NJUAI 南京大学 人工智能学院
SCHOOL OF ARTIFICIAL INTELLIGENCE, NANJING UNIVERSITY

# Lecture 11: Learning 2

# Previously...

Learning

- Decision tree learning
- Nearest Neighbors
- Naive Bayes

Question:

*why we can learn?*

# Classification

## what can be observed:

on examples/training data:
$\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_m, y_m)\} \quad y_i = f(\boldsymbol{x}_i)$

e.g. training error

$$\epsilon_t = \frac{1}{m} \sum_{i=1}^{m} I(h(\boldsymbol{x}_i) \neq y_i)$$

## what is expected:

over the whole distribution: generalization error

$$\epsilon_g = \mathbb{E}_x[I(h(\boldsymbol{x}) \neq f(\boldsymbol{x}))]$$

$$= \int_{\mathcal{X}} p(x) I(h(\boldsymbol{x}) \neq f(\boldsymbol{x}))] \mathrm{d}x$$

# Regression

## what can be observed:

on examples/training data:
$\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_m, y_m)\} \quad y_i = f(\boldsymbol{x}_i)$

e.g. training mean square error/MSE

$$\epsilon_t = \frac{1}{m} \sum_{i=1}^{m} (h(\boldsymbol{x}_i) - y_i)^2$$

## what is expected:

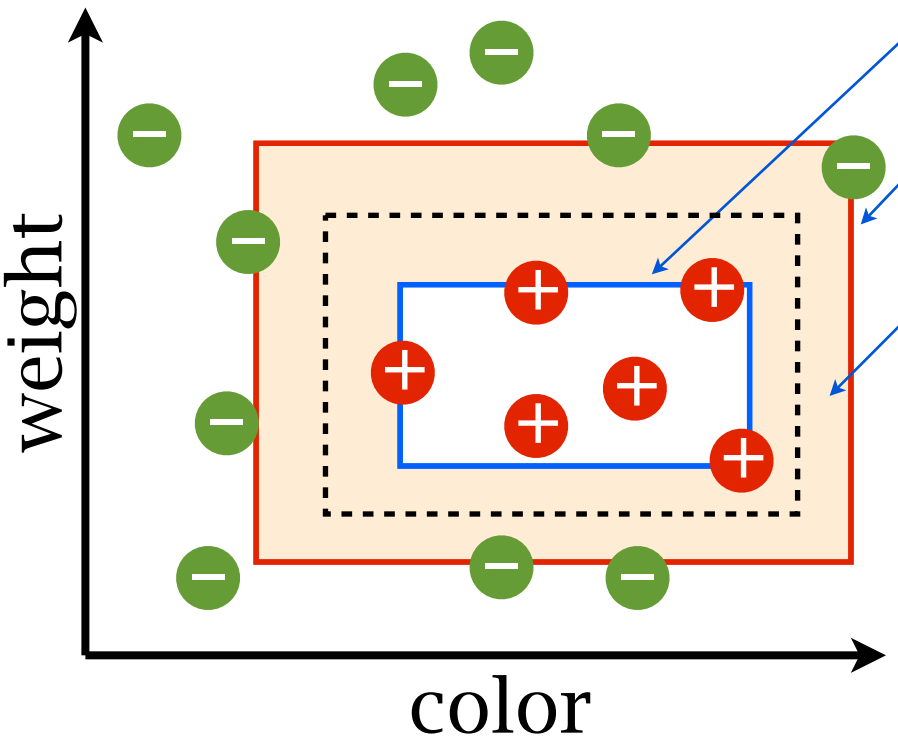
over the whole distribution: generalization MSE

$$\epsilon_g = \mathbb{E}_x (h(\boldsymbol{x}) \neq f(\boldsymbol{x}))^2$$

$$= \int_{\mathcal{X}} p(x)(h(\boldsymbol{x}) - f(\boldsymbol{x}))^2 \mathrm{d}x$$

# The version space algorithm

an abstract view of learning algorithms




S: most specific hypothesis

G: most general hypothesis

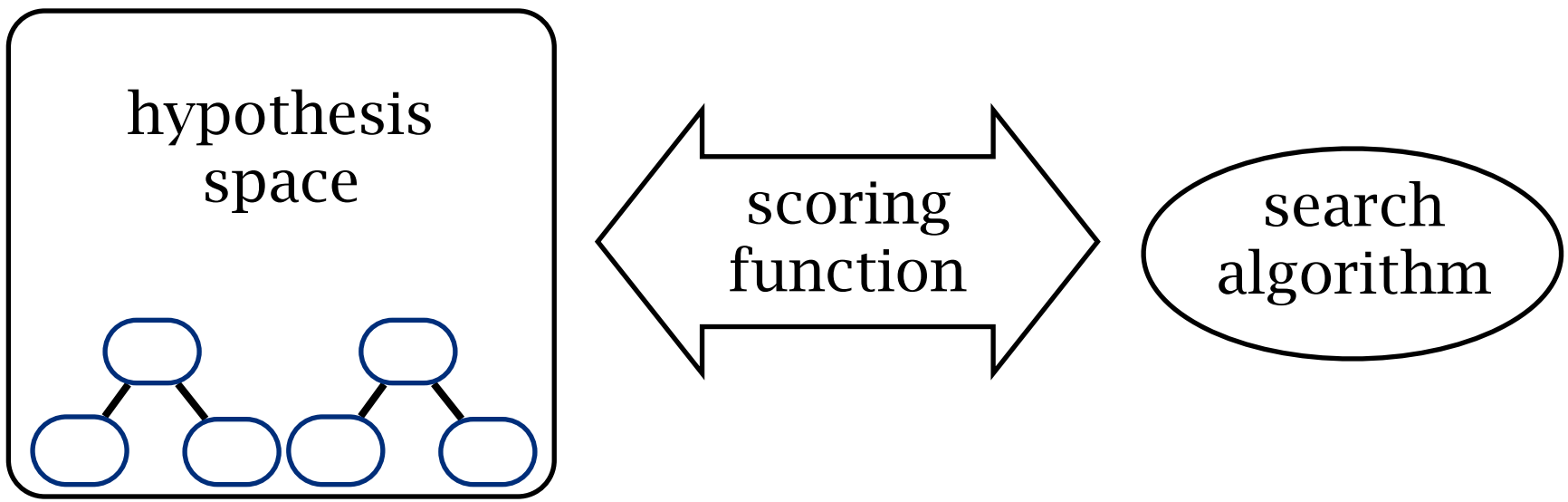
version space: consistent hypotheses [Mitchell, 1997]

*remove the hypothesis that are inconsistent with the data, select a hypothesis according to learner's bias*

# The version space algorithm

an abstract view of learning algorithms

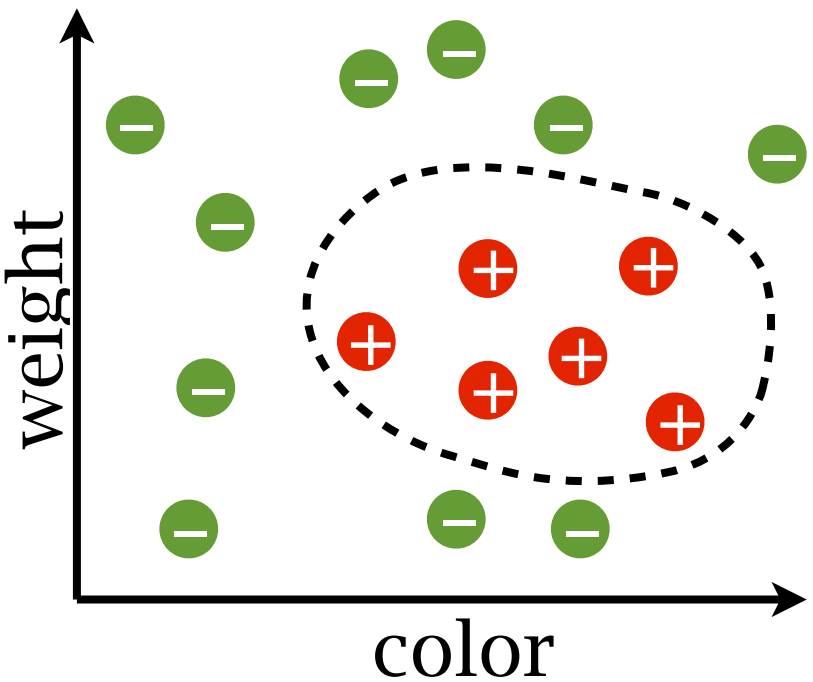
## three components of a learning algorithm

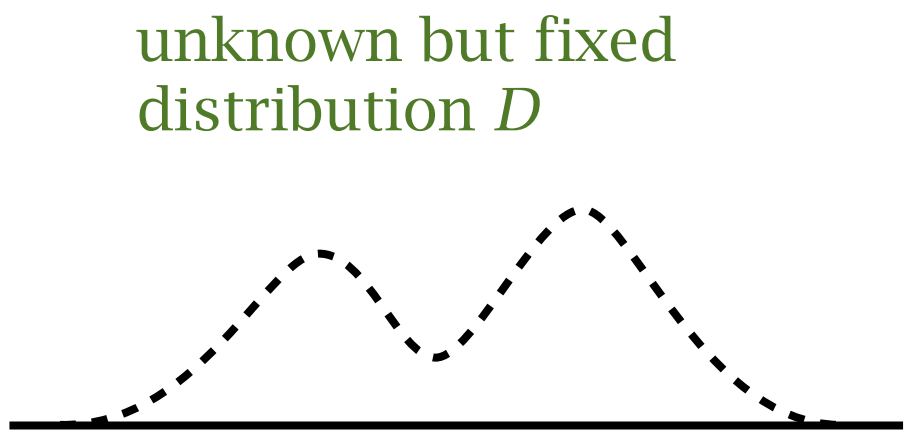


# Theories

The i.i.d. assumption:

all training examples and future (test) examples are drawn *independently* from an *identical distribution*, the label is assigned by a *fixed ground-truth function*



# Bias-variance dilemma

Suppose we have 100 training examples
but there can be different training sets

Start from the expected training MSE:

$$E_D[\epsilon_t] = E_D\left[\frac{1}{m}\sum_{i=1}^{m}(h(\boldsymbol{x}_i) - y_i)^2\right] = \frac{1}{m}\sum_{i=1}^{m} E_D\left[(h(\boldsymbol{x}_i) - y_i)^2\right]$$

(assume no noise)

$$\begin{aligned}
&E_D\left[(h(\boldsymbol{x}) - f(\boldsymbol{x}))^2\right] \\
&= E_D\left[(h(\boldsymbol{x}) - E_D[h(\boldsymbol{x})] + E_D[h(\boldsymbol{x})] - f(\boldsymbol{x}))^2\right] \\
&= E_D\left[(h(\boldsymbol{x}) - E_D[h(\boldsymbol{x})])^2\right] + E_D\left[(E_D[h(\boldsymbol{x})] - f(\boldsymbol{x}))^2\right] \\
&\quad + E_D\left[2(h(\boldsymbol{x}) - E_D[h(\boldsymbol{x})])(E_D[h(\boldsymbol{x})] - f(\boldsymbol{x}))\right] \\
&= E_D\left[(h(\boldsymbol{x}) - E_D[h(\boldsymbol{x})])^2\right] + E_D\left[(E_D[h(\boldsymbol{x})] - f(\boldsymbol{x}))^2\right]
\end{aligned}$$

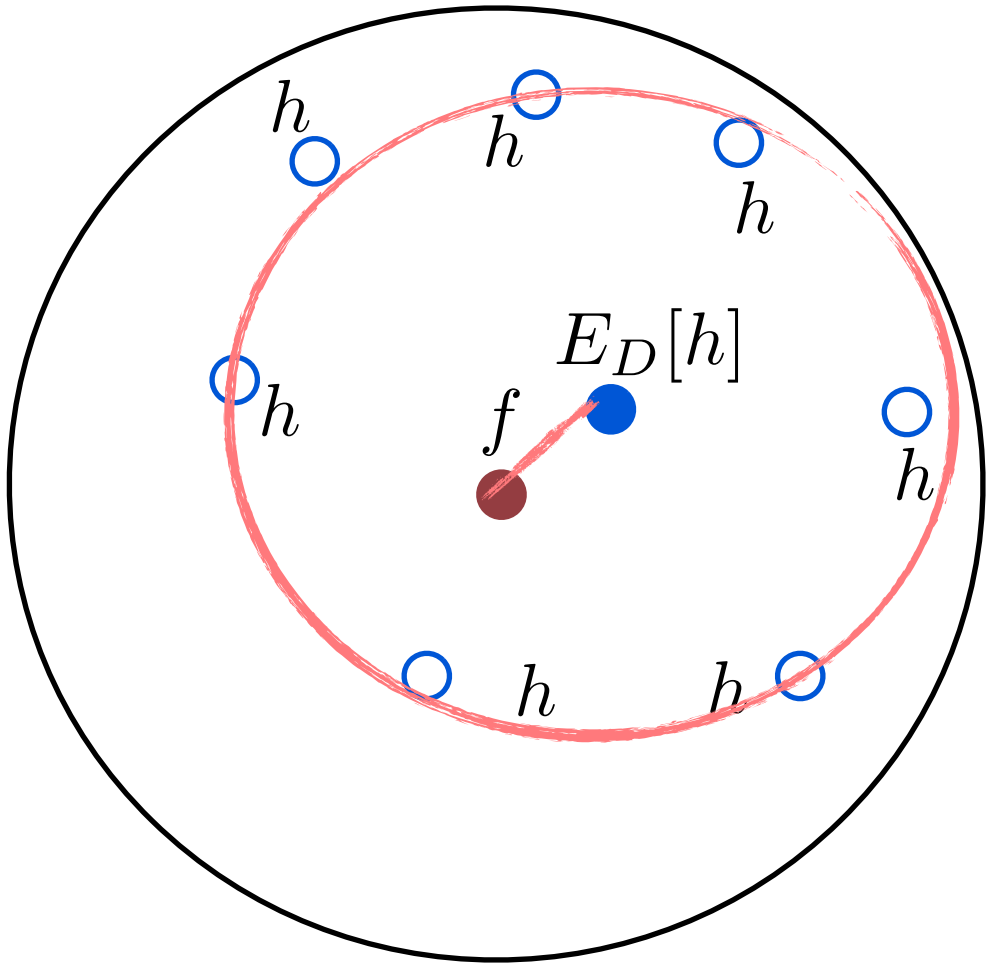
variance bias^2

# Bias-variance dilemma

$E_D\left[(h(\boldsymbol{x}) - E_D[h(\boldsymbol{x})])^2\right]$ variance

$E_D\left[(E_D[h(\boldsymbol{x})] - f(\boldsymbol{x}))^2\right]$ bias^2

larger hypothesis space
=>
lower bias
but higher variance



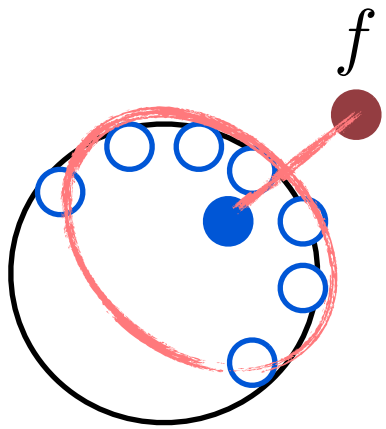

hypothesis space

# Bias-variance dilemma

$E_D\left[(h(\boldsymbol{x}) - E_D[h(\boldsymbol{x})])^2\right]$ $E_D\left[(E_D[h(\boldsymbol{x})] - f(\boldsymbol{x}))^2\right]$

variance bias^2

smaller hypothesis space
=>
smaller variance
but higher bias

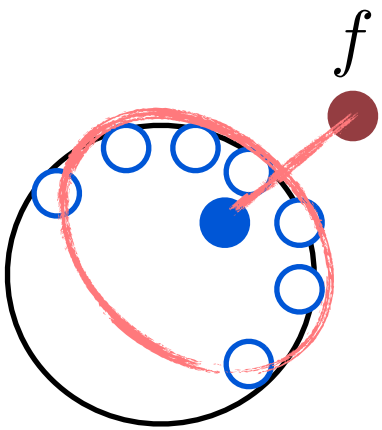


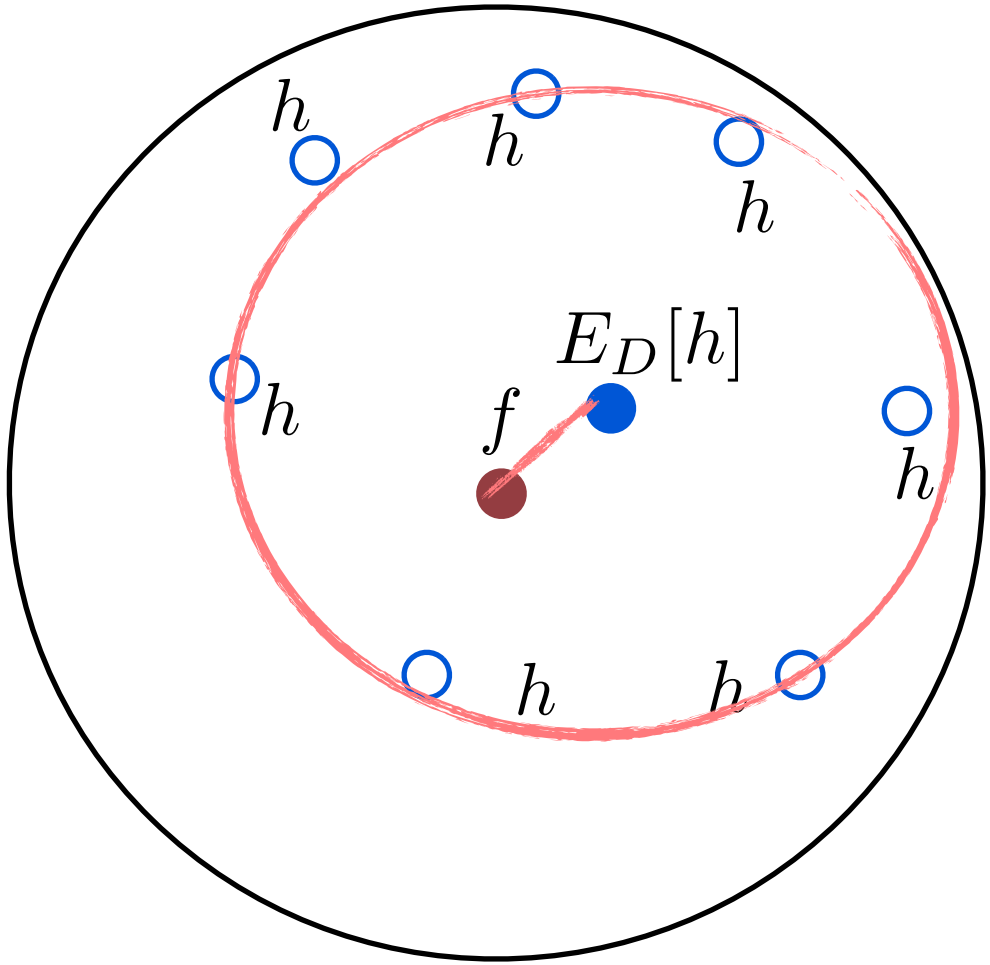

hypothesis space

# Bias-variance dilemma

$E_D\left[(h(\boldsymbol{x}) - E_D[h(\boldsymbol{x})])^2\right]$ variance

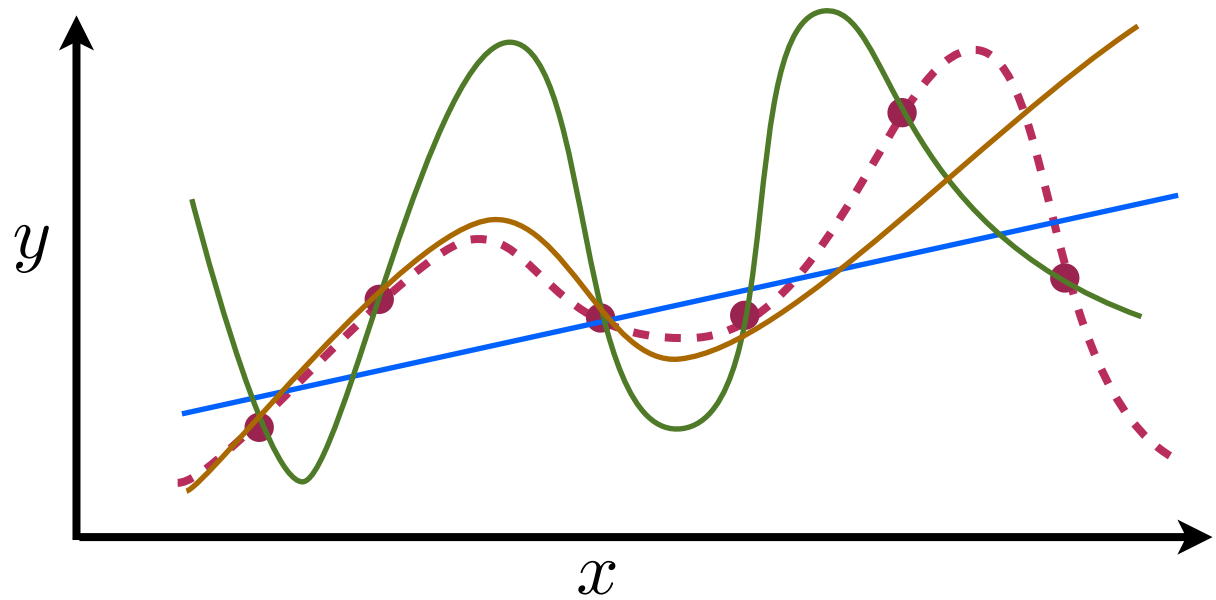
$E_D\left[(E_D[h(\boldsymbol{x})] - f(\boldsymbol{x}))^2\right]$ bias^2



# Overfitting and underfitting

training error v.s. hypothesis space size




linear functions: high training error, small space

$$\{y = a + bx \mid a, b \in \mathbb{R}\}$$

higher polynomials: moderate training error, moderate space

$$\{y = a + bx + cx^2 + dx^3 \mid a, b, c, d \in \mathbb{R}\}$$

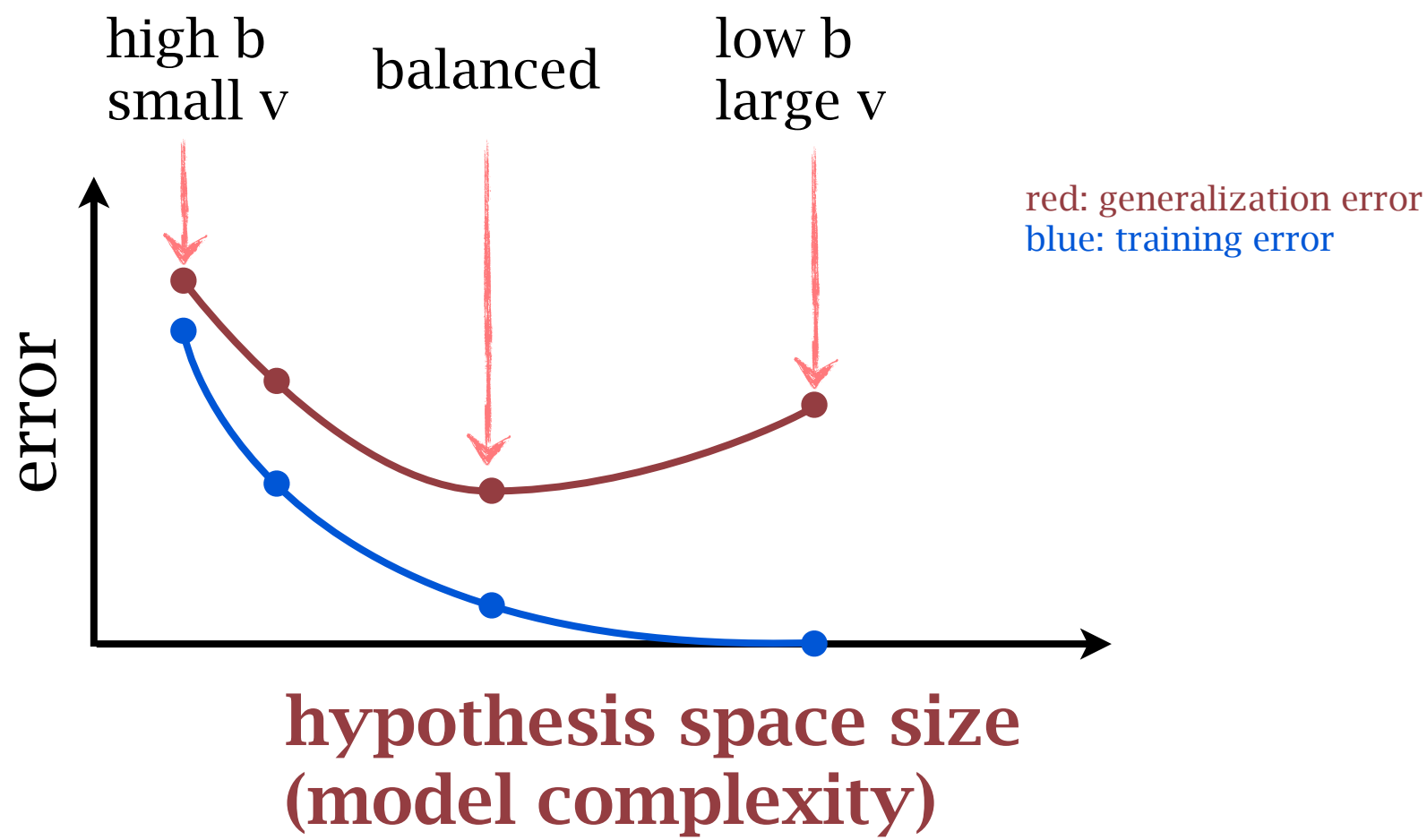
even higher order: no training error, large space

$$\{y = a + bx + cx^2 + dx^3 + ex^4 + fx^5 \mid a, b, c, d, e, f \in \mathbb{R}\}$$

# Overfitting and bias-variance dilemma

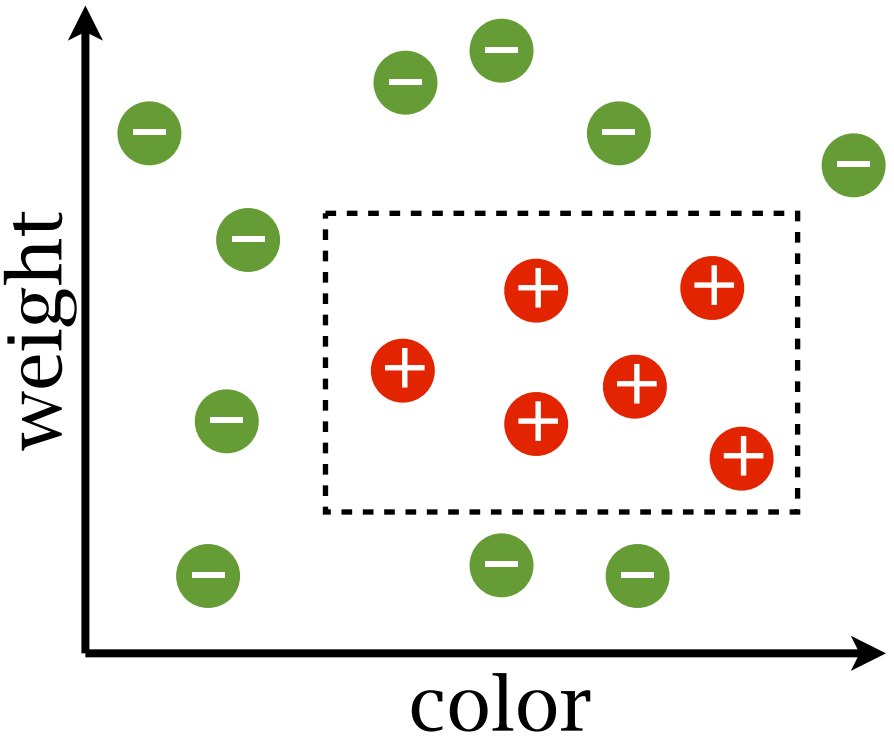
$E_D\left[(h(\boldsymbol{x}) - E_D[h(\boldsymbol{x})])^2\right]$ variance

$E_D\left[(E_D[h(\boldsymbol{x})] - f(\boldsymbol{x}))^2\right]$ bias^2



# Generalization error

assume i.i.d. examples, and the ground-truth hypothesis is a box




the error of picking a consistent hypothesis:

with probability at least $1-\delta$

$$\epsilon_g < \frac{1}{m} \cdot (\ln |\mathcal{H}| + \ln \frac{1}{\delta})$$

smaller generalization error:

- more examples
- smaller hypothesis space

# Generalization error

for one $h$

What is the probability of $\begin{array}{l} h \text{ is consistent} \\ \epsilon_g(h) \geq \epsilon \end{array}$

assume $h$ is ***bad***: $\epsilon_g(h) \geq \epsilon$

$h$ is consistent with 1 example:

$$P \leq 1 - \epsilon$$

$h$ is consistent with $\boldsymbol{m}$ example:

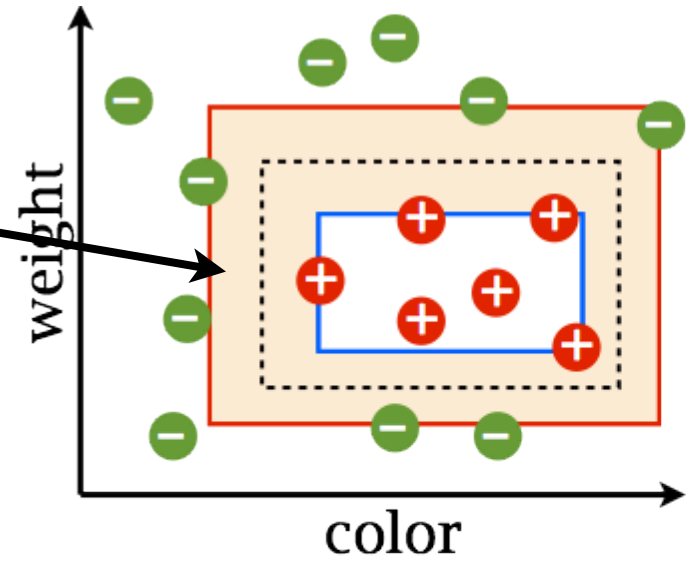
$$P \leq (1 - \epsilon)^m$$

# Generalization error

$h$ is consistent with $\boldsymbol{m}$ example:

$$P \leq (1-\epsilon)^m$$

There are $\boldsymbol{k}$ consistent hypotheses




Probability of choosing a bad one:

$h_1$ is chosen and $h_1$ is bad $P \leq (1-\epsilon)^m$

$h_2$ is chosen and $h_2$ is bad $P \leq (1-\epsilon)^m$

...

$h_k$ is chosen and $h_k$ is bad $P \leq (1-\epsilon)^m$

overall:

$\exists h$: $h$ can be chosen (consistent) but is bad

# Generalization error

$h_1$ is chosen and $h_1$ is bad $P \leq (1-\epsilon)^m$

$h_2$ is chosen and $h_2$ is bad $P \leq (1-\epsilon)^m$

...

$h_k$ is chosen and $h_k$ is bad $P \leq (1-\epsilon)^m$

overall:

$\exists h$: $h$ can be chosen (consistent) but is bad

Union bound: $P(A \cup B) \leq P(A) + P(B)$

$$P(\exists h \text{ is consistent but bad}) \leq k \cdot (1-\epsilon)^m \leq |\mathcal{H}| \cdot (1-\epsilon)^m$$

# Generalization error

$$P(\exists h \text{ is consistent but bad}) \leq k \cdot (1-\epsilon)^m \leq |\mathcal{H}| \cdot (1-\epsilon)^m$$

$$\Downarrow$$

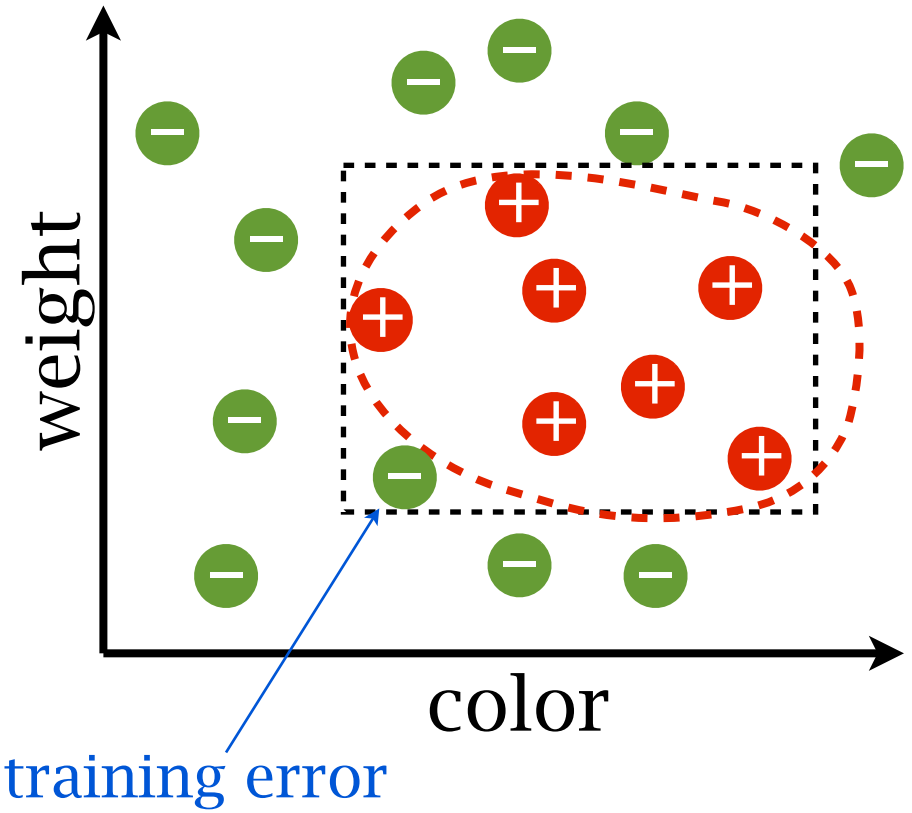
$$P(\epsilon_g \geq \epsilon) \leq \underline{|\mathcal{H}| \cdot (1-\epsilon)^m}$$
$$\delta$$

with probability at least $1-\delta$

$$\epsilon_g < \frac{1}{m} \cdot (\ln |\mathcal{H}| + \ln \frac{1}{\delta})$$

# Inconsistent hypothesis

What if the ground-truth hypothesis is NOT a box: non-zero training error




with probability at least $1-\delta$

$$\epsilon_g < \epsilon_t + \sqrt{\frac{1}{m}(\ln|\mathcal{H}| + \ln\frac{1}{\delta})}$$

smaller generalization error:

- more examples
- smaller hypothesis space
- **smaller training error**

# Hoeffding's inequality

$X$ be an i.i.d. random variable
$X_1, X_2, \ldots, X_m$ be $m$ samples $\qquad X_i \in [a, b]$

$$\frac{1}{m}\sum_{i=1}^{m} X_i - \mathbb{E}[X] \leftarrow \text{ difference between sum and expectation}$$

$$P(\frac{1}{m}\sum_{i=1}^{m} X_i - \mathbb{E}[X] \geq \epsilon) \leq \exp\left(-\frac{2\epsilon^2 m}{(b-a)^2}\right)$$

# Generalization error

for one $h$

$$X_i = I(h(x_i) \neq f(x_i)) \in [0, 1]$$

$$\frac{1}{m}\sum_{i=1}^{m} X_i \to \epsilon_t(h) \qquad\qquad \mathbb{E}[X_i] \to \epsilon_g(h)$$

$$P(\epsilon_t(h) - \epsilon_g(h) \geq \epsilon) \leq \exp\left(-2\epsilon^2 m\right)$$

$$P(\epsilon_t - \epsilon_g \geq \epsilon)$$

$$\leq P(\exists h \in |\mathcal{H}| : \epsilon_t(h) - \epsilon_g(h) \geq \epsilon) \leq \underline{|\mathcal{H}| \exp\left(-2\epsilon^2 m\right)}$$

$$\delta$$

with probability at least $1 - \delta$

$$\epsilon_g < \epsilon_t + \sqrt{\frac{1}{2m} \cdot (\ln |\mathcal{H}| + \ln \frac{1}{\delta})}$$

# Generalization error: Summary

assume i.i.d. examples
consistent hypothesis case:

$$\text{with probability at least } 1-\delta$$

$$\epsilon_g < \frac{1}{m} \cdot (\ln |\mathcal{H}| + \ln \frac{1}{\delta})$$

inconsistent hypothesis case:

$$\text{with probability at least } 1-\delta$$

$$\epsilon_g < \epsilon_t + \sqrt{\frac{1}{m}(\ln |\mathcal{H}| + \ln \frac{1}{\delta})}$$

generalization error:

number of examples $m$
training error $\epsilon_t$
hypothesis space complexity $\ln |\mathcal{H}|$

# PAC-learning

## Probably approximately correct (PAC):

with probability at least $1-\delta$

$$\epsilon_g < \epsilon_t + \sqrt{\frac{1}{2m} \cdot (\ln |\mathcal{H}| + \ln \frac{1}{\delta})}$$

Leslie Valiant
Turing Award (2010)
EATCS Award (2008)
Knuth Prize (1997)
Nevanlinna Prize (1986)

**PAC-learnable:** [Valiant, 1984]

A concept class $\mathcal{C}$ is PAC-learnable if exists a learning algorithm $A$ such that for all $f \in \mathcal{C}, \epsilon > 0, \delta > 0$ and distribution $D$

$$P_D(\epsilon_g \leq \epsilon) \geq 1-\delta$$

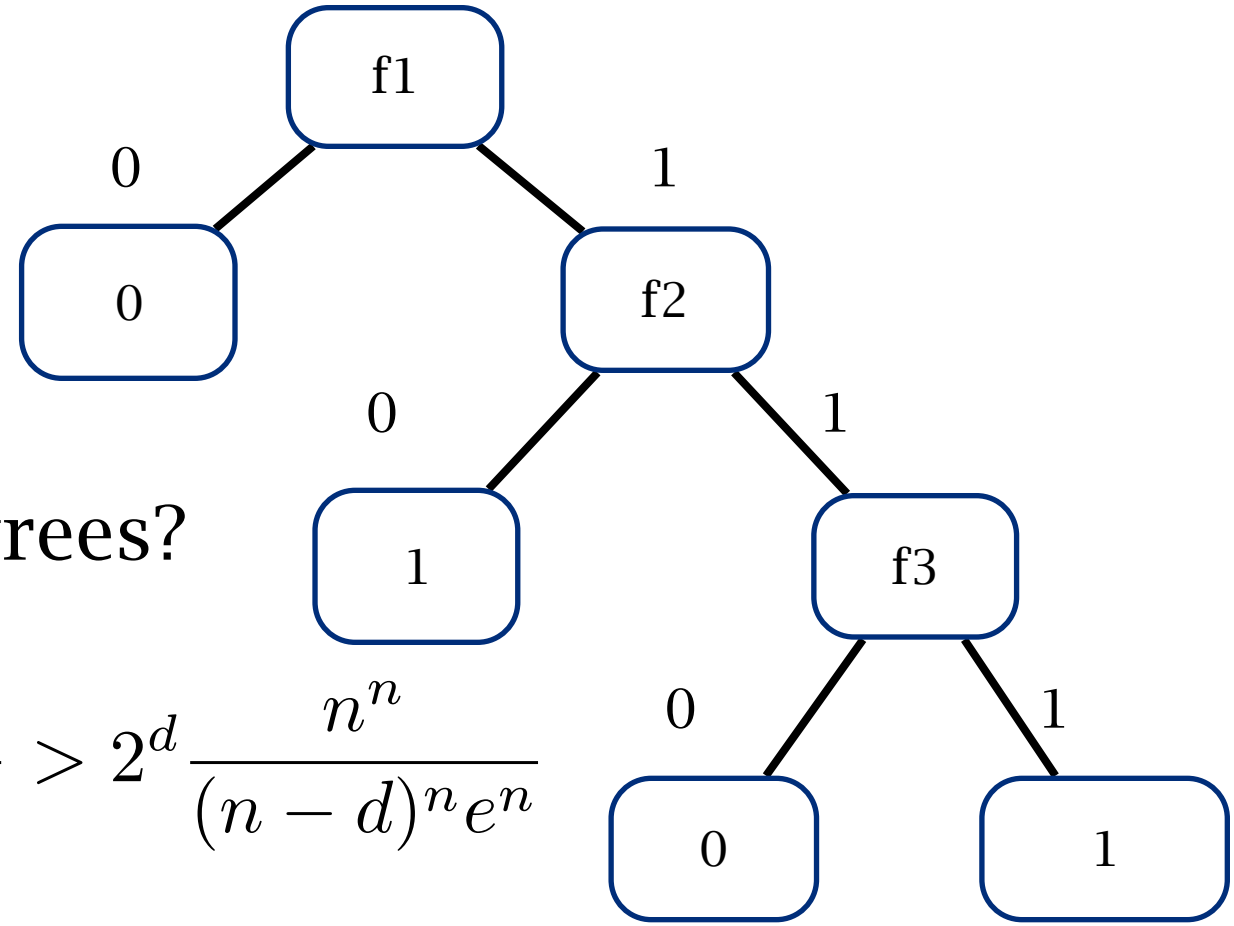
using $m = poly(1/\epsilon, 1/\delta)$ examples and polynomial time.

# Learning algorithms revisit

Decision Tree

# Tree depth and the possibilities

features: $n$
feature type: binary
depth: $d<n$




How many different trees?

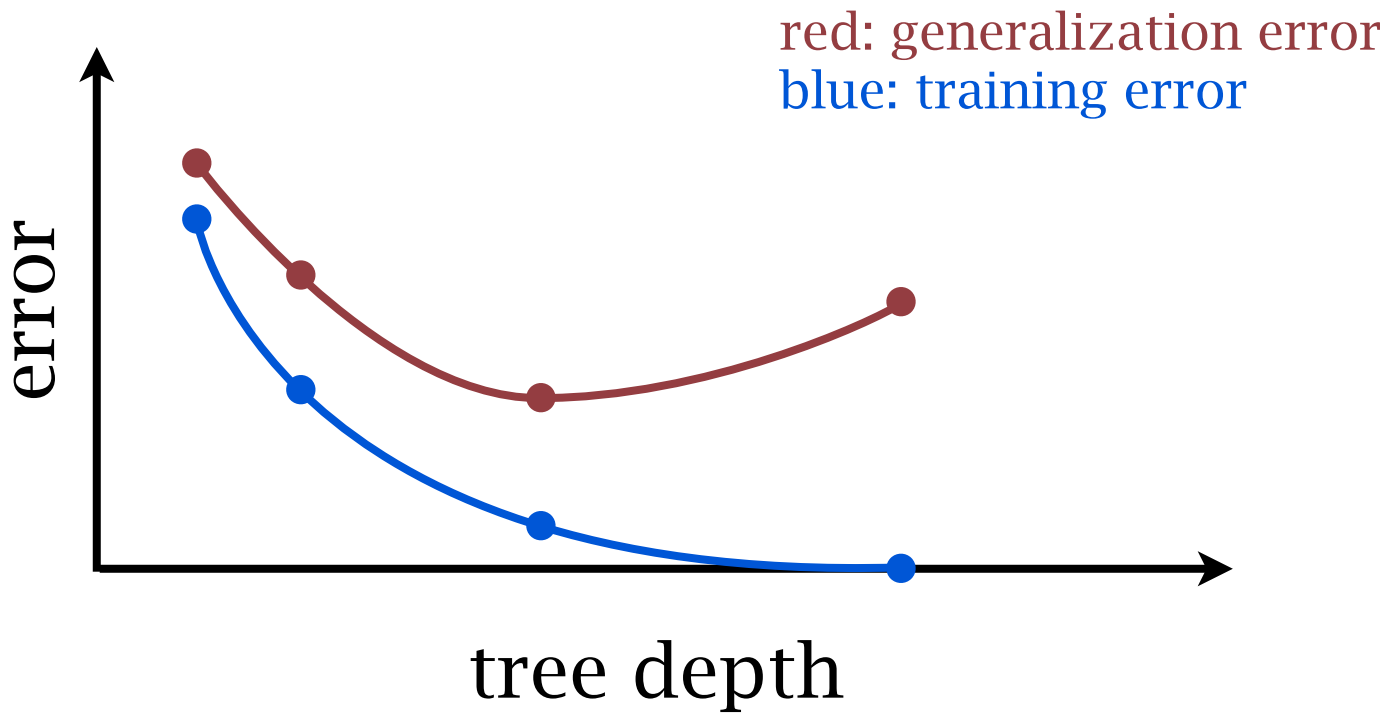
one-branch: $2^d \frac{n!}{(n-d)!} > 2^d \frac{n^n}{(n-d)^n e^n}$

full-tree: $2^{2^d} \prod_{i=0}^{d-1} \frac{(n-i)!}{(n-d-i)!}$

the possibility of trees grows very fast with $d$

# The overfitting phenomena

-- the divergence between infinite and finite samples



# Pruning

To make decision tree less complex

**Pre-pruning**: early stop

- minimum data in leaf
- maximum depth
- maximum accuracy

**Post-pruning**: prune full grown DT

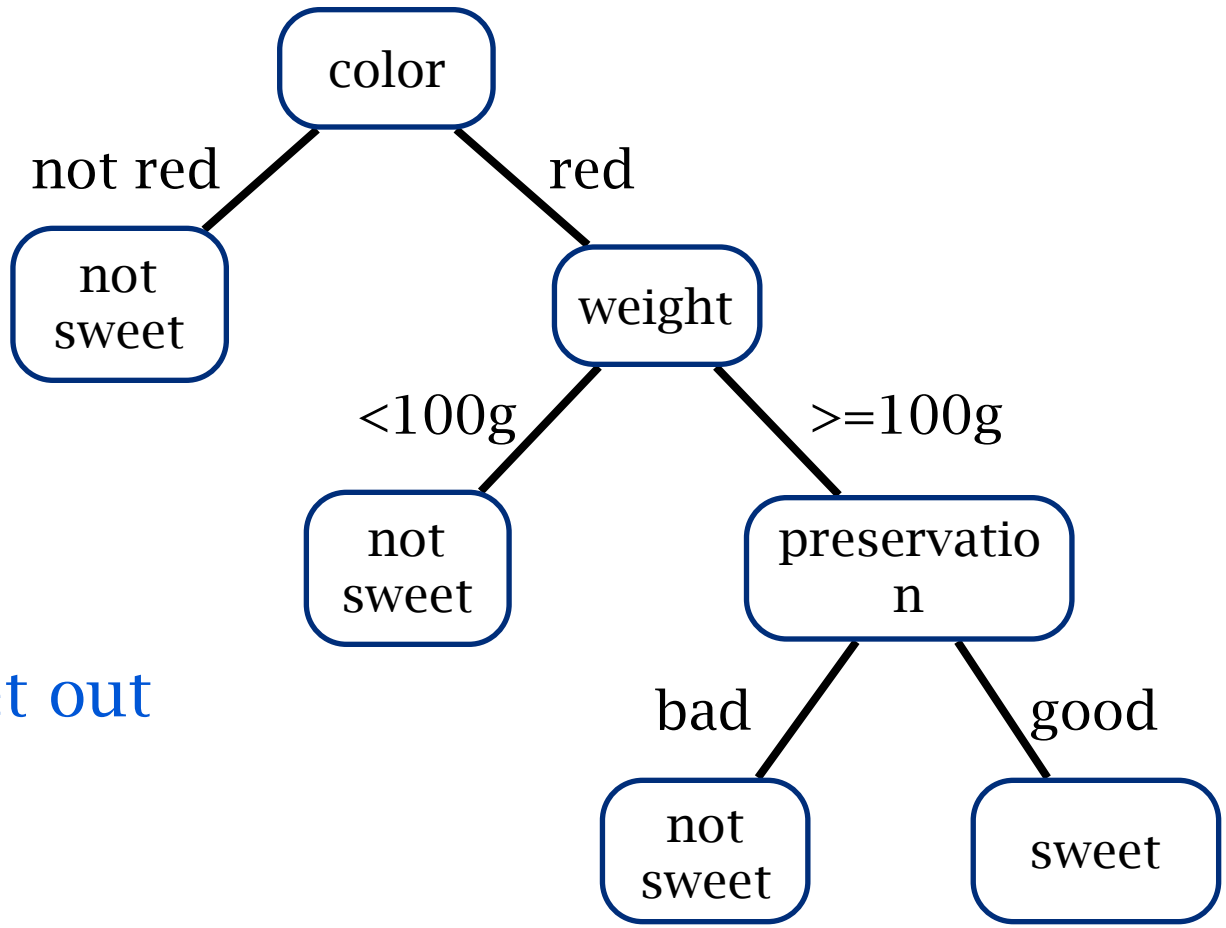
reduced error pruning

# Reduced error pruning

1. Grow a decision tree

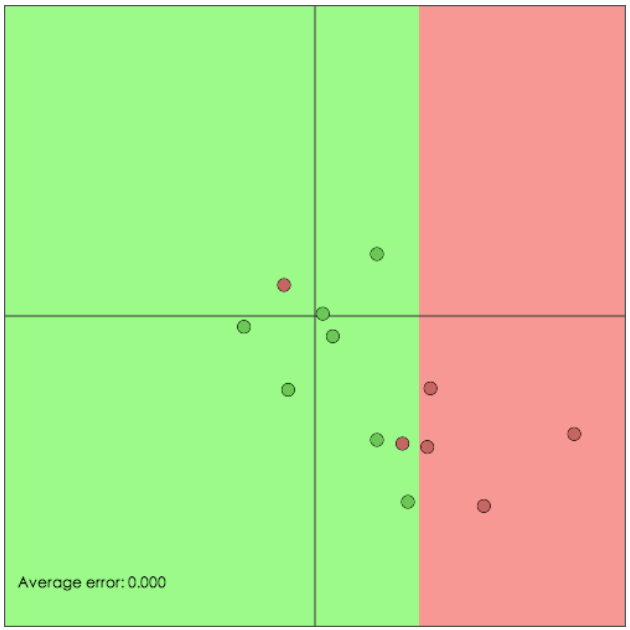
2. For every node starting from the leaves

3. Try to make the node leaf, if does not increase the error, keep as the leaf

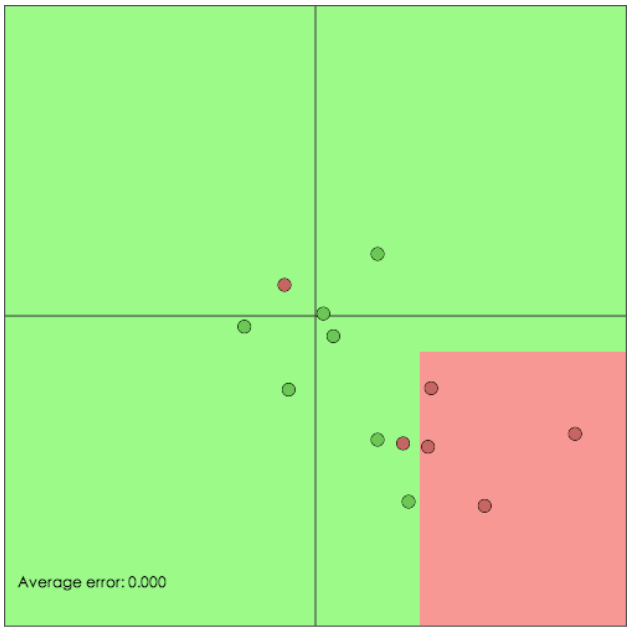



could split a validation set out from the training set to evaluate the error

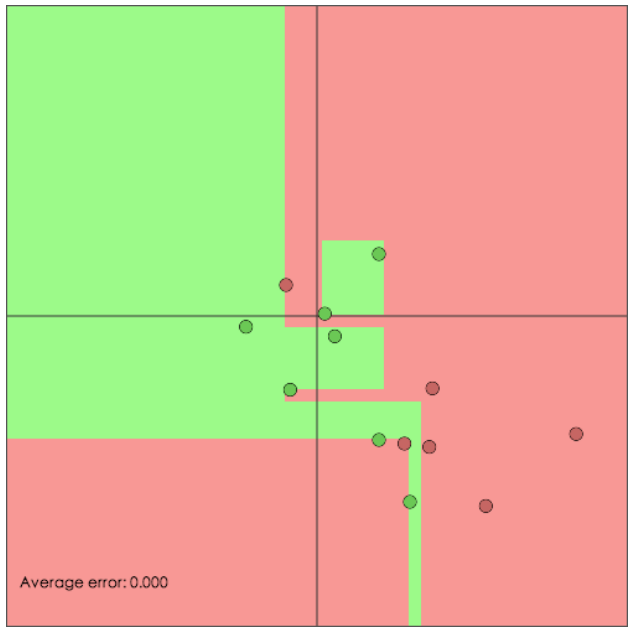
# DT boundary visualization




decision stump




max depth=2




max depth=12

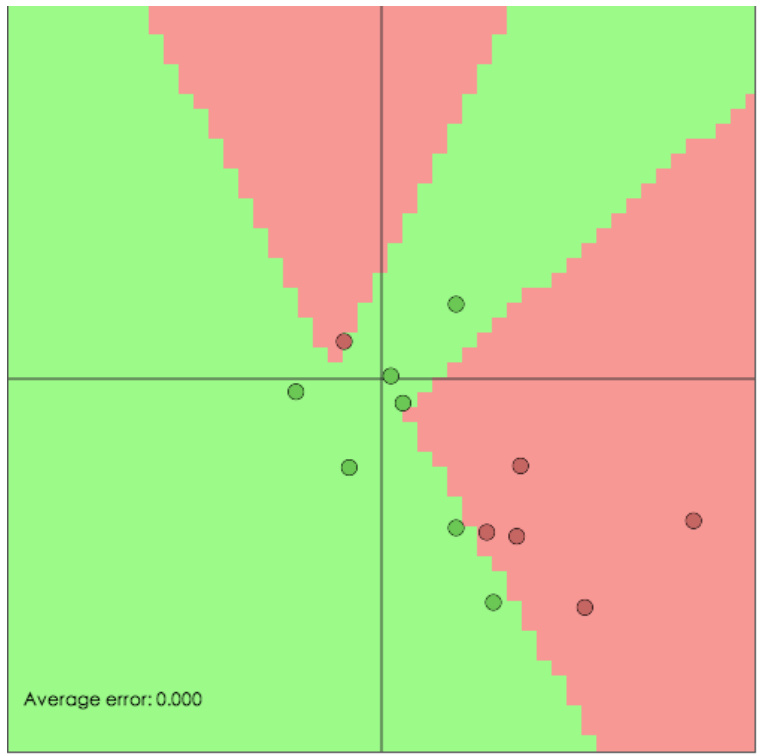
# Oblique decision tree

choose a linear combination in each node:

axis parallel:
$X_1 > 0.5$

oblique:
$0.2\ X_1 + 0.7\ X_2 + 0.1\ X_3 > 0.5$

*was hard to train*



# Linear Models

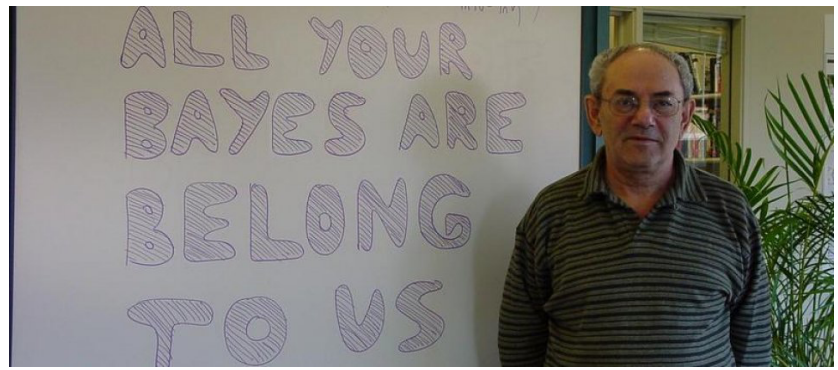
# Linear model

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$$

$$\boldsymbol{w} = \ w_1, w_2, \ldots, w_n \quad b$$

$$w_1 \cdot x_1 + w_2 \cdot x_2 + \ldots + w_n \cdot x_n + b$$

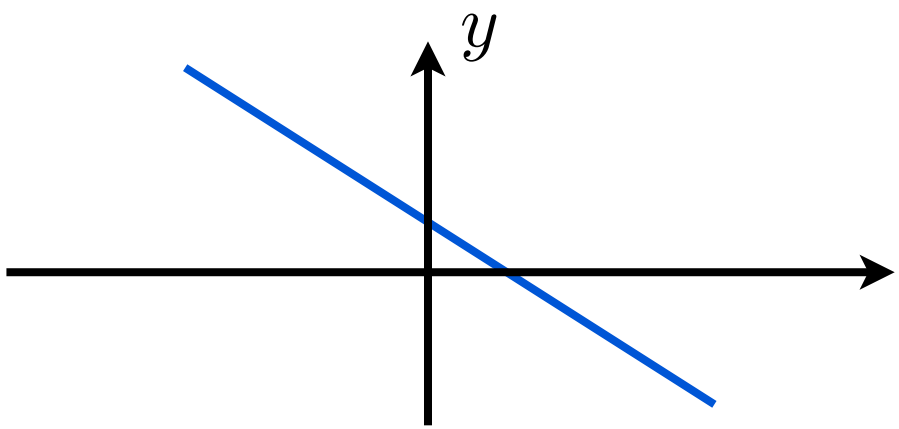
$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$$



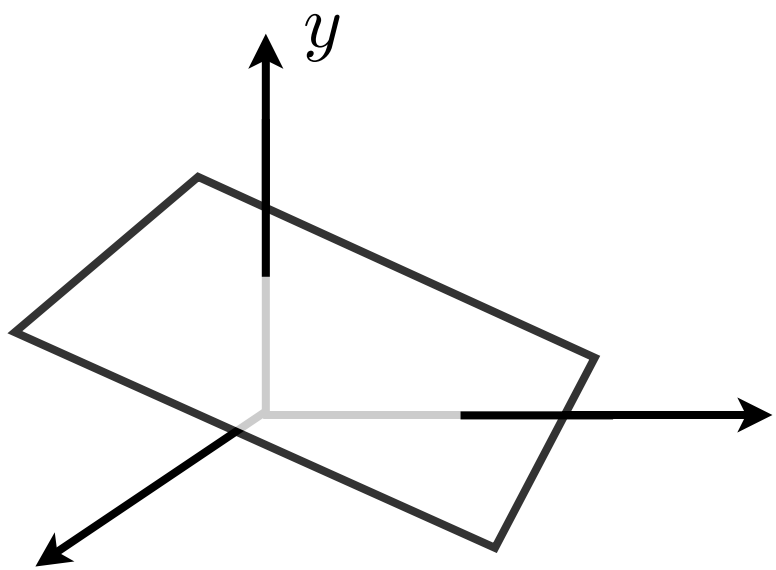

Vladimir Vapnik

# Linear model

$y = ax + b$




$y = w_1 \cdot x_1 + w_2 \cdot x_2 + b$




is the following a linear model?

$y = w_1 \cdot x + w_2 \cdot x^2 + b$

yes, the parameters are linear

# Least square regression

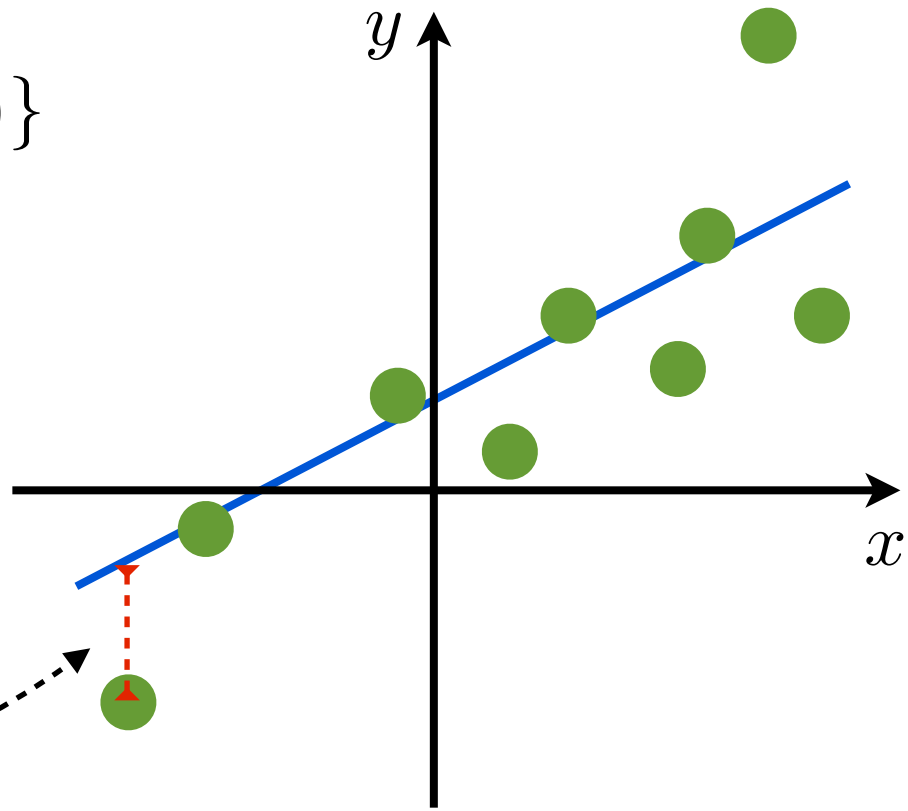
Regression: $y \in \mathbb{R}$

Training data:

$$\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), (\boldsymbol{x}_m, y_m)\}$$

Least square loss:

$$\frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)^2$$



# Least square regression

$$L(\boldsymbol{w}, b) = \frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)^2$$

$$\frac{\partial L(\boldsymbol{w}, b)}{\partial b} = \frac{1}{m}\sum_{i=1}^{m} 2(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i) = 0$$

$$\frac{\partial L(\boldsymbol{w}, b)}{\partial \boldsymbol{w}} = \frac{1}{m}\sum_{i=1}^{m} 2(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)\boldsymbol{x}_i^\top = 0$$

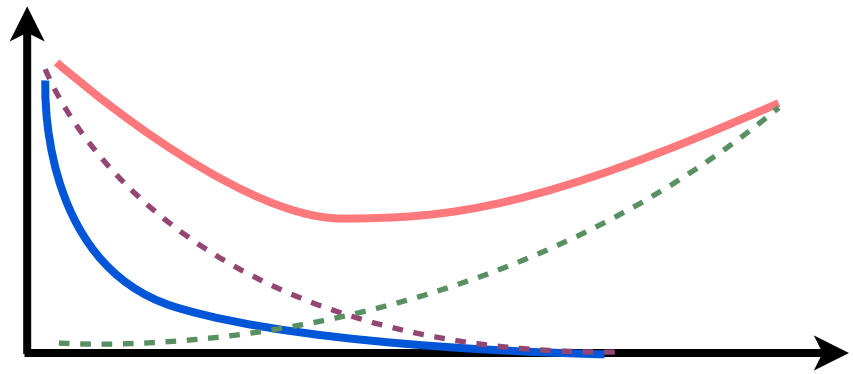
$$b = \frac{1}{m}\sum_{i=1}^{m}(y_i - \boldsymbol{w}^\top \boldsymbol{x}_i) = \bar{y} - \boldsymbol{w}^\top \bar{\boldsymbol{x}}$$

$$\boldsymbol{w} = \left(\frac{1}{m}\sum_{i=1}^{m}\boldsymbol{x}_i\boldsymbol{x}_i^\top - \bar{\boldsymbol{x}}\bar{\boldsymbol{x}}^\top\right)^{-1}\left(\frac{1}{m}\sum_{i=1}^{m}(y_i\boldsymbol{x}_i) - \bar{y}\bar{\boldsymbol{x}}\right)$$

$$= var(\boldsymbol{x})^{-1} cov(\boldsymbol{x}, y) = (X^\top X)^{-1} X^\top Y$$

*closed form solution*

# Complexity of linear models



complexity

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x}$$

possibility of $\boldsymbol{w}$

# Regularization

make hypothesis space small

→ better generalization ability

make numerical analysis stable

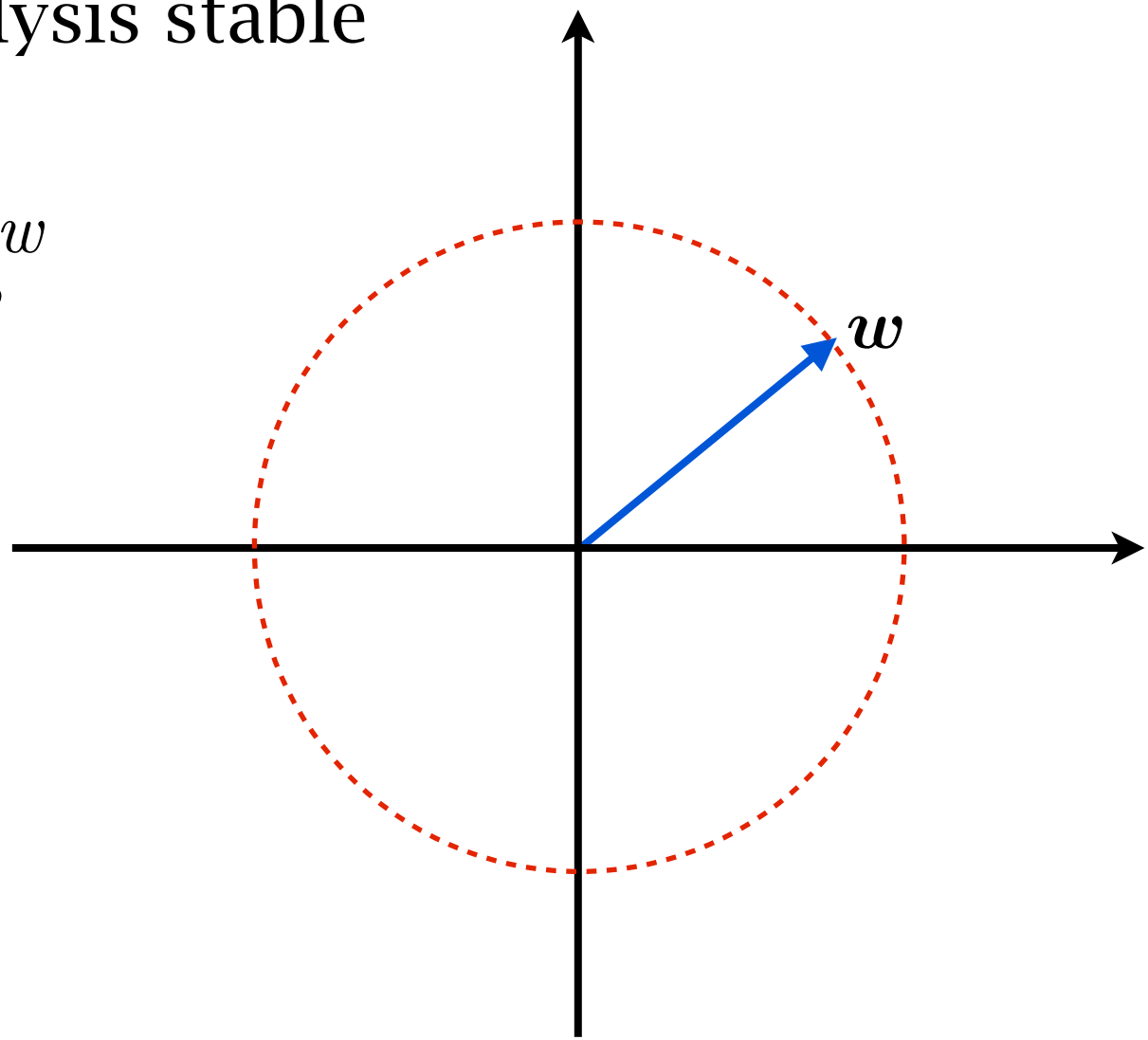
restrict the norm of $w$

$$\|\boldsymbol{w}\|_p = \left( \sum_{i=1}^{n} |w_i|^p \right)^{1/p}$$

$$\|\boldsymbol{w}\|_2 = \sqrt{\sum_{i=1}^{n} w_i^2}$$

$$\|\boldsymbol{w}\|_1 = \sum_{i=1}^{n} |w_i|$$

$$\|\boldsymbol{w}\|_\infty = \max_{i=1,\ldots,n} |w_i|$$



# Ridge regression

Regression: $y \in \mathbb{R}$
Training data:

$$\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), (\boldsymbol{x}_m, y_m)\}$$

objective:

$$\arg\min_{\boldsymbol{w},b} \frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)^2$$

$$s.t. \qquad \|\boldsymbol{w}\|_2 \leq \theta$$

or:

$$\arg\min_{\boldsymbol{w},b} \frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)^2 + \lambda\|\boldsymbol{w}\|_2$$

# Ridge regression

centered data, no bias:

$$\arg\min_{\boldsymbol{w}} \frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^\top \boldsymbol{x}_i - y_i)^2 + \lambda\|\boldsymbol{w}\|_2$$

closed form solution:

$$\boldsymbol{w} = \left(\frac{1}{m}\sum_{i=1}^{m}\boldsymbol{x}_i\boldsymbol{x}_i^\top - \bar{\boldsymbol{x}}\bar{\boldsymbol{x}}^\top + \lambda\boldsymbol{I}\right)^{-1}\left(\frac{1}{m}\sum_{i=1}^{m}(y_i\boldsymbol{x}_i) - \bar{y}\bar{\boldsymbol{x}}\right)$$

$$= (var(\boldsymbol{x}) + \lambda\boldsymbol{I})^{-1} cov(\boldsymbol{x}, y)$$

$$= (X^\top X + \lambda I)^{-1} X^\top Y$$

0

$\boldsymbol{I}$ is the identity matrix

# Least square v.s. ridge regression

$$\boldsymbol{w} = \left(\frac{1}{m}\sum_{i=1}^{m}\boldsymbol{x}_i\boldsymbol{x}_i^\top - \bar{\boldsymbol{x}}\bar{\boldsymbol{x}}^\top\right)^{-1}\left(\frac{1}{m}\sum_{i=1}^{m}(y_i\boldsymbol{x}_i) - \bar{y}\bar{\boldsymbol{x}}\right)$$

$$= var(\boldsymbol{x})^{-1}cov(\boldsymbol{x}, y) = (X^\top X)^{-1}X^\top Y$$

$$\boldsymbol{w} = \left(\frac{1}{m}\sum_{i=1}^{m}\boldsymbol{x}_i\boldsymbol{x}_i^\top - \bar{\boldsymbol{x}}\bar{\boldsymbol{x}}^\top + \lambda\boldsymbol{I}\right)^{-1}\left(\frac{1}{m}\sum_{i=1}^{m}(y_i\boldsymbol{x}_i) - \bar{y}\bar{\boldsymbol{x}}\right)$$

$$= (var(\boldsymbol{x}) + \lambda\boldsymbol{I})^{-1}cov(\boldsymbol{x}, y)$$

$$= (X^\top X + \lambda I)^{-1}X^\top Y$$

stable solution

# Least absolute shrinkage and selection operator (LASSO)

Regression: $y \in \mathbb{R}$

Training data:

$$\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), (\boldsymbol{x}_m, y_m)\}$$

objective:

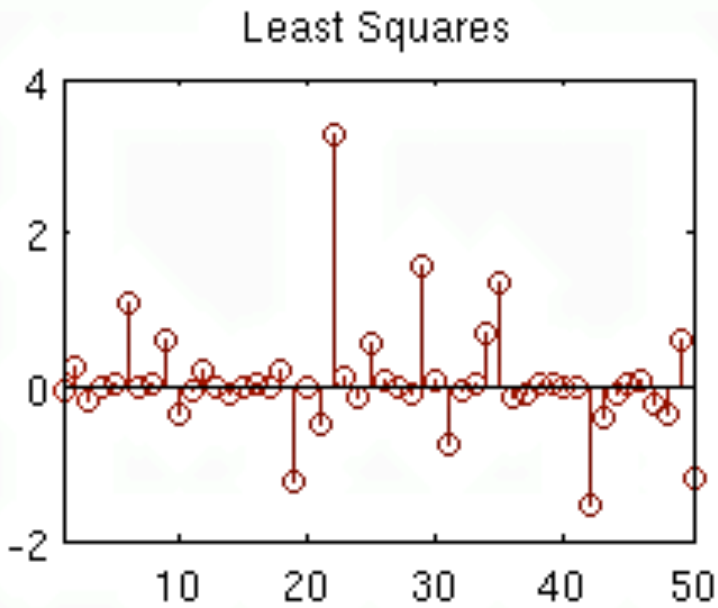
$$\arg\min_{\boldsymbol{w},b} \frac{1}{m} \sum_{i=1}^{m} (\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)^2$$

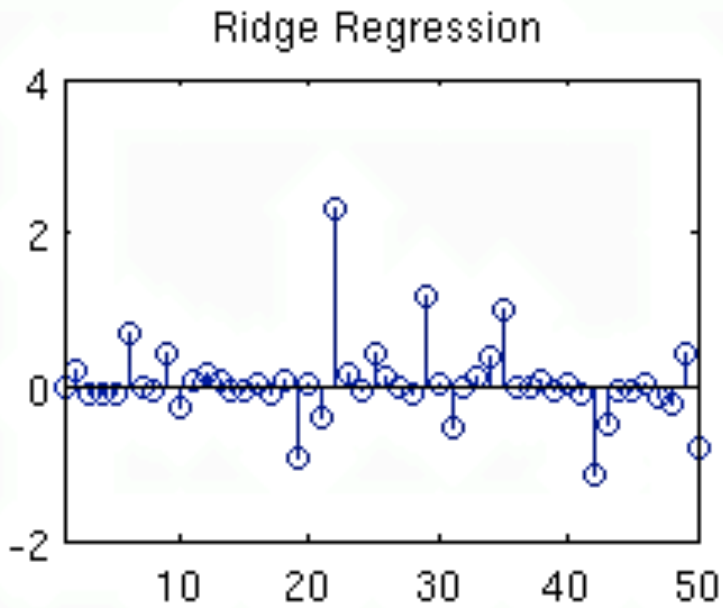
$$s.t. \qquad \|\boldsymbol{w}\|_1 \leq \theta$$

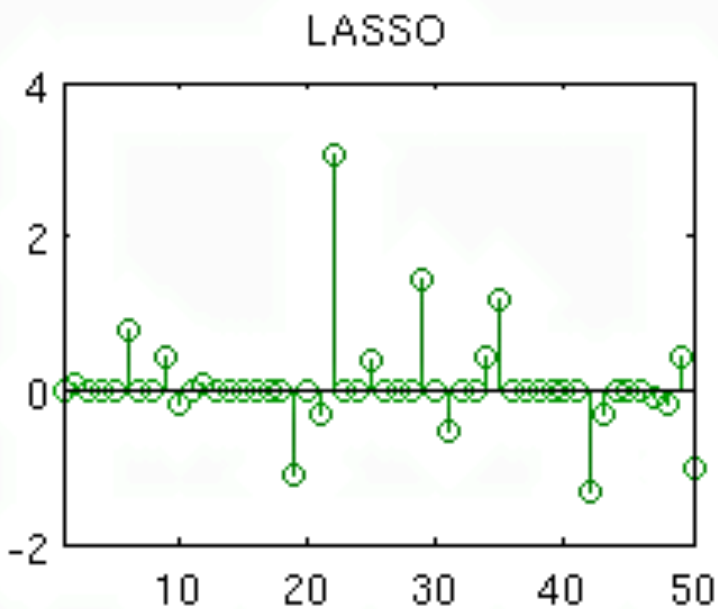
or:

$$\arg\min_{\boldsymbol{w},b} \frac{1}{m} \sum_{i=1}^{m} (\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)^2 + \lambda \|\boldsymbol{w}\|_1$$

# Comparing different regressions




[Pictures from www.cs.ubc.ca/~schmidtm/Software/L1General/examples.html]

# A general framework

objective function:

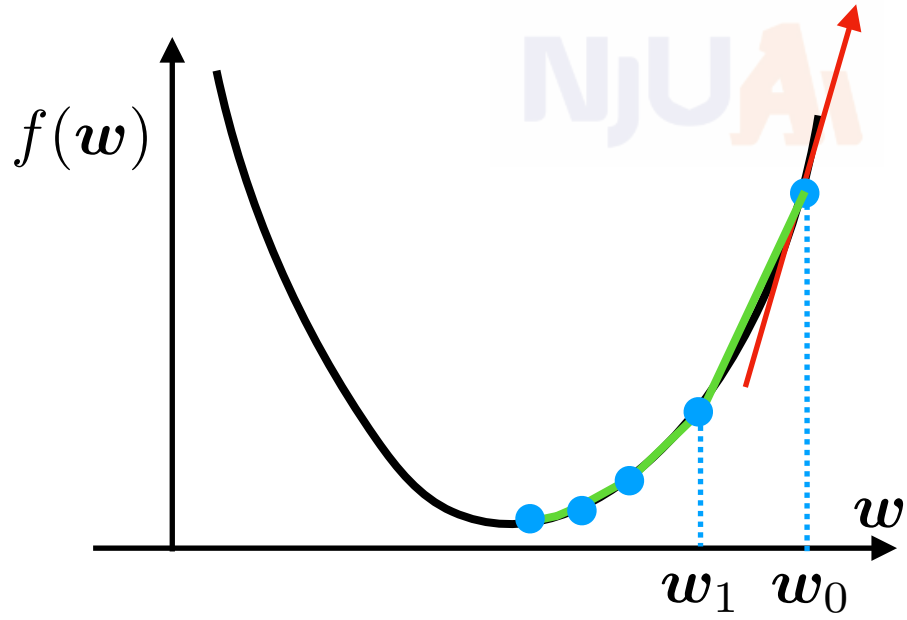
$$\arg\min_{\boldsymbol{w},b} L(\boldsymbol{w},b) + \|\boldsymbol{w}\|_p$$

how to solve the parameters?

a generally applied technique: gradient-descent

# Gradient descent

(steepest descent)



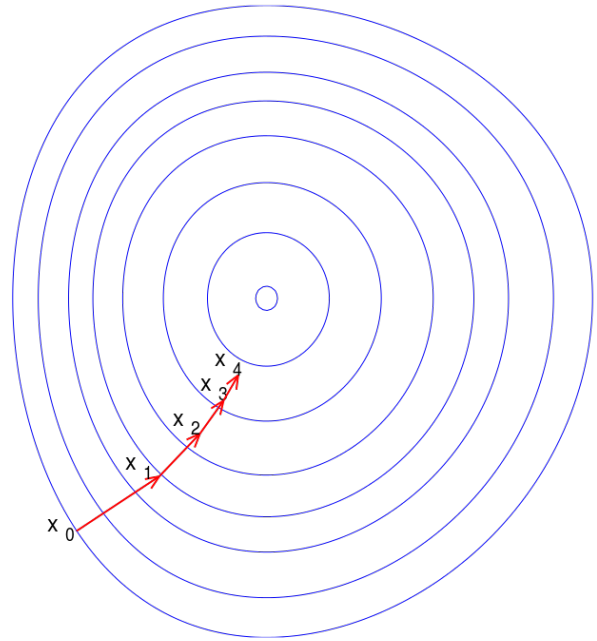

for a differentiable function $f$

$$\arg\min_{\boldsymbol{w}} f(\boldsymbol{w})$$

can be solved by

1. start from an arbitrary initial point $\boldsymbol{w}_0$
2. loop from $t$=0
3. $$\boldsymbol{w}_{t+1} = \boldsymbol{w} - \eta \frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}}$$

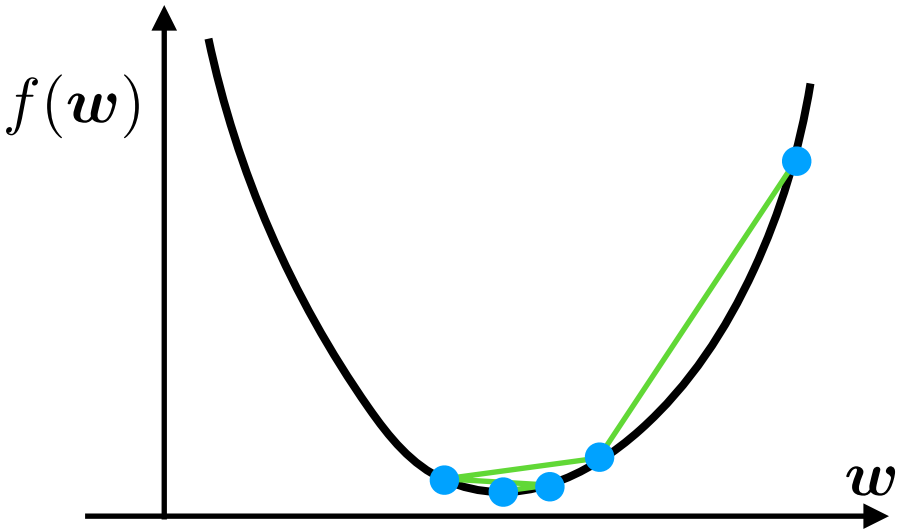
   or $\boldsymbol{w}_{t+1} = \boldsymbol{w} - \eta \nabla_{\boldsymbol{w}} f(\boldsymbol{w})$
4. until convergence $\|\nabla_{\boldsymbol{w}} f(\boldsymbol{w})\| < \epsilon$

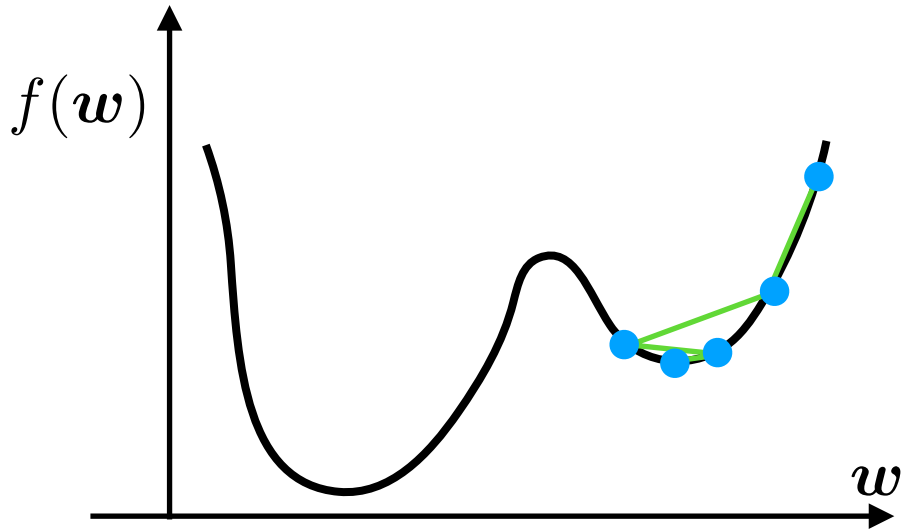



[image from wikipedia]

# Gradient descent

$$\boldsymbol{w}_{t+1} = \boldsymbol{w} - \eta \nabla_{\boldsymbol{w}} f(\boldsymbol{w})$$

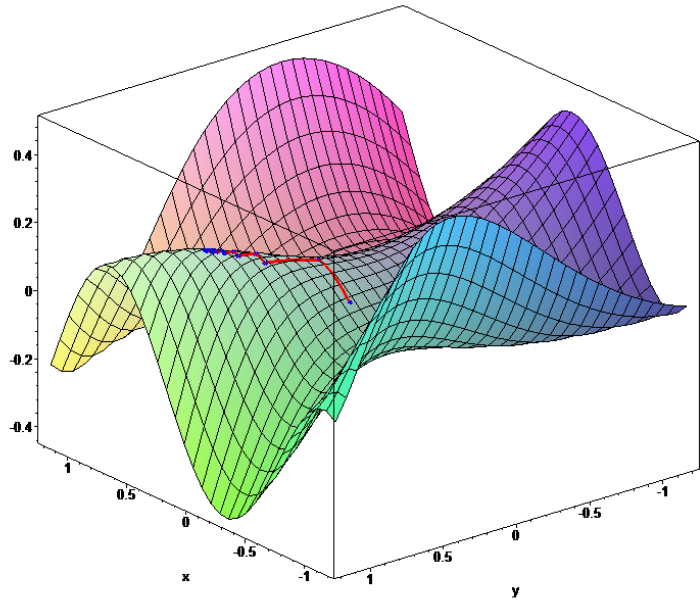



for convex functions:
converge to global optima

$$f(\alpha \boldsymbol{w}_1 + (1-\alpha)\boldsymbol{w}_2)) \geq \alpha f(\boldsymbol{w}_1) + (1-\alpha) f(\boldsymbol{w}_2)$$

for other functions:
converge to stationary points




[image from wikipedia]

# A general framework

objective function:

$$\arg\min_{\boldsymbol{w},b} L(\boldsymbol{w},b) + \|\boldsymbol{w}\|_p$$

how to solve the parameters?

general optimization: gradient descent

$$(\boldsymbol{w},b) - = \eta \frac{\partial(L(\boldsymbol{w},b) + \|\boldsymbol{w}\|_p)}{\partial(\boldsymbol{w},b)}$$

# Linear classifier

model space: $\mathbb{R}^{n+1}$

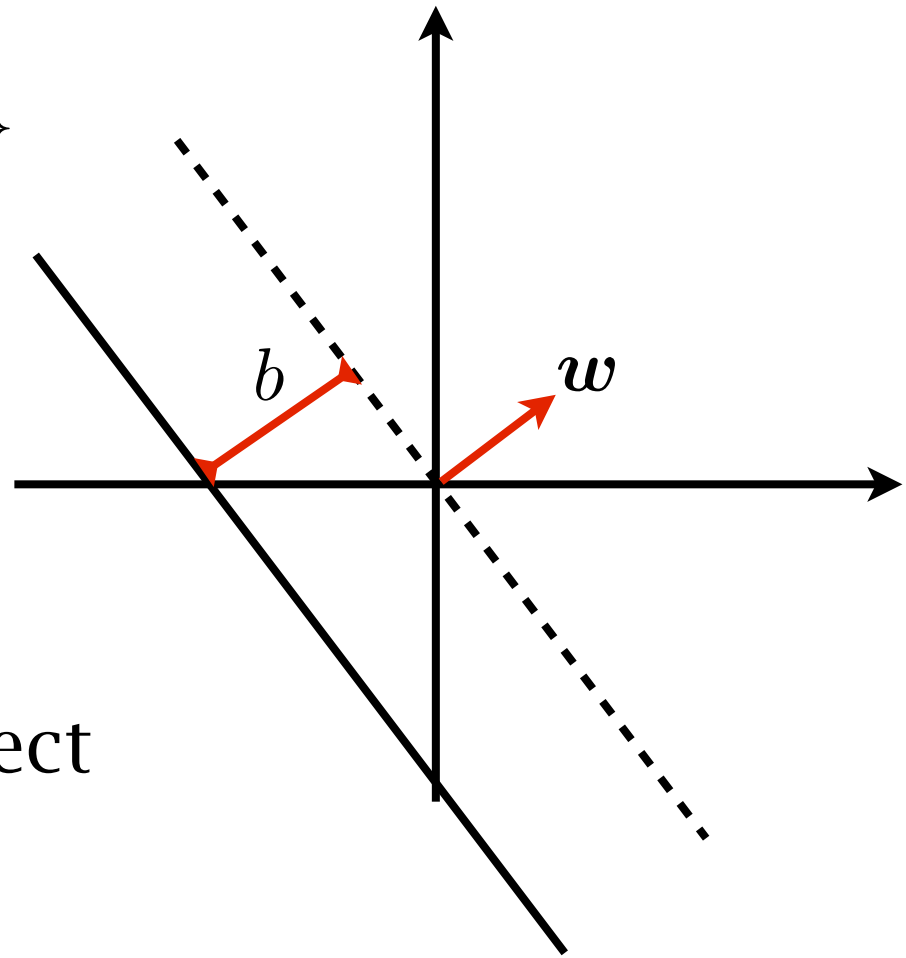
$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$$

for classification $y \in \{-1, +1\}$

we predict an instance by

$$\text{sign}(\boldsymbol{w}^\top \boldsymbol{x} + b)$$

$$= \begin{cases} +1, & \boldsymbol{w}^\top \boldsymbol{x} + b > 0 \\ -1, & \boldsymbol{w}^\top \boldsymbol{x} + b < 0 \\ \text{random}, & otherwise \end{cases}$$




for an example $(\boldsymbol{x}, y)$, a correct prediction means

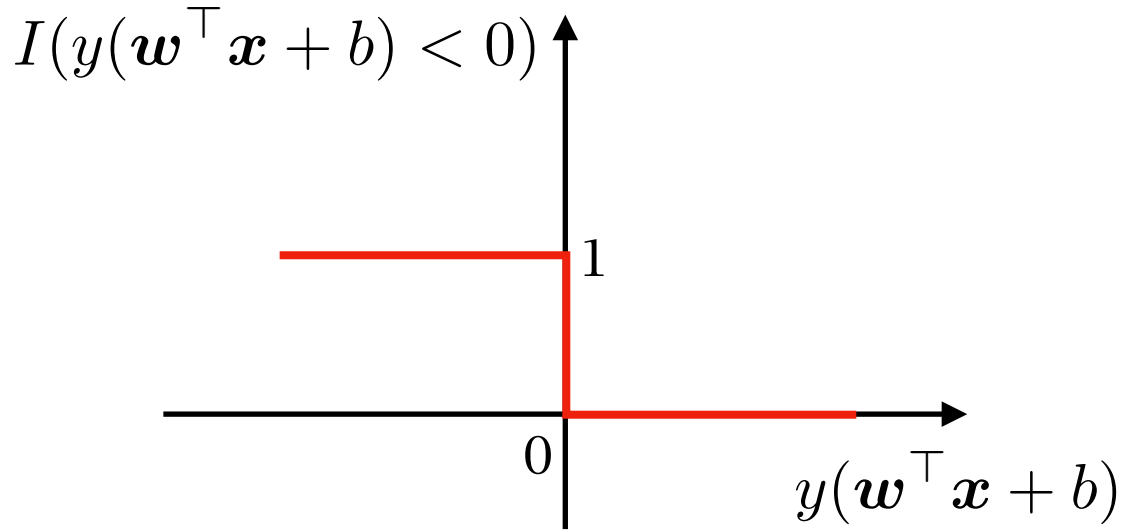
$$y(\boldsymbol{w}^\top \boldsymbol{x} + b) > 0$$

# Ideal classifier

$$\arg\min_{\boldsymbol{w},b} \sum_i I(y(\boldsymbol{w}^\top \boldsymbol{x} + b) \leq 0)$$

non-differentiable

hard to solve by gradient descent



# Prototype

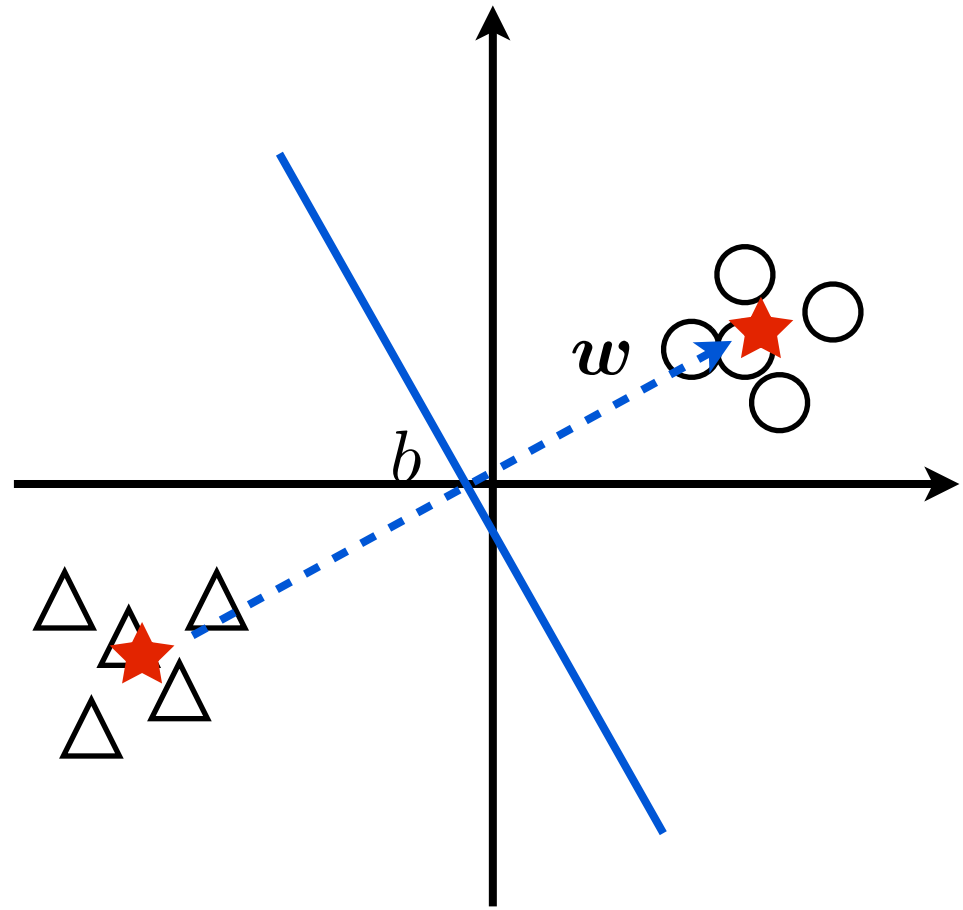
simple, but too restricted

$$\bar{\boldsymbol{x}}^{+} = \frac{1}{\sum_{i:y_i=+1} 1} \sum_{i:y_i=+1} \boldsymbol{x}_i$$

$$\bar{\boldsymbol{x}}^{-} = \frac{1}{\sum_{i:y_i=-1} 1} \sum_{i:y_i=-1} \boldsymbol{x}_i$$

$$\boldsymbol{w} = \bar{\boldsymbol{x}}^{+} - \bar{\boldsymbol{x}}^{-}$$

$$b = -\boldsymbol{w}^{\top} \cdot \frac{\bar{\boldsymbol{x}}^{+} + \bar{\boldsymbol{x}}^{-}}{2}$$



# Perceptron

perception loss

$$\arg\min_{\boldsymbol{w},b} \sum_i \max\{-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), 0\}$$

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$$

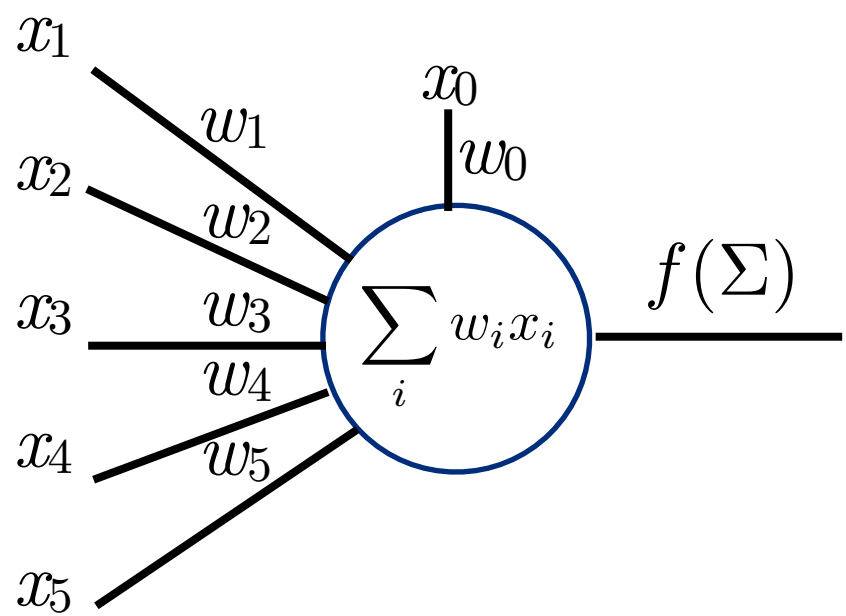
gradient ascent

$$\frac{\partial y \boldsymbol{w}^\top \boldsymbol{x}}{\partial \boldsymbol{w}} = y\boldsymbol{x}$$

feed training examples one by one

1. $\boldsymbol{w} = 0$
2. for each example $(\boldsymbol{x}, y)$
   if $\text{sign}(y\boldsymbol{w}^\top \boldsymbol{x}) < 0$

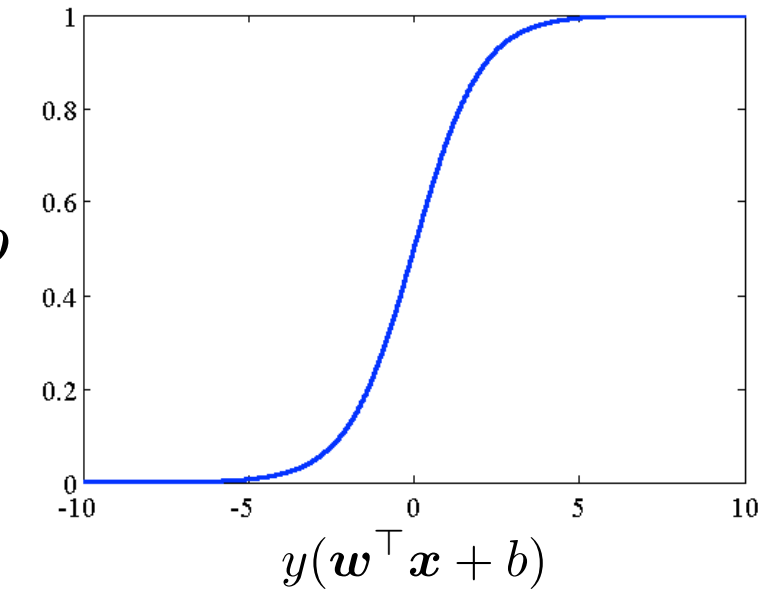
$$\boldsymbol{w} = \boldsymbol{w} + y\boldsymbol{x}$$



# Logistic regression

assume logit model: for a positive example

$$\boldsymbol{w}^\top \boldsymbol{x} = \log \frac{p(+1 \mid \boldsymbol{x})}{1 - p(+1 \mid \boldsymbol{x})}$$

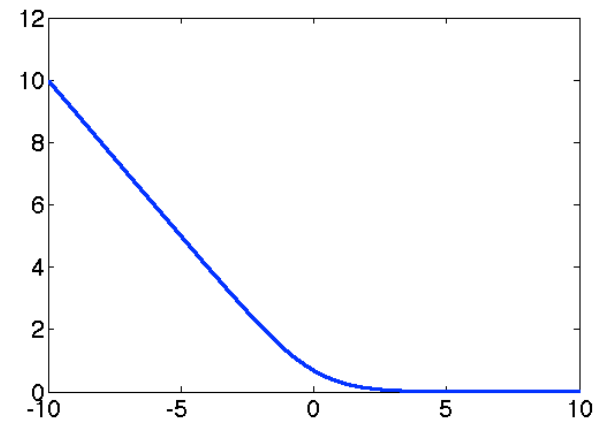



so that $p(y \mid \boldsymbol{x}, \boldsymbol{w}) = \dfrac{1}{1 + e^{-y(\boldsymbol{w}^\top \boldsymbol{x})}}$

minimize negative log-likelihood:

$$\arg\min_{\boldsymbol{w},b} -\log \prod_{i=1}^{m} p(y_i \mid \boldsymbol{x}_i, \boldsymbol{w}) = -\sum_i \log p(y_i \mid \boldsymbol{x}_i, \boldsymbol{w})$$

$$= \sum_i \log \left(1 + e^{-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i)}\right)$$




convex

# Linear classifier revisit

model space: $\mathbb{R}^{n+1}$

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$$

for classification $y \in \{-1, +1\}$

## Original objective:

$$\arg\min_{\boldsymbol{w},b} \sum_i I(y(\boldsymbol{w}^\top \boldsymbol{x} + b) \leq 0)$$

0-1 loss
hard to optimize

## Surrogate objective:

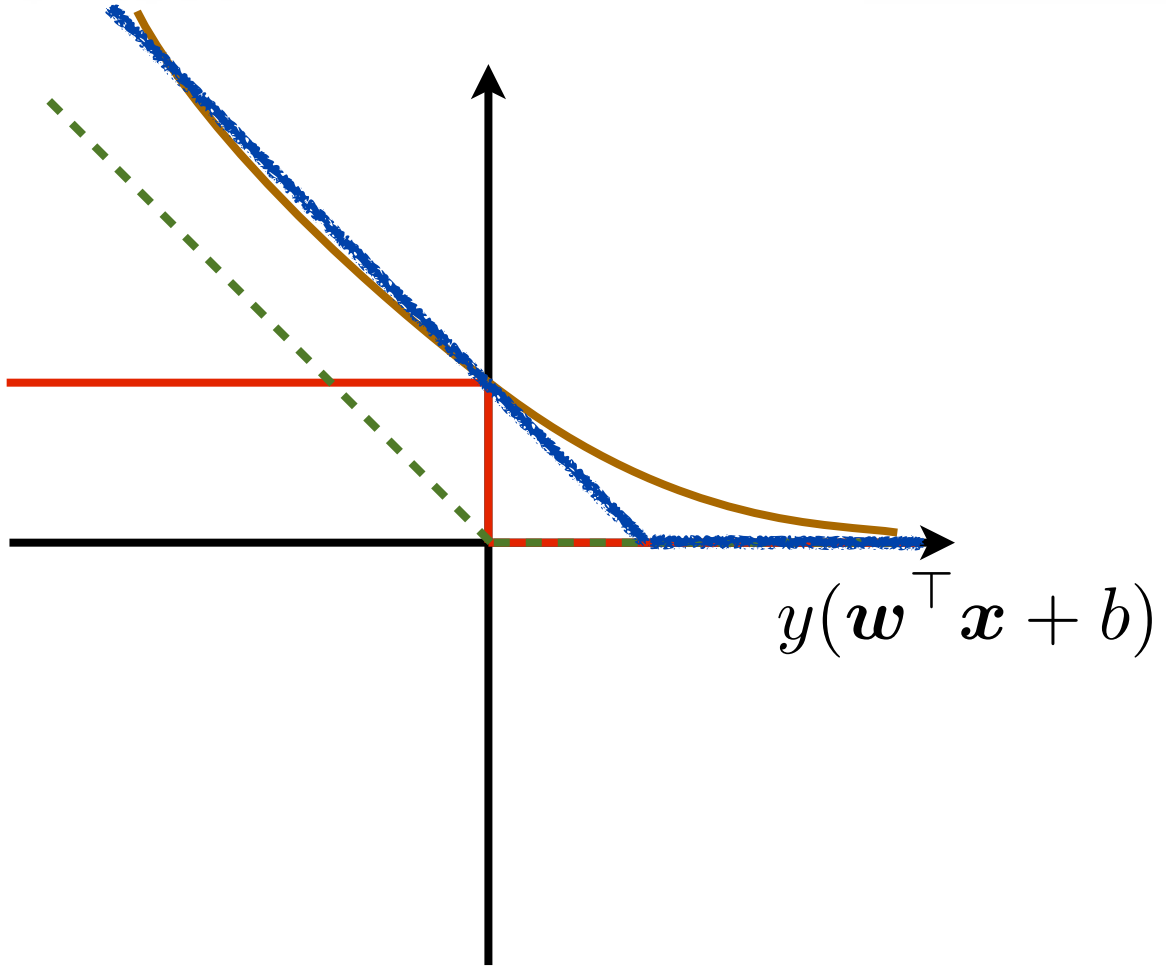
$$\arg\min_{\boldsymbol{w},b} \sum_i \log\left(1 + e^{-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)}\right)$$

logistic regression

$$\arg\min_{\boldsymbol{w},b} \sum_i \max\{-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), 0\}$$

perceptron

# Linear classifier revisit




0-1 loss

$I(y(\boldsymbol{w}^\top \boldsymbol{x} + b) \leq 0)$

logistic regression

$\log_2(1 + e^{-y(\boldsymbol{w}^\top \boldsymbol{x} + b)})$

perceptron

$\max\{-y(\boldsymbol{w}^\top \boldsymbol{x} + b), 0\}$

hinge loss

$\max\{1 - y(\boldsymbol{w}^\top \boldsymbol{x} + b), 0\}$

# Support vector machines (SVM)

hinge loss + L2-norm

$$\arg\min_{\boldsymbol{w},b} \sum_i \max(1 - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), 0) + \lambda \|\boldsymbol{w}\|_2$$

$$\max(1 - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), 0) = \xi_i$$
$$\xi_i \geq 1 - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$$
$$\xi_i \geq 0$$

$$\arg\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2 + C\sum_i \xi_i$$

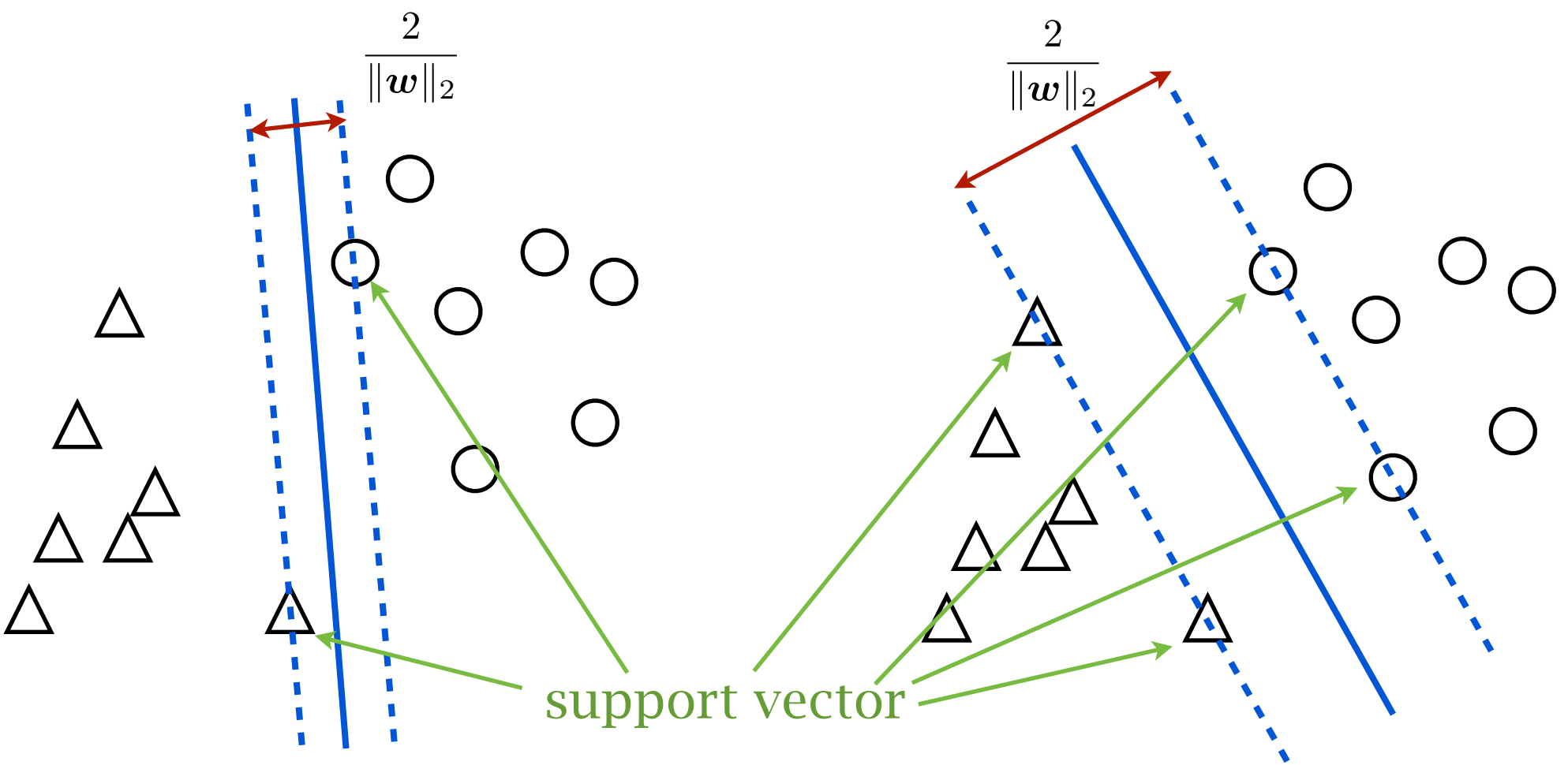
$$s.t. \quad y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i$$
$$\xi_i \geq 0$$

quadratic

# Support vector machines (SVM)

$$\arg\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2$$

$$s.t. \qquad y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1$$



# Scoring functions

$$\frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)^2 \quad \text{least square regression}$$

$$\frac{1}{m}\sum_{i=1}^{m}|\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i| \quad \text{LAD regression}$$

$$\frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)^2 + \lambda\|\boldsymbol{w}\|_2 \quad \text{ridge regression}$$

$$\frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i)^2 + \lambda\|\boldsymbol{w}\|_1 \quad \text{LASSO}$$

# Scoring functions

$$\sum_i I(y(\boldsymbol{w}^\top \boldsymbol{x} + b) > 0)$$ 0-1 loss

$$\sum_i \max\{-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), 0\}$$ perceptron

$$\sum_i \log\left(1 + e^{-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)}\right)$$ logistic regression

$$\sum_i \log\left(1 + e^{-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)}\right) + \lambda\|\boldsymbol{w}\|_2$$ regularized LR

$$\sum_i \max(1 - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), 0) + \lambda\|\boldsymbol{w}\|_2$$ SVM

minimize loss + regularization