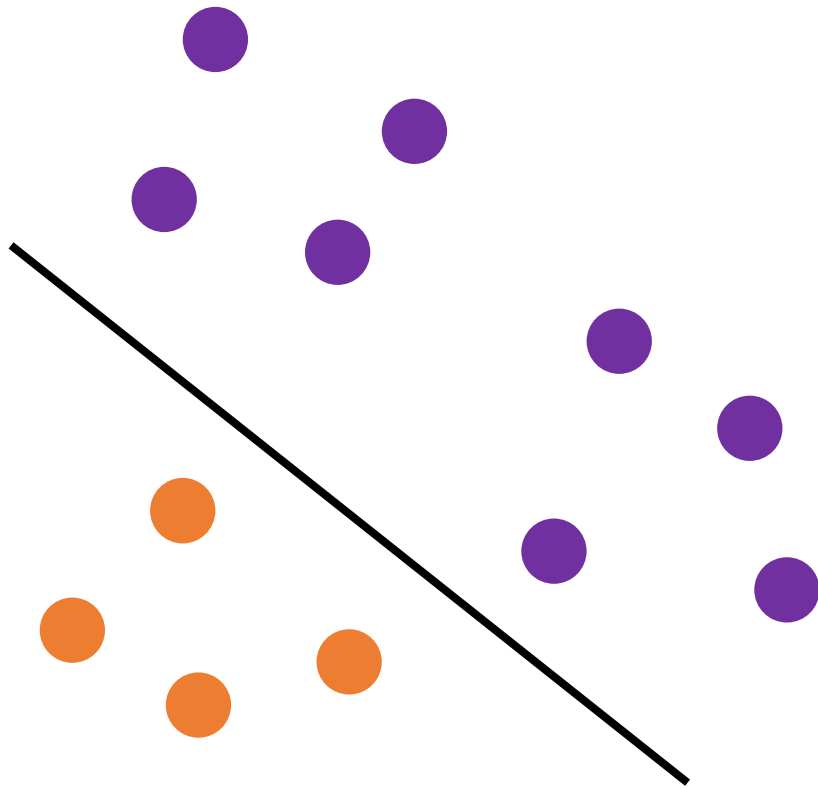
$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$



Multi-classification Loss Functions

One-vs-All (OVA)

$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$



Multi-classification Loss Functions

One-vs-All (OVA)

$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$

$$\hat{\mathbf{w}}^j = \arg \min_{\mathbf{w}} \sum_{i=1}^n \ell(y^{i,(j)}, \langle \mathbf{w}, \mathbf{x}^i \rangle)$$
$$y^{i,(j)} = \begin{cases} 1 & ; y^i = j \\ -1 & ; y^i \neq j \end{cases}$$

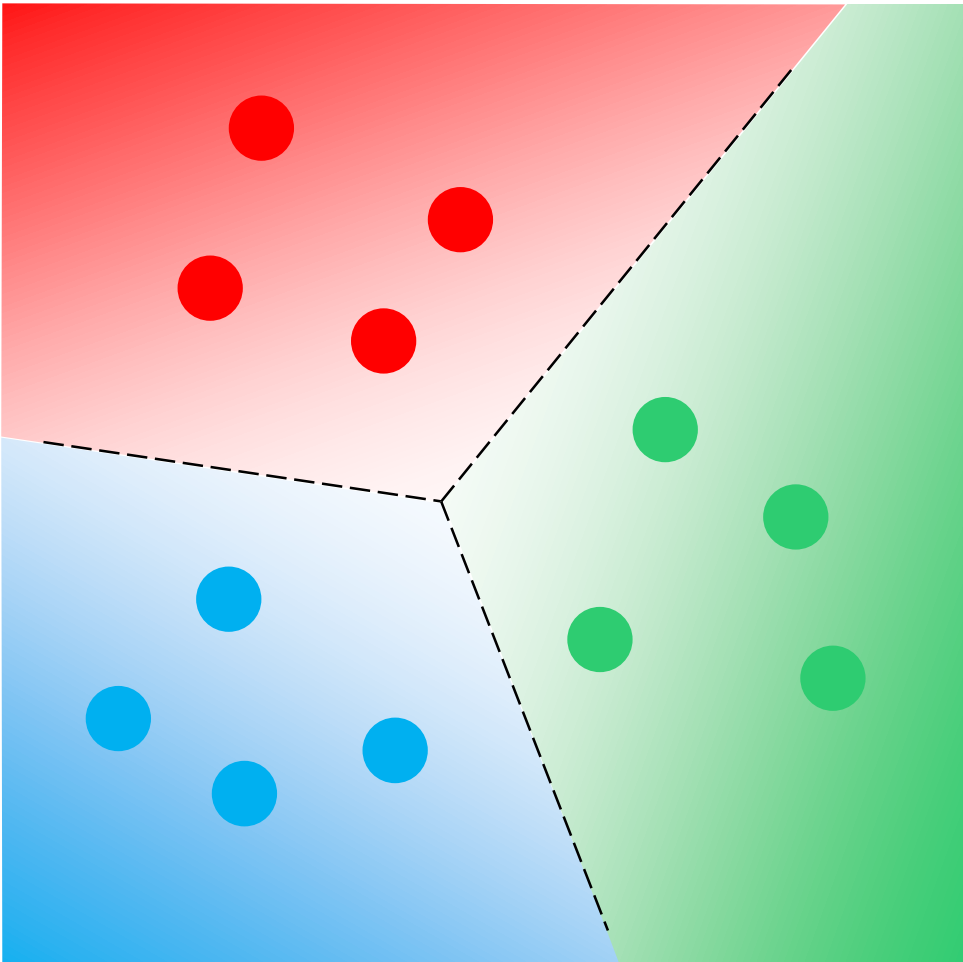
hinge, logistic etc



Multi-classification Loss Functions

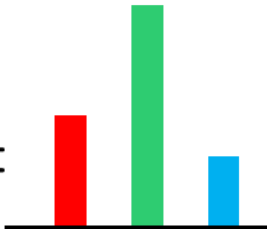
One-vs-All (OVA)

$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$



Multi-classification using MLE

- $K > 2$ classes – need more detailed parameters
- For each point, its label profile is a vector

$$\boldsymbol{\eta}(\mathbf{x}) =$$


$$\mathbb{P} [y^i = k \mid \mathbf{x}^i, \{\mathbf{w}^l\}_{1,\dots,K}] \propto \exp(\langle \mathbf{w}^k, \mathbf{x}^i \rangle)$$

$$\mathbb{P} [y^i = k \mid \mathbf{x}^i, \{\mathbf{w}^l\}_{1,\dots,K}] = \frac{\exp(\langle \mathbf{w}^k, \mathbf{x}^i \rangle)}{\sum_{l=1}^K \exp(\langle \mathbf{w}^l, \mathbf{x}^i \rangle)}$$

- Likelihood function is multinomial instead of binomial

$$\mathbb{P} [\mathbf{y} \mid \mathbf{X}, \mathbf{w}] = \prod_{i=1}^n \hat{\eta}_{y^i}^i(\mathbf{x}) \quad \hat{\eta}_k^i(\mathbf{x}) = \frac{\exp(\langle \mathbf{w}^k, \mathbf{x}^i \rangle)}{\sum_{l=1}^K \exp(\langle \mathbf{w}^l, \mathbf{x}^i \rangle)}$$

Softmax Regression

Multi-classification Loss Functions

One-vs-All (OVA)

$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$

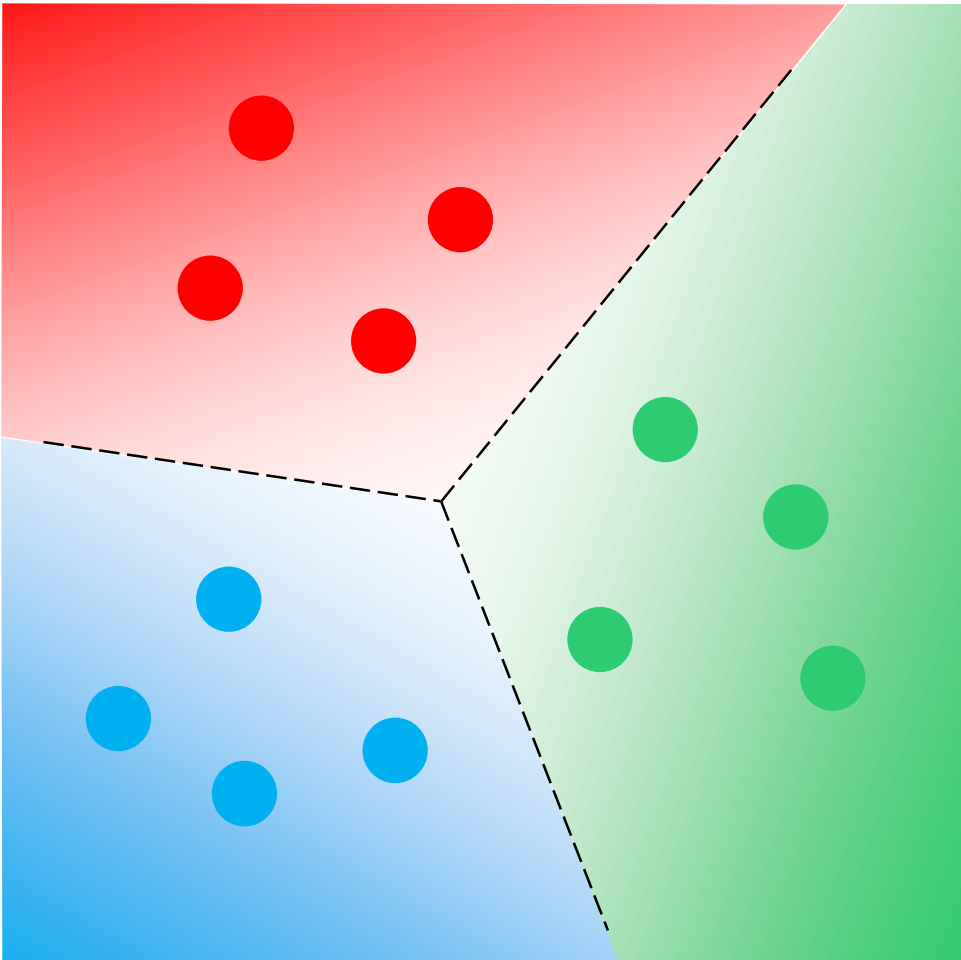
$$\mathbf{W} = [\mathbf{w}^1, \mathbf{w}^2, \dots, \mathbf{w}^K]$$

$$\hat{\mathbf{W}}_{\text{MLE}} = \arg \min_{\mathbf{W}} \sum_{i=1}^n \ell_{\text{sm}}(y^i, \langle \mathbf{W}, \mathbf{x}^i \rangle)$$

$$\langle \mathbf{W}, \mathbf{x} \rangle = \boldsymbol{\eta} = [\eta_1, \eta_2, \dots, \eta_K]$$

$$\ell_{\text{sm}}(y, \{\eta_j\}) = -\log \frac{\exp(\eta_y)}{\sum_{k=1}^K \exp(\eta_k)}$$

Softmax loss
function



Multi-classification Loss Functions

One-vs-All (OVA)

$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$

Slack variable

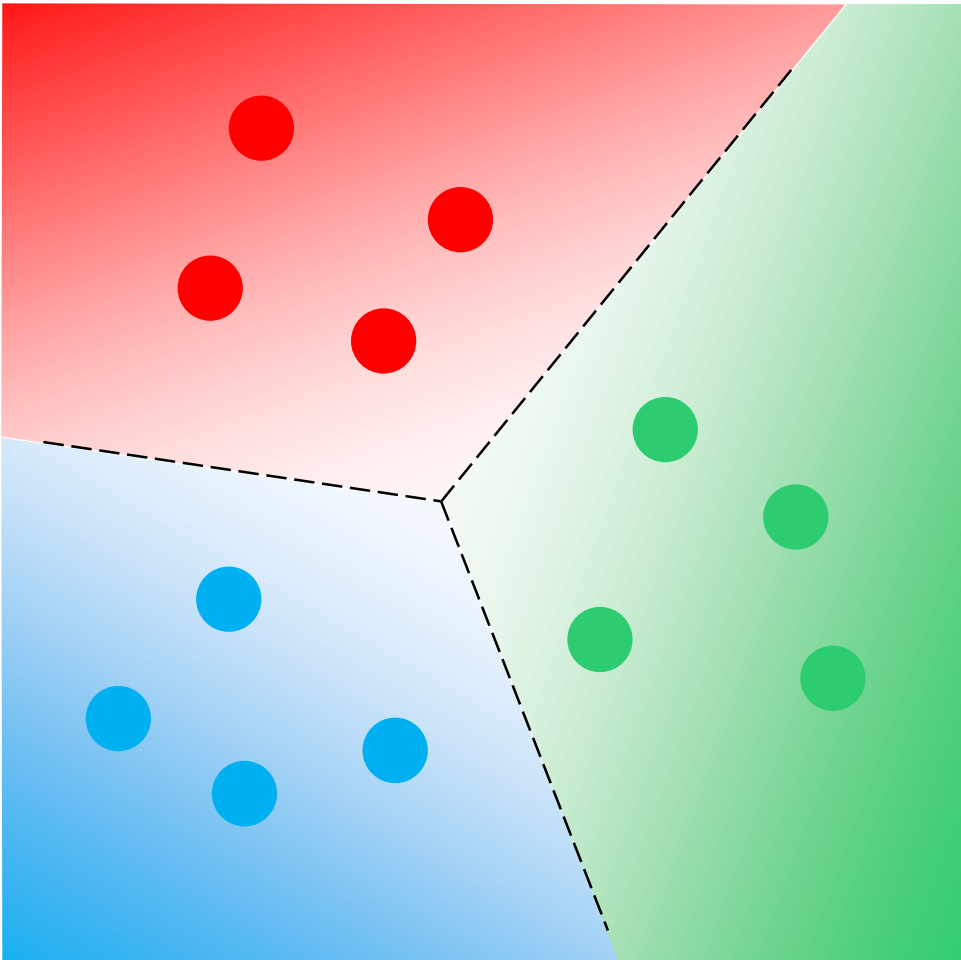
$$\mathbf{W} = [\mathbf{w}^1, \mathbf{w}^2, \dots, \mathbf{w}^K]$$

$$\hat{\mathbf{W}} = \arg \min_{\mathbf{W}, \{\xi_i\}} \sum_{k=1}^K \|\mathbf{w}^k\|_2^2 + \sum_{i=1}^n \xi_i$$

$$\text{s.t. } \langle \mathbf{w}^{y^i}, \mathbf{x}^i \rangle \geq \langle \mathbf{w}^k, \mathbf{x}^i \rangle + 1 - \xi_i$$

$$\xi_i \geq 0$$

$$\forall k \neq y^i$$



Multi-classification Loss Functions

One-vs-All (OVA)

$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$

$$\mathbf{W} = [\mathbf{w}^1, \mathbf{w}^2, \dots, \mathbf{w}^K]$$

$$\hat{\mathbf{W}} = \arg \min_{\mathbf{W}} \sum_{k=1}^K \|\mathbf{w}^k\|_2^2 + \sum_{i=1}^n \ell_{\text{cs}}(y^i, \langle \mathbf{W}, \mathbf{x}^i \rangle)$$

$$\langle \mathbf{W}, \mathbf{x} \rangle = \boldsymbol{\eta} = [\eta_1, \eta_2, \dots, \eta_K]$$

$$\ell_{\text{cs}}(y, \{\eta_j\}) = [1 + \max_{k \neq y} \eta_k - \eta_y]_+$$

Exercise

Crammer-Singer
loss function

Multi-classification Loss Functions

One-vs-All (OVA)

$$\hat{y}^i = \arg \max_{j \in [K]} \langle \mathbf{w}^j, \mathbf{x}^i \rangle$$

$$\mathbf{W} = [\mathbf{w}^1, \mathbf{w}^2, \dots, \mathbf{w}^K]$$

$$\mathbf{X} = [\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^n]$$

$$\|\mathbf{X}\|_* = \sum \sigma_i(\mathbf{X})$$

$$\|\mathbf{X}\|_{p,q} = \left\| \|\mathbf{x}^1\|_p, \|\mathbf{x}^2\|_p, \dots, \|\mathbf{x}^n\|_p \right\|_q$$

$$\arg \min_{\mathbf{W}} \sum_{k=1}^K \|\mathbf{w}^k\|_2^2 + \sum_{i=1}^n \ell_{\text{cs}}(y^i, \langle \mathbf{W}, \mathbf{x}^i \rangle)$$

$$\langle \mathbf{W}, \mathbf{x} \rangle = \boldsymbol{\eta} = [\eta_1, \eta_2, \dots, \eta_K]$$

$$\ell_{\text{cs}}(y, \{\eta_j\}) = \max_{k \neq y} \eta_k - \eta_y + 1$$

Exercise

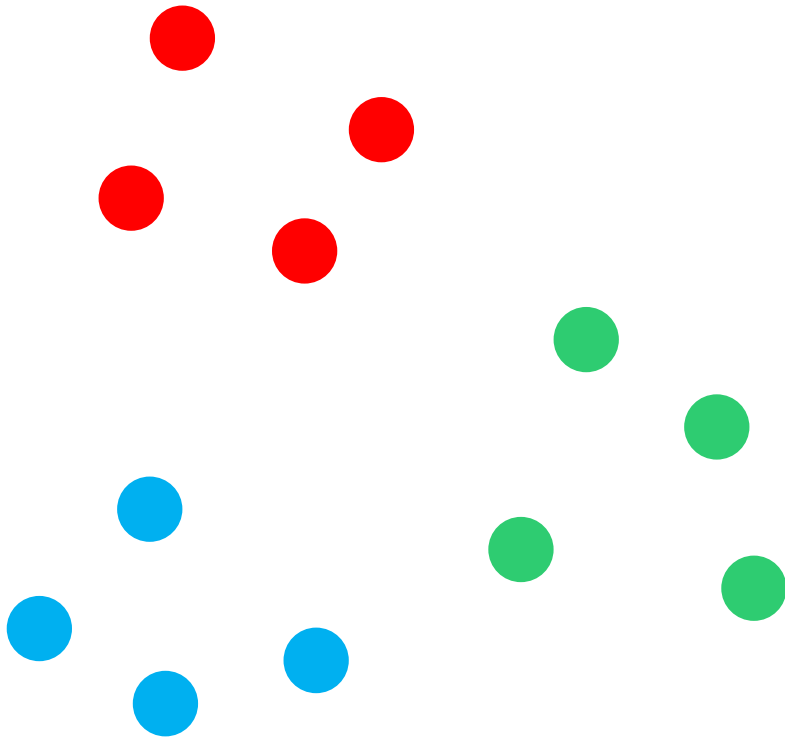
More powerful regularizers?

$$\|\mathbf{W}\|_F^2, \|\mathbf{W}\|_*, \|\mathbf{W}\|_0, \|\mathbf{W}^\top\|_{2,0}$$

Multi-classification Loss Functions

All-vs-All (AVA)

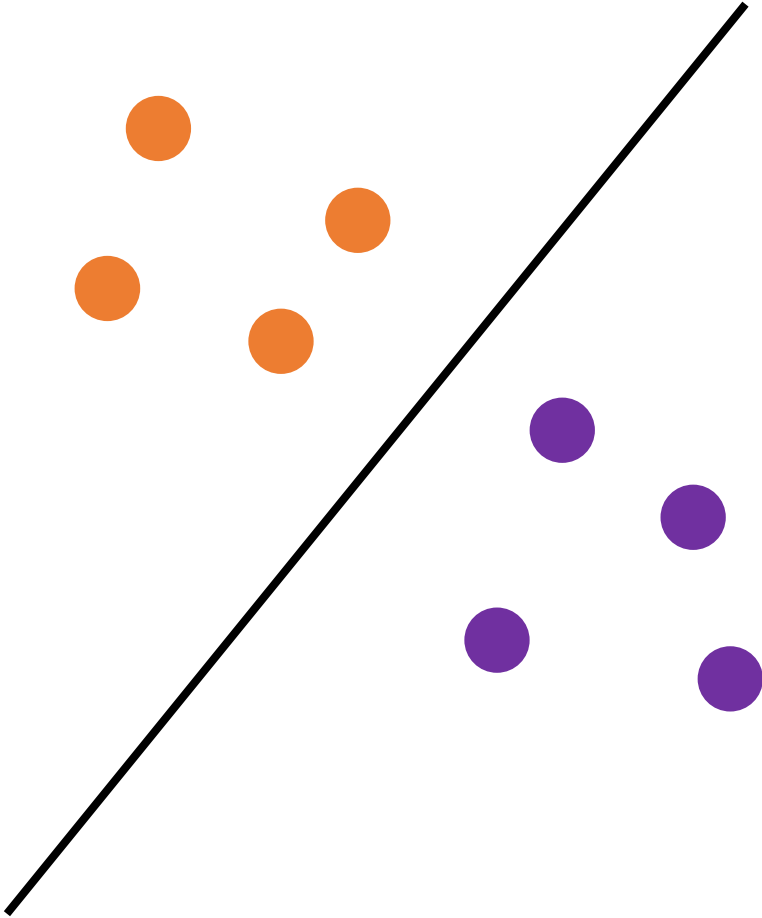
$$\hat{y}^i = \arg \max_{j \in [K]} \sum_{k=1}^K \langle \mathbf{w}^{j,k}, \mathbf{x}^i \rangle$$



Multi-classification Loss Functions

All-vs-All (AVA)

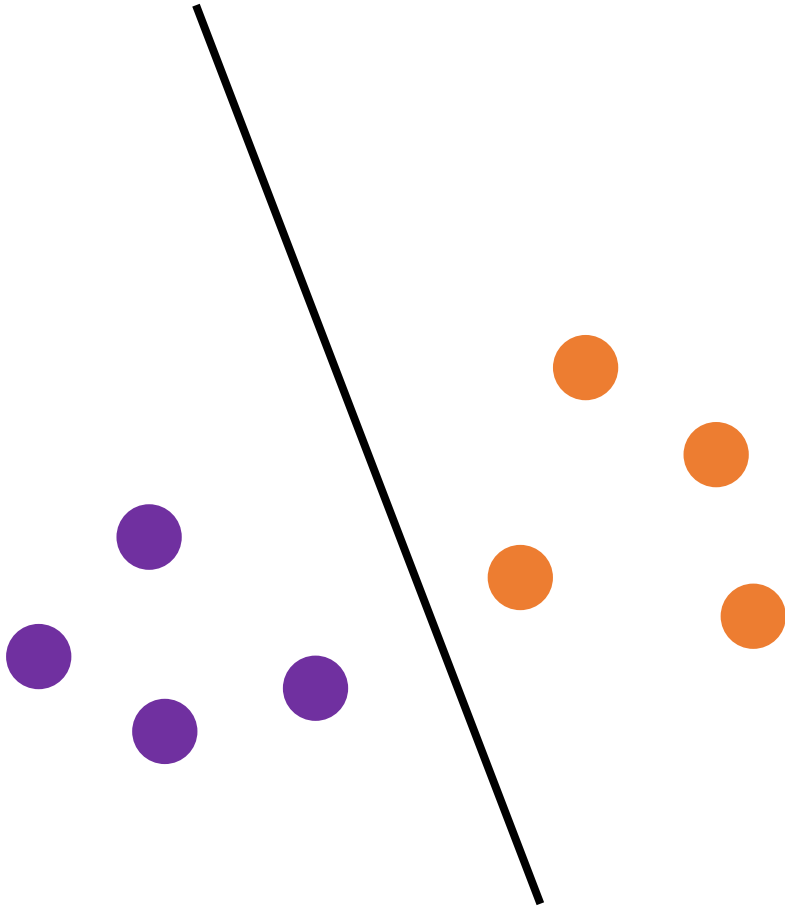
$$\hat{y}^i = \arg \max_{j \in [K]} \sum_{k=1}^K \langle \mathbf{w}^{j,k}, \mathbf{x}^i \rangle$$



Multi-classification Loss Functions

All-vs-All (AVA)

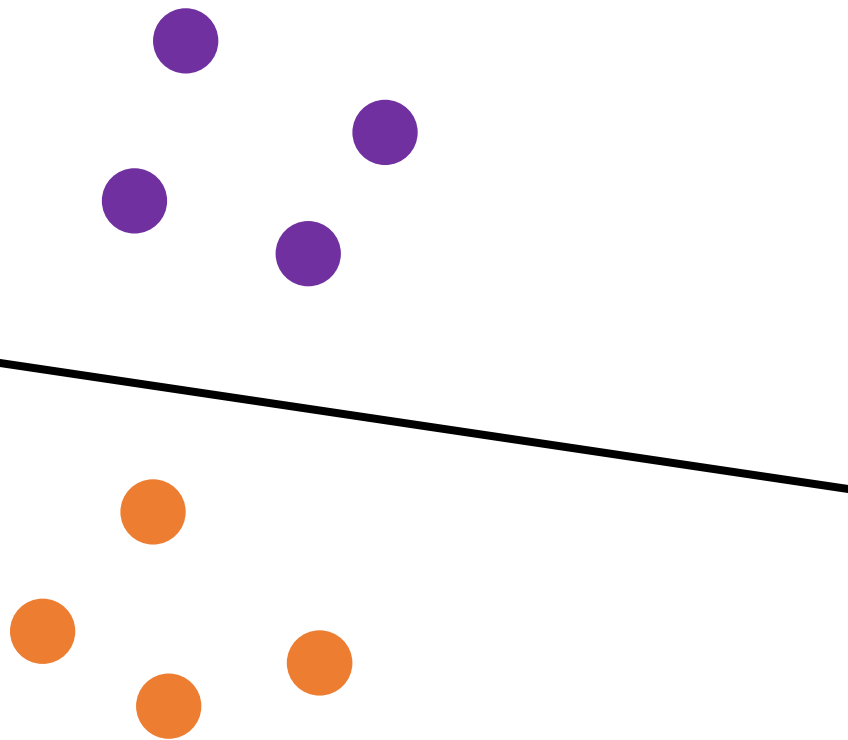
$$\hat{y}^i = \arg \max_{j \in [K]} \sum_{k=1}^K \langle \mathbf{w}^{j,k}, \mathbf{x}^i \rangle$$



Multi-classification Loss Functions

All-vs-All (AVA)

$$\hat{y}^i = \arg \max_{j \in [K]} \sum_{k=1}^K \langle \mathbf{w}^{j,k}, \mathbf{x}^i \rangle$$



Multi-classification Loss Functions

All-vs-All (AVA)

$$\hat{y}^i = \arg \max_{j \in [K]} \sum_{k=1}^K \langle \mathbf{w}^{j,k}, \mathbf{x}^i \rangle$$

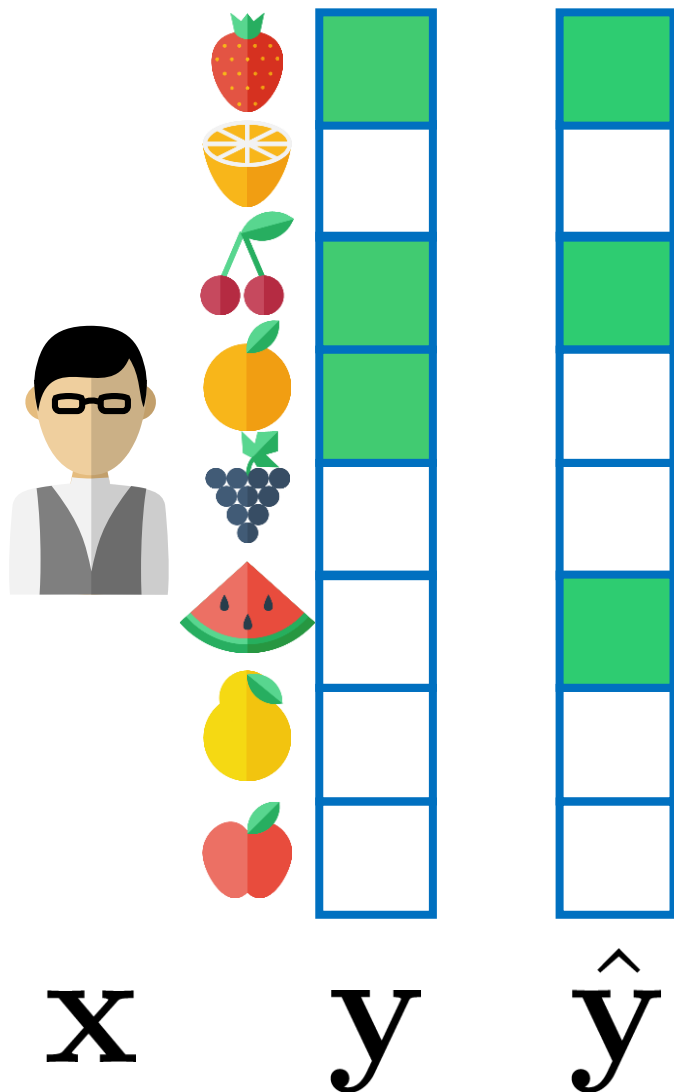
Expensive!

Other techniques: use
DT to eliminate classes,
error correcting codes

Exercise: Develop
training techniques for
AVA classifiers



Multi-label Classification Loss Functions



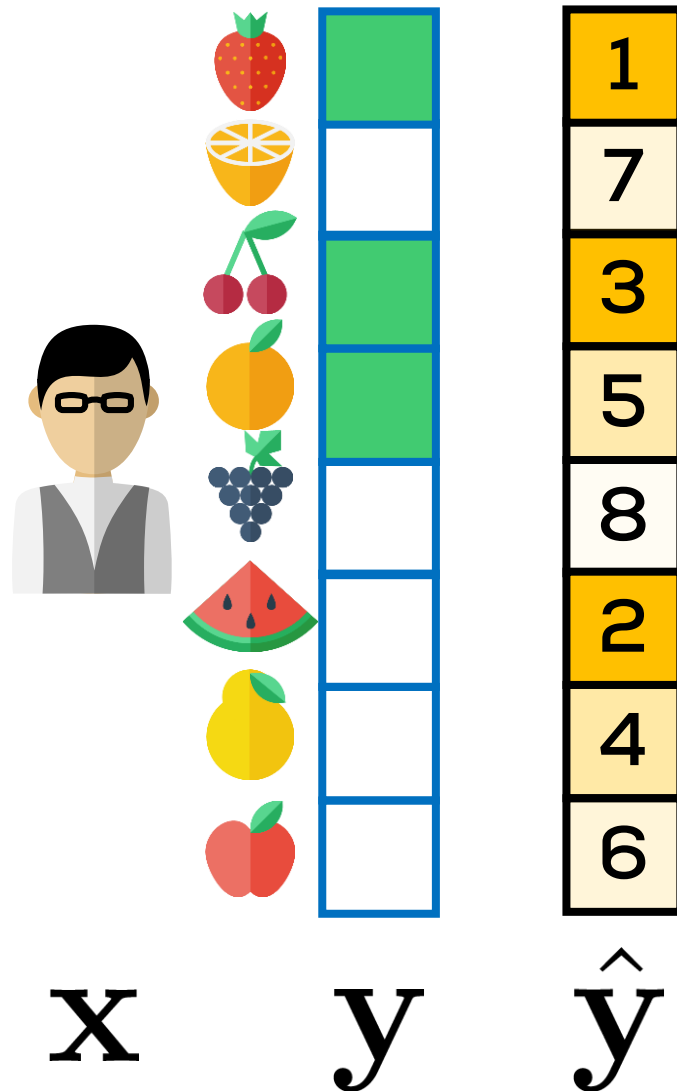
$\left \left\{ i : \begin{array}{l} \hat{y}_i = 1 \\ y_i = 1 \end{array} \right\} \right $	$\left \left\{ i : \begin{array}{l} \hat{y}_i = 1 \\ y_i = -1 \end{array} \right\} \right $
$\left \left\{ i : \begin{array}{l} \hat{y}_i = -1 \\ y_i = 1 \end{array} \right\} \right $	$\left \left\{ i : \begin{array}{l} \hat{y}_i = -1 \\ y_i = -1 \end{array} \right\} \right $

$$\ell_{\text{Hamming}}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{b + c}{a + b + c + d}$$

$$r_{\text{Precision}}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{a}{a + b} \quad r_{\text{Recall}}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{a}{a + c}$$

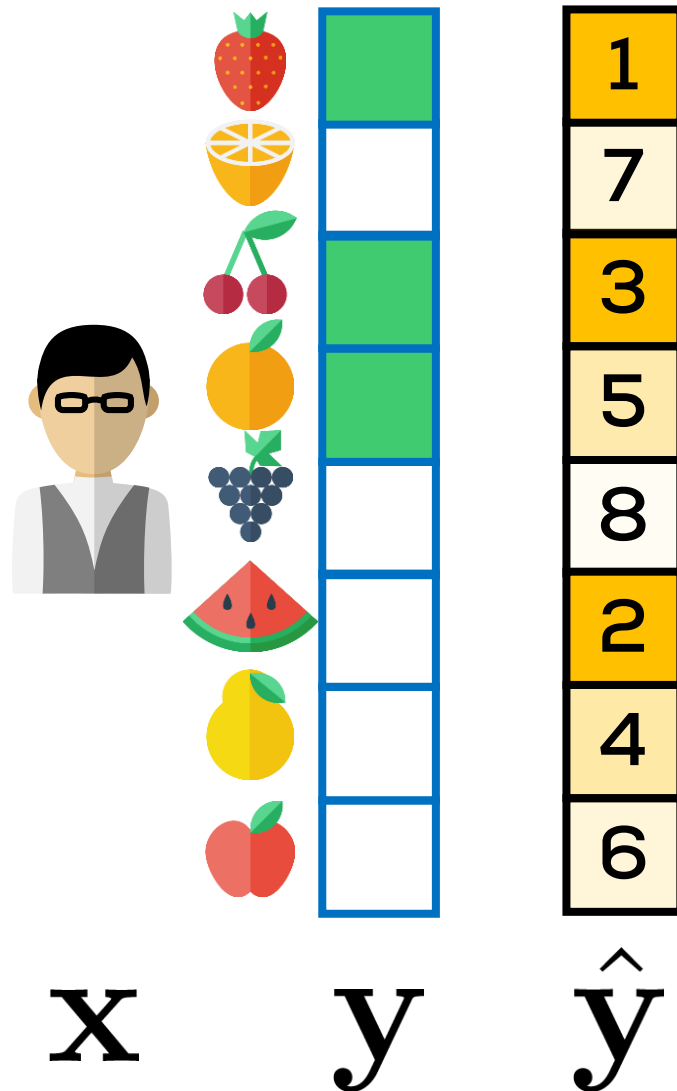
$$F(\mathbf{y}, \hat{\mathbf{y}}) = \frac{2r_{\text{Prec}} \cdot r_{\text{Rec}}}{r_{\text{Prec}} + r_{\text{Rec}}} = \frac{2a}{2a + b + c}$$

Multi-label Classification Loss Functions



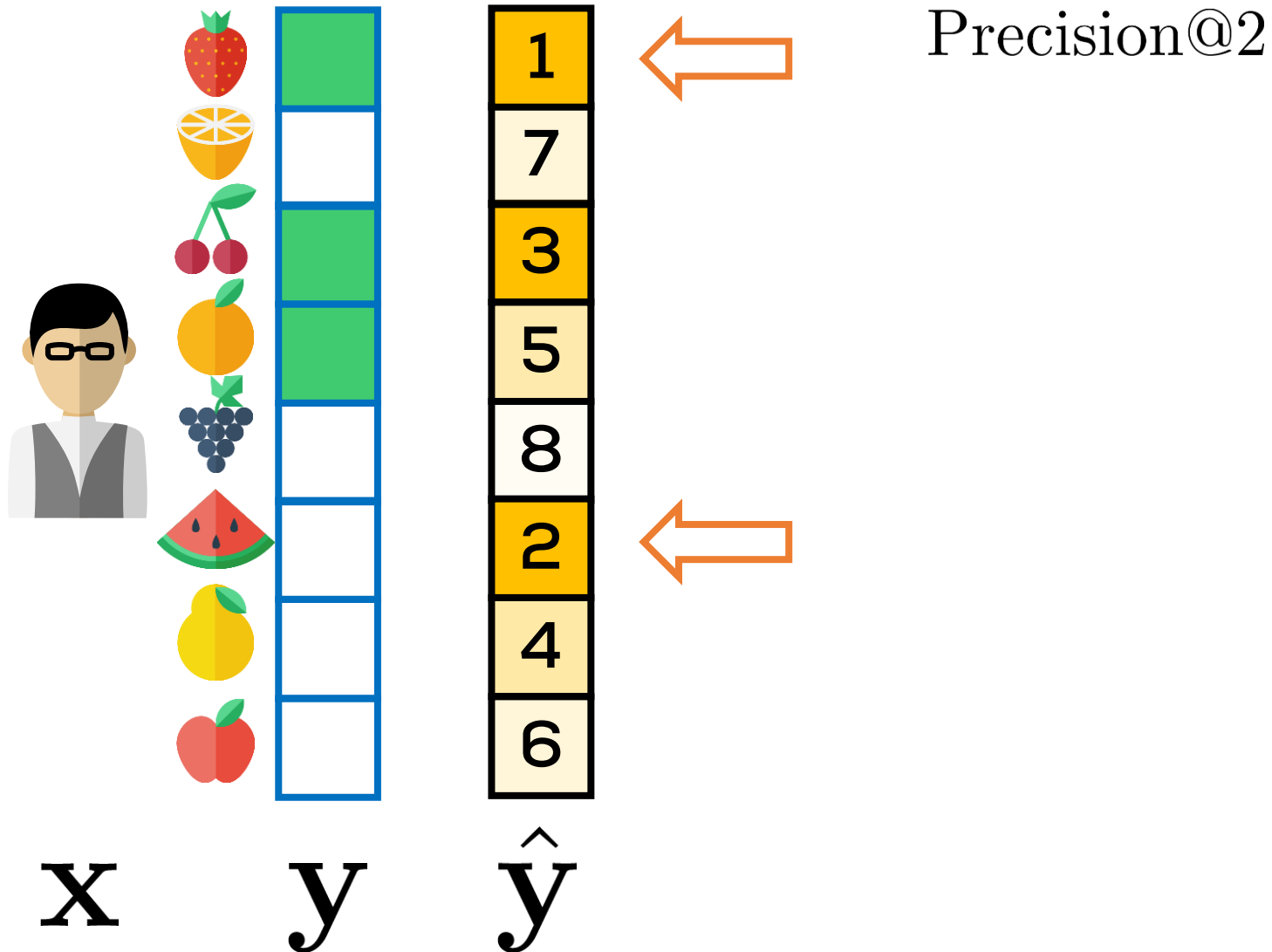
Precision@ k

Multi-label Classification Loss Functions

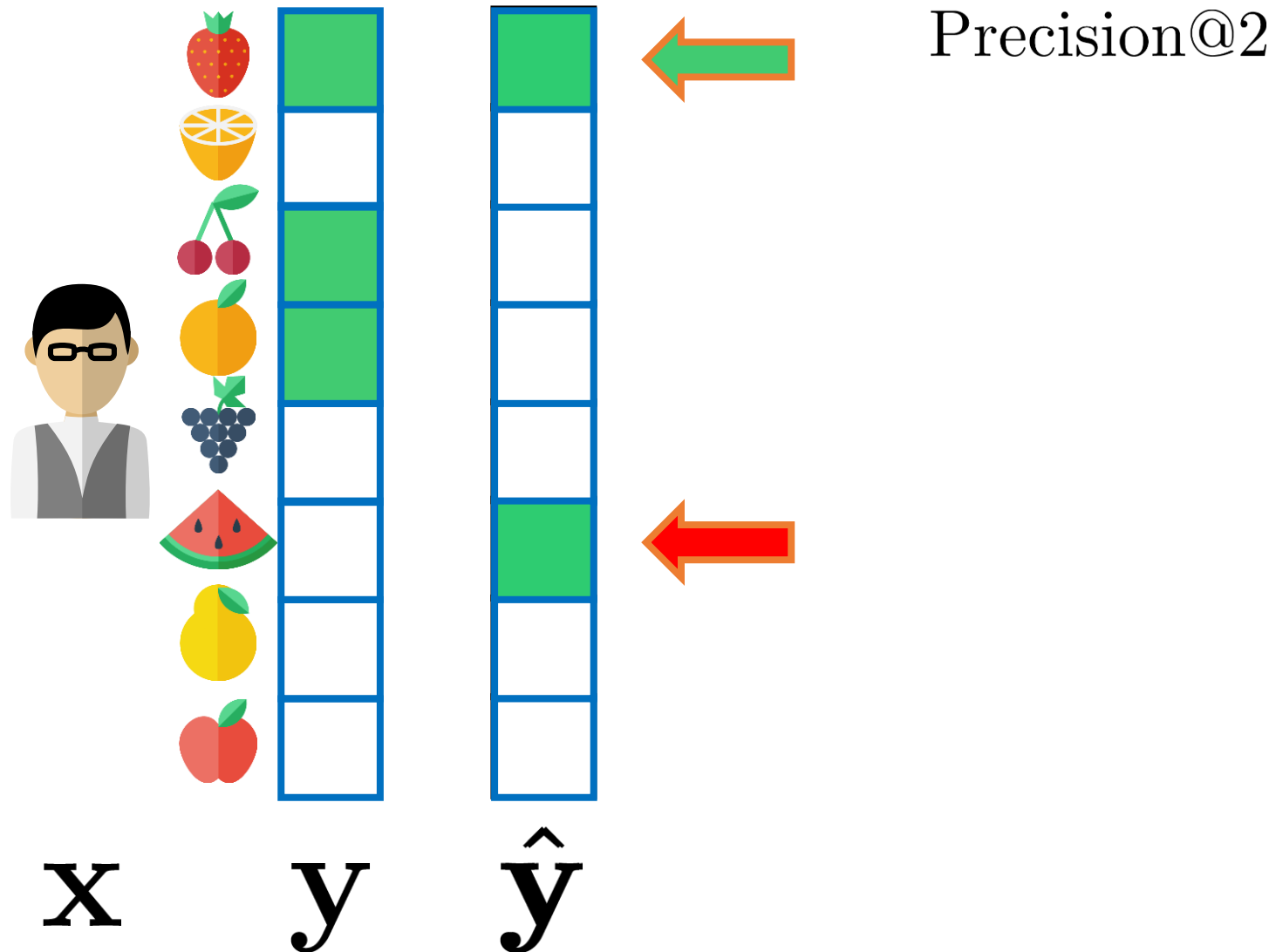


Precision@2

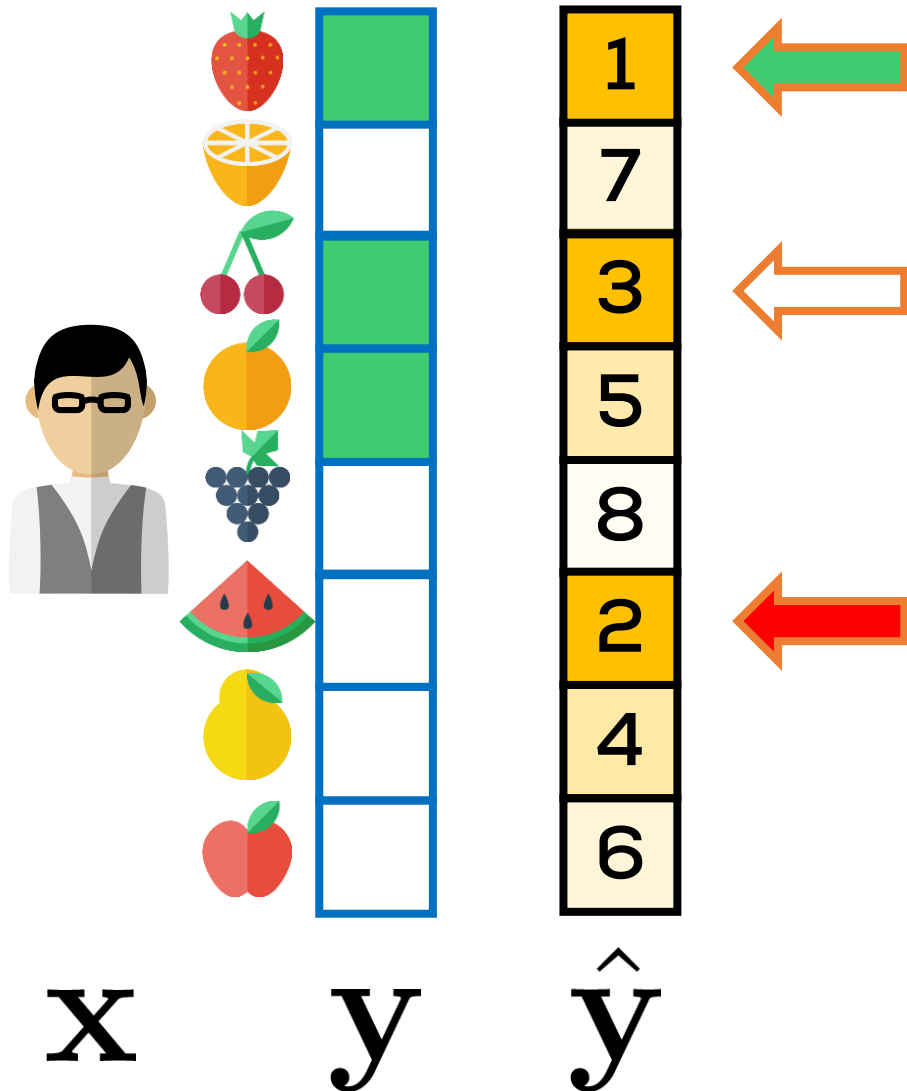
Multi-label Classification Loss Functions



Multi-label Classification Loss Functions

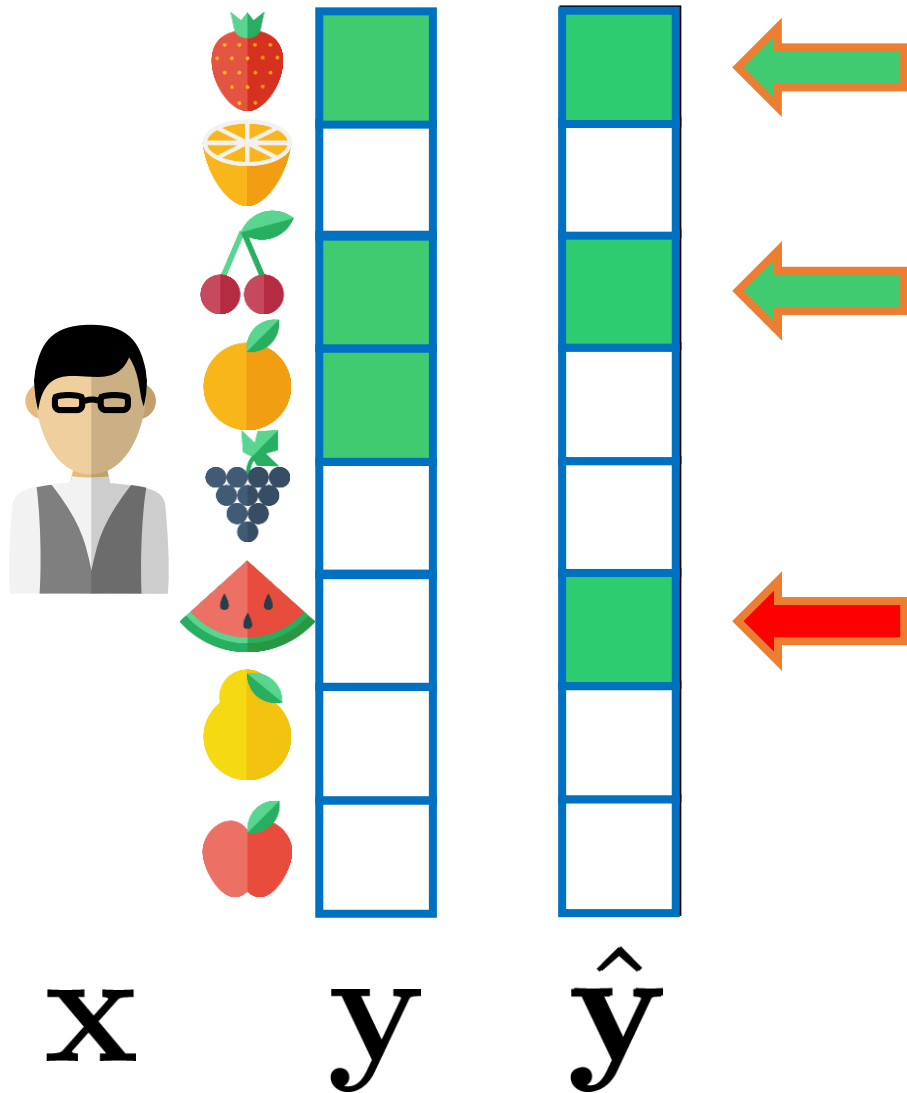


Multi-label Classification Loss Functions



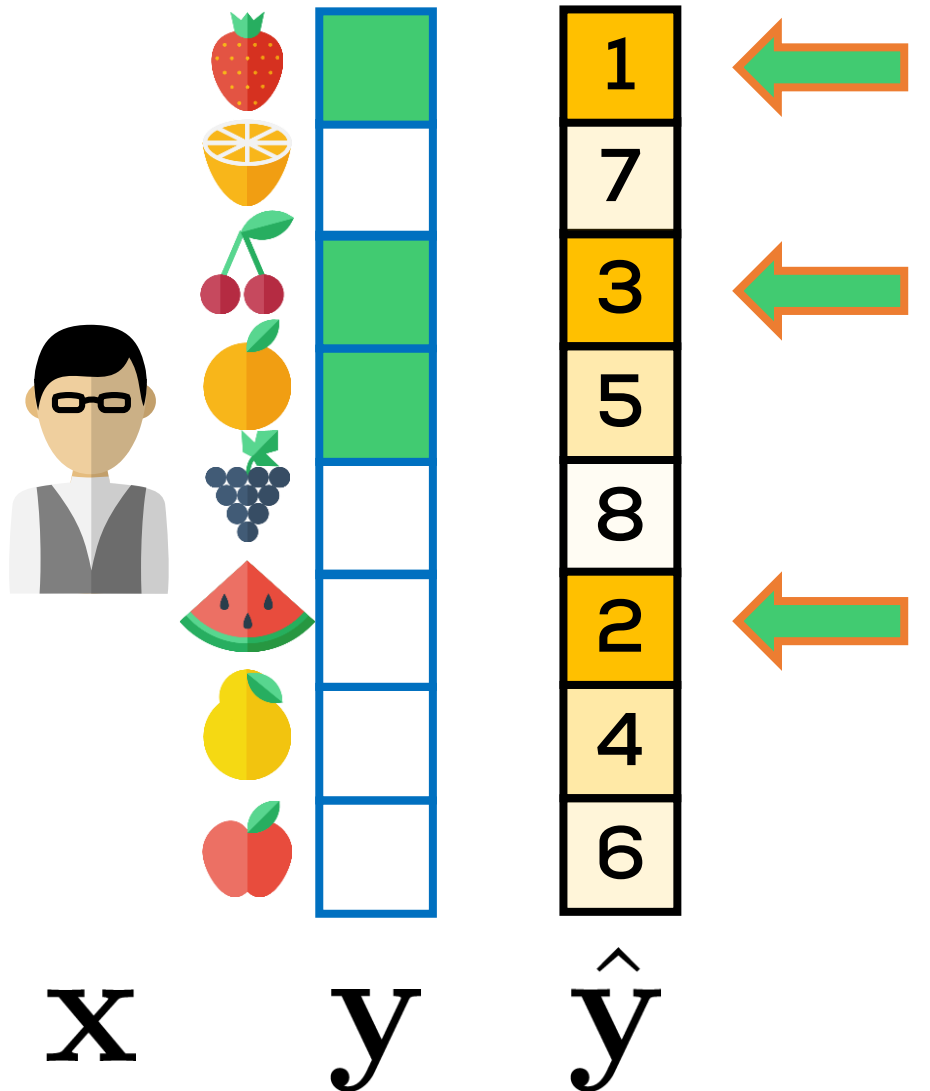
Precision@3

Multi-label Classification Loss Functions



Precision@3

Multi-label Classification Loss Functions



Precision@3

Recall@ k

Many, many, more
NDCG, AUC, MRR, MAP

How to optimize these losses?

... aka where is the rest of the Math?

Some Fundamentals

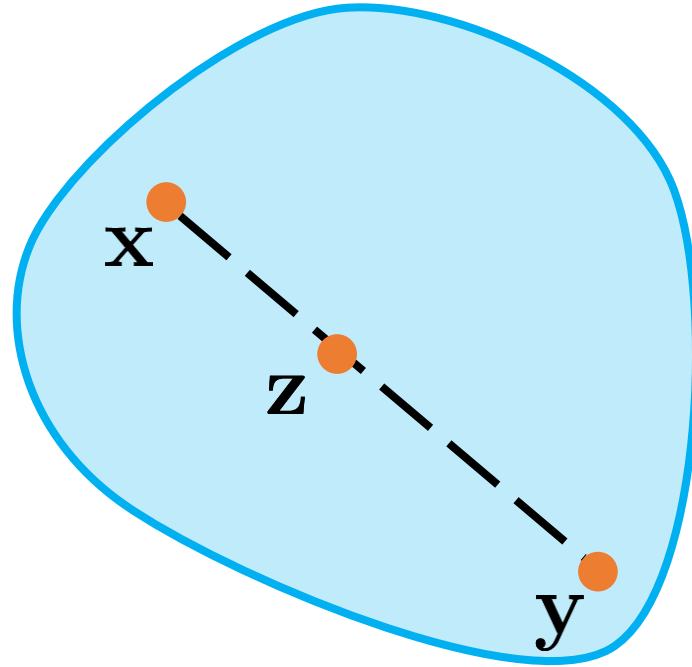
Functional Analysis, Optimization Theory

August 25, 2017

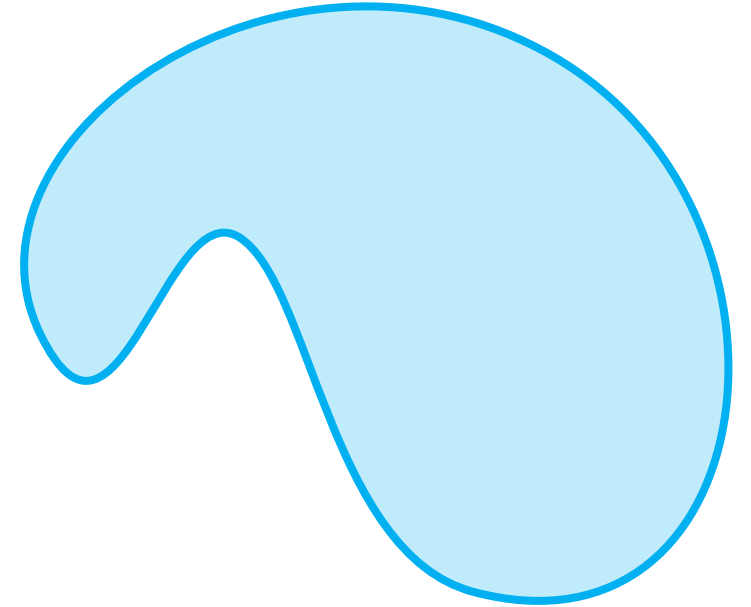


Convex Sets

$$\mathcal{C} \subseteq \mathbb{R}^d$$



CONVEX SET



NON-CONVEX SET

$$\forall \mathbf{x}, \mathbf{y} \in \mathcal{C}$$

$$\forall \lambda \in [0, 1]$$

$$\mathbf{z} = \lambda \cdot \mathbf{x} + (1 - \lambda) \cdot \mathbf{y} \in \mathcal{C}$$

Convex
combination

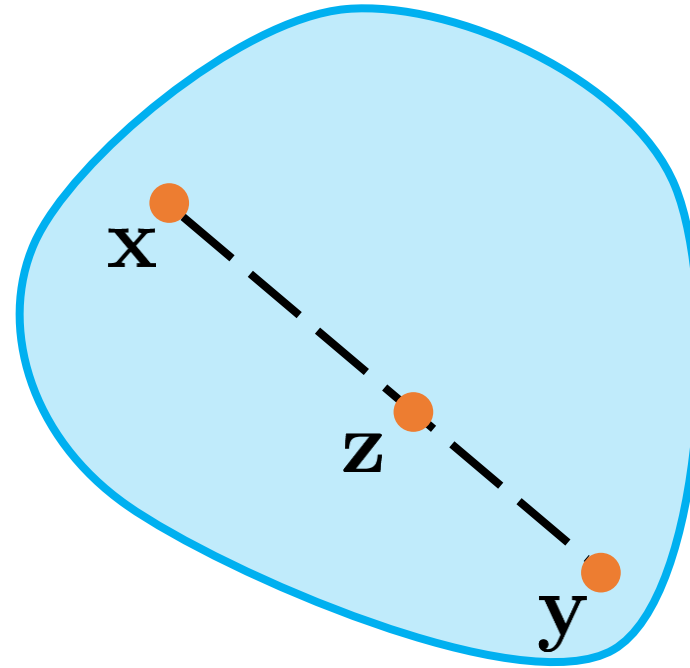
Convex Sets

$$\mathcal{C} \subseteq \mathbb{R}^d$$

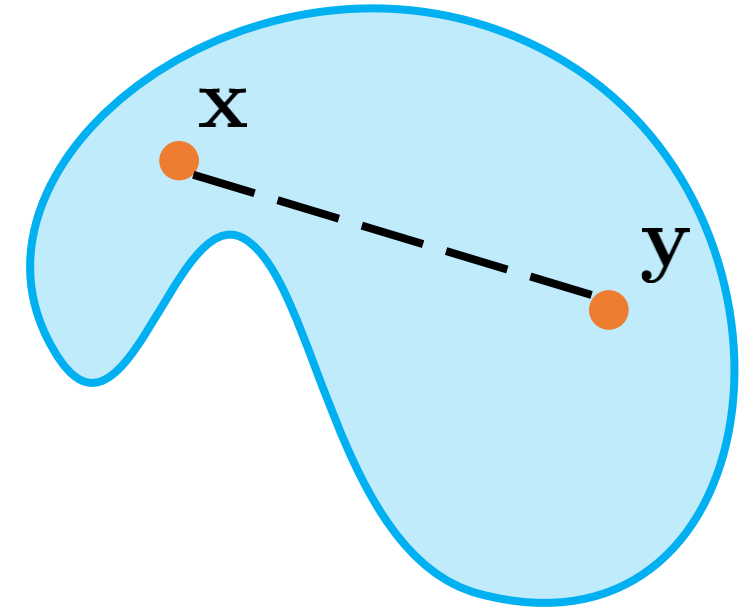
$$\forall \mathbf{x}, \mathbf{y} \in \mathcal{C}$$

$$\forall \lambda \in [0, 1]$$

$$\mathbf{z} = \lambda \cdot \mathbf{x} + (1 - \lambda) \cdot \mathbf{y} \in \mathcal{C}$$



CONVEX SET



NON-CONVEX SET

Convex
combination

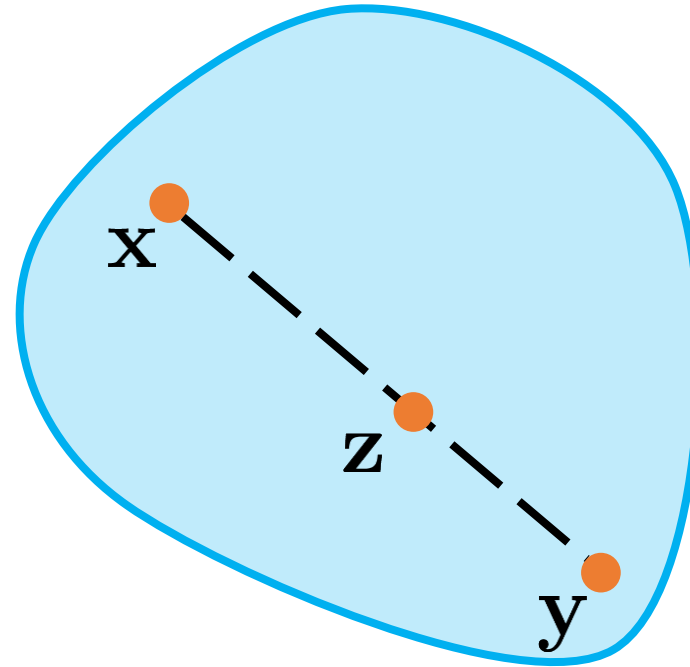
Convex Sets

$$\mathcal{C} \subseteq \mathbb{R}^d$$

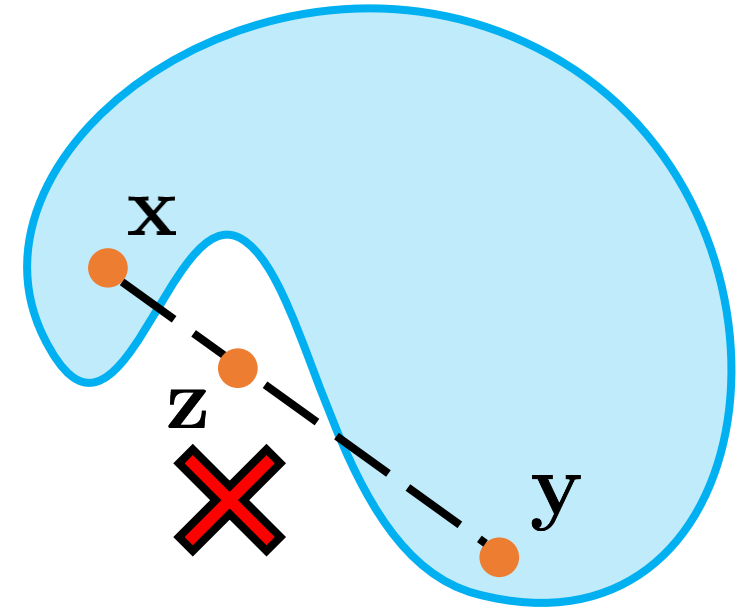
$$\forall \mathbf{x}, \mathbf{y} \in \mathcal{C}$$

$$\forall \lambda \in [0, 1]$$

$$\mathbf{z} = \lambda \cdot \mathbf{x} + (1 - \lambda) \cdot \mathbf{y} \in \mathcal{C}$$



CONVEX SET

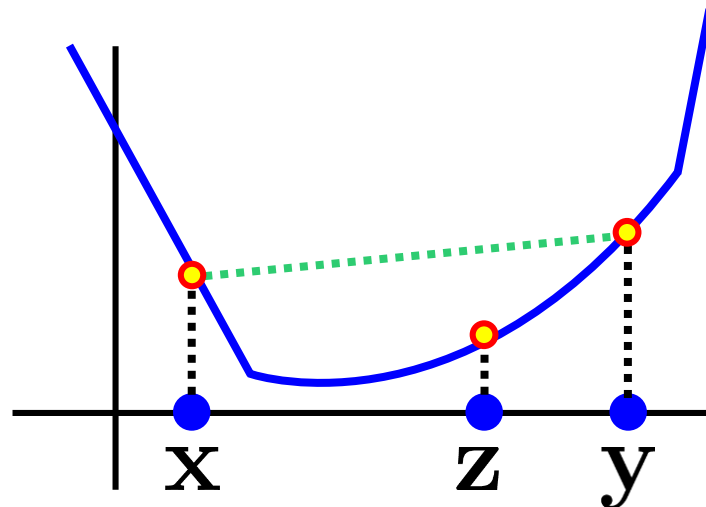


NON-CONVEX SET

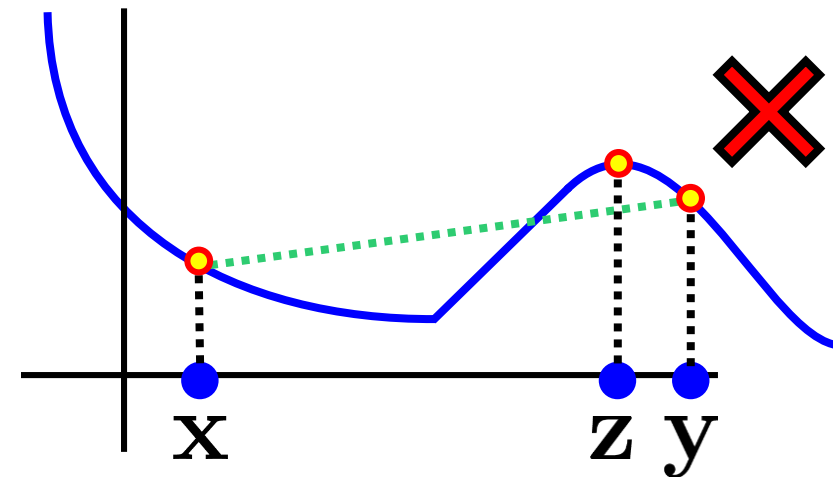
Convex
combination

Convex Functions

$$f : \mathbb{R}^d \rightarrow \mathbb{R}$$



CONVEX FUNCTION



NON-CONVEX
FUNCTION

$$\forall \mathbf{x}, \mathbf{y}$$

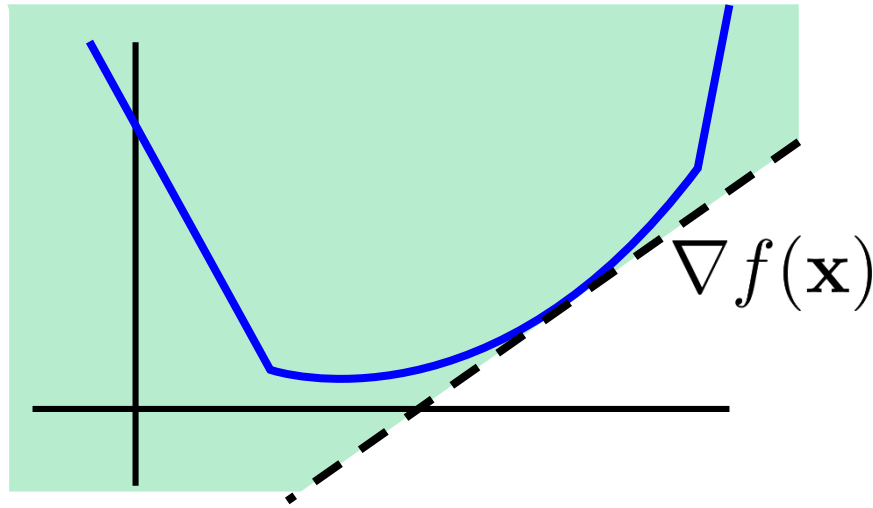
$$\forall \lambda \in [0, 1]$$

$$\mathbf{z} = \lambda \cdot \mathbf{x} + (1 - \lambda) \cdot \mathbf{y}$$

$$f(\mathbf{z}) \leq \lambda \cdot f(\mathbf{x}) + (1 - \lambda) \cdot f(\mathbf{y})$$

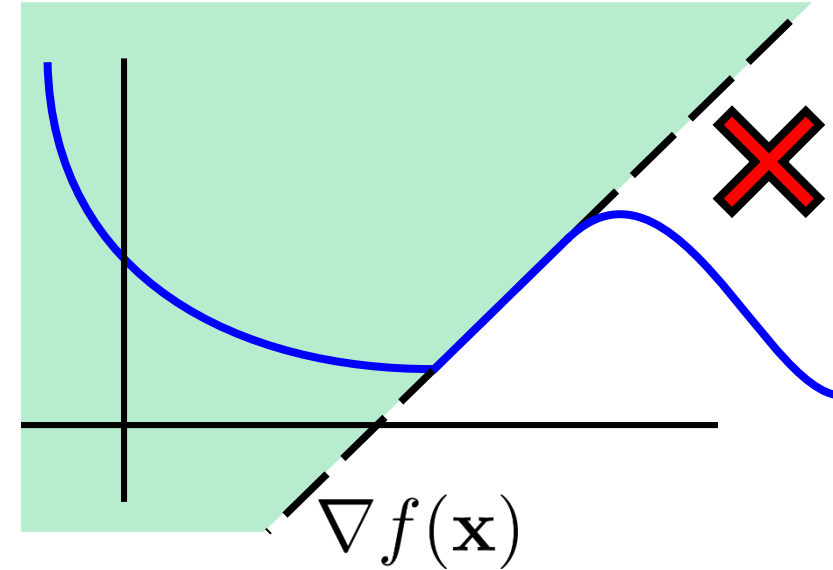
Convex Functions

$$f : \mathbb{R}^d \rightarrow \mathbb{R}$$



$$\forall \mathbf{x}, \mathbf{y}$$

CONVEX FUNCTION

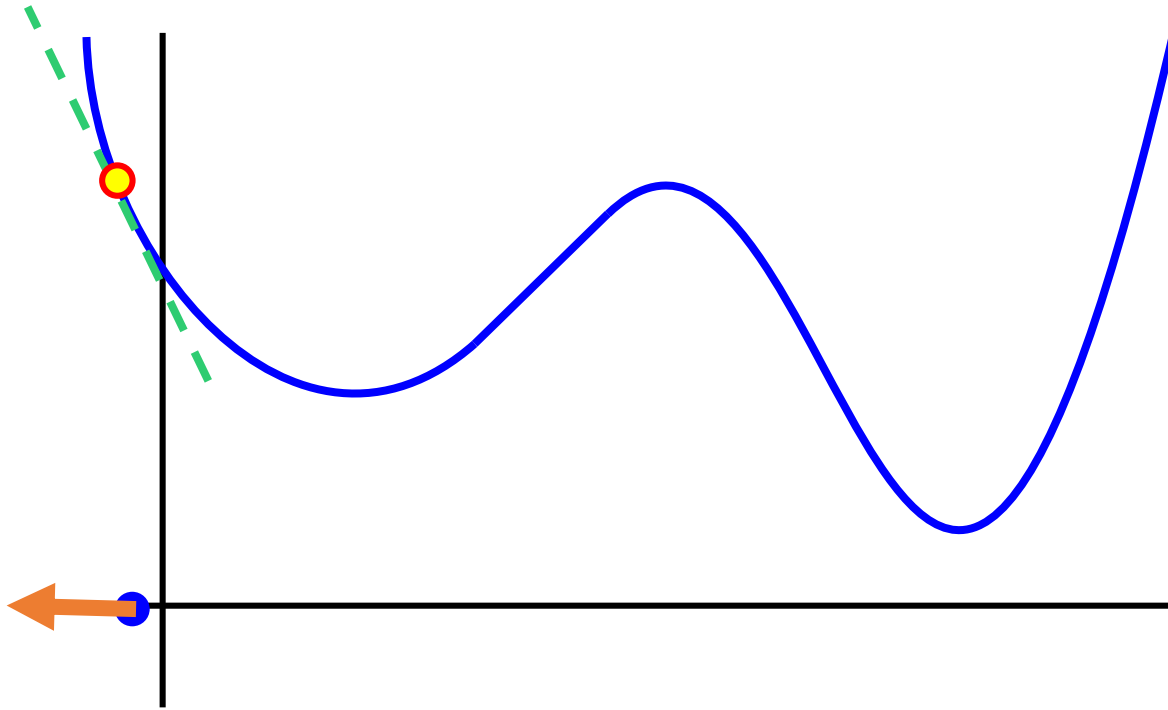


**NON-CONVEX
FUNCTION**

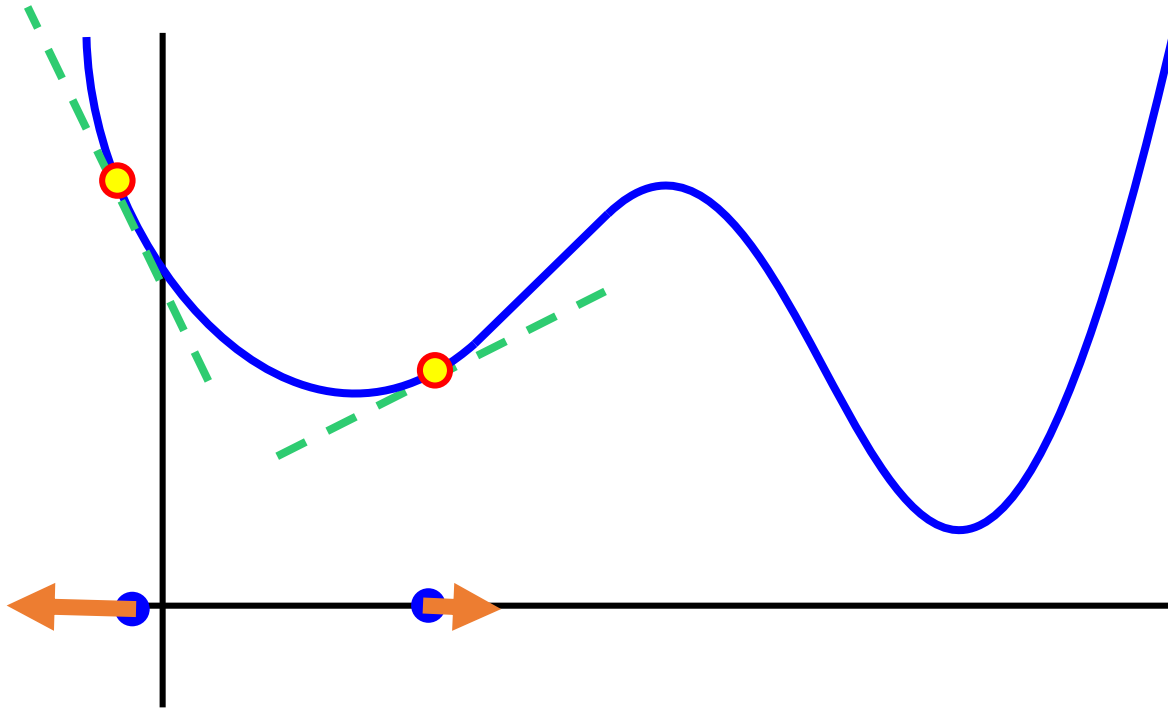
Differentiable function!

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$$

Gradients

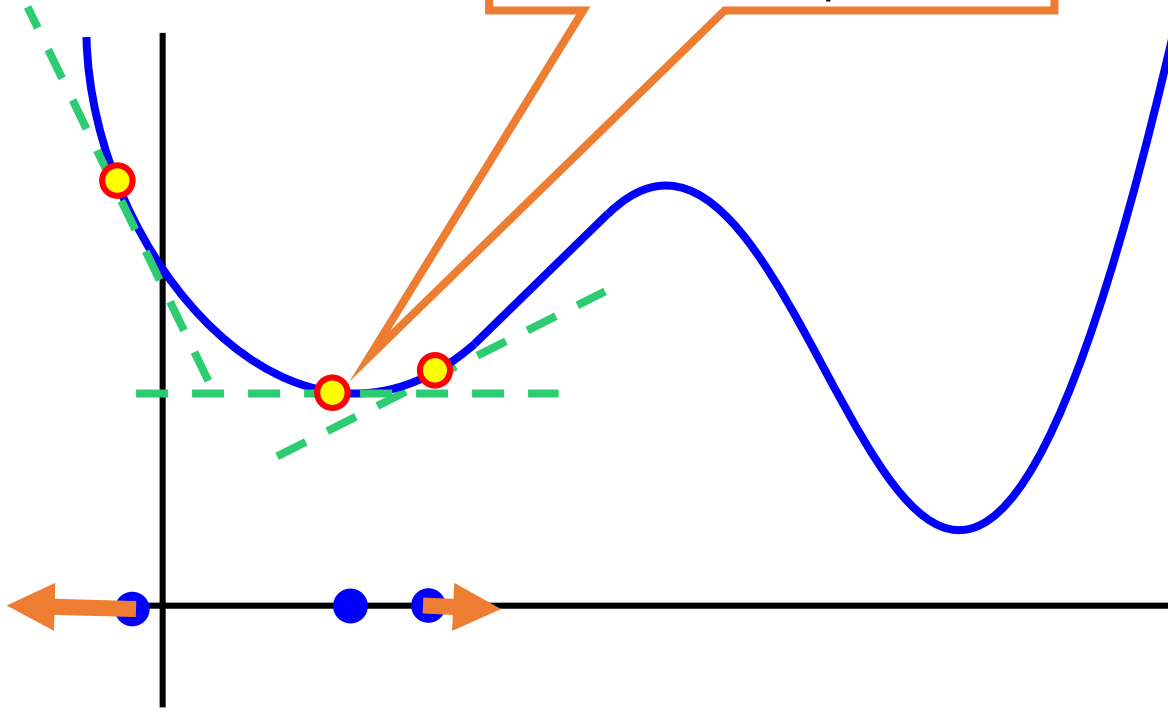


Gradients



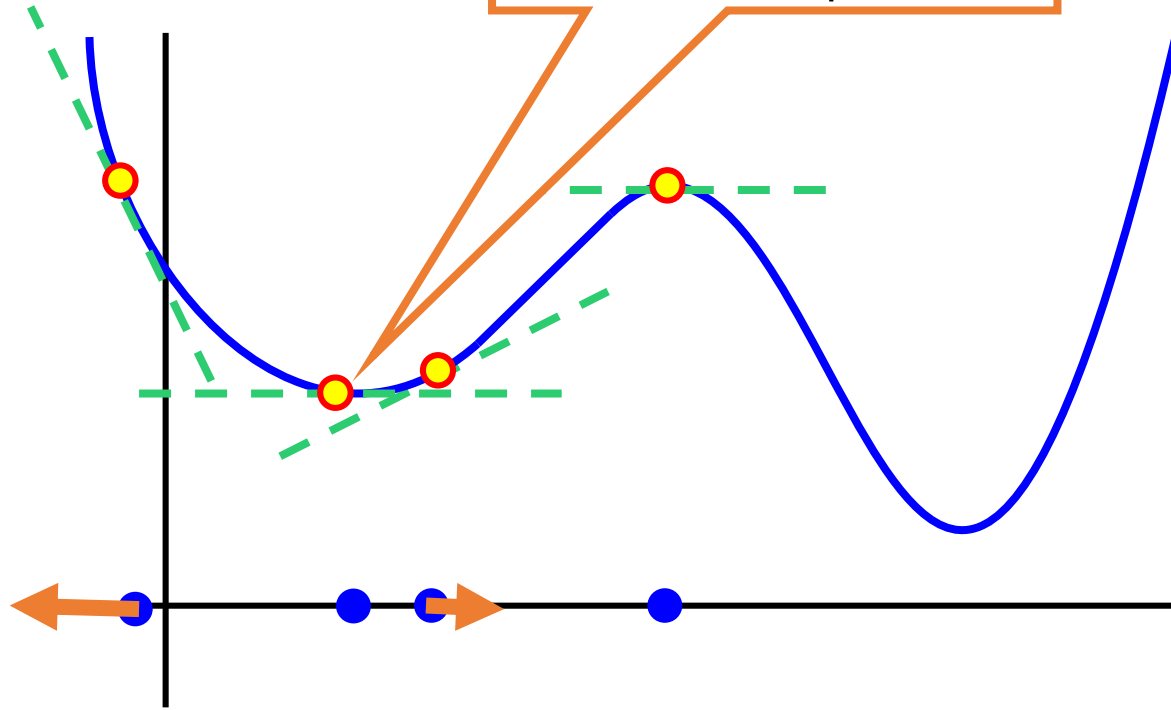
Gradients

Gradients vanish
at local optima



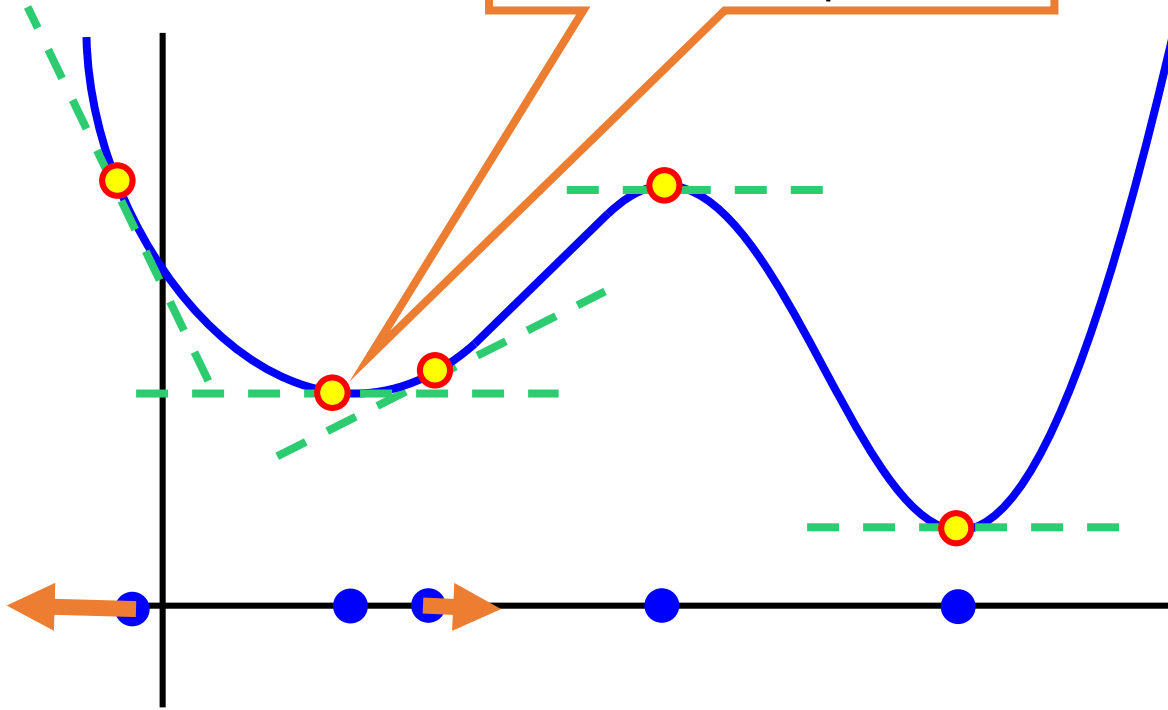
Gradients

Gradients vanish
at local optima

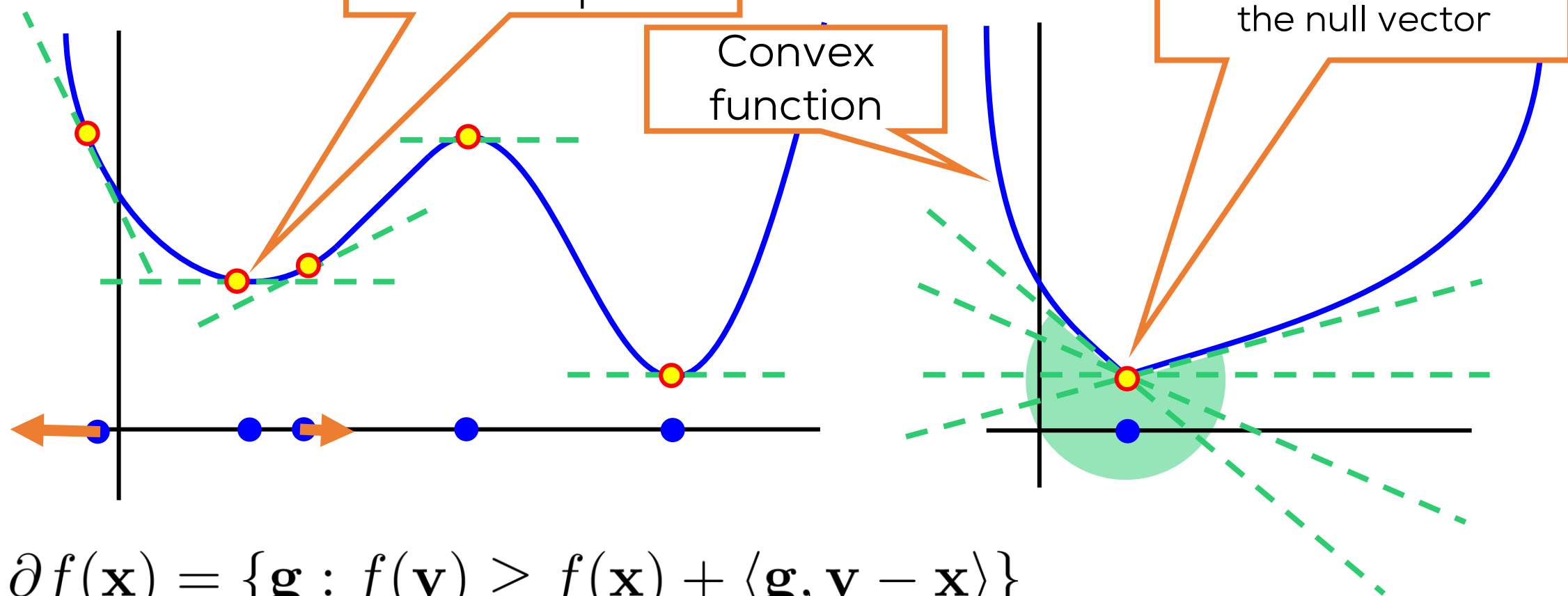


Gradients

Gradients vanish
at local optima



Gradients

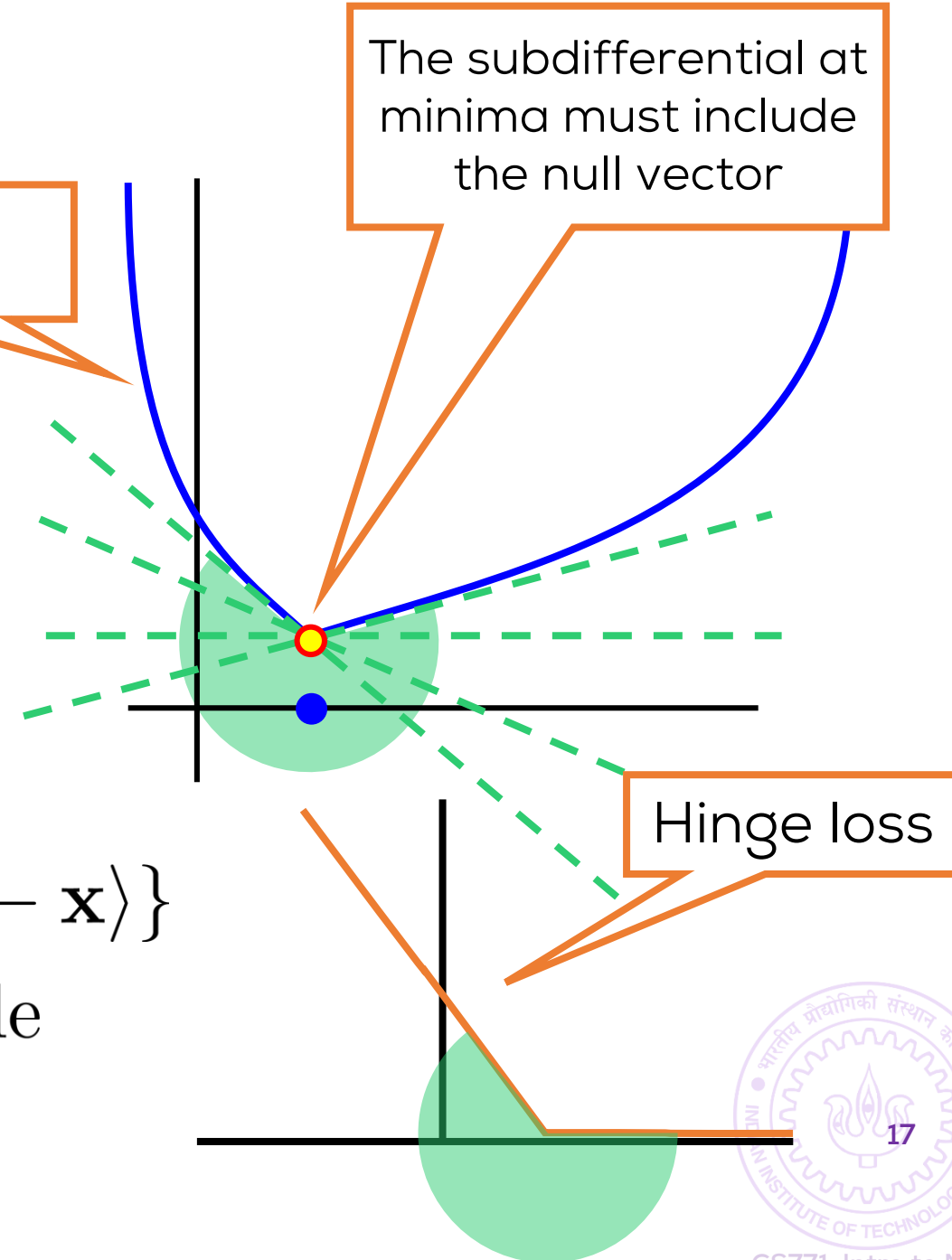
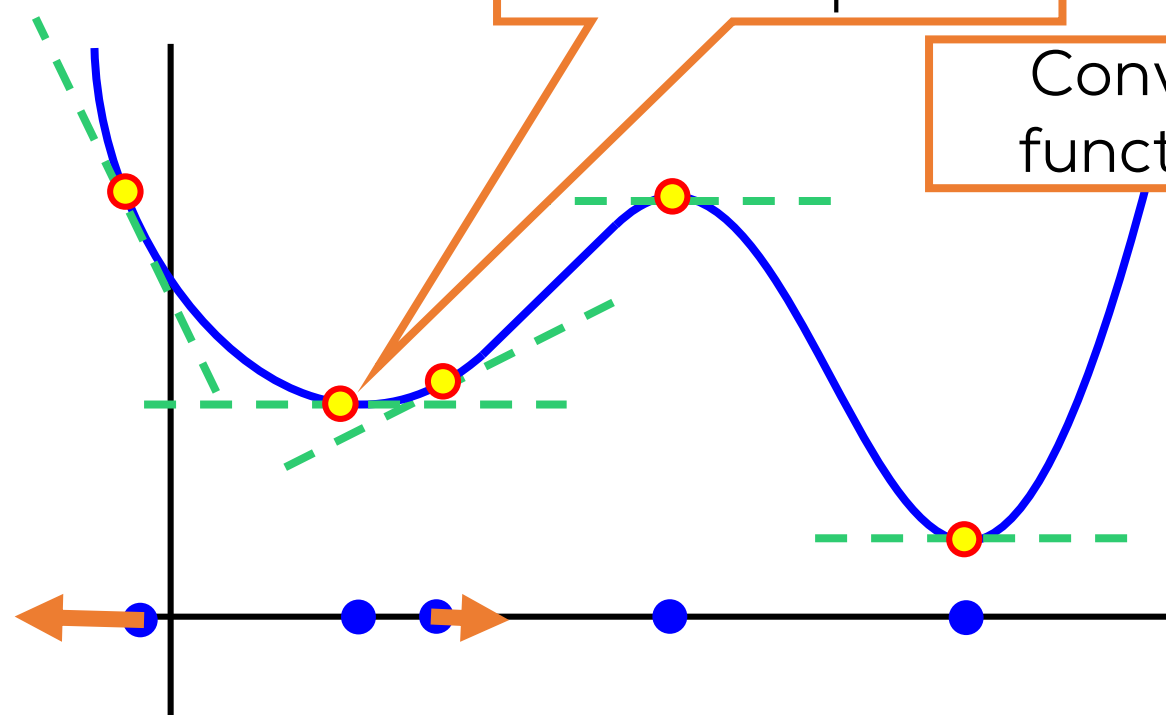


$$\partial f(\mathbf{x}) = \{\mathbf{g} : f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \mathbf{g}, \mathbf{y} - \mathbf{x} \rangle\}$$

$$\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\} \text{ if } f \text{ differentiable}$$

$$\partial f(\mathbf{x}) \ni \mathbf{0} \text{ if } \mathbf{x} \text{ minimum}$$

Gradients



$$\partial f(\mathbf{x}) = \{\mathbf{g} : f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \mathbf{g}, \mathbf{y} - \mathbf{x} \rangle\}$$

$$\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\} \text{ if } f \text{ differentiable}$$

$$\partial f(\mathbf{x}) \ni \mathbf{0} \text{ if } \mathbf{x} \text{ minimum}$$

So ...

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

Empirical
Loss function

Regularizer

Examples

$$f(\mathbf{w}) = \sum_{i=1}^n (y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle)^2 \quad f(\mathbf{w}) = \sum_{i=1}^n \log (1 + \exp(-y^i \langle \mathbf{w}, \mathbf{x}^i \rangle))$$

$$r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_1 \quad r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_2^2$$

Use Optimality Condition

Only works for the simplest of problems

First-order Optimality

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

$$\nabla f(\mathbf{w}) + \nabla r(\mathbf{w}) = \mathbf{0}$$

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

$$r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_2^2$$

$$2 \sum_{i=1}^n (\langle \mathbf{w}, \mathbf{x}^i \rangle - y^i) \cdot \mathbf{x}_i + 2\lambda \cdot \mathbf{w} = \mathbf{0}$$

First-order Optimality

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

$$\nabla f(\mathbf{w}) + \nabla r(\mathbf{w}) = \mathbf{0}$$

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

$$r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\left(\sum_{i=1}^n (\mathbf{x}^i)(\mathbf{x}^i)^\top + \lambda \cdot I \right) \mathbf{w} = \sum_{i=1}^n y^i \mathbf{x}^i$$

First-order Optimality

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

$$\nabla f(\mathbf{w}) + \nabla r(\mathbf{w}) = \mathbf{0}$$

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

$$r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_2^2$$

$$\mathbf{w} = (\mathbf{X}\mathbf{X}^\top + \lambda \cdot I)^{-1} \mathbf{X}\mathbf{y}$$

First-order Optimality

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

$$\nabla f(\mathbf{w}) + \nabla r(\mathbf{w}) = \mathbf{0}$$

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

$$r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_1$$

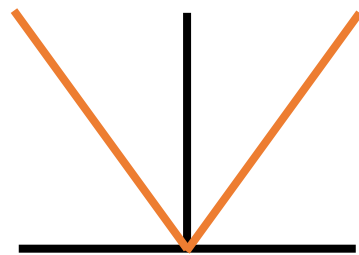
First-order Optimality

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

$$\nabla f(\mathbf{w}) + \nabla r(\mathbf{w}) = \mathbf{0}$$

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

$$r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_1$$



$$\partial f(\mathbf{w}) + \partial r(\mathbf{w}) \ni \mathbf{0}$$

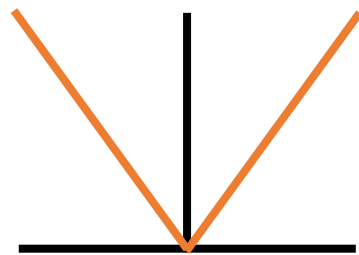
First-order Optimality

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

$$\nabla f(\mathbf{w}) + \nabla r(\mathbf{w}) = \mathbf{0}$$

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

$$r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_1$$



$$\nabla f(\mathbf{w}) + \partial r(\mathbf{w}) \ni \mathbf{0}$$

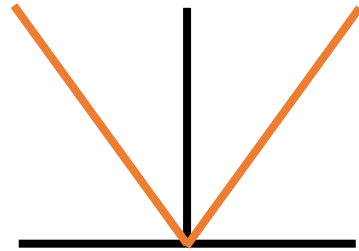
First-order Optimality

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

$$\nabla f(\mathbf{w}) + \nabla r(\mathbf{w}) = \mathbf{0}$$

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

$$r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_1$$

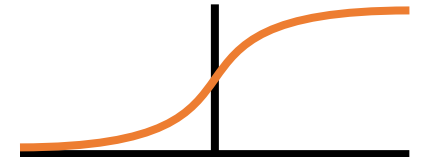


$$\nabla f(\mathbf{w}) + \partial r(\mathbf{w}) \ni \mathbf{0}$$

$$f(\mathbf{w}) = \sum_{i=1}^n \log (1 + \exp(-y^i \langle \mathbf{w}, \mathbf{x}^i \rangle))$$

$$r(\mathbf{w}) = \lambda \cdot \|\mathbf{w}\|_2^2$$

?

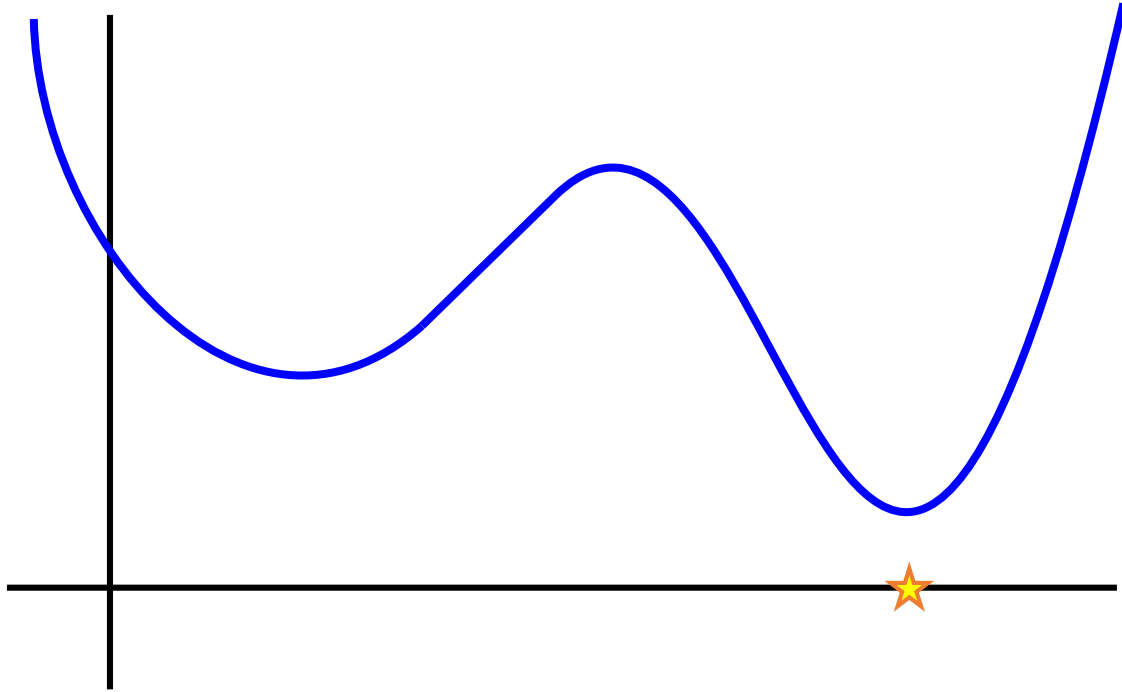


$$\sum_{i=1}^n (1 - \sigma(y^i \langle \mathbf{w}, \mathbf{x}^i \rangle)) y^i \cdot \mathbf{x}^i + 2\lambda \cdot \mathbf{w} = \mathbf{0}$$

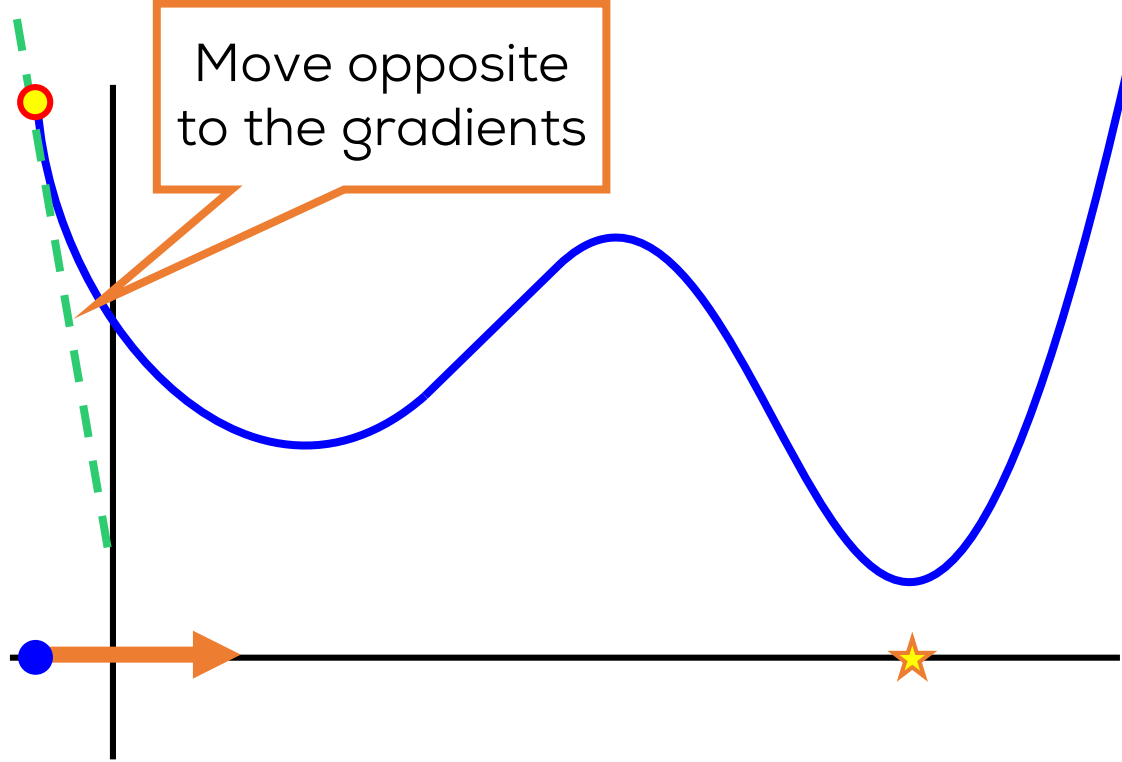
Use (sub)Gradient Descent

Use this technique before anything else

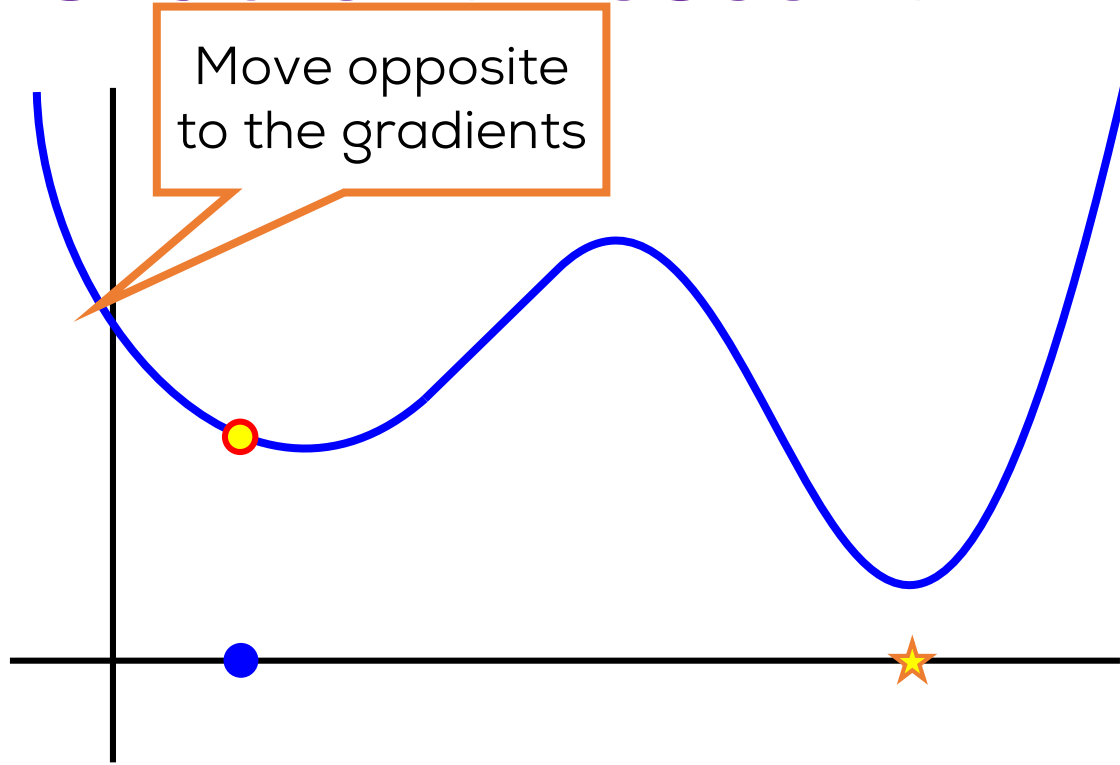
Gradient Descent



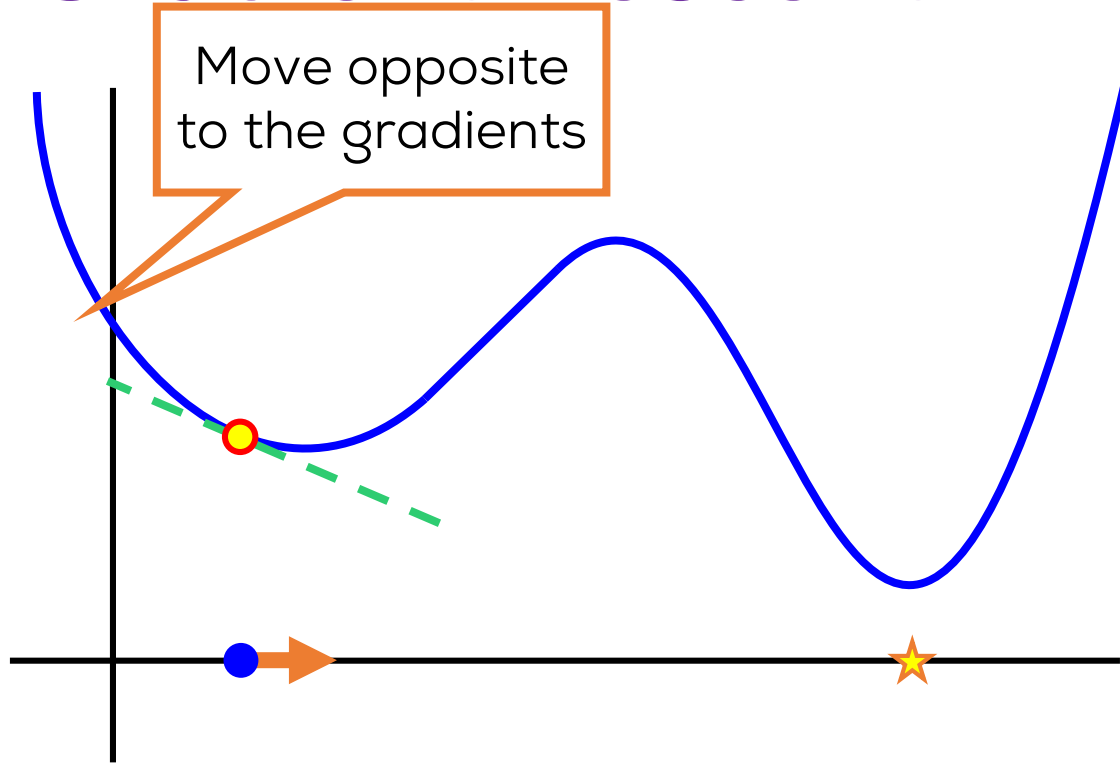
Gradient Descent



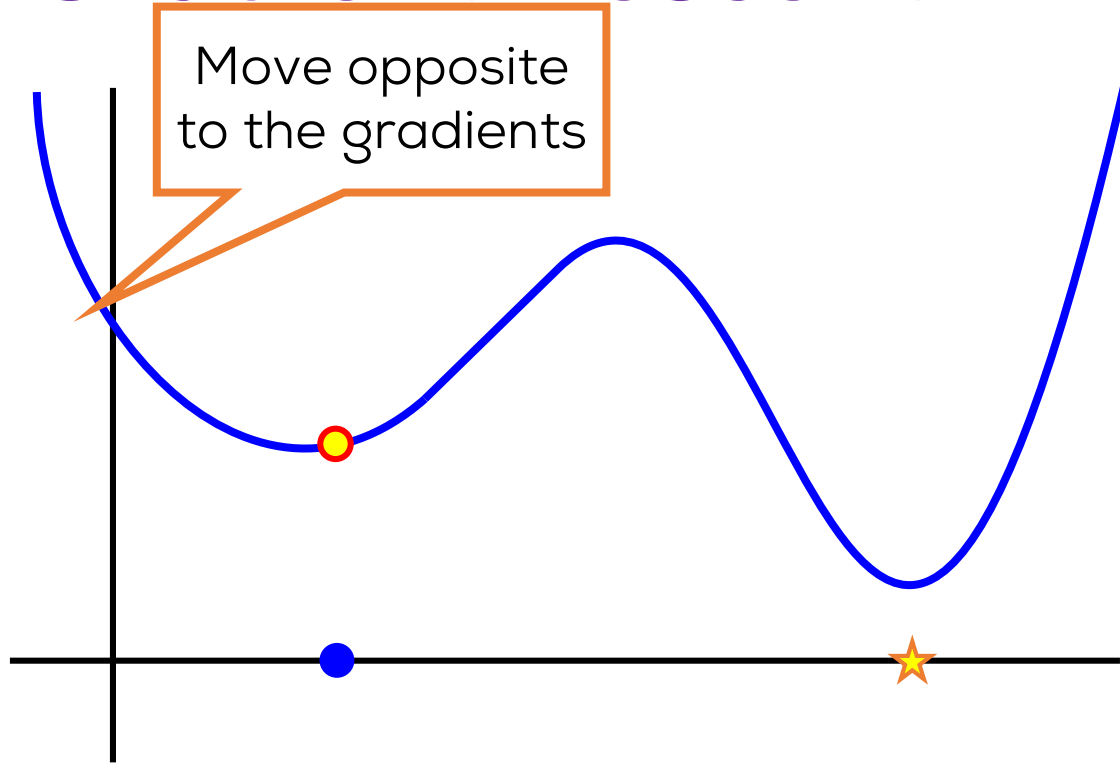
Gradient Descent



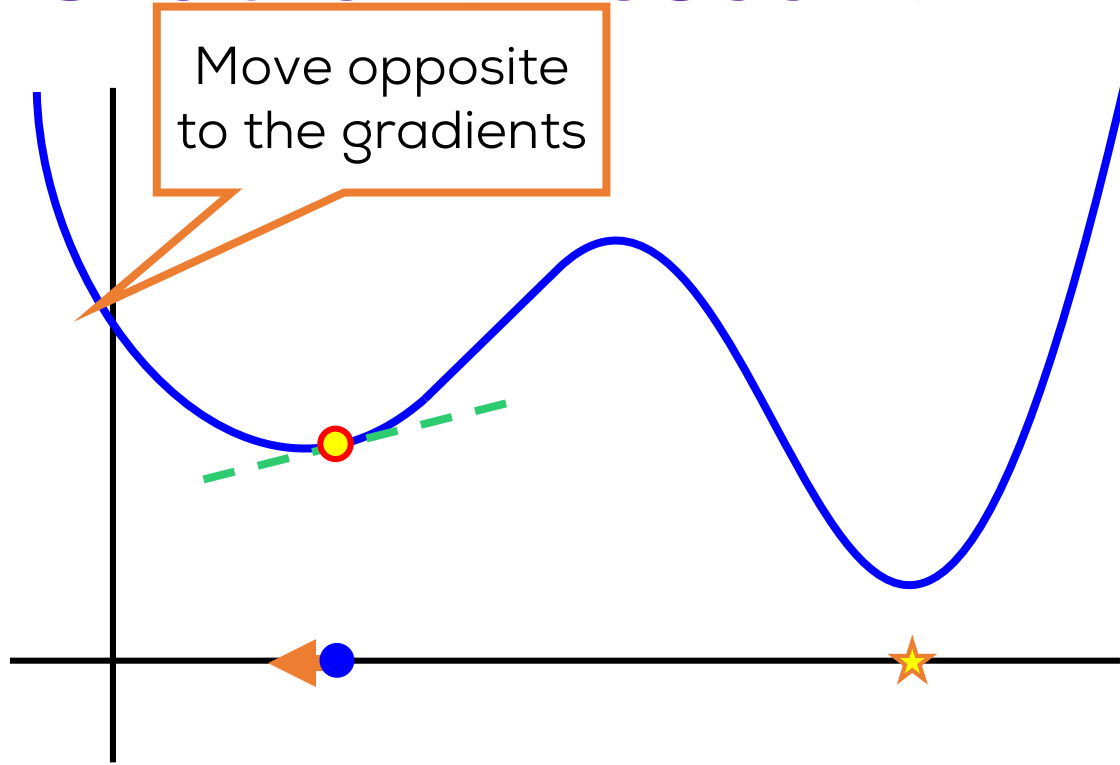
Gradient Descent



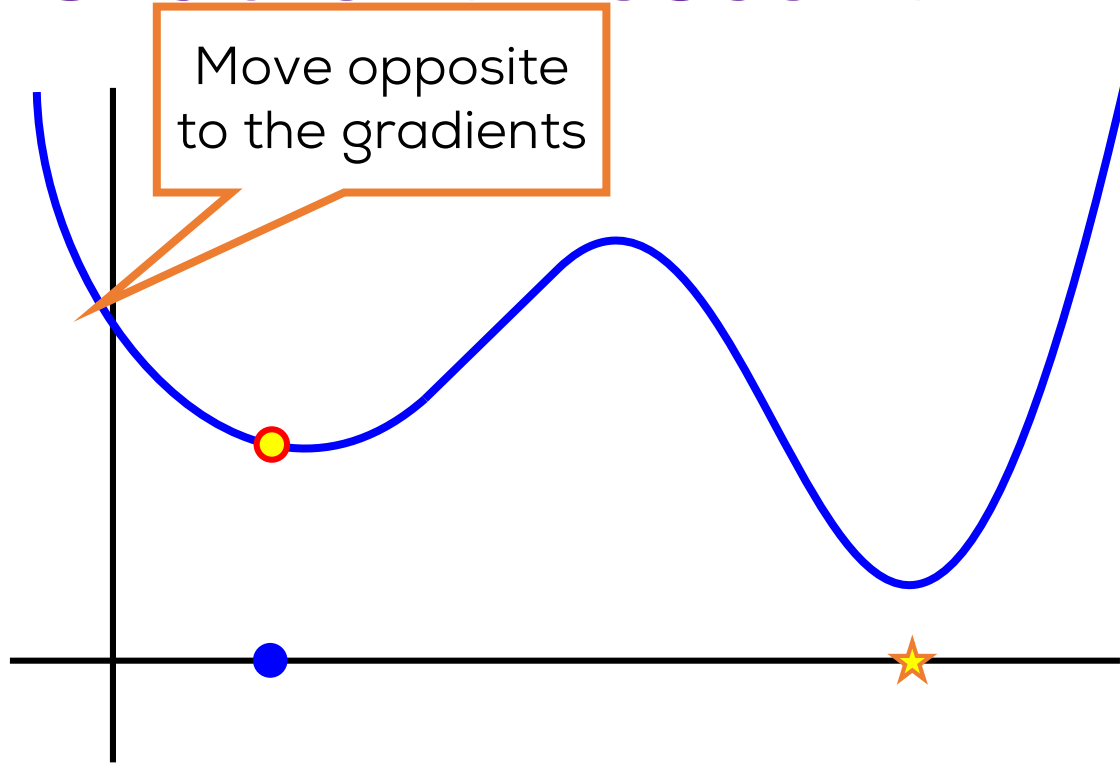
Gradient Descent



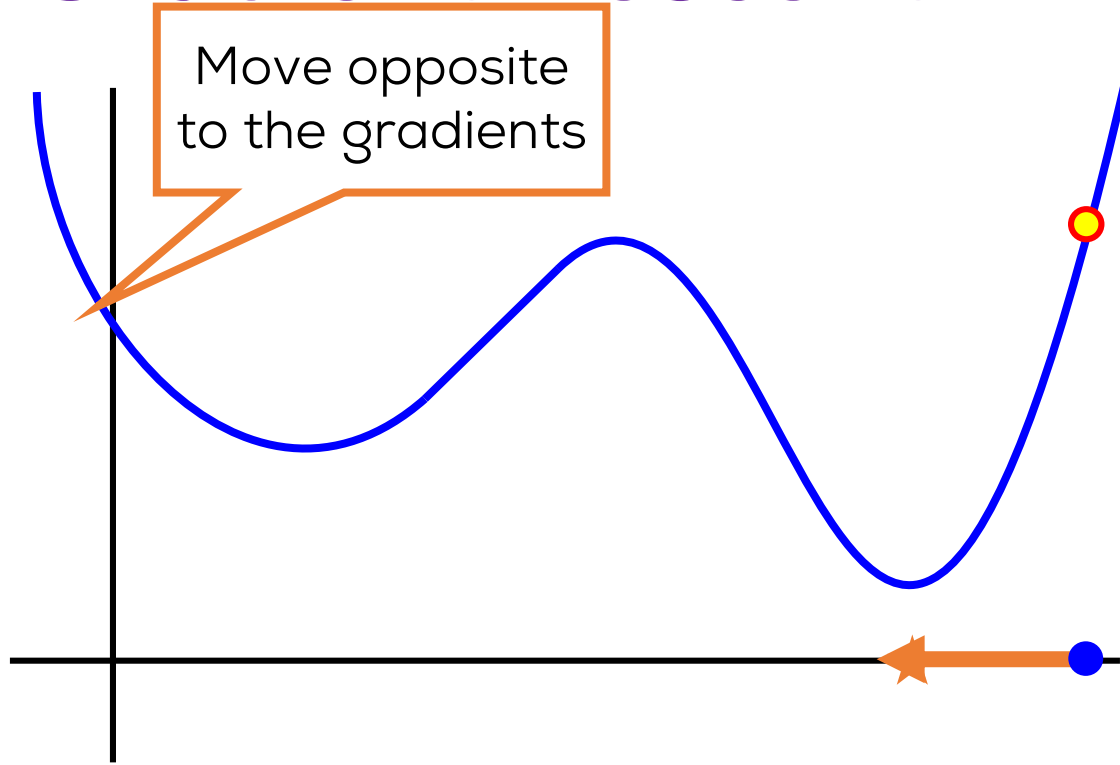
Gradient Descent



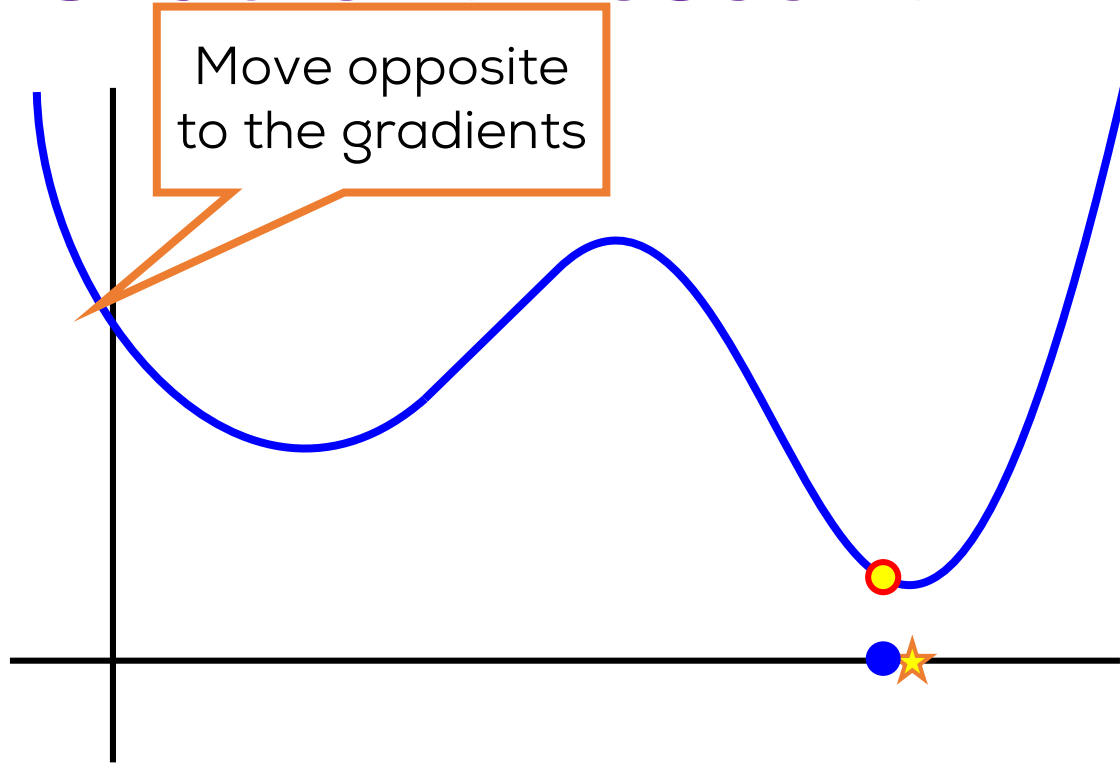
Gradient Descent



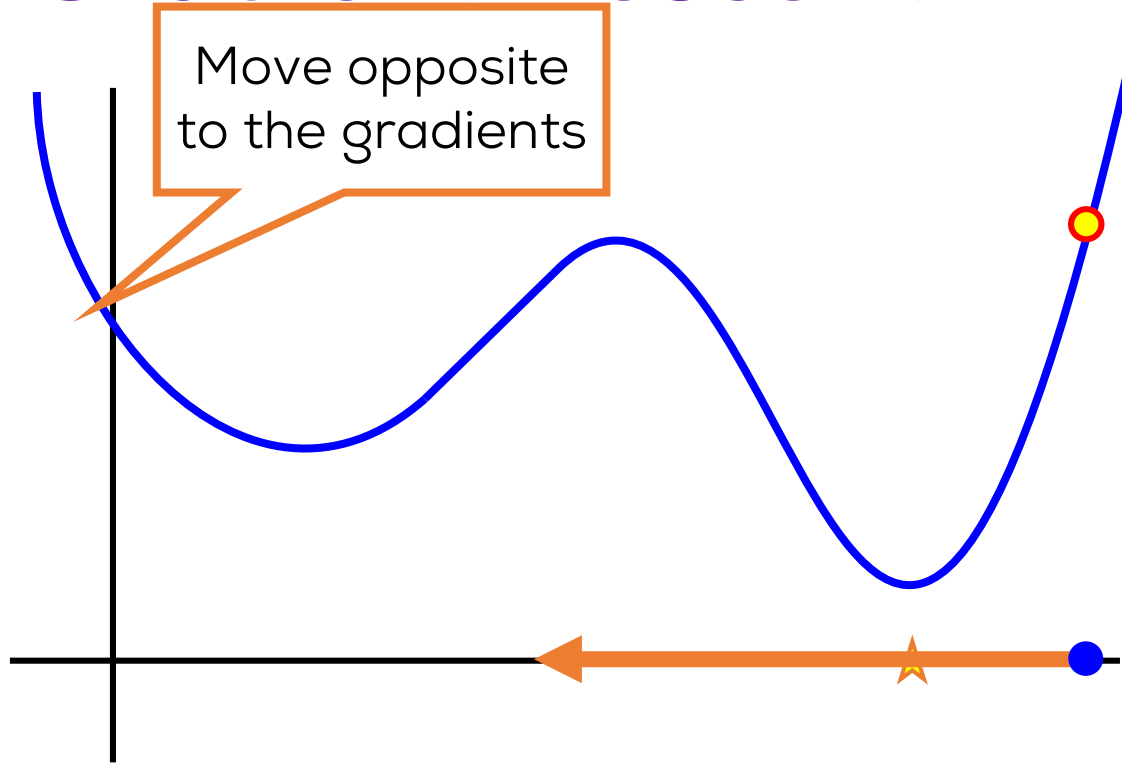
Gradient Descent



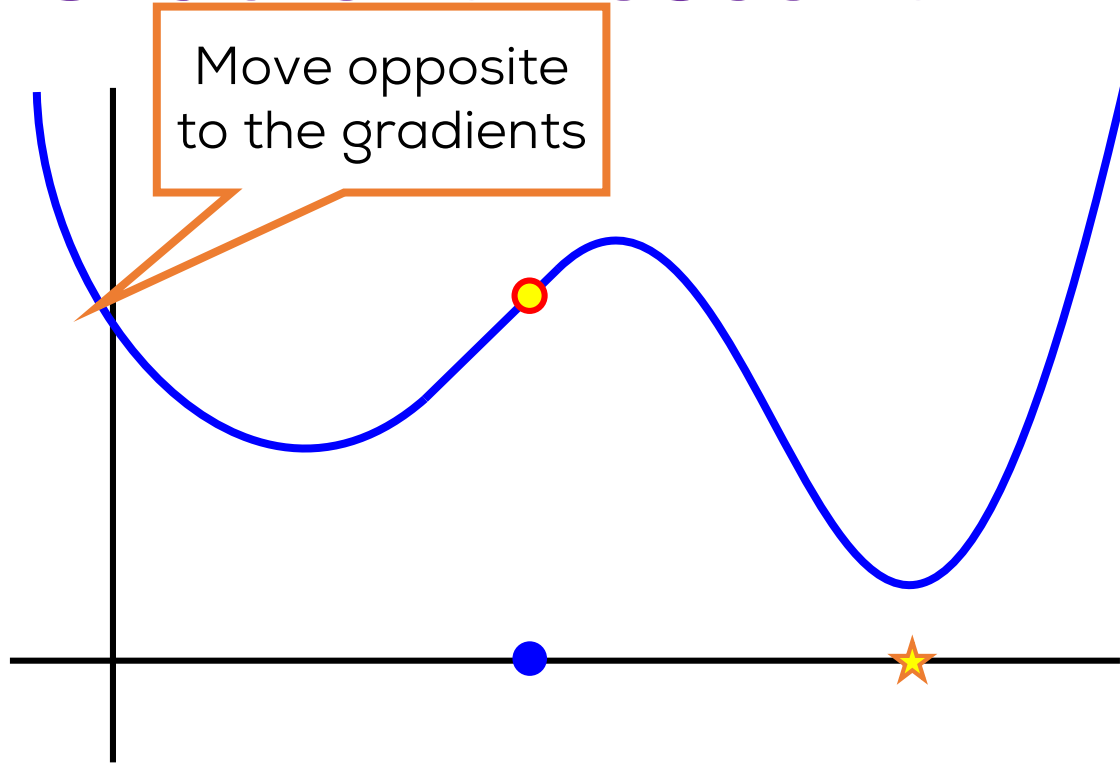
Gradient Descent



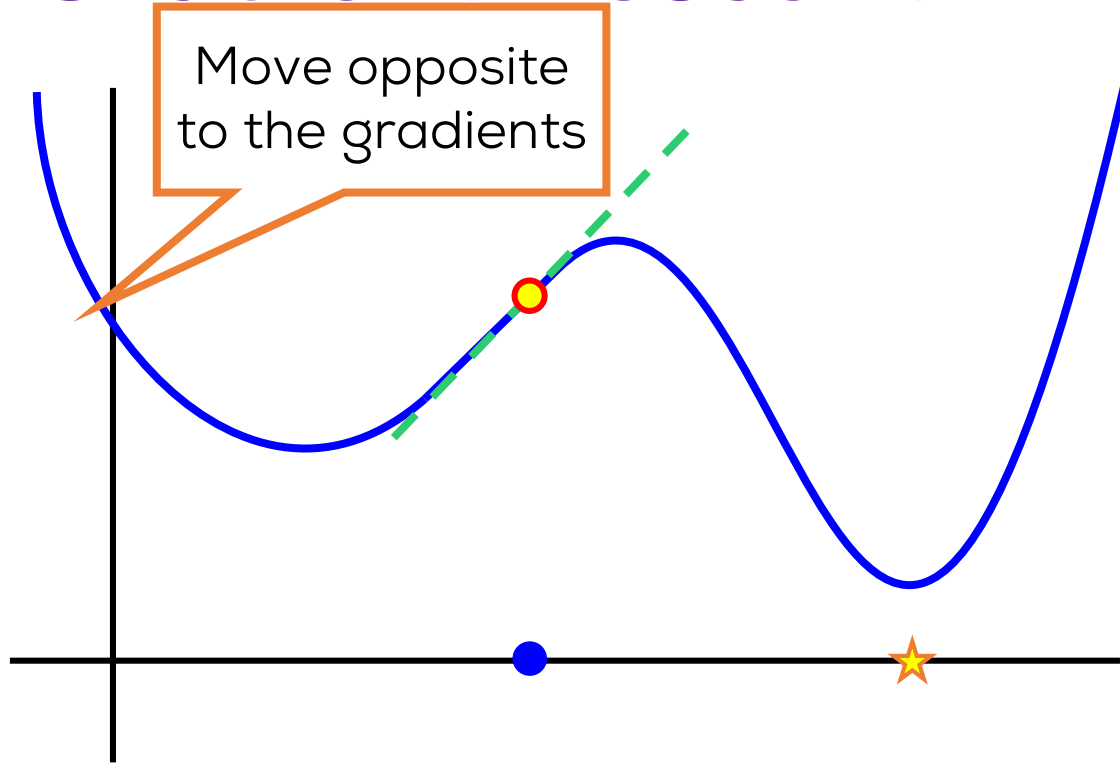
Gradient Descent



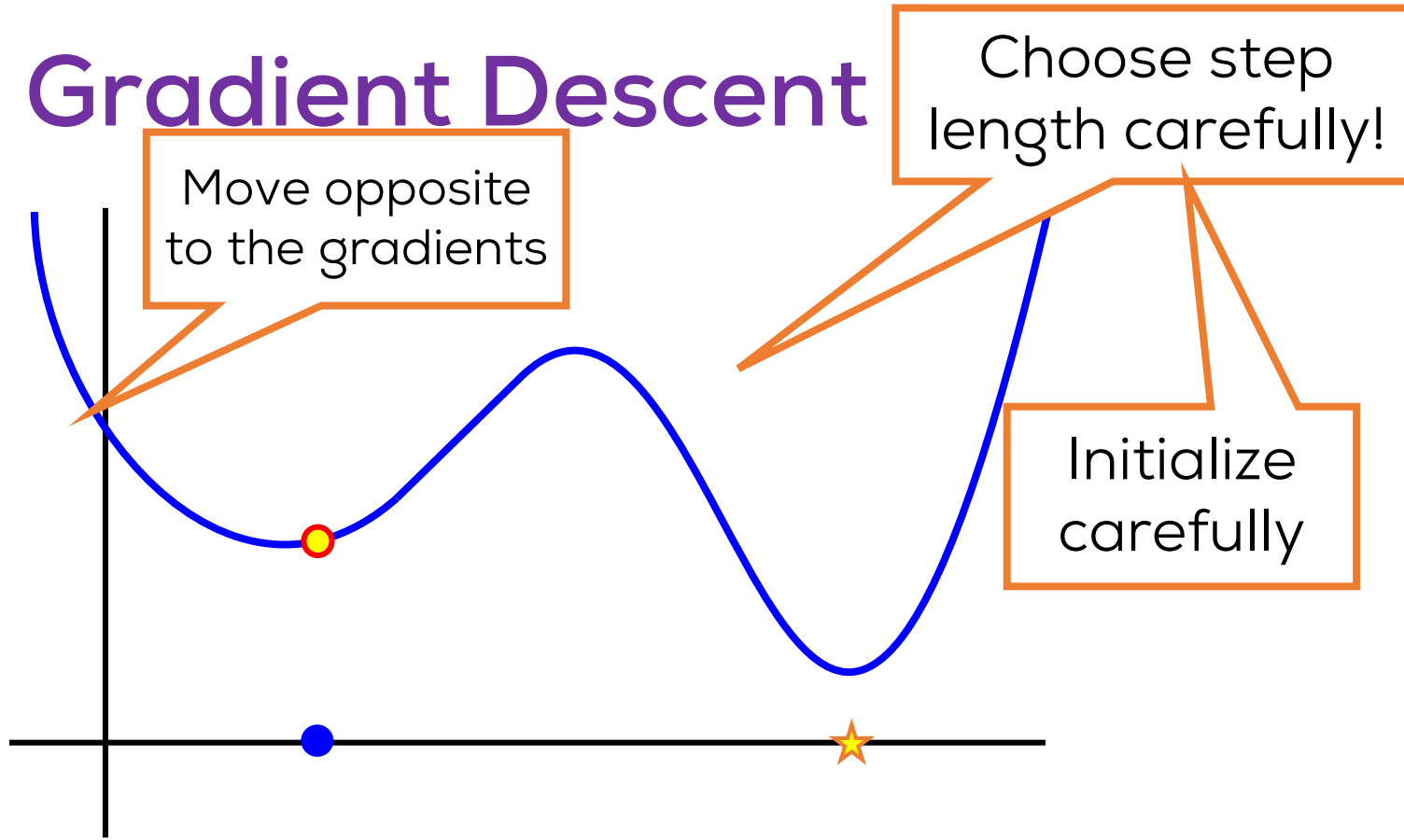
Gradient Descent



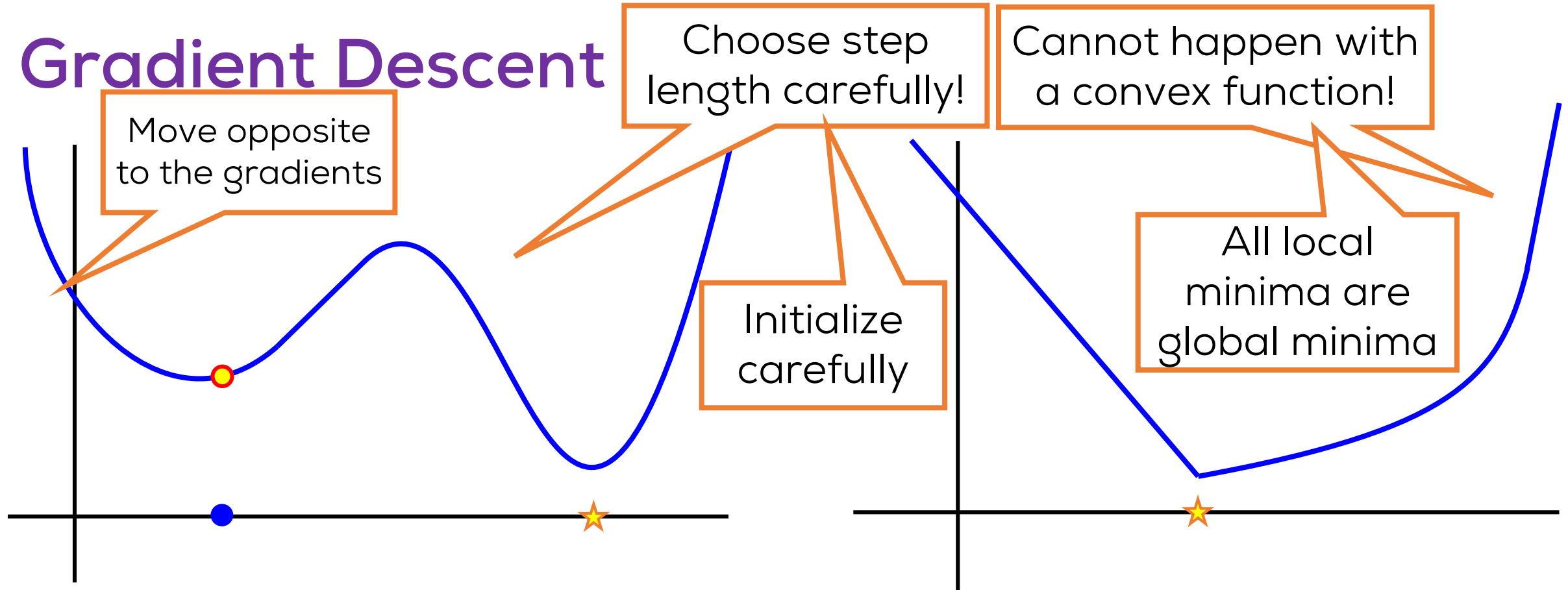
Gradient Descent



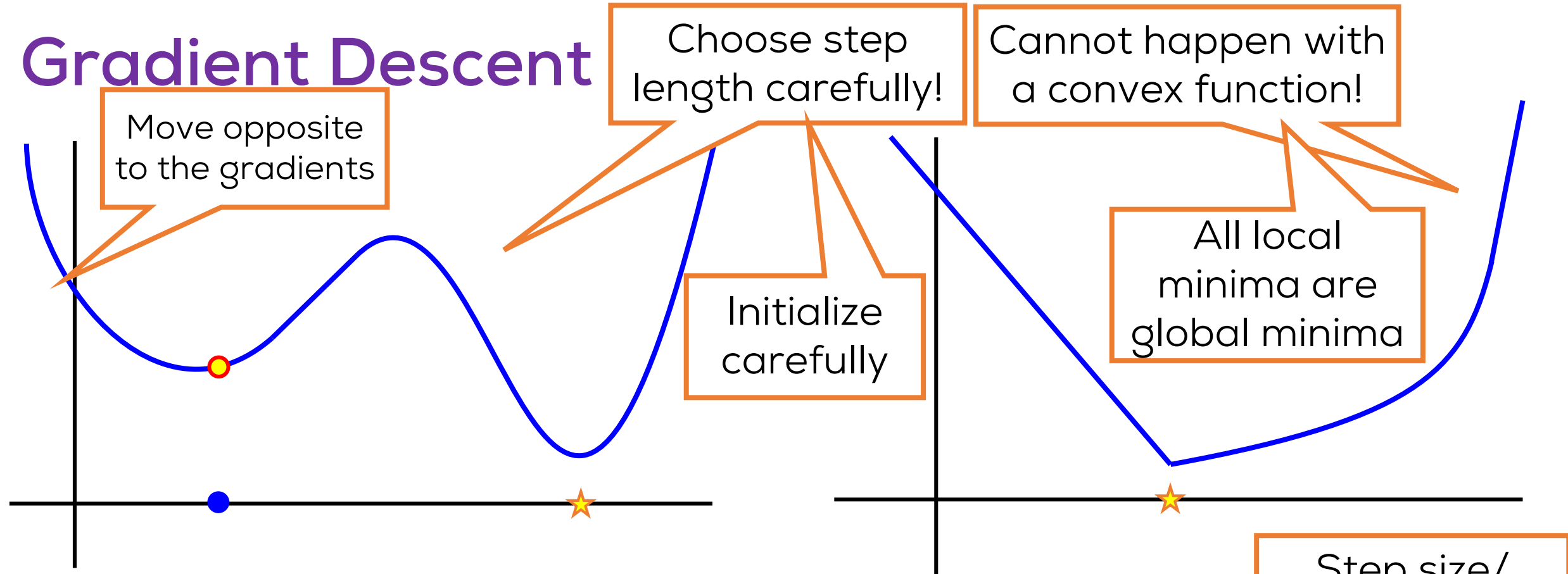
Gradient Descent



Gradient Descent



Gradient Descent



Many convergence criteria – length of gradient, performance threshold, dual criteria

GRADIENT DESCENT

1. Initialize \mathbf{w}^0
2. Update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta_t \cdot \mathbf{g}^t$
3. Repeat until convergence

Step size/
learning rate

Tuned carefully
CV, Adam,
Adagrad

Please give your Feedback

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