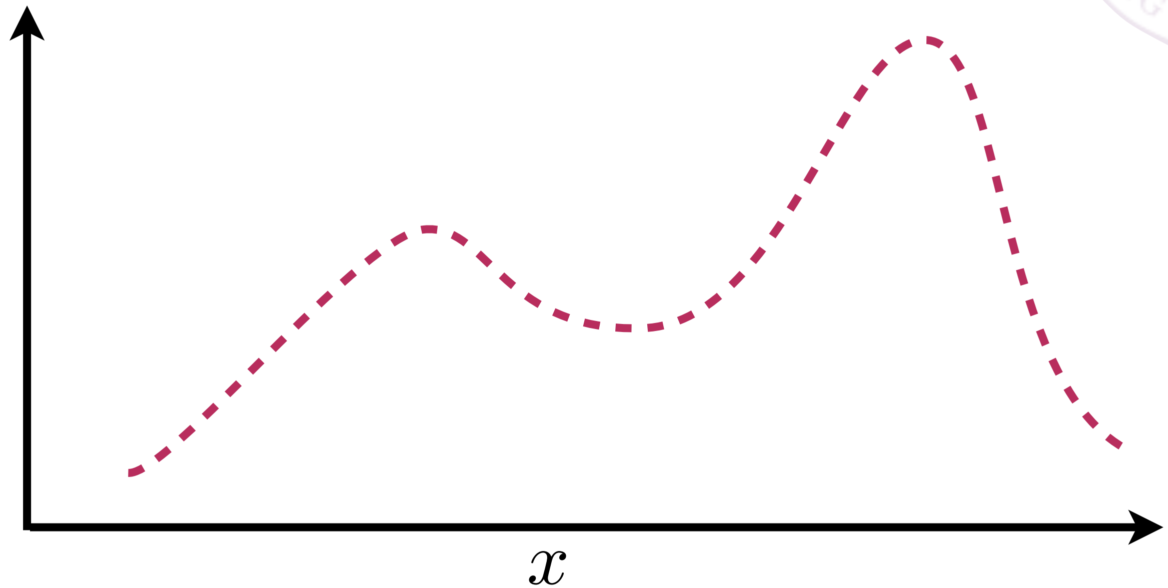
Lecture 4: Machine Learning II

Principle of Learning

http://cs.nju.edu.cn/yuy/course_dm14ms.ashx

The core of all the problems

infinite samples

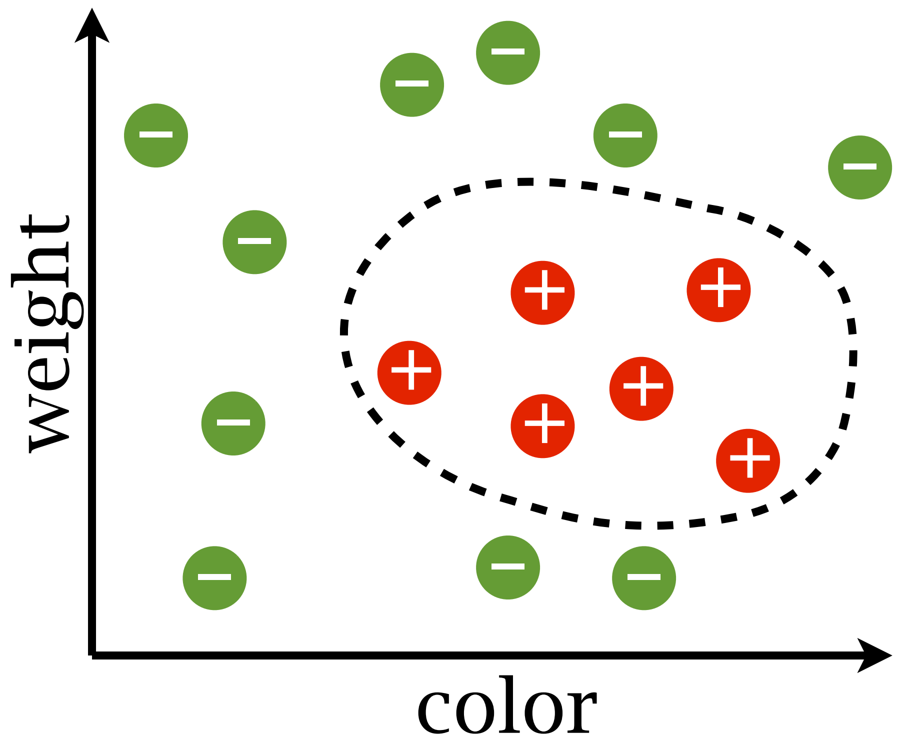


v.s.

finite samples

Classification

Features: color, weight
Label: taste is sweet (positive/+) or not (negative/-)



(color, weight) $\rightarrow$ sweet ?

$\mathcal{X} \quad\quad \rightarrow \{-1, +1\}$

ground-truth function f

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

learning: find an f' that is <u>close</u> to f

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:

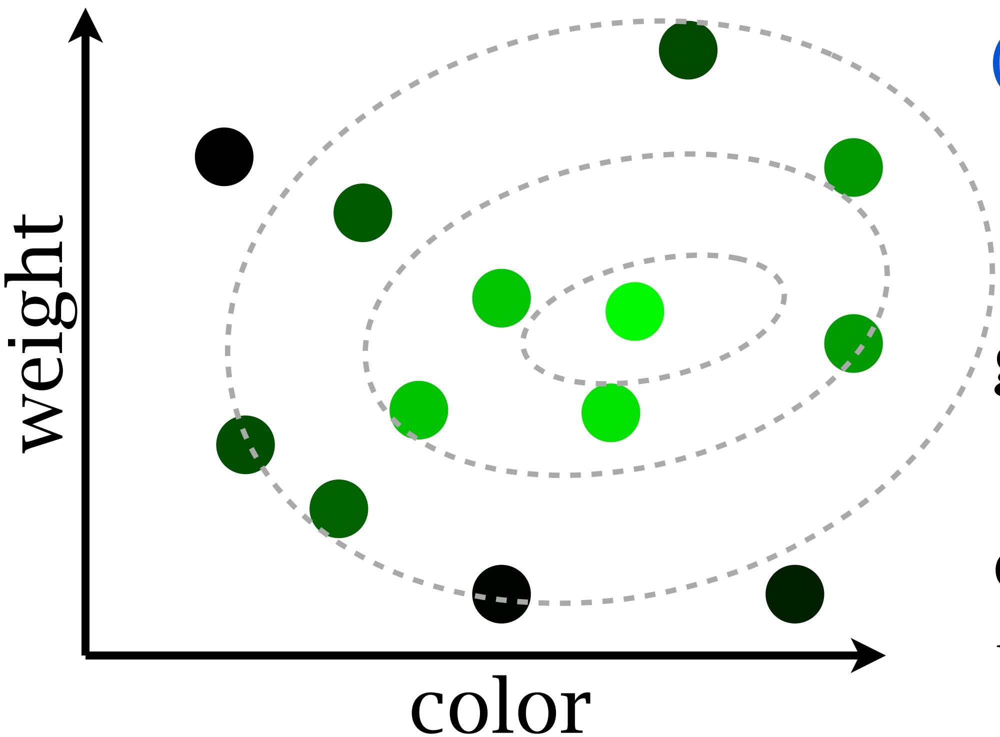
over the whole distribution: generalization error

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

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

Regression

Features: color, weight
Label: price [0,1]



(color, weight) → price

$\mathcal{X} \rightarrow [0, +1]$

ground-truth function f

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

learning: find an f' that is close to f

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:

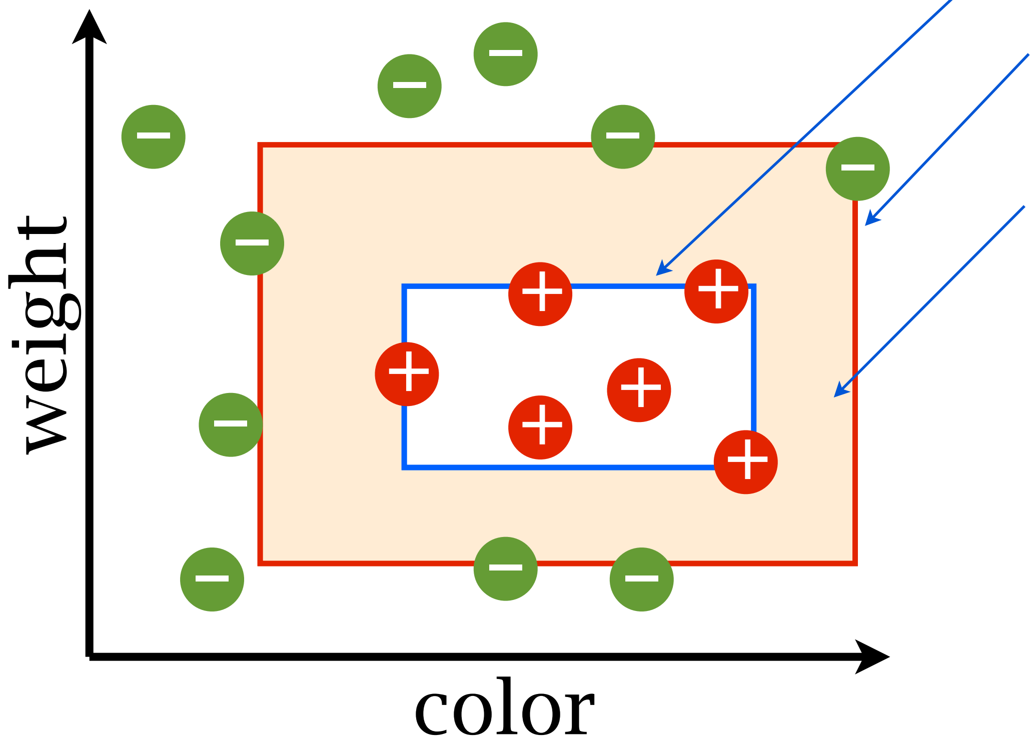
over the whole distribution: generalization MSE

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

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

The version space algorithm

an abstract view of learning algorithms



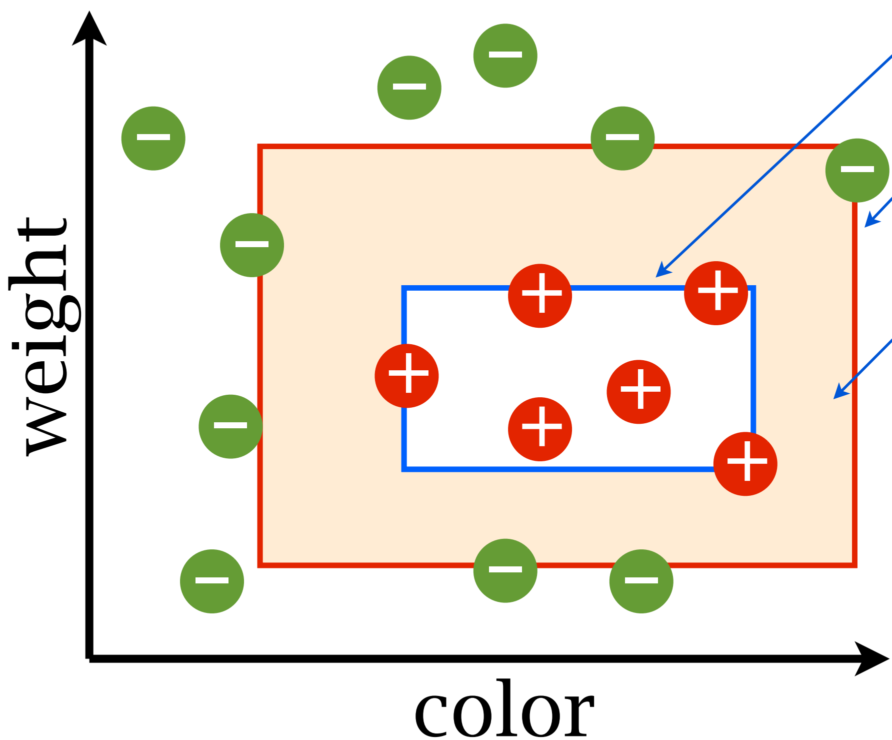
S: most specific hypothesis

G: most general hypothesis

version space: consistent hypotheses [Mitchell, 1997]

The version space algorithm

an abstract view of learning algorithms



S: most specific hypothesis

G: most general hypothesis

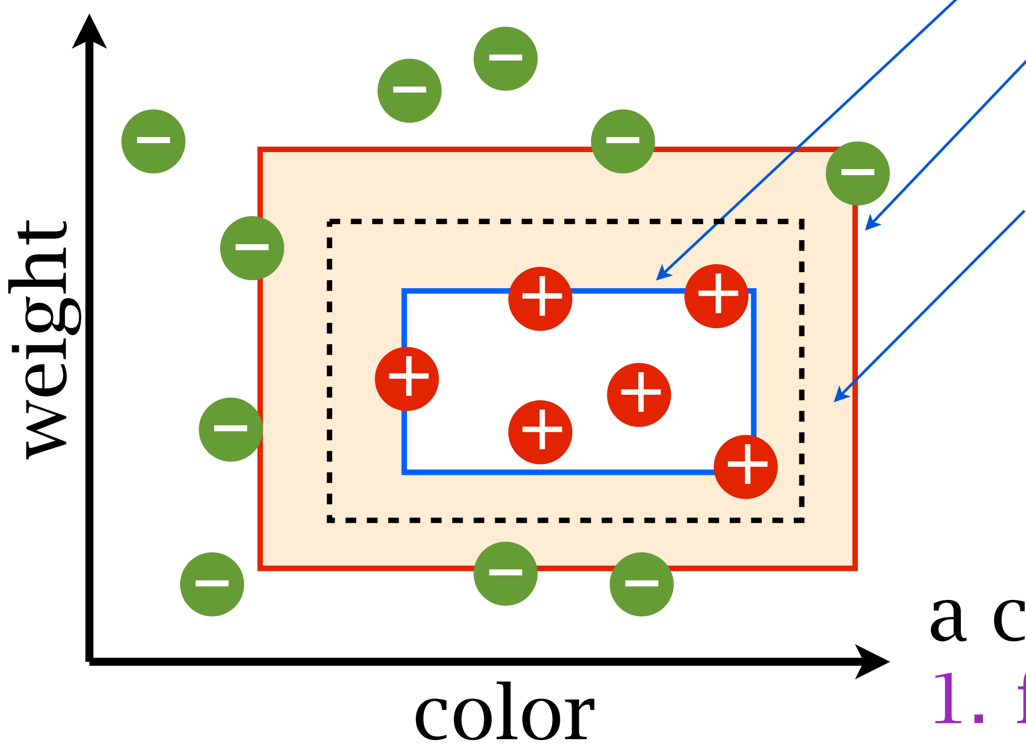
version space: consistent hypotheses [Mitchell, 1997]

a conceptual algorithm:

1. for every example, remove the conflict boxes
2. find S in remaining boxes
3. find G in remaining boxes
4. output the mean of S and G

The version space algorithm

an abstract view of learning algorithms



S: most specific hypothesis

G: most general hypothesis

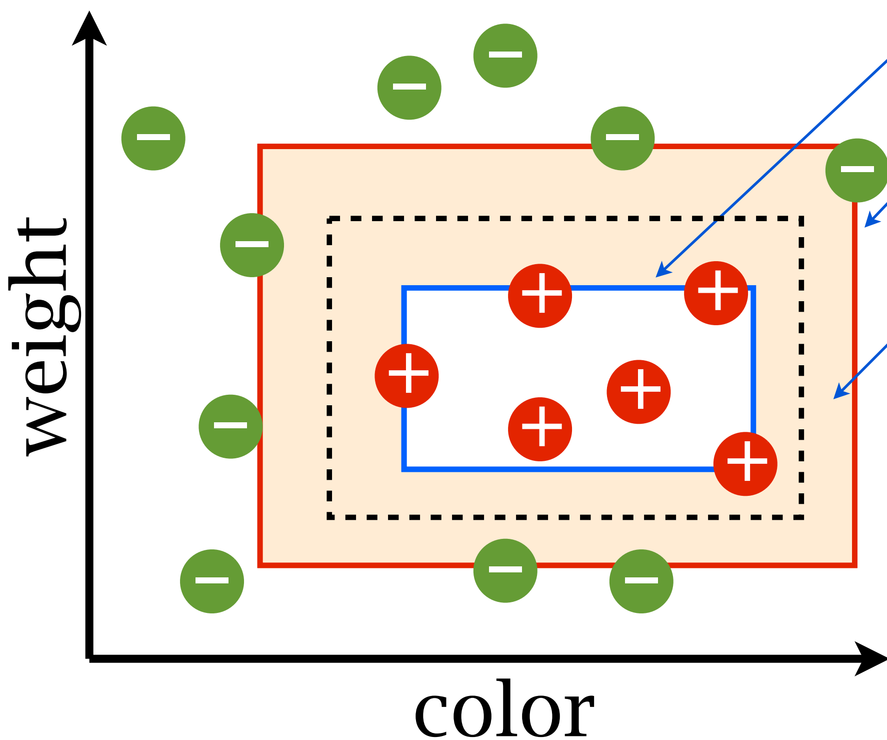
version space: consistent hypotheses [Mitchell, 1997]

a conceptual algorithm:

1. for every example, remove the conflict boxes
2. find S in remaining boxes
3. find G in remaining boxes
4. output the mean of S and G

The version space algorithm

an abstract view of learning algorithms



S: most specific hypothesis

G: most general hypothesis

version space: consistent hypotheses [Mitchell, 1997]

selection a hypothesis according to learner's bias

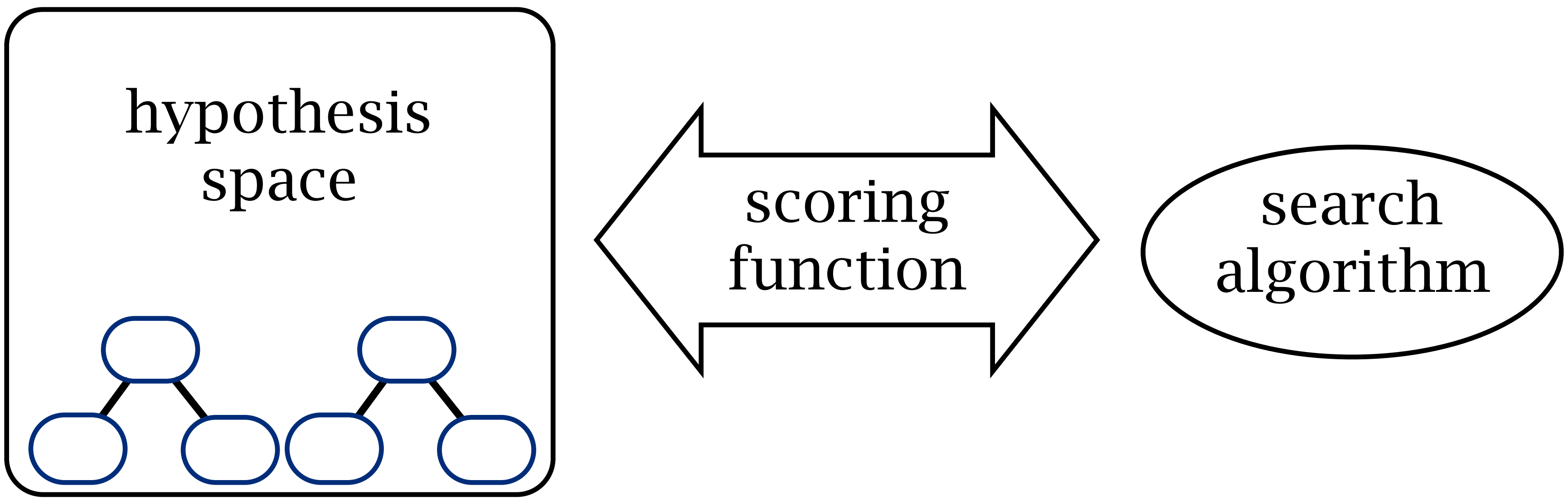
a conceptual algorithm:

1. for every example, remove the conflict boxes
2. find S in remaining boxes
3. find G in remaining boxes
4. output the mean of S and G

The version space algorithm

an abstract view of learning algorithms

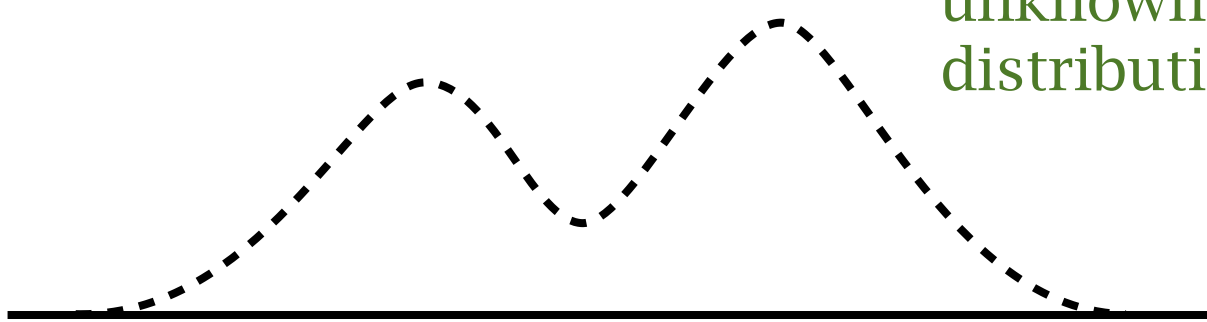
three components of a learning algorithm



Theories

The i.i.d. assumption:

all training examples and future (test) examples are drawn *independently* from an *identical distribution*



bias-variance dilemma (regression)

generalization bound (classification)

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)

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

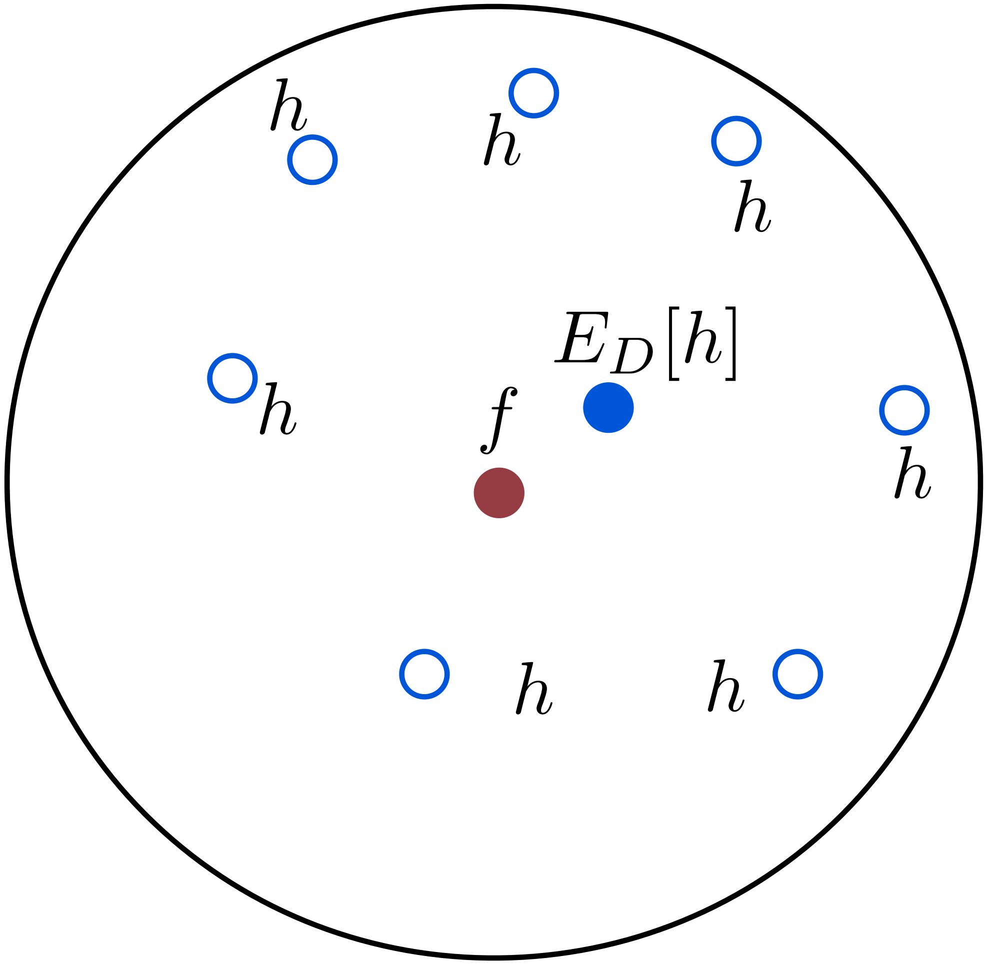
variance bias^2

Bias-variance dilemma

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

variance bias^2

larger hypothesis space
=>
lower bias
but higher variance



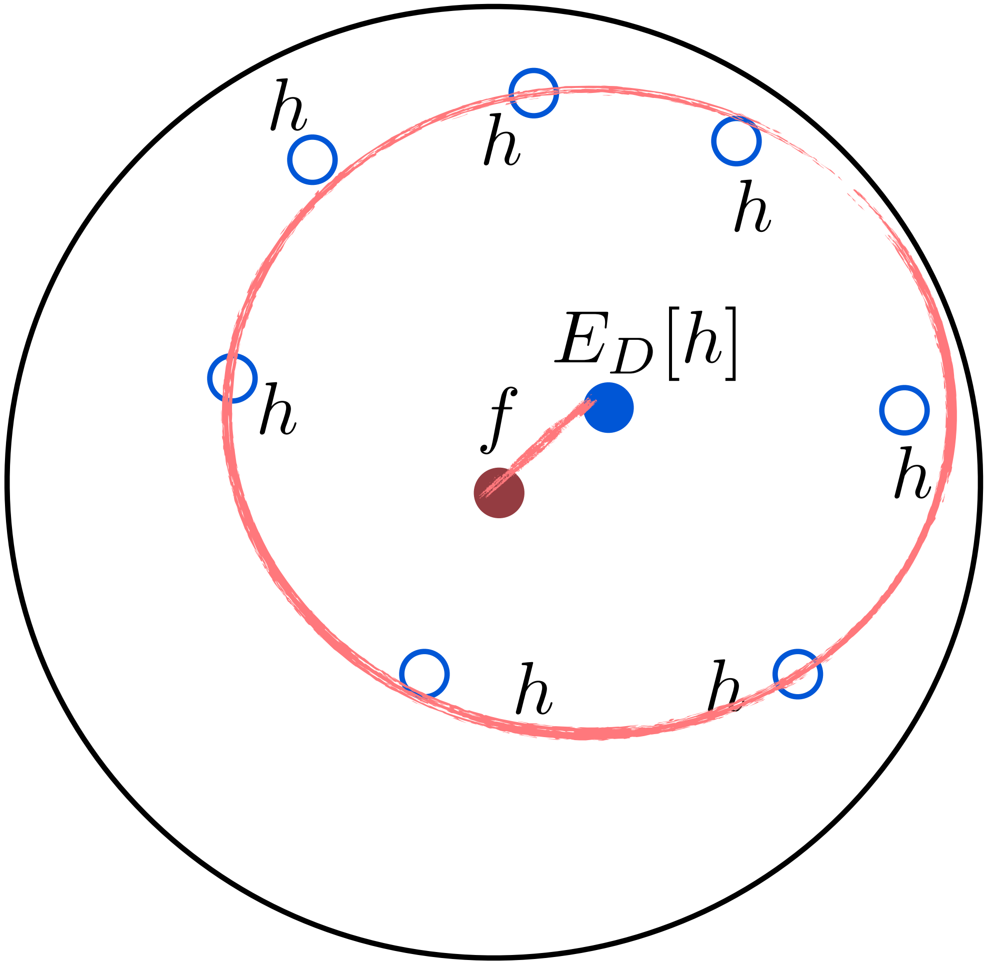
hypothesis space

Bias-variance dilemma

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

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

larger hypothesis space
=>
lower bias
but higher variance



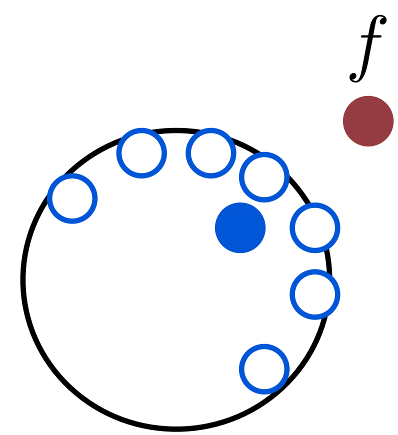
hypothesis space

Bias-variance dilemma

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

variance   bias^2

smaller hypothesis space
=>
smaller variance
but higher bias



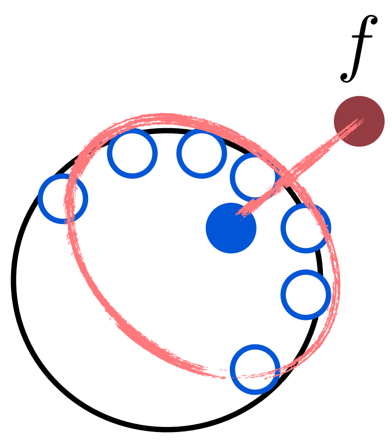
hypothesis space

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

smaller hypothesis space
=>
smaller variance
but higher bias



hypothesis space

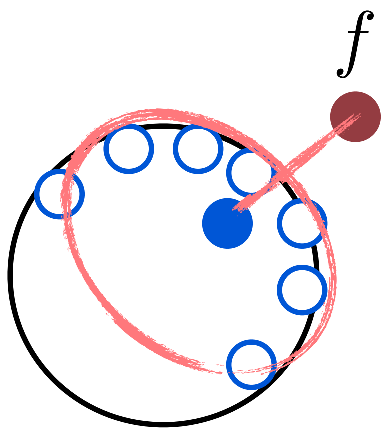
Bias-variance dilemma

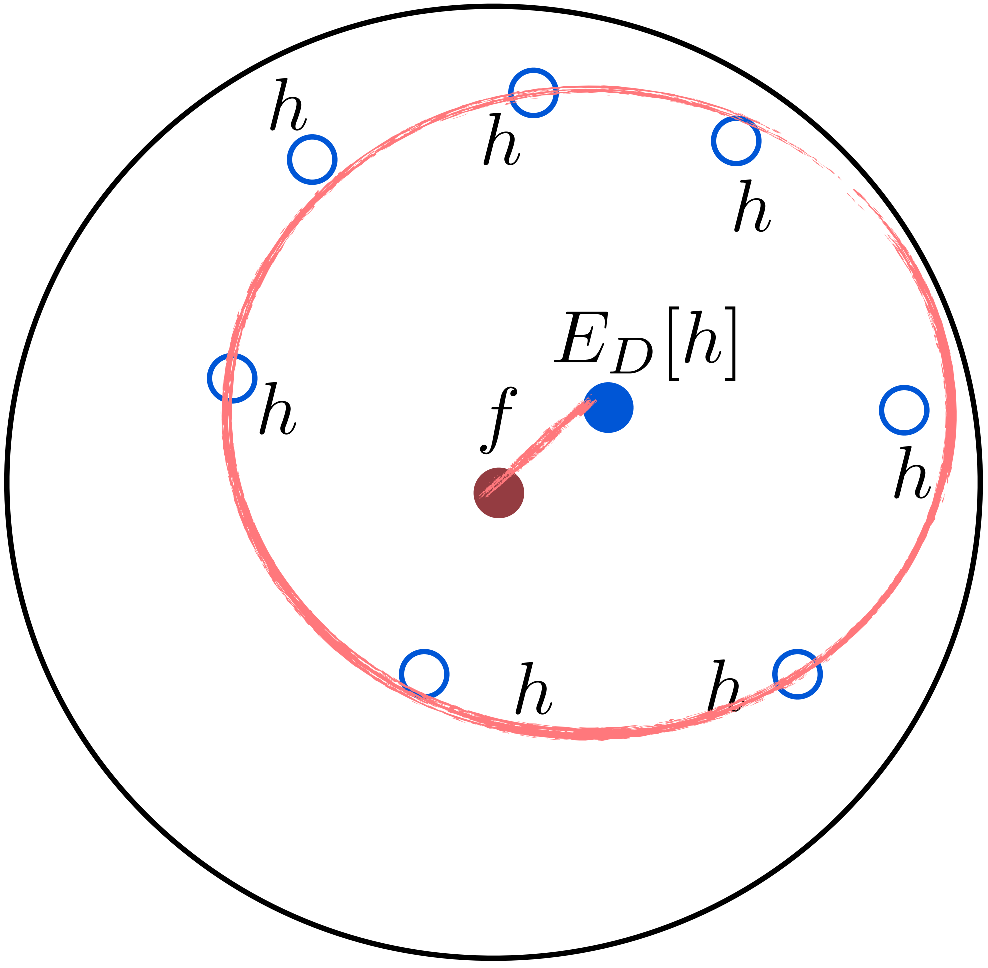
$$E_D\left[(h(\boldsymbol{x}) - E_D[h(\boldsymbol{x})])^2\right]$$

variance

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

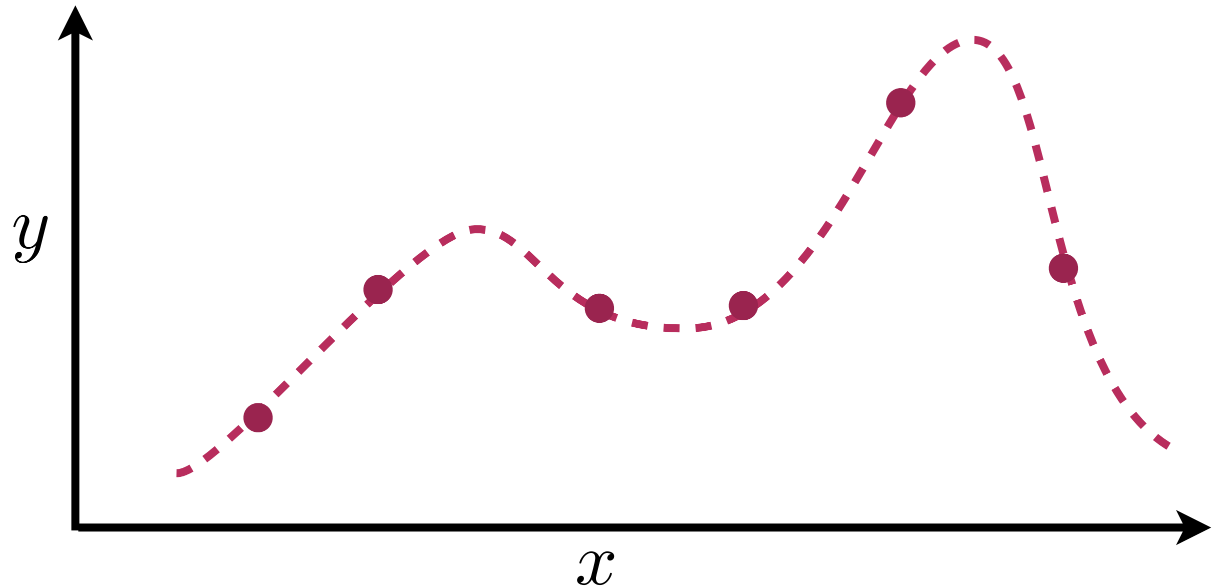
bias^2





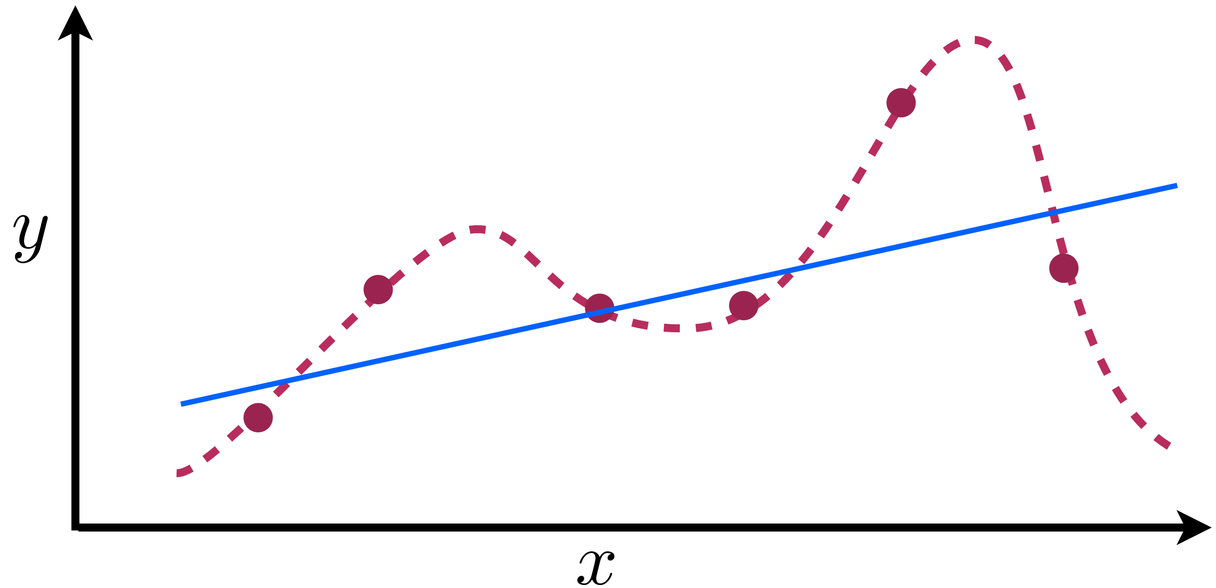
Overfitting and underfitting

training error v.s. hypothesis space size



Overfitting and underfitting

training error v.s. hypothesis space size

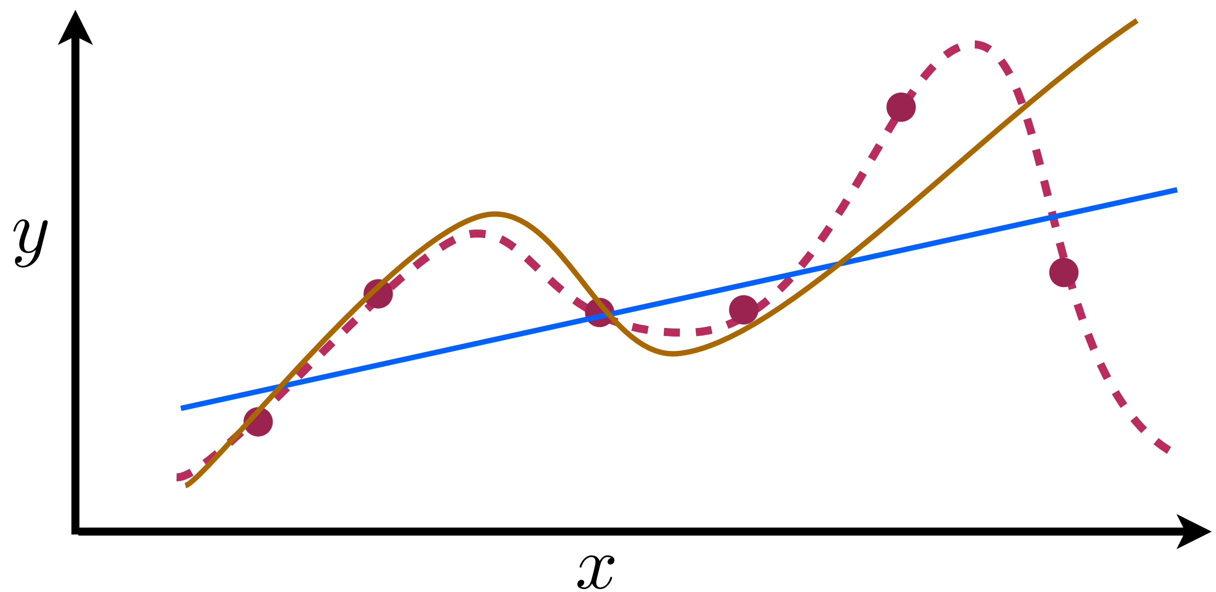


linear functions: high training error, small space

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

Overfitting and underfitting

training error v.s. hypothesis space size



linear functions: high training error, small space

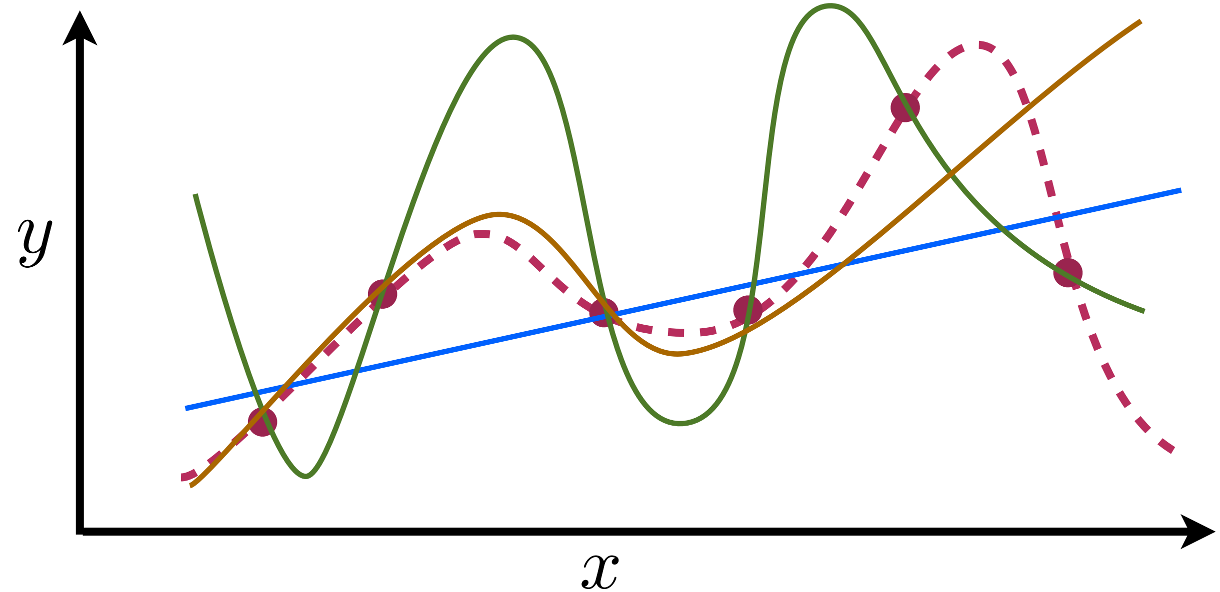
$$\{y = a + bx \mid a, b \in \mathbb{R}\}$$

higher polynomials: moderate training error, moderate space

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

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

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

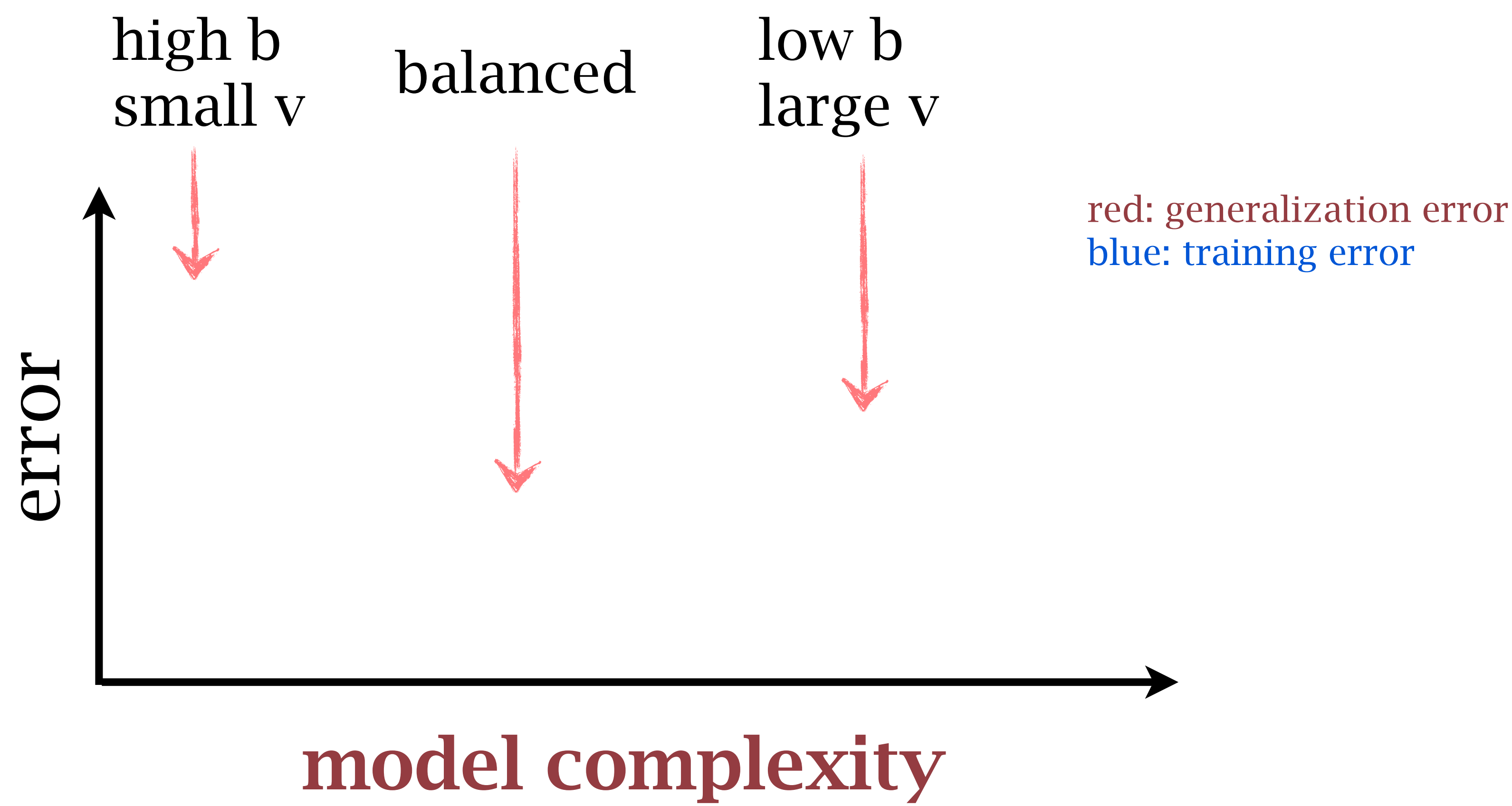
Overfitting and bias-variance dilemma

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

variance

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

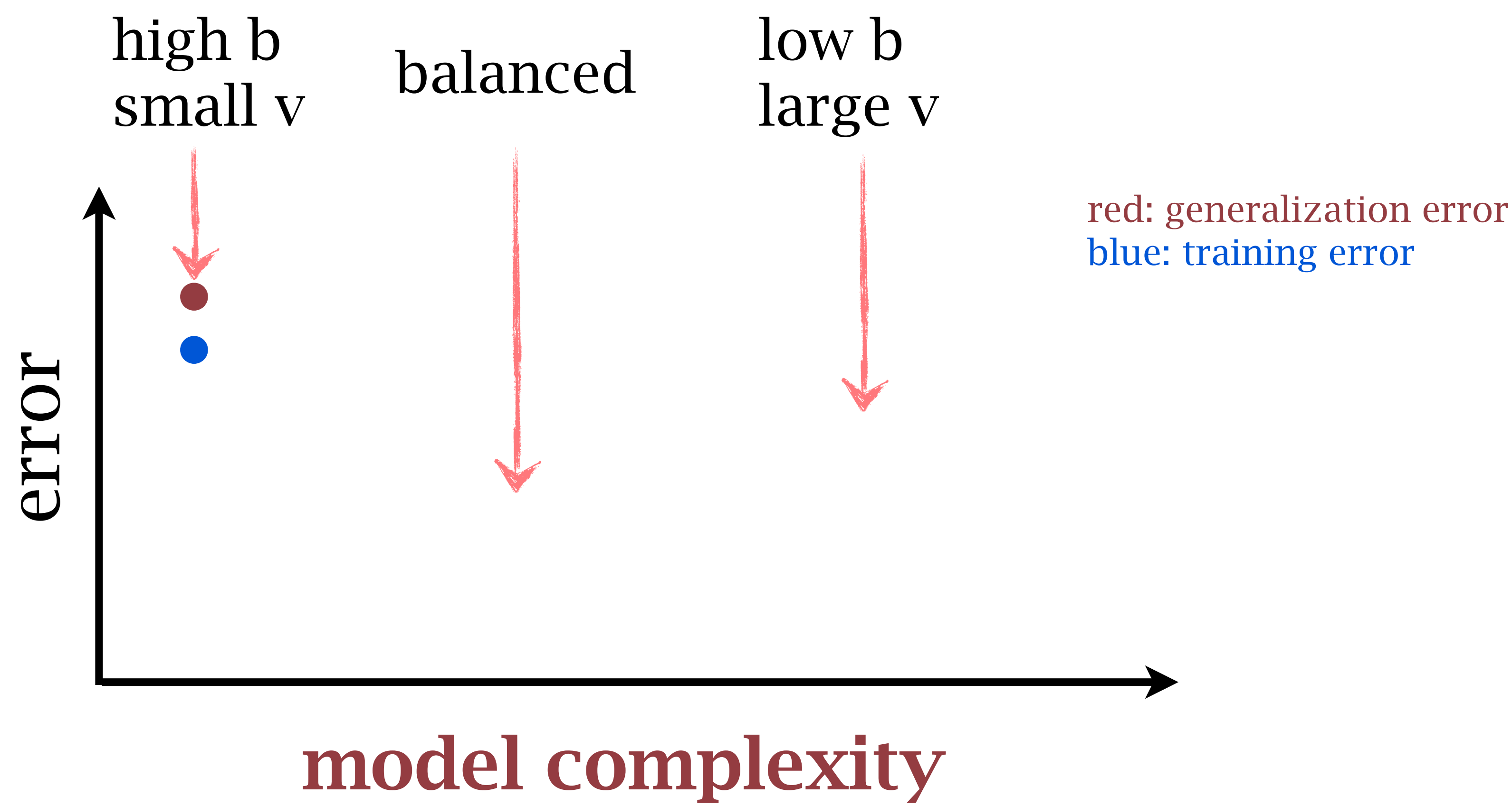
bias^2



Overfitting and bias-variance dilemma

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

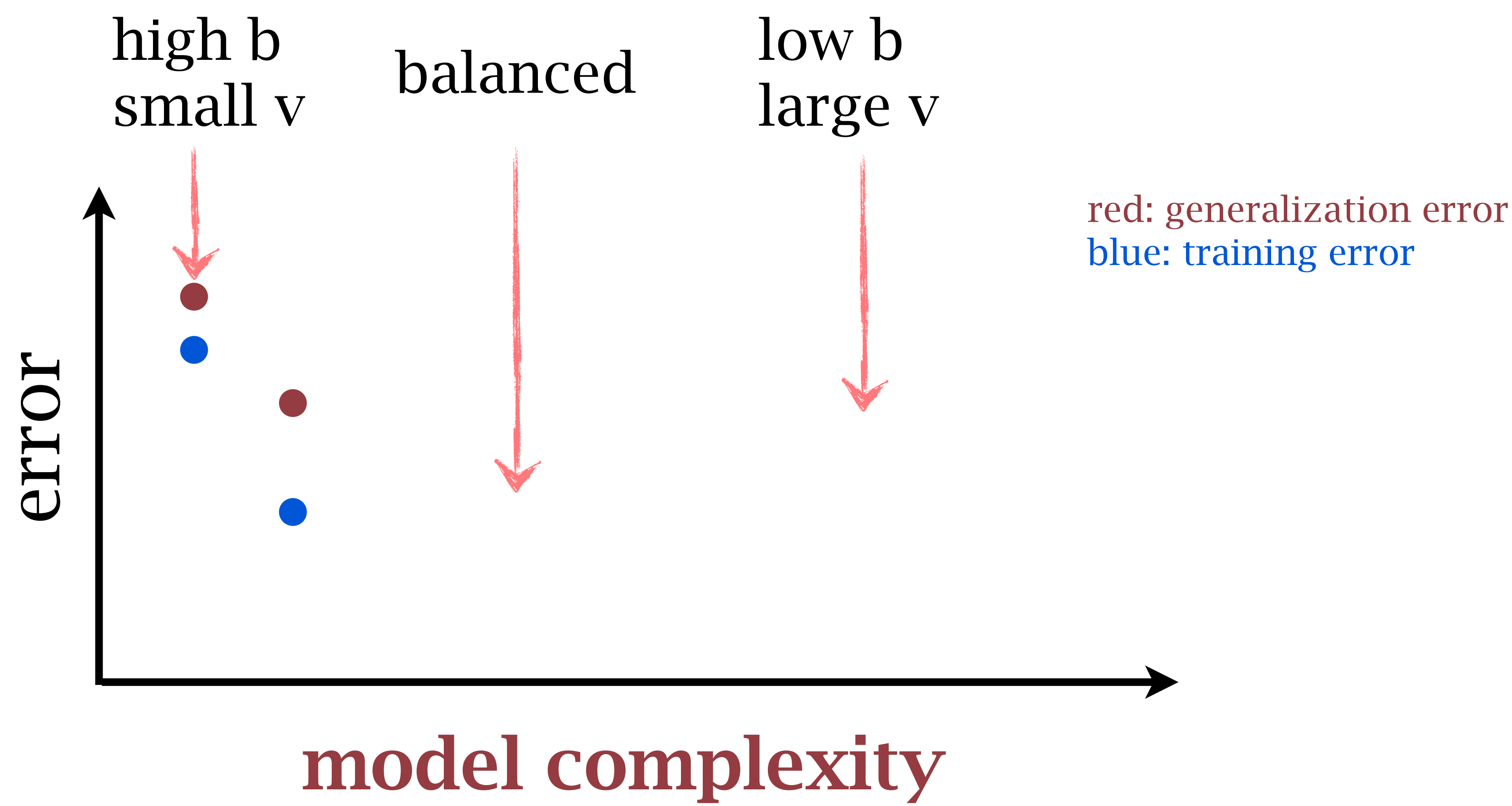
variance　　　　bias^2



Overfitting and bias-variance dilemma

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

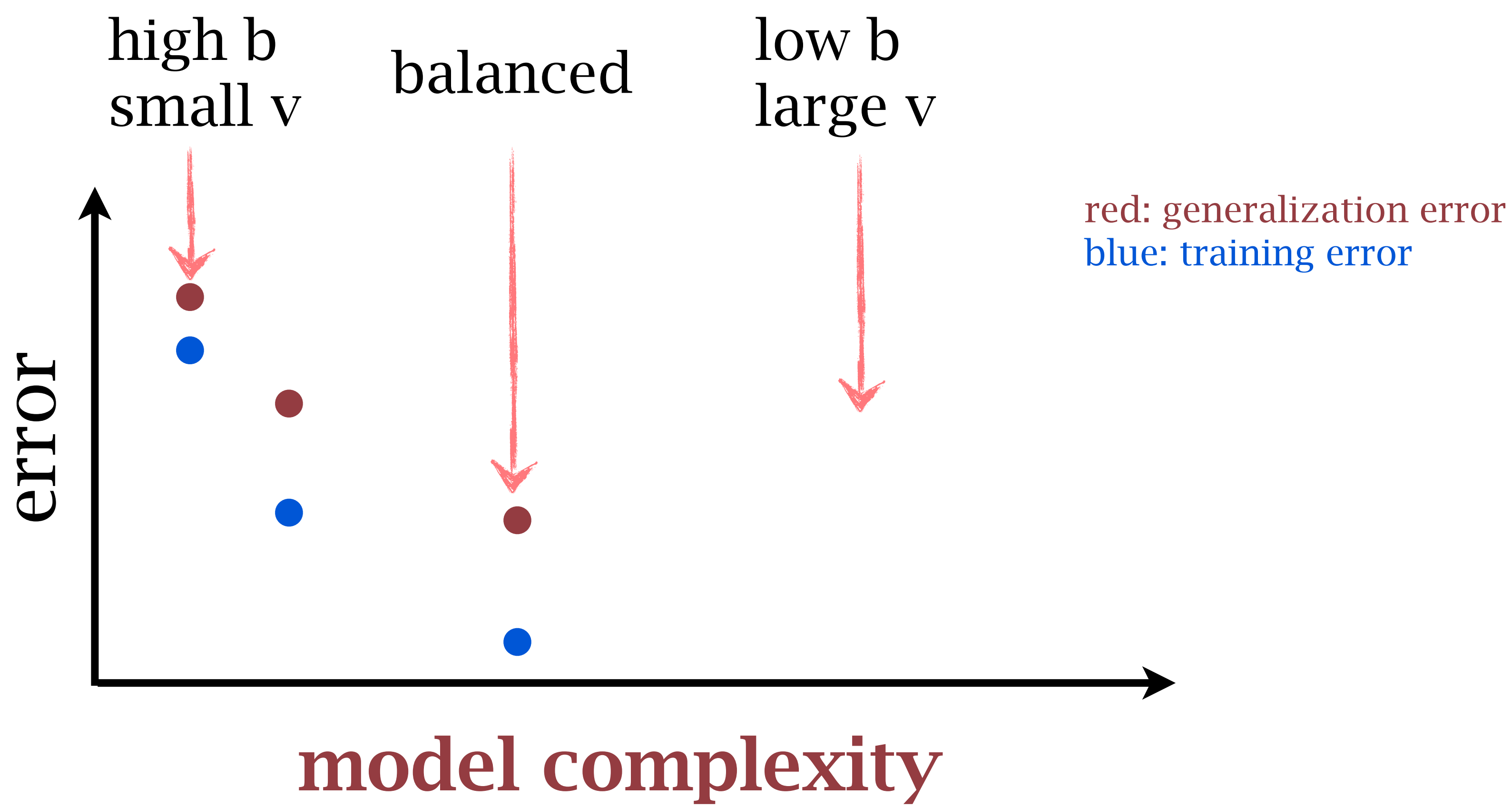
variance bias^2



Overfitting and bias-variance dilemma

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

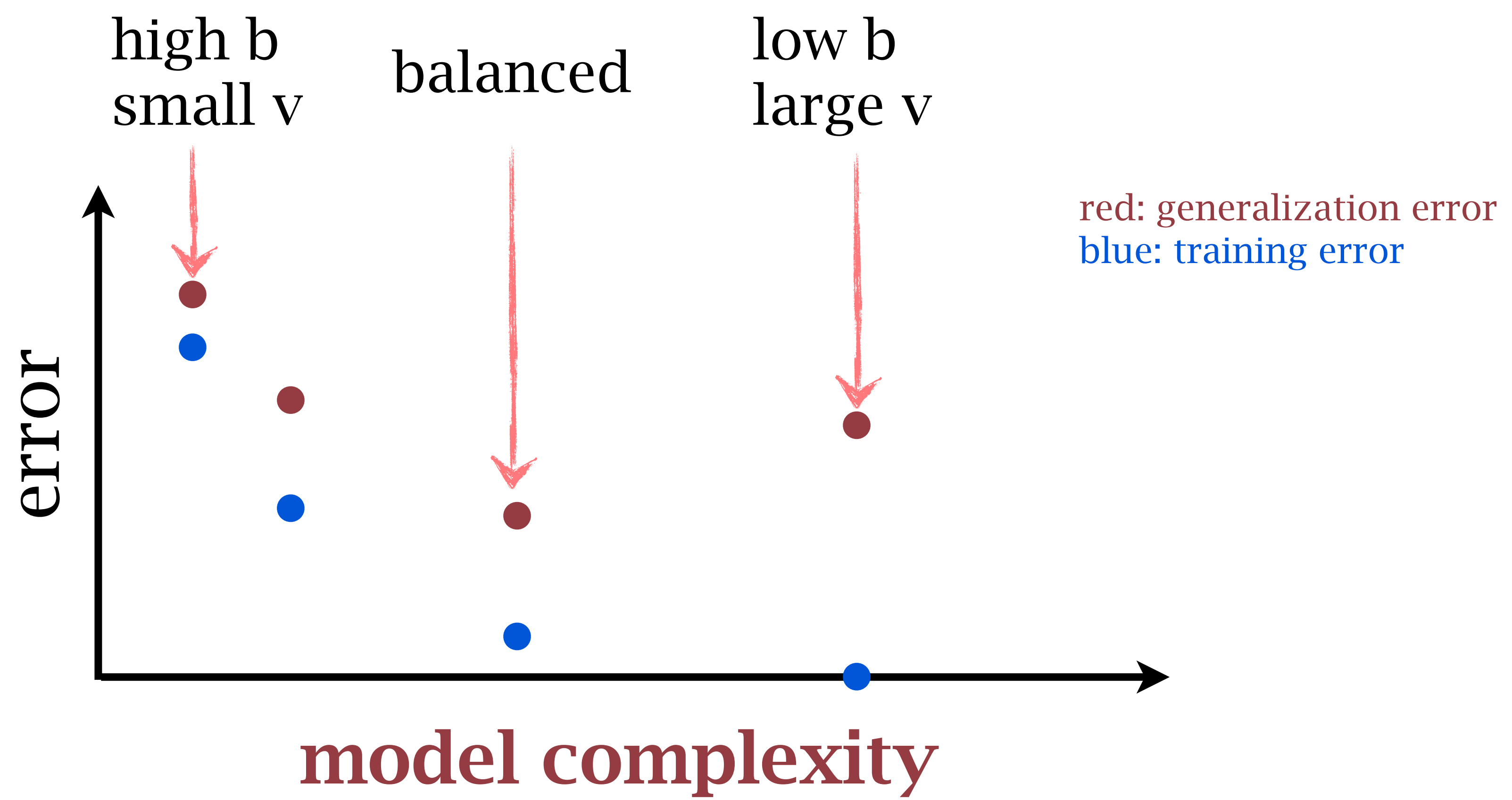
variance bias^2



Overfitting and bias-variance dilemma

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

variance bias^2



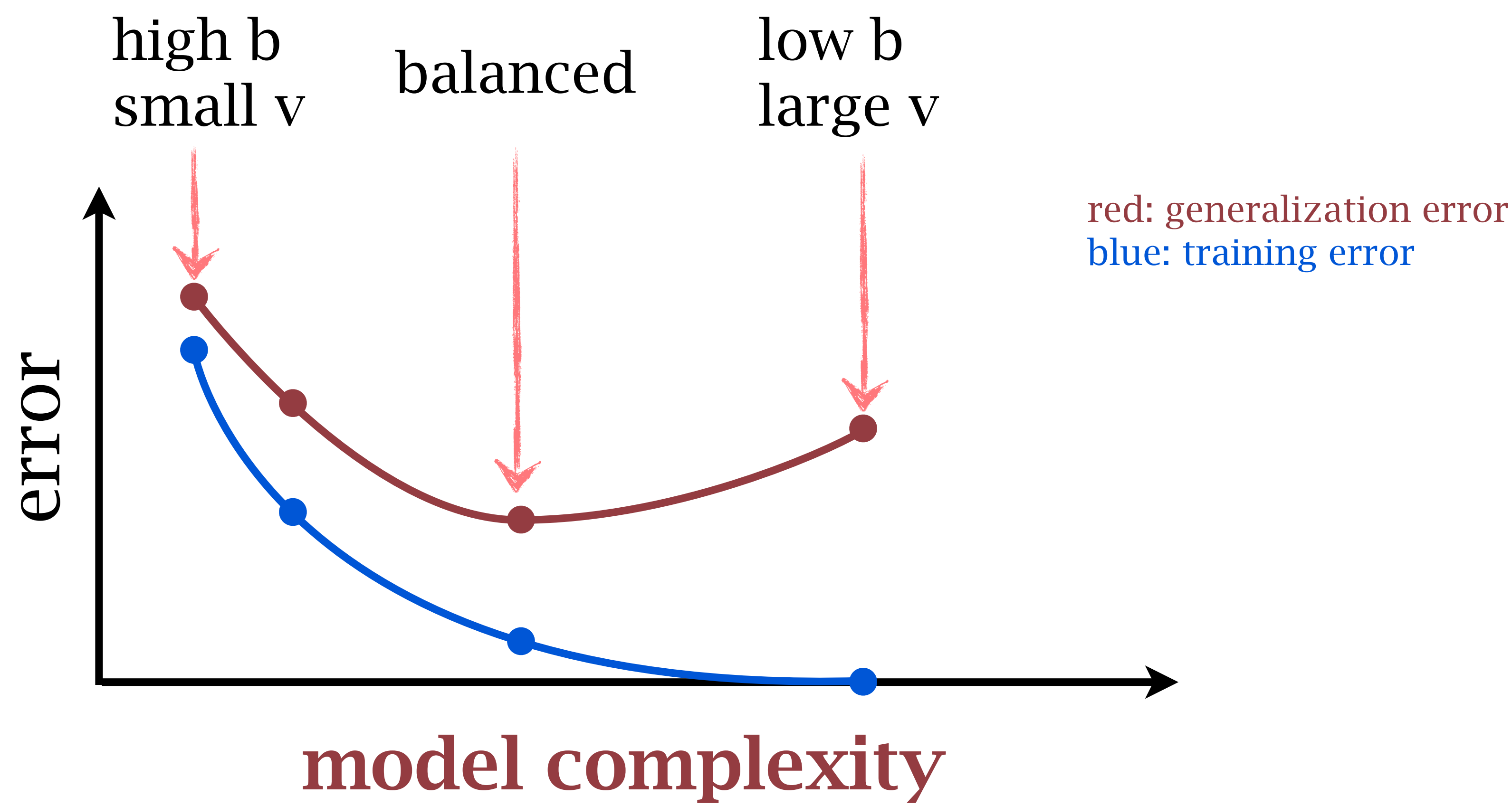
Overfitting and bias-variance dilemma

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

variance

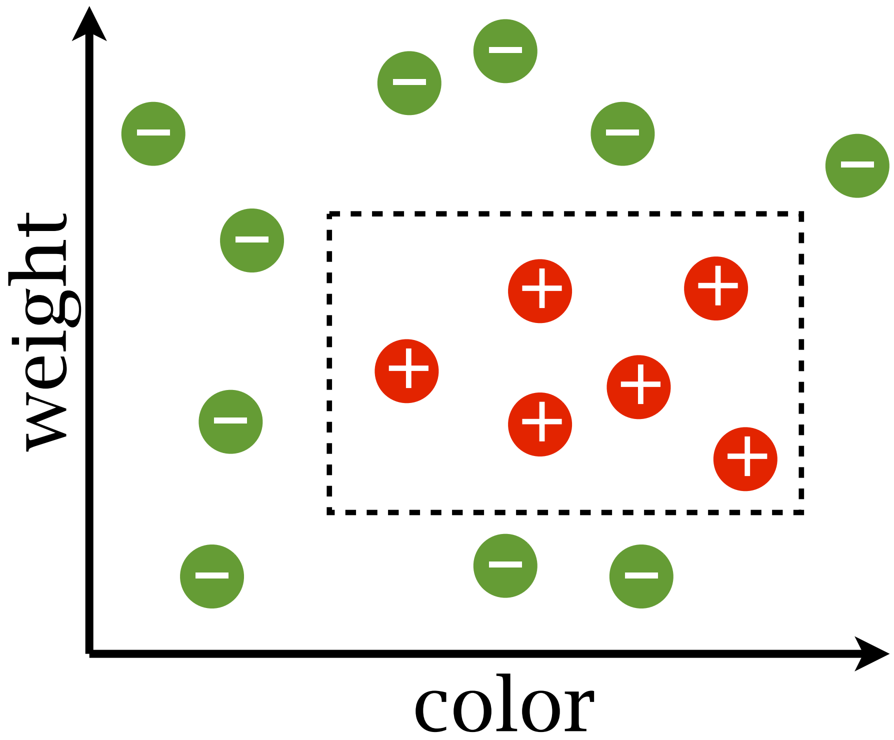
$$E_D\left[(E_D[h(\boldsymbol{x})] - f(\boldsymbol{x}))^2\right]$$

bias^2



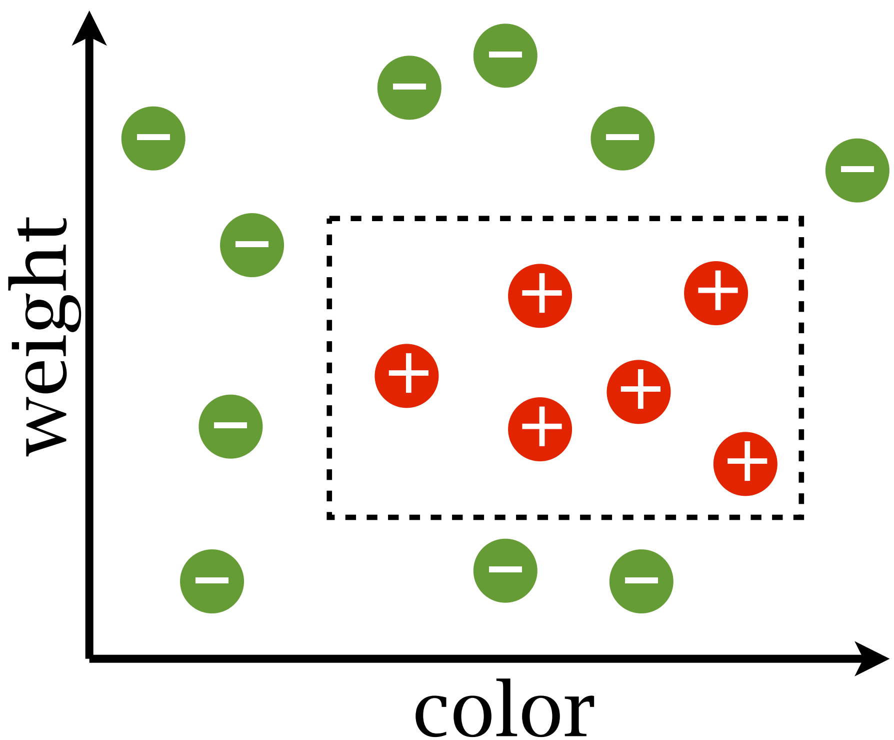
Generalization error

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



Generalization error

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



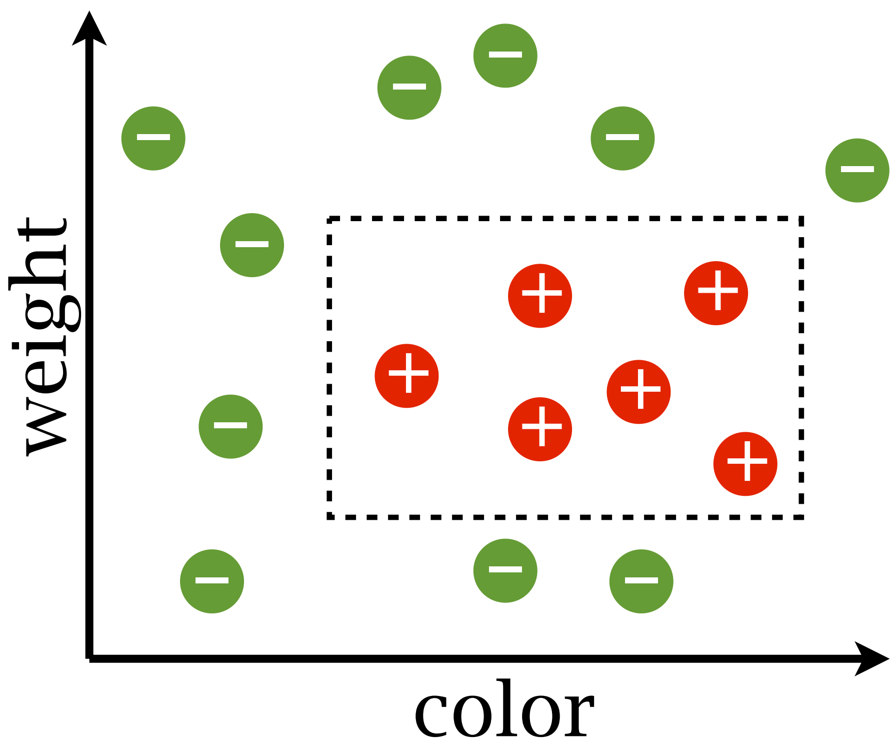
the error of picking a consistent hypothesis:

with probability at least $1-\delta$

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

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 h is consistent

$\epsilon_g(h) \geq \epsilon$

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

Generalization error

for one *h*

What is the probability of h is consistent, $\epsilon_g(h) \geq \epsilon$

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

h is consistent with 1 example:

Generalization error

for one *h*

What is the probability of h is consistent
$\epsilon_g(h) \geq \epsilon$

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

h is consistent with 1 example:

$$P \leq 1 - \epsilon$$

Generalization error

for one *h*

What is the probability of h is consistent, $\epsilon_g(h) \geq \epsilon$

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

h is consistent with 1 example:

$$P \leq 1 - \epsilon$$

h is consistent with ***m*** example:

Generalization error

for one *h*

What is the probability of h is consistent, $\epsilon_g(h) \geq \epsilon$

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

h is consistent with 1 example:

$$P \leq 1 - \epsilon$$

h is consistent with ***m*** example:

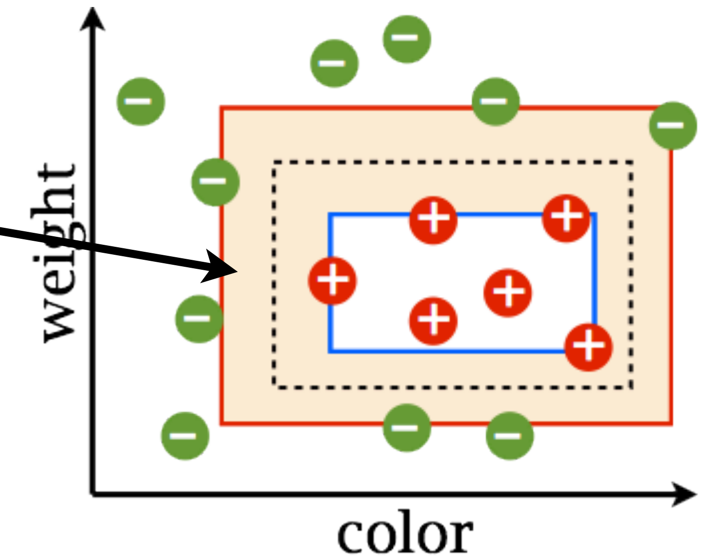
$$P \leq (1 - \epsilon)^m$$

Generalization error

h is consistent with ***m*** example:

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

There are ***k*** consistent hypotheses

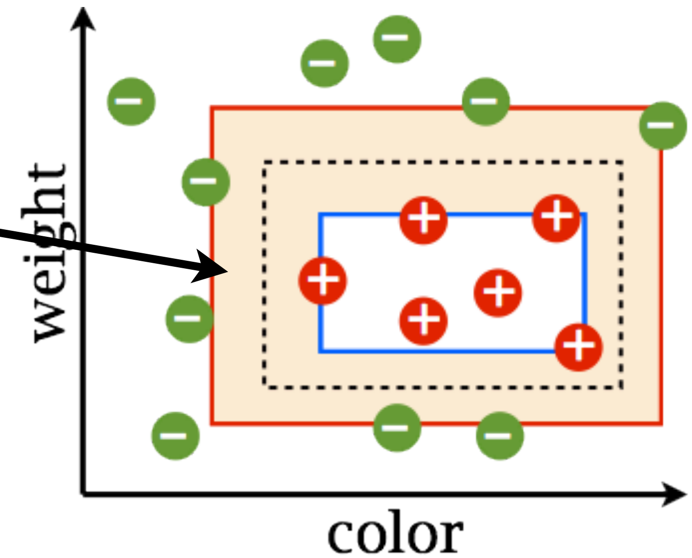


Generalization error

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

...

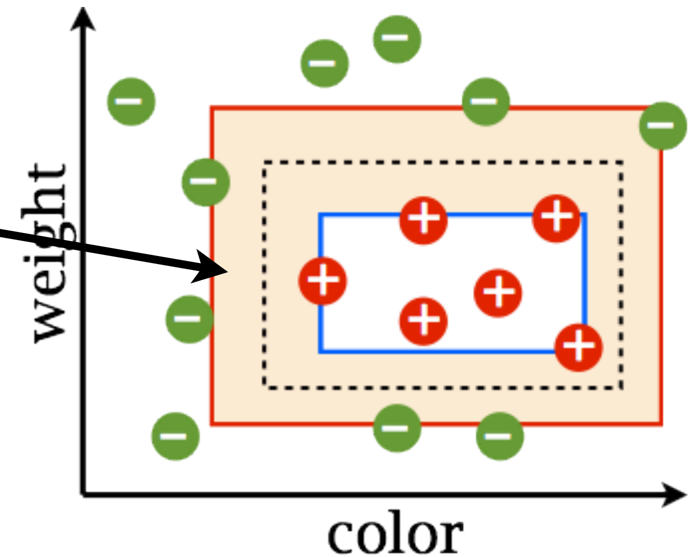
h_k is chosen and h_k is bad $P \leq (1-\epsilon)^m$

Generalization error

h is consistent with ***m*** example:

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

There are ***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

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

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

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

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

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

Generalization error

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

$$\Downarrow$$

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

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

Generalization error

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

$$\Downarrow$$

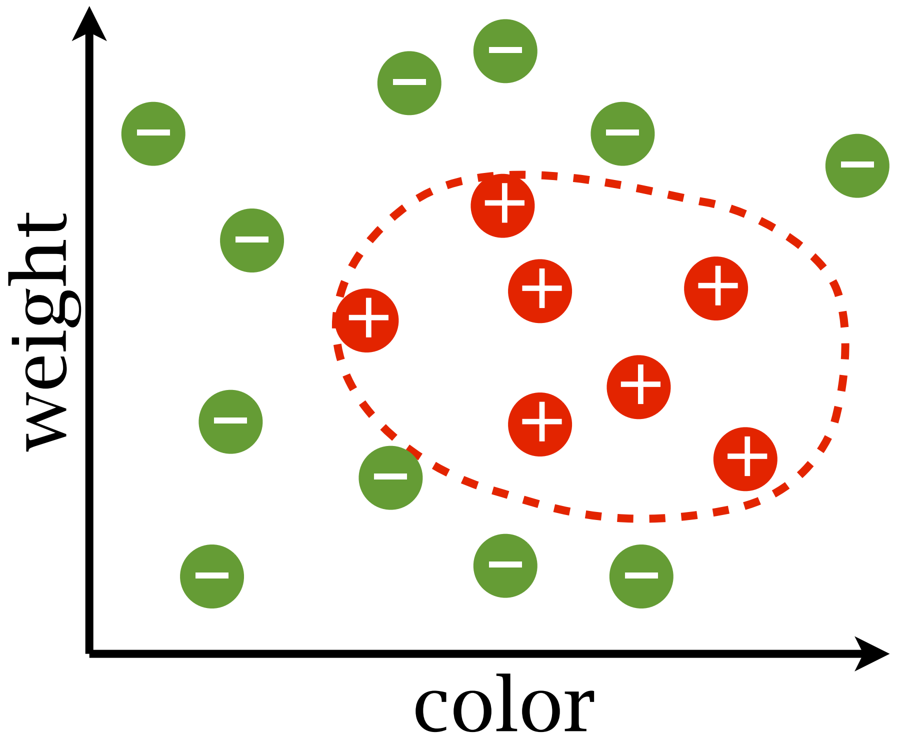
$$P(\epsilon_g \geq \epsilon) \leq \underbrace{|\mathcal{H}| \cdot (1-\epsilon)^m}_{\delta}$$

with probability at least $1-\delta$

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

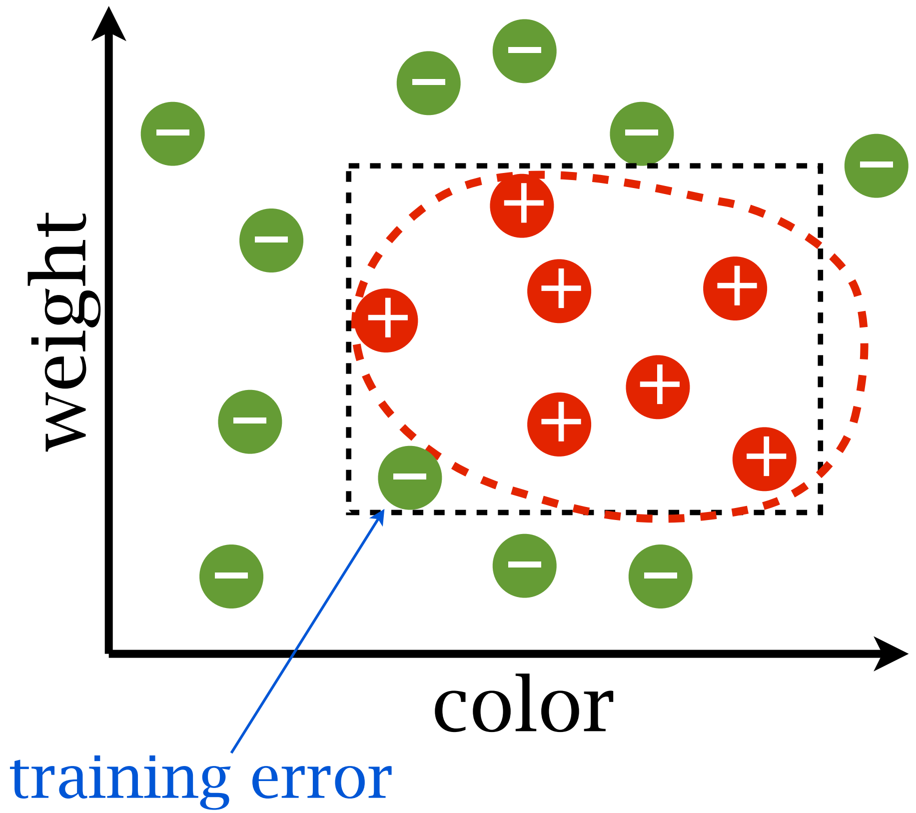
Inconsistent hypothesis

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



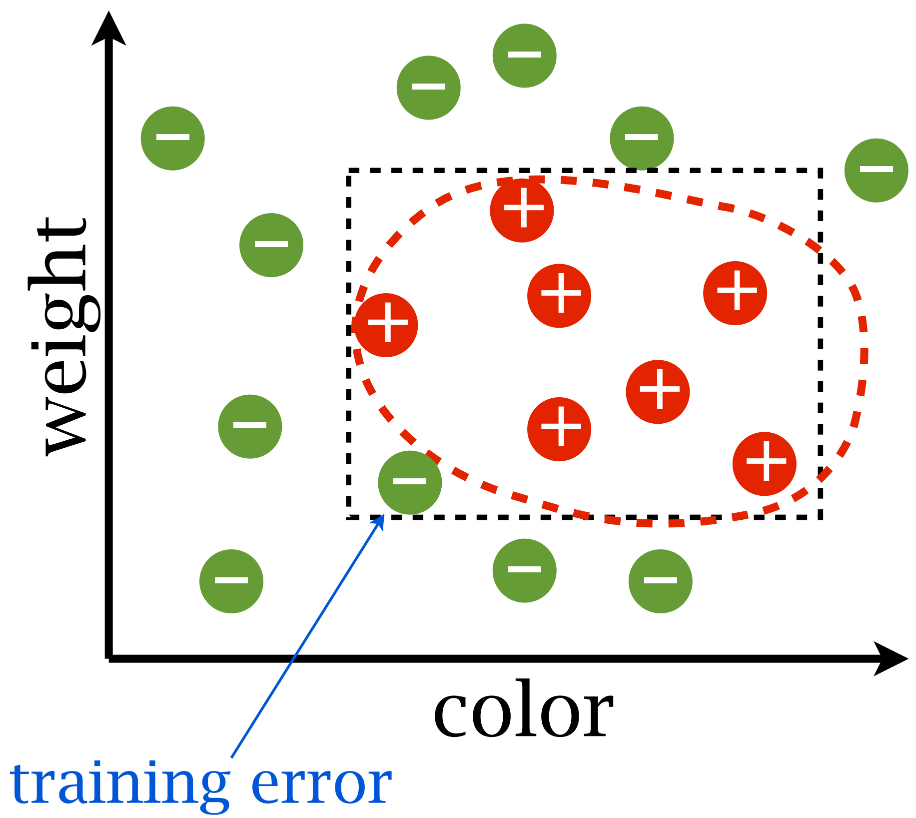
Inconsistent hypothesis

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



Inconsistent hypothesis

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

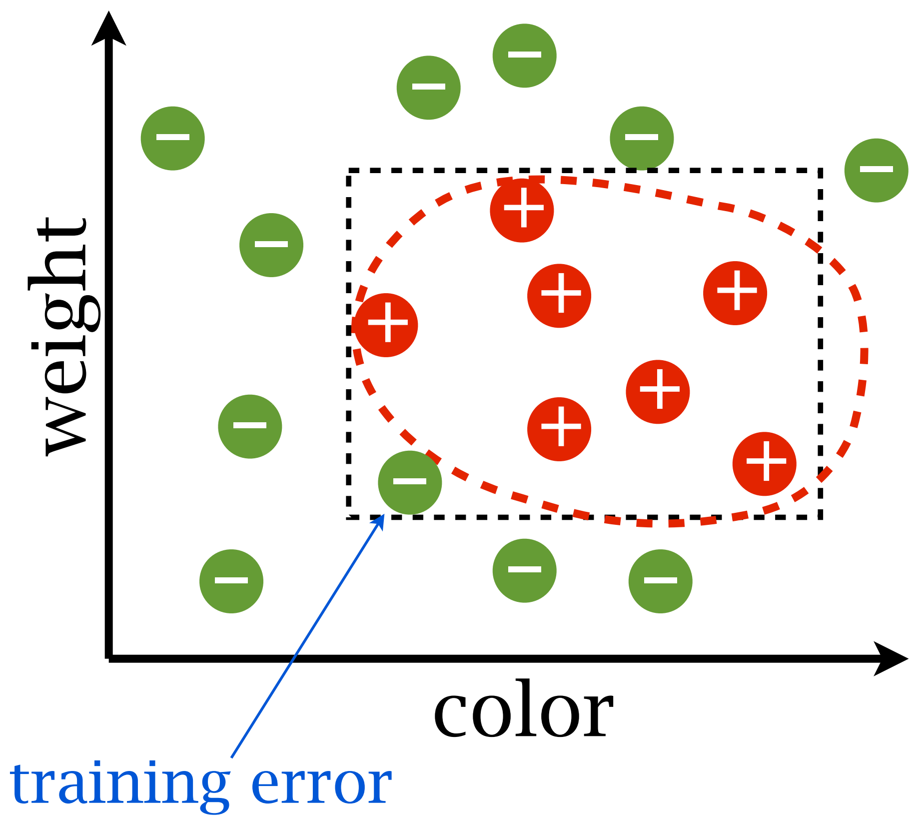


with probability at least $1-\delta$

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

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

$$\text{for one } h$$

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

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

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

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

Generalization error

for one h

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

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

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

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

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:

with probability at least $1-\delta$

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

inconsistent hypothesis case:

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 ϵ_t
- hypothesis space complexity $\ln |\mathcal{H}|$

PAC-learning

Probably approximately correct (PAC):

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

PAC-learning

Probably approximately correct (PAC):

with probability at least $1 - \delta$

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

PAC-learning

Probably approximately correct (PAC):

with probability at least $1-\delta$

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

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.

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)

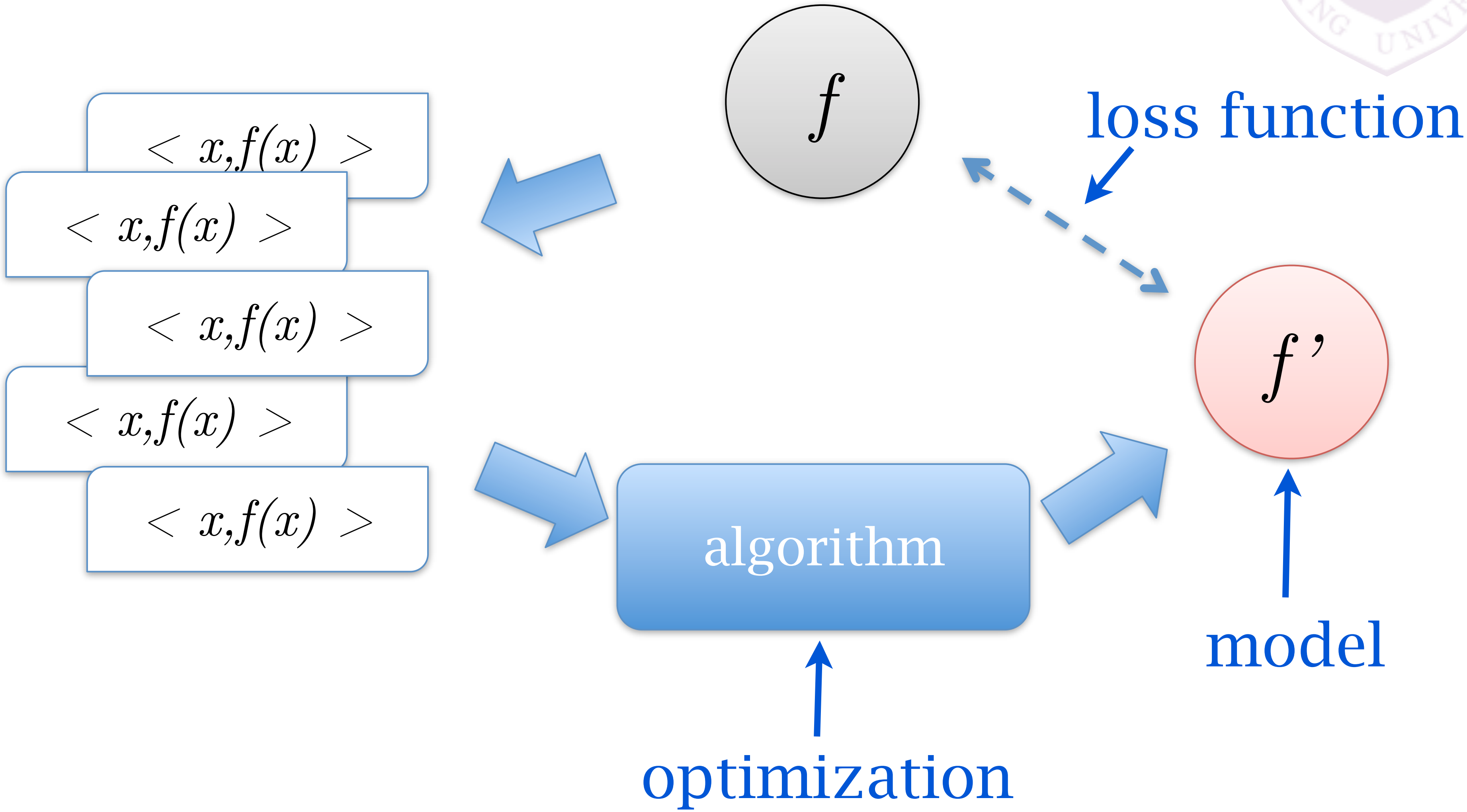
PAC-learnable: [Valiant, 1984]

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.

Dimensions of modeling

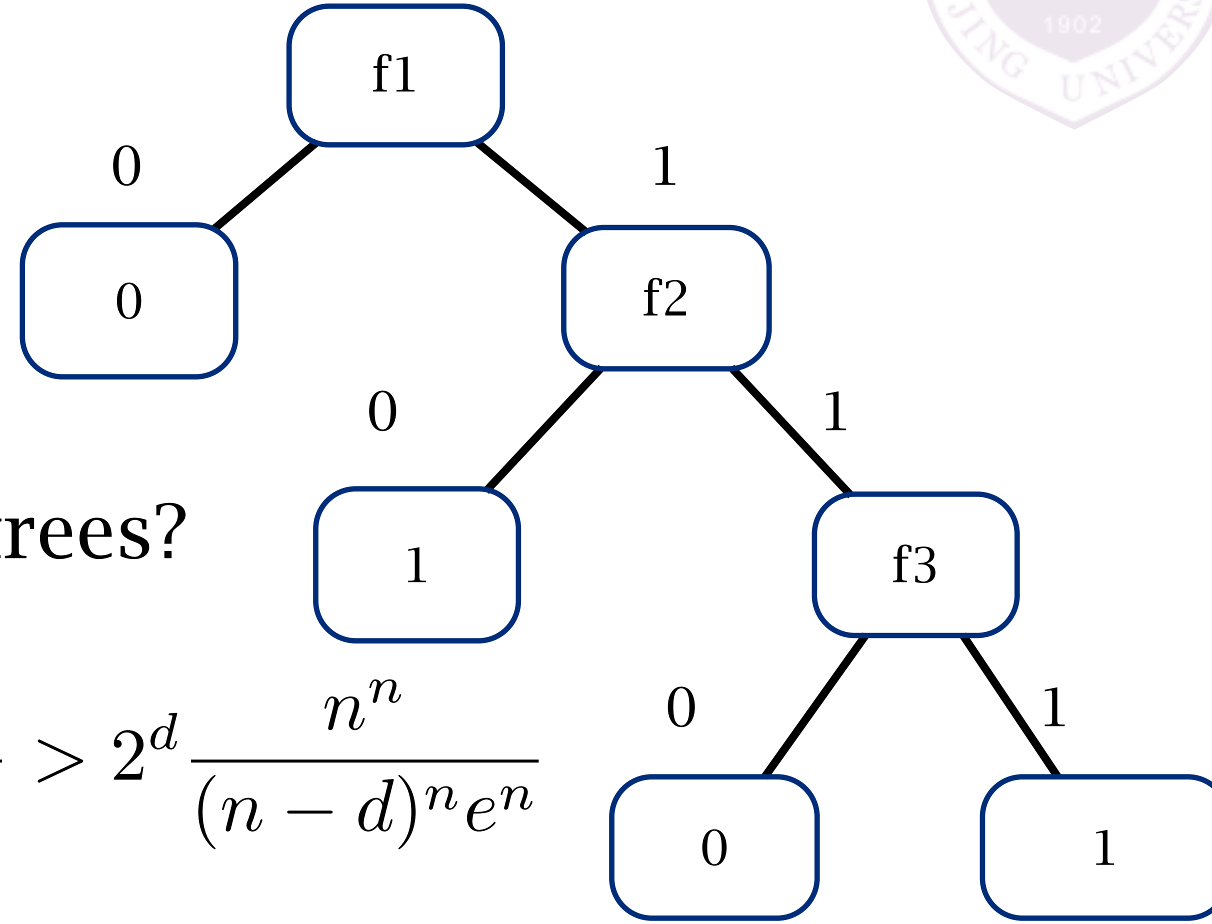


Learning algorithms revisit

Decision Tree

Tree depth and the possibilities

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



How many different trees?

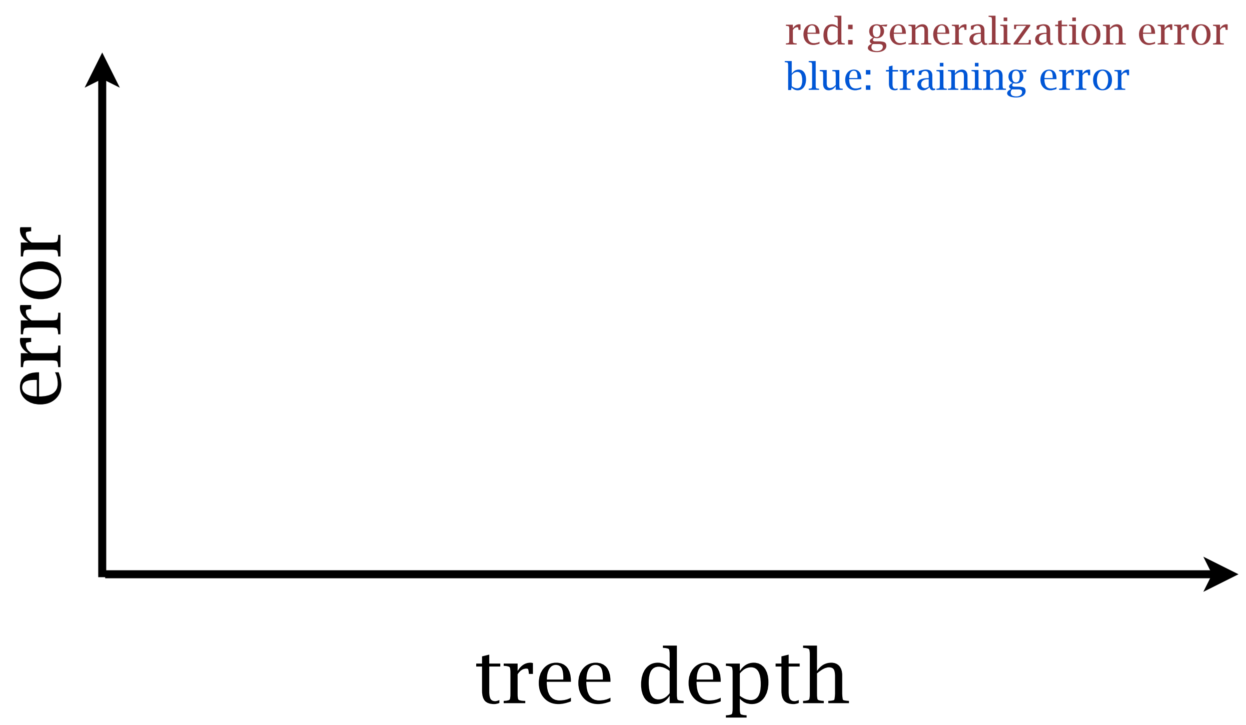
one-branch: $2^d \dfrac{n!}{(n-d)!} > 2^d \dfrac{n^n}{(n-d)^n e^n}$

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

the possibility of trees grows very fast with *d*

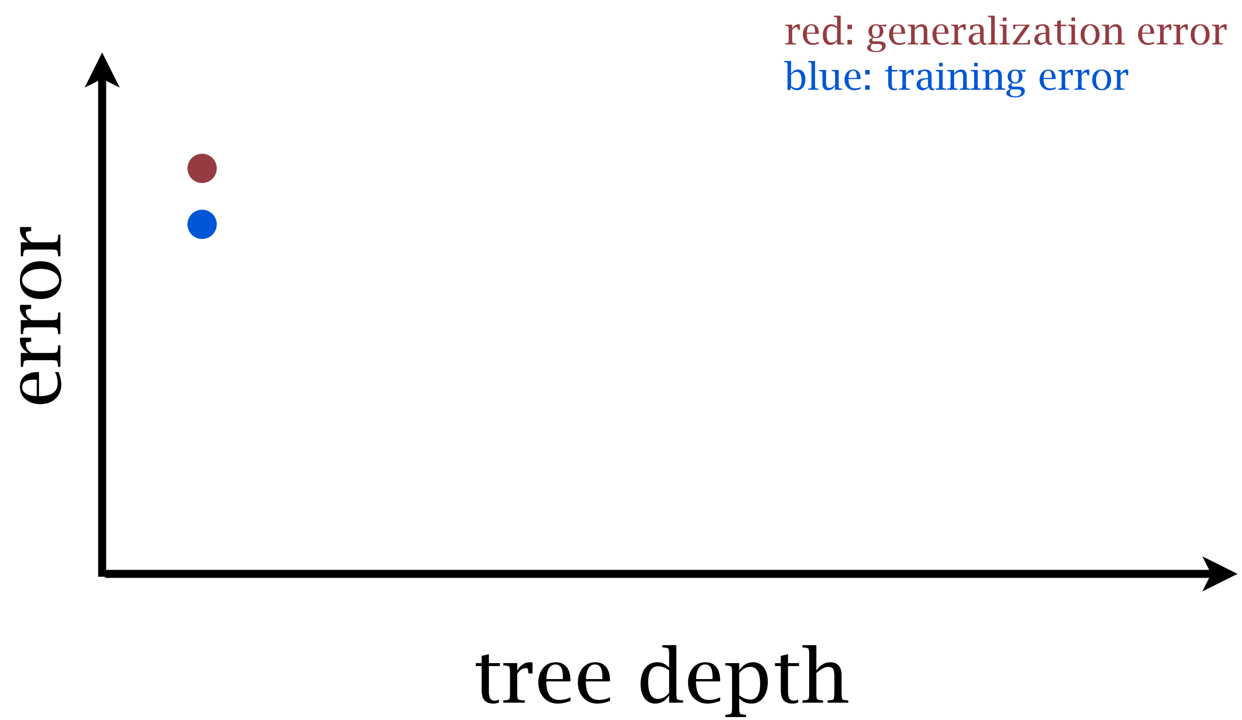
The overfitting phenomena

-- the divergence between infinite and finite samples



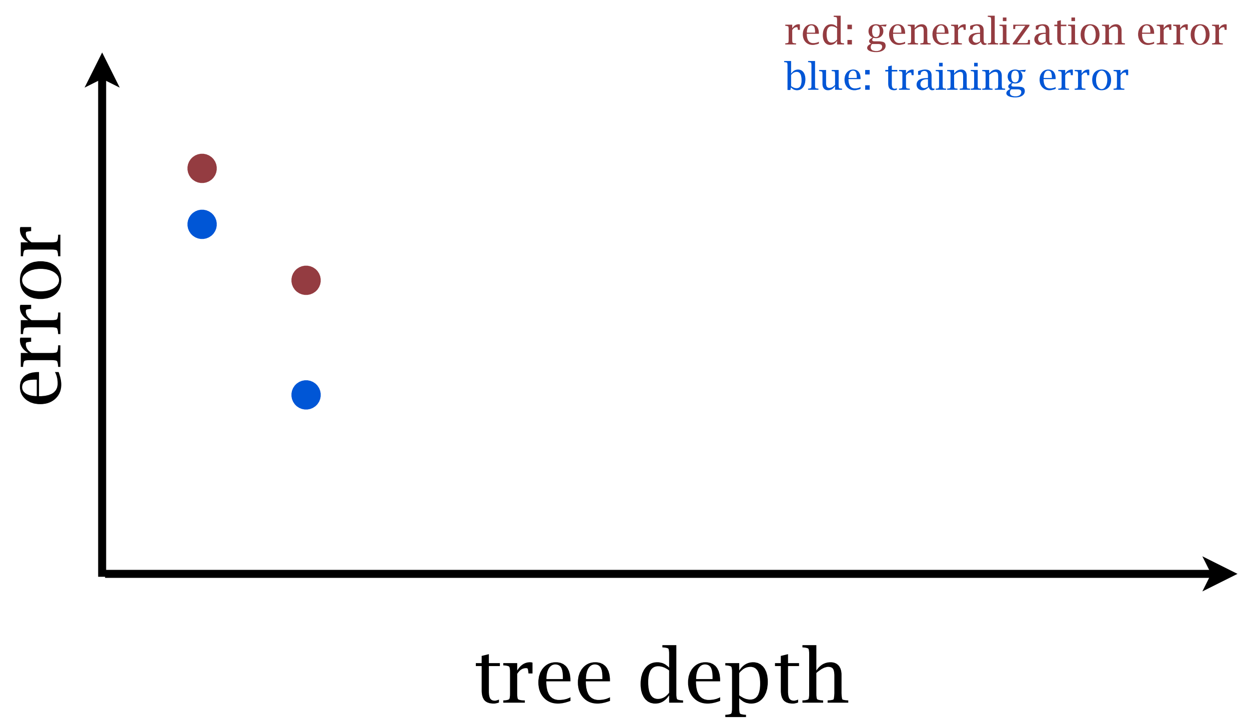
The overfitting phenomena

-- the divergence between infinite and finite samples



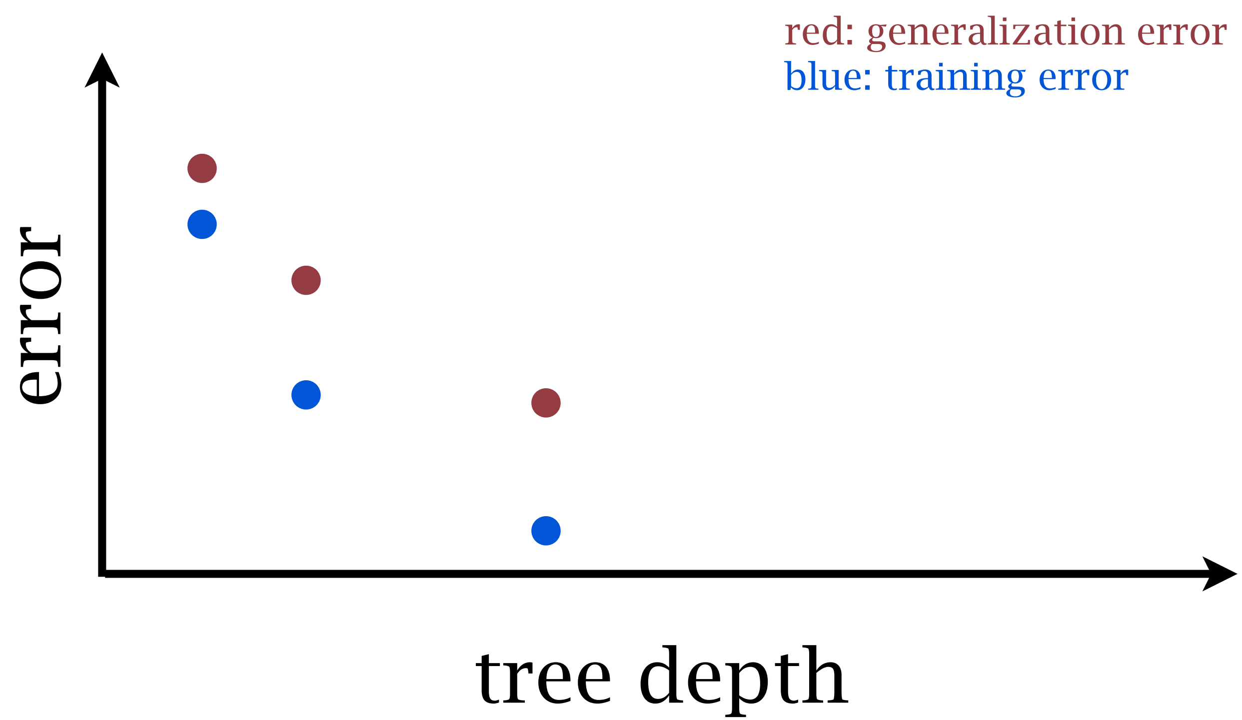
The overfitting phenomena

-- the divergence between infinite and finite samples



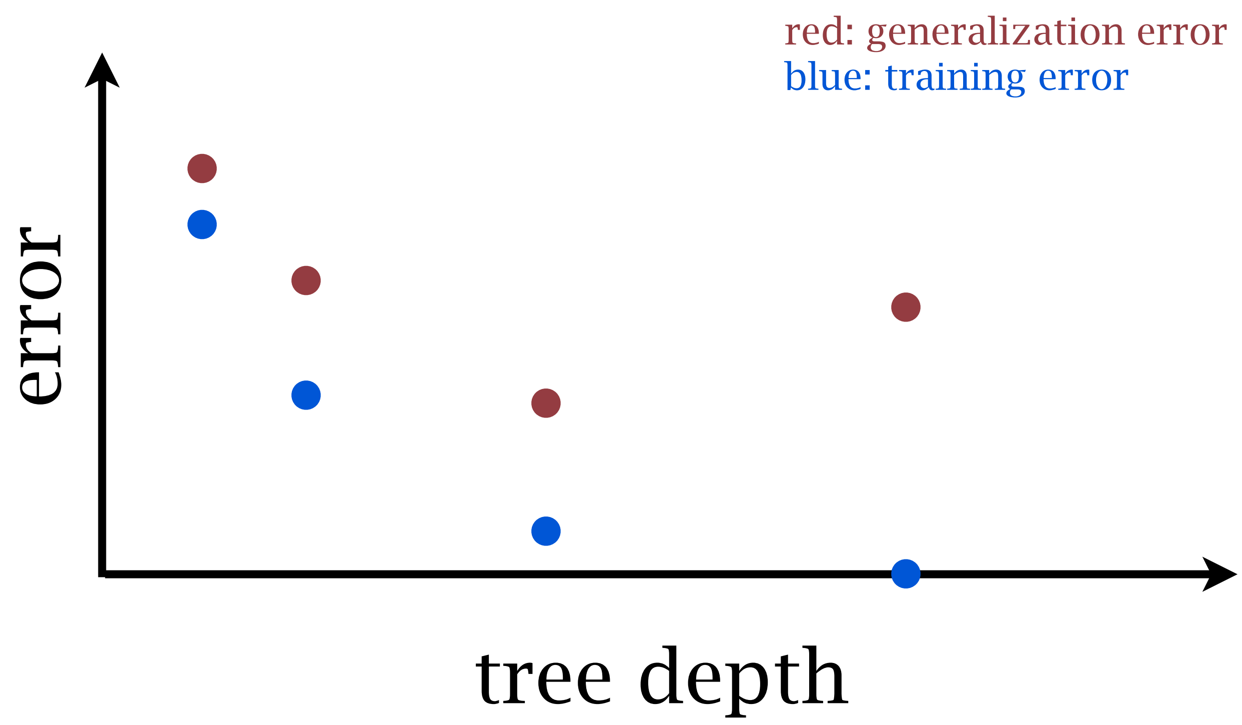
The overfitting phenomena

-- the divergence between infinite and finite samples



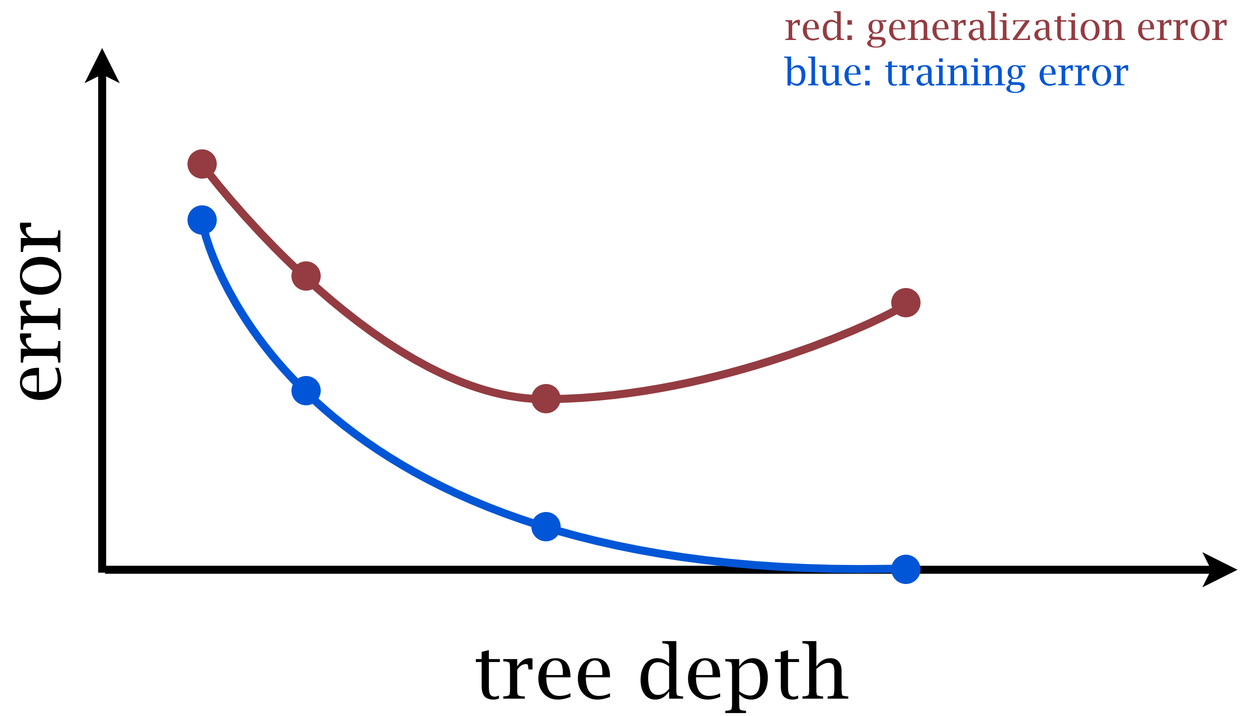
The overfitting phenomena

-- the divergence between infinite and finite samples



The overfitting phenomena

-- the divergence between infinite and finite samples



Pruning

To make decision tree less complex

Pre-pruning: early stop

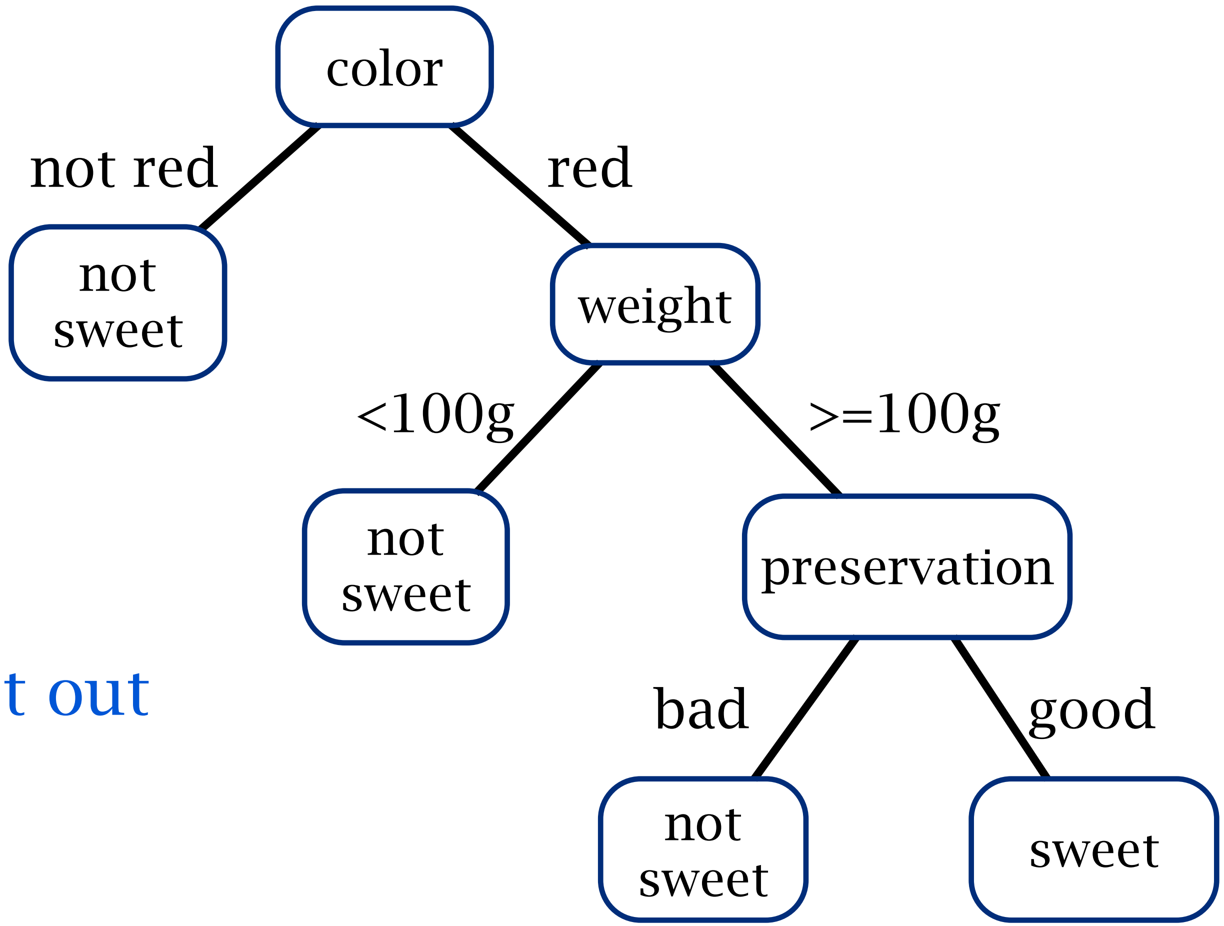
- minimum data in leaf
- maximum depth
- maximum accuracy

Post-pruning: prune full grown DT

reduced error pruning

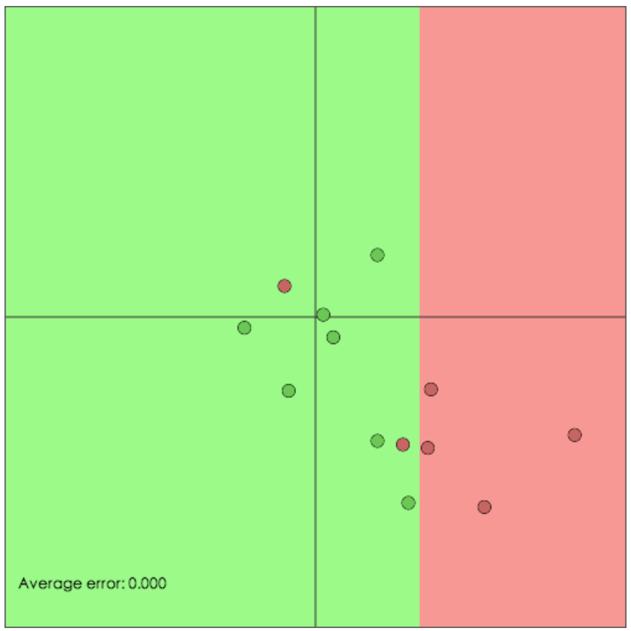
Reduced error pruning

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

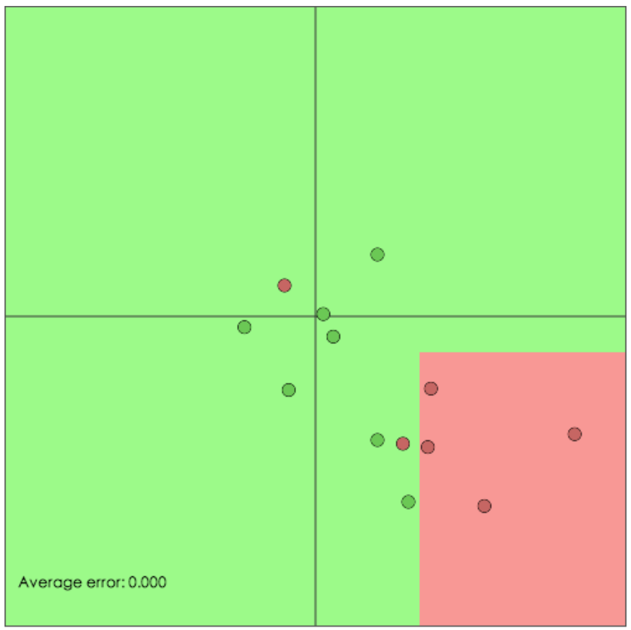


could split a validation set out from the training set to evaluate the error

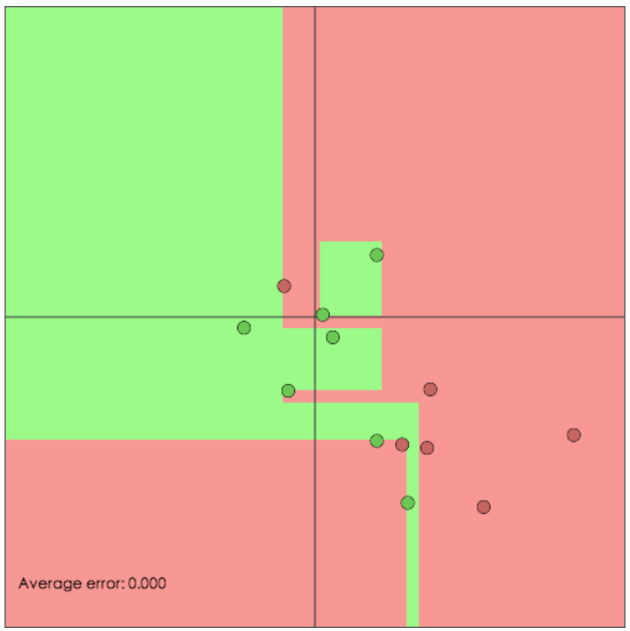
DT boundary visualization



decision stump



max depth=2



max depth=12

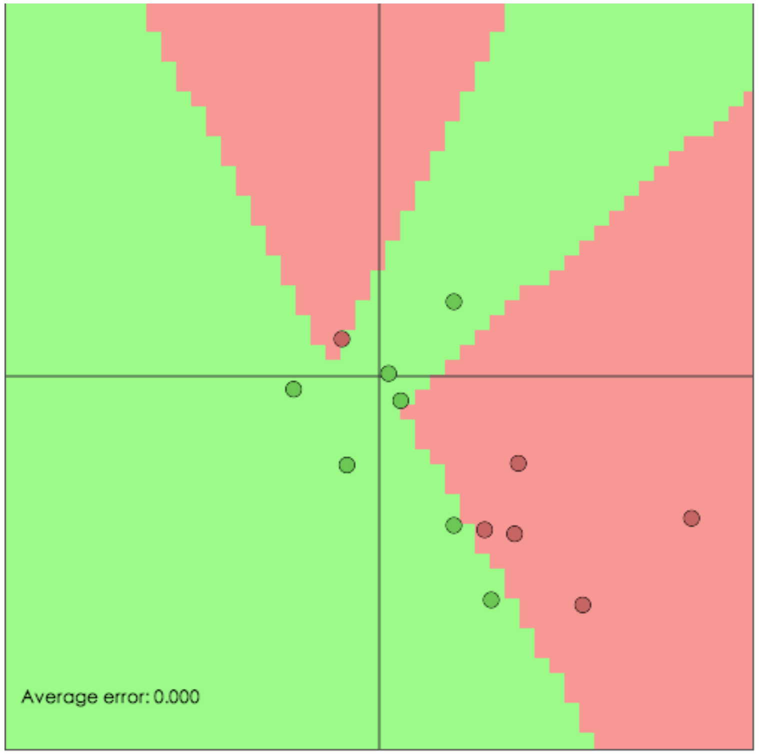
Oblique decision tree

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



Learning algorithms revisit

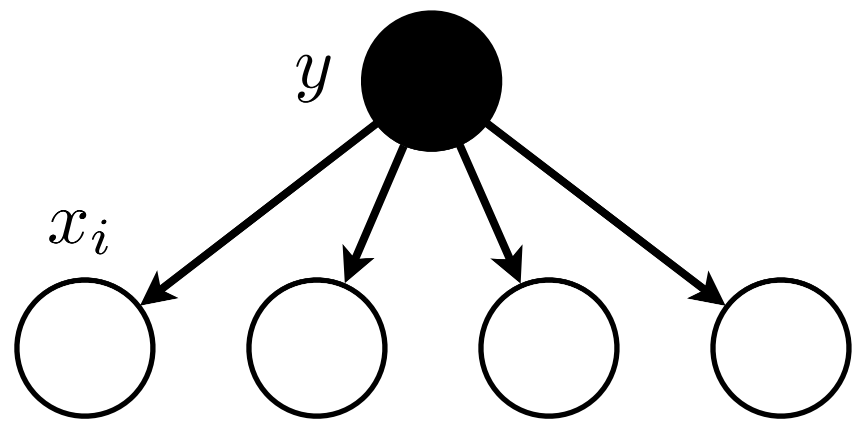
Naive Bayes

Naive Bayes

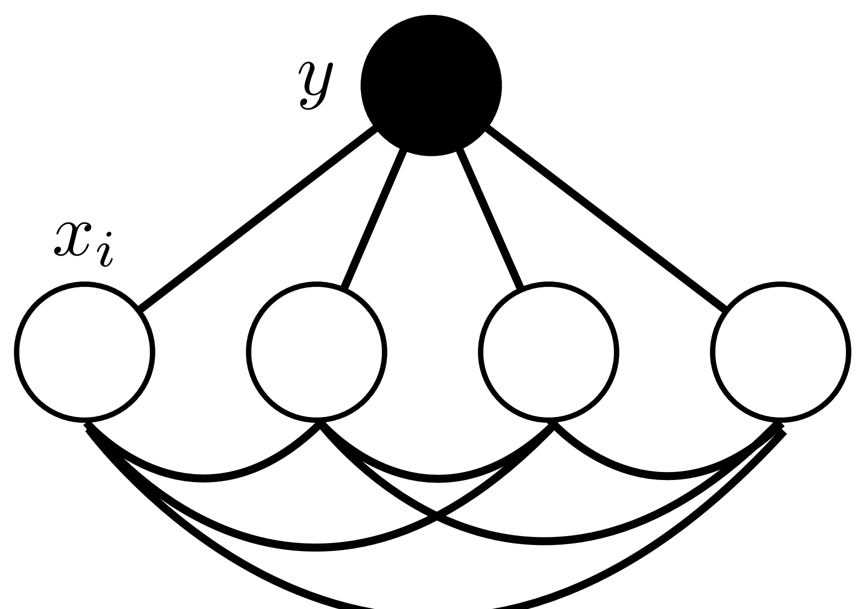
graphic representation

naive Bayes assumption:

$$P(\boldsymbol{x} \mid y) = \prod_i P(x_i \mid y)$$



no assumption:



Relaxation of naive Bayes assumption

assume features are conditional independence given the class

if the assumption holds, naive Bayes classifier will have excellence performance

if the assumption does not hold ...

Relaxation of naive Bayes assumption

assume features are conditional independence given the class

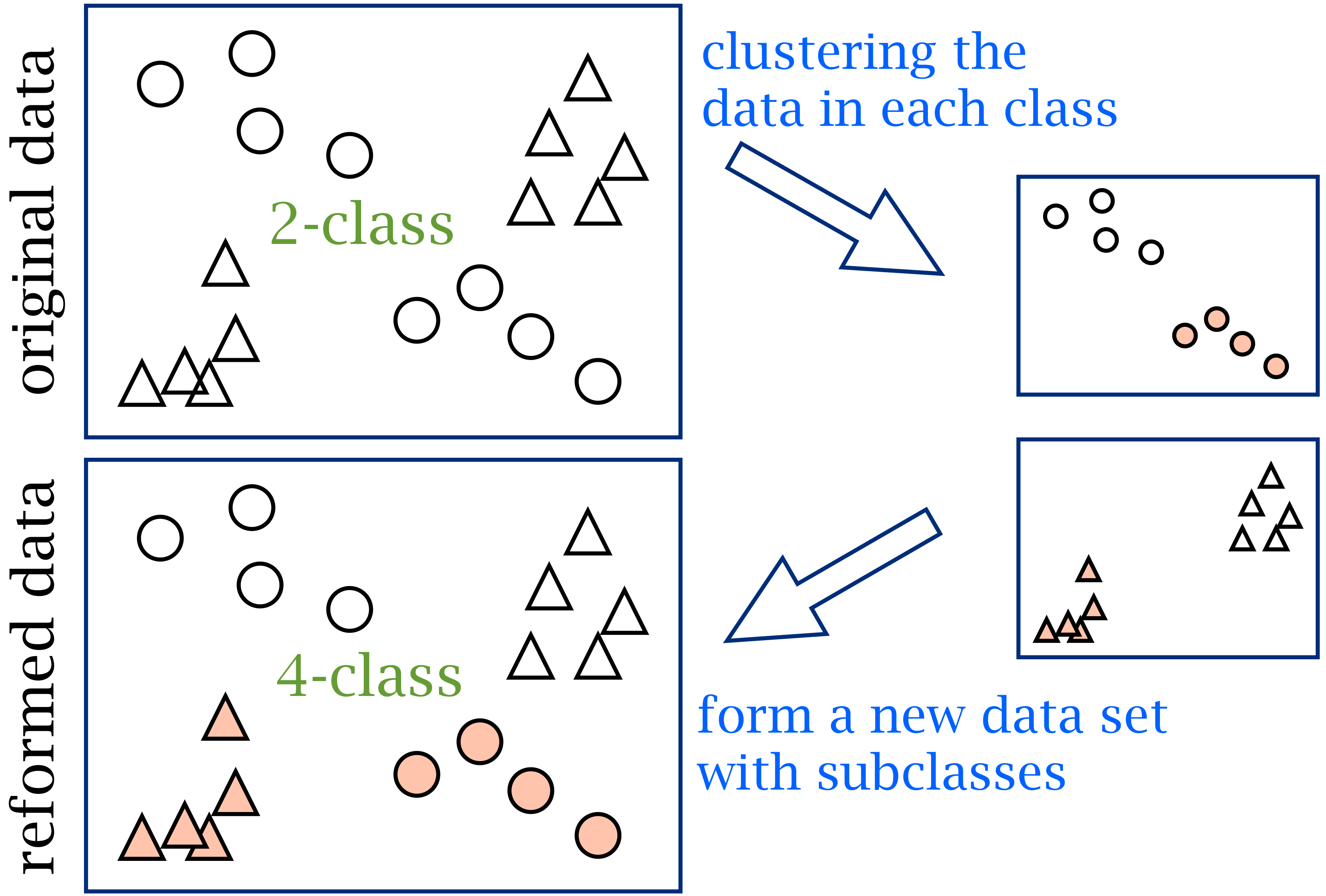
if the assumption holds, naive Bayes classifier will have excellence performance

if the assumption does not hold ...

- Naive Bayes classifier may also have good performance
- Reform the data to satisfy the assumption
- Invent algorithms to relax the assumption

Reform the data

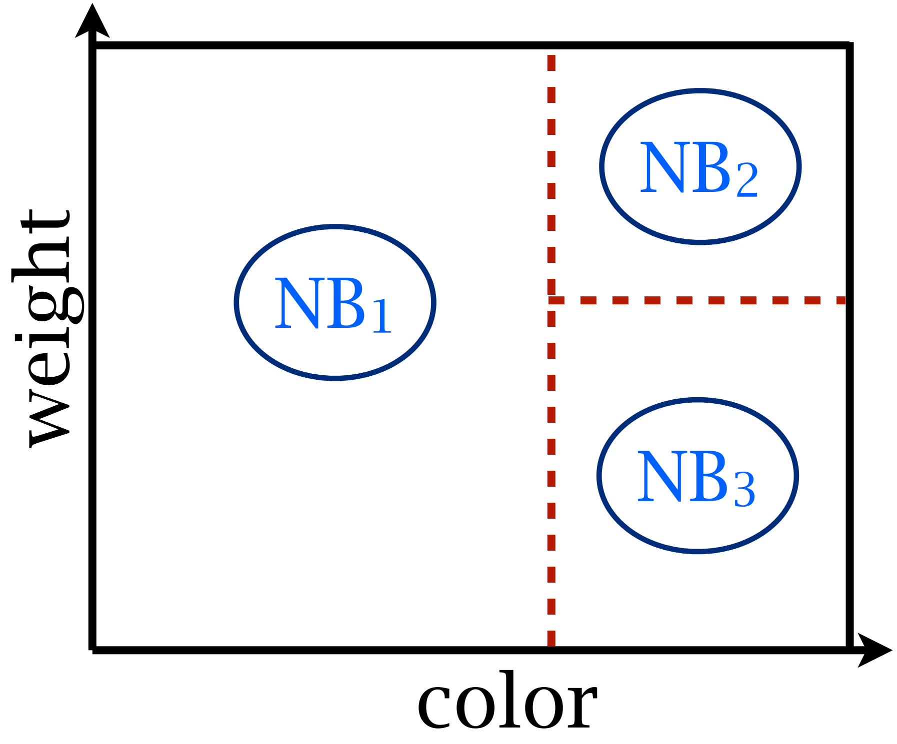
clustering to generate data with subclasses



Semi-naive Bayes classifiers

TreeNB

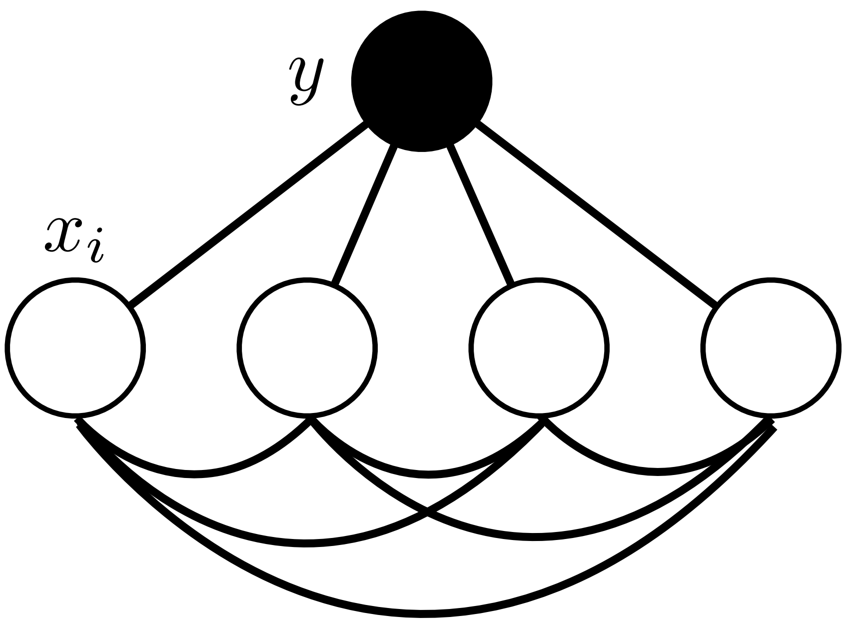
train an NB classifier in each leaf node of a rough decision tree



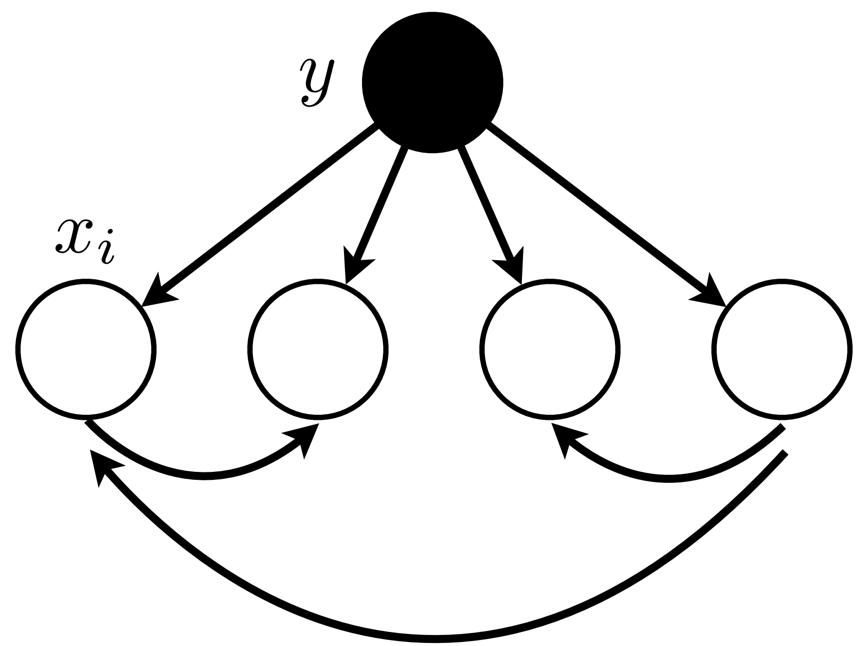
Semi-naive Bayes classifiers

TAN (Tree Augmented NB)

extends NB by allowing every feature to have one more parent feature other than the class, which forms a tree structure



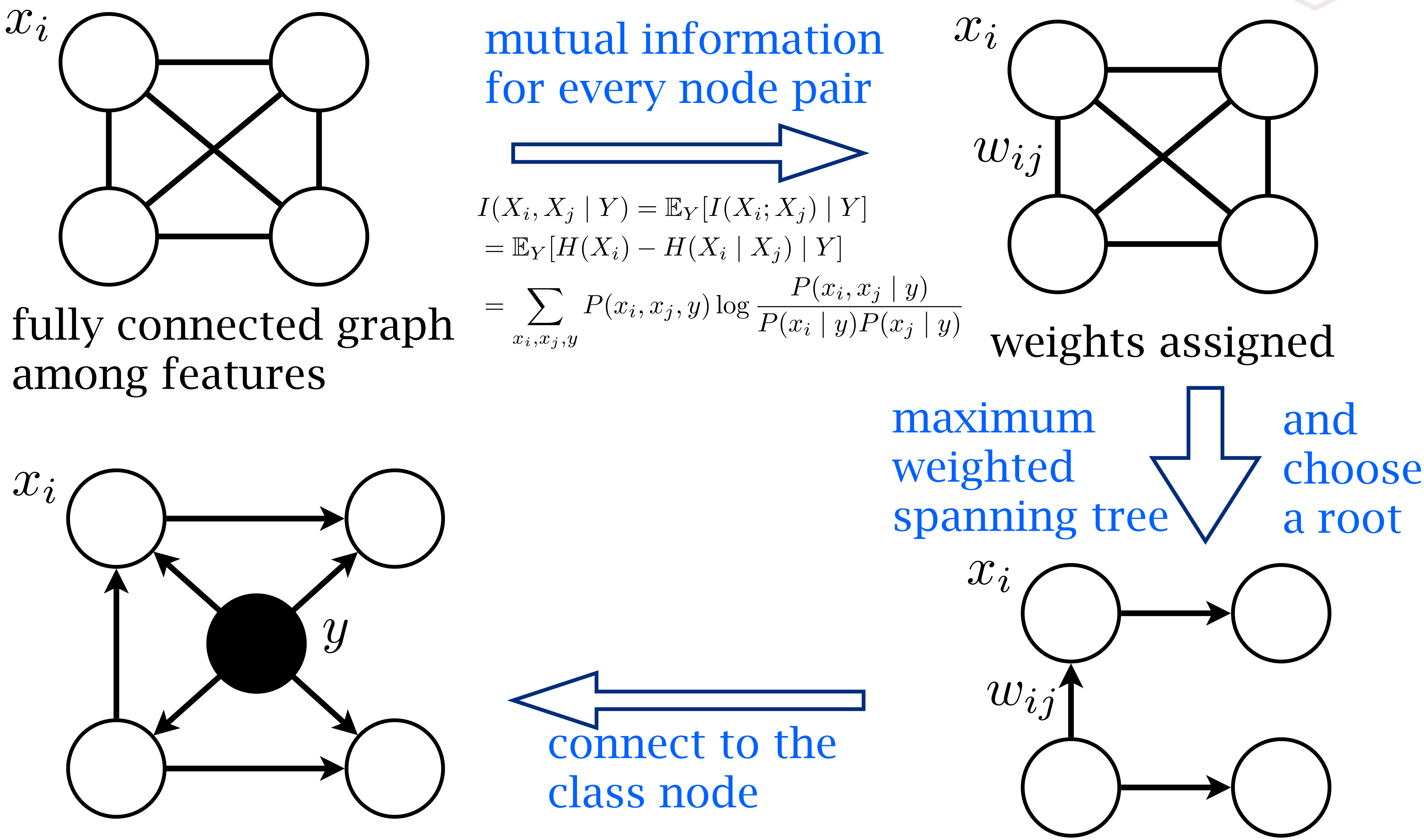
fully connected



TAN

Semi-naive Bayes classifiers

TAN (Tree Augmented NB)



$$
\begin{aligned}
I(X_i, X_j \mid Y) &= \mathbb{E}_Y[I(X_i; X_j) \mid Y] \\
&= \mathbb{E}_Y[H(X_i) - H(X_i \mid X_j) \mid Y] \\
&= \sum_{x_i, x_j, y} P(x_i, x_j, y) \log \frac{P(x_i, x_j \mid y)}{P(x_i \mid y) P(x_j \mid y)}
\end{aligned}
$$

Semi-naive Bayes classifiers

AODE (average one-dependent estimators)

expand a posterior probability with one-dependent estimators (ODEs)

compare with NB:

$$P(\boldsymbol{x} \mid y) = \prod_i P(x_i \mid y)$$

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

$$= P(x_1 \mid y) \prod_i P(x_i \mid x_1, y)$$

- the conditional independency is less important
- harder to estimate (fewer data)

AODE: average ODEs

$$f(x) = \arg\max_y \sum_i I(\text{count}(x_i \geq m)) \cdot \tilde{P}(y) \cdot \tilde{P}(x_i \mid y) \cdot \prod_j \tilde{P}(x_j \mid x_i, y)$$

Handling numerical features

Discretization

recall what we have talked about in Lecture 2

Estimate probability density (P(X) → p(x))

Gaussian model:

$$p(x) = \frac{1}{\sqrt{2\pi\delta^2}} e^{-\frac{(x-\mu)^2}{2\delta^2}}$$

$$p(x_1, \ldots, x_n) = \frac{1}{(2\pi)^{k/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \Sigma^{-1} (\boldsymbol{x}-\boldsymbol{\mu})}$$

training: calculate mean and covariance
test: calculate density

Bayesian networks

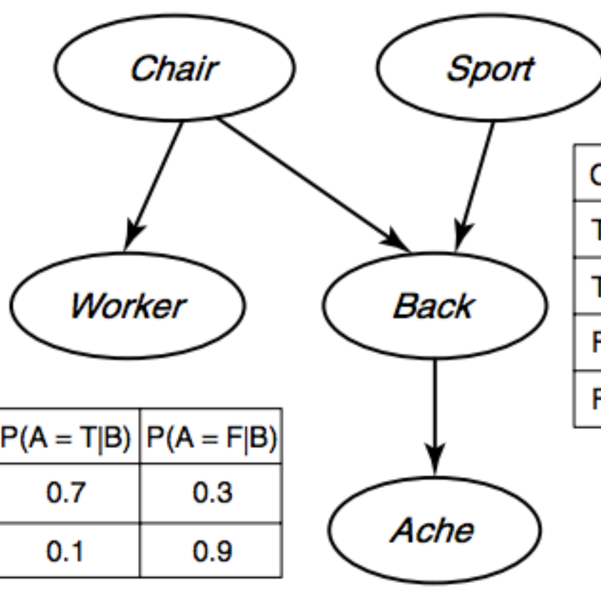
inference in a graphic model representation

a model simplified by conditional independence

a clear description of how things are going

P(C = T)	P(C = F)
0.8	0.2

P(S = T)	P(S = F)
0.02	0.98



C	P(W = T|C)	P(W = F|C)
T	0.9	0.1
F	0.01	0.99

C	S	P(B = T|C,S)	P(B = F|C,S)
T	T	0.9	0.1
T	F	0.2	0.8
F	T	0.9	0.1
F	F	0.01	0.99

B	P(A = T|B)	P(A = F|B)
T	0.7	0.3
F	0.1	0.9

Judea Pearl
Turing Award 2011

"for fundamental contributions to artificial intelligence through the development of a calculus for probabilistic and causal reasoning"

习题

监督学习的目标是否是最小化训练误差?

PAC-learning泛化界对于任意的潜在分布是否都成立?

解释过配(overfitting)和欠配(underfitting)现象。

解释 Bias-Variance 困境。

一数据集用以下两个多项式函数空间都可以得到0训练错误率，使用哪个函数空间的泛化错误可能更低?

$$\mathcal{F}_1 = \{y = a + bx + cx^2 \mid a, b, c \in \mathbb{R}\}$$

$$\mathcal{F}_2 = \{y = a + ax + bx^2 + bx^3 + (a+b)x^4 \mid a, b \in \mathbb{R}\}$$

朴素贝叶斯假设不满足时，朴素贝叶斯的性能一定不好?