Lecture 7: Evaluation

COMP 411, Fall 2021
Victoria Manfredi

WESLEYAN
UNIVERSITY

Acknowledgements: These slides are based primarily on material from the book Machine Learning by Tom Mitchell (and associated slides), the book Machine Learning, An Applied Mathematics Introduction by Paul Wilmott, slides created by Vivek Srikumar (Utah) and Dan Roth (Penn)

Today's Topics

Evaluation

- Cross-validation
- Bias-variance tradeoff

Evaluation

CROSS-VALIDATION

Model selection

Very broadly: choosing the best model using given data

What makes a model:

1. Features
2. Hyper-parameters that control the hypothesis space
   - Example: depth of a decision tree, neural network architecture, etc.
3. The learning algorithm, which may have its own hyperparameters
4. Actual model itself

The learning algorithms we see in this class only find the last one

– What about the rest?

Model selection strategies

Choose model that performs best on a hold-out test dataset

Cross-validation

- estimate model performance using resampling technique

VC dimension and risk minimization

Probabilistic statistical measures

- Akaike Information Criterion (AIC)
- Bayesian Information Criterion (BIC)
- Minimum Description Length (MDL)

Cross-validation

We want to train a classifier using a given dataset

We know how to train given features and hyper-parameters

How do we know what the best feature set and hyper-parameters are?

K-fold cross-validation

Given a particular feature set and hyper-parameter setting

1. Split the data into K (say 5 or 10) equal sized parts

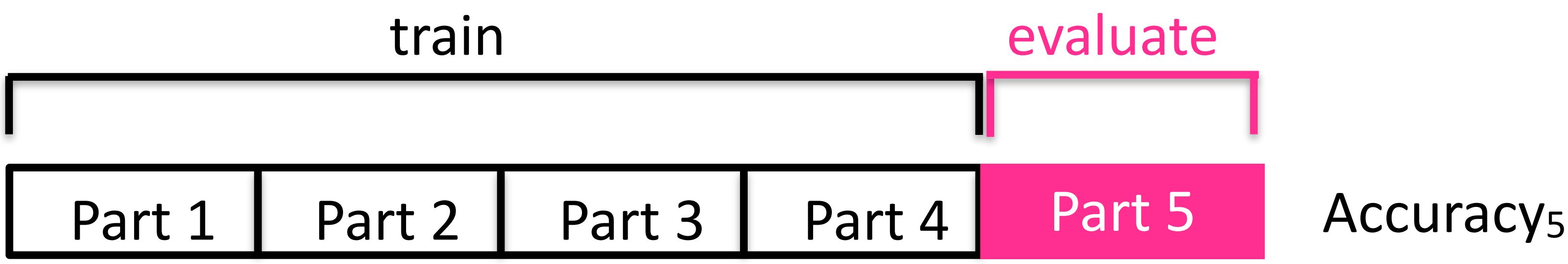
Part 1	Part 2	Part 3	Part 4	Part 5

K-fold cross-validation

Given a particular feature set and hyper-parameter setting

1. Split the data into K (say 5 or 10) equal sized parts

2. Train a classifier on four parts and evaluate it on the fifth one



K-fold cross-validation

Given a particular feature set and hyper-parameter setting

1. Split the data into K (say 5 or 10) equal sized parts

2. Train a classifier on four parts and evaluate it on the fifth one

3. Repeat this using each of the K parts as the validation set

Part 1	Part 2	Part 3	Part 4	**Part 5**	Accuracy$_5$
Part 1	Part 2	Part 3	**Part 4**	Part 5	Accuracy$_4$
Part 1	Part 2	**Part 3**	Part 4	Part 5	Accuracy$_3$
Part 1	**Part 2**	Part 3	Part 4	Part 5	Accuracy$_2$
Part 1	Part 2	Part 3	Part 4	Part 5	Accuracy$_1$

K-fold cross-validation

Given a particular feature set and hyper-parameter setting

1. Split the data into K (say 5 or 10) equal sized parts

2. Train a classifier on four parts and evaluate it on the fifth one

3. Repeat this using each of the K parts as the validation set

4. The quality of this feature set/hyper-parameter is the average of these K estimates

$$\text{Performance} = (\text{Accuracy}_1 + \text{Accuracy}_2 + \text{Accuracy}_3 + \text{Accuracy}_4 + \text{Accuracy}_5) / 5$$

K-fold cross-validation

Given a particular feature set and hyper-parameter setting

1. Split the data into K (say 5 or 10) equal sized parts

2. Train a classifier on four parts and evaluate it on the fifth one

3. Repeat this using each of the K parts as the validation set

4. The quality of this feature set/hyper-parameter is the average of these K estimates

   Performance = $(\text{Accuracy}_1 + \text{Accuracy}_2 + \text{Accuracy}_3 + \text{Accuracy}_4 + \text{Accuracy}_5) / 5$

5. Repeat for every feature set/ hyper-parameter choice

Cross-validation

We want to train a classifier using a given dataset

We know how to train given features and hyper-parameters

How do we know what the best feature set and hyper-parameters are?

1. Evaluate every feature set and hyper-parameter using cross-validation (could be computationally expensive)

2. Pick the best according to cross-validation performance

3. Train on full data using this setting

Evaluation

BIAS AND VARIANCE INFORMALLY

Bias

Every learning algorithm requires assumptions about the hypothesis space.

E.g., “my hypothesis space is

- linear”
- decision trees with 5 nodes”
- a three layer neural network with rectifier hidden units”

Bias

Every learning algorithm requires assumptions about the hypothesis space.

E.g., “my hypothesis space is

- linear”
- decision trees with 5 nodes”
- a three layer neural network with rectifier hidden units”

Bias is the true error (loss) of the best predictor in the hypothesis set

- What will the bias be if the hypothesis set cannot represent the target function? (high or low?): bias will be non-zero, possibly high
- Underfitting: when bias is high

Variance

The performance of a classifier is dependent on the specific training set we have. Perhaps the model will change if we slightly change the training set

Variance: describes how much the best classifier depends on the training set

Overfitting: high variance

Variance:

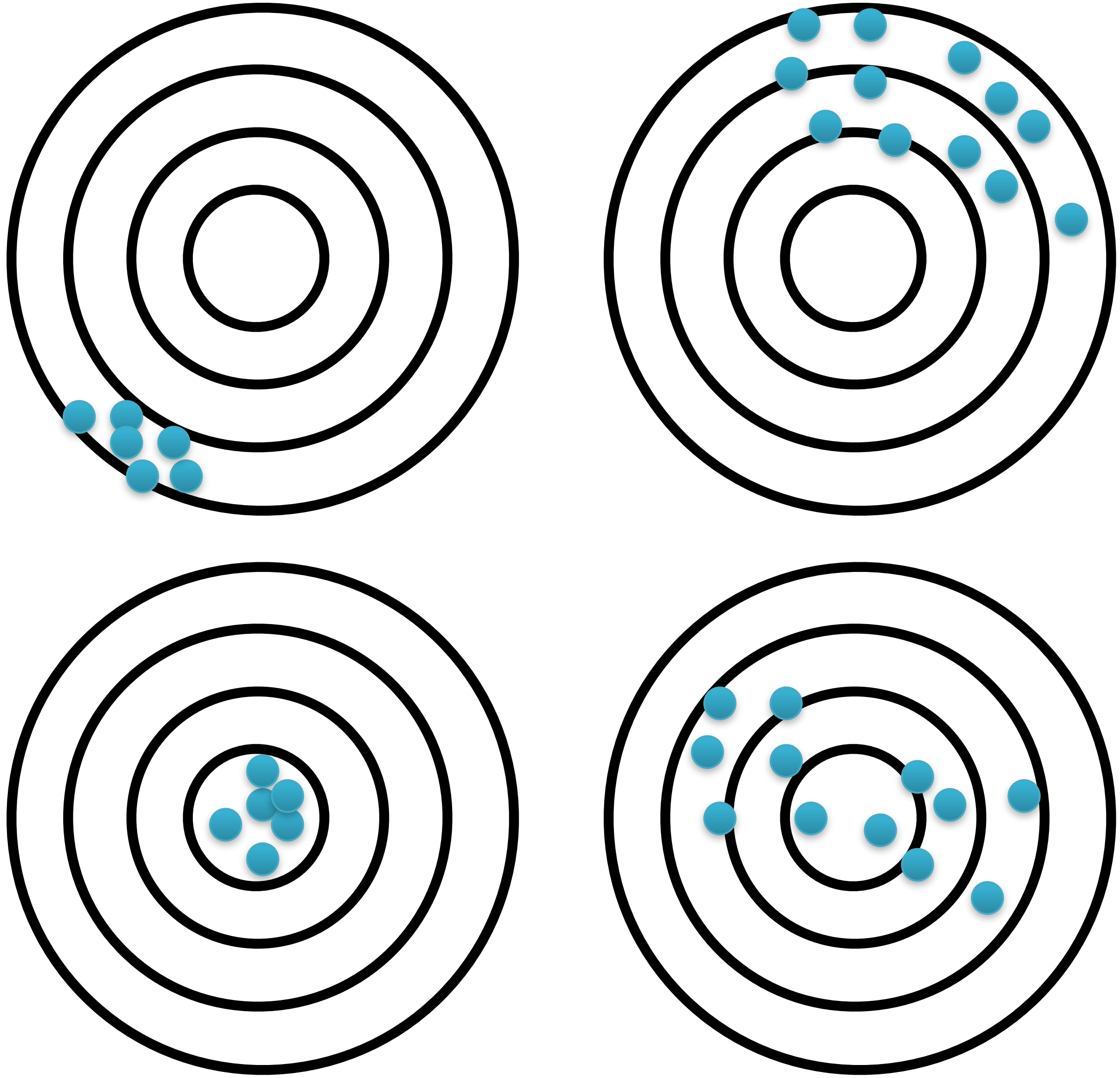
- Increases when the classifiers become more complex
- Decreases with larger training sets

Let’s play darts

Suppose the true concept is the center

High bias

Low bias



Low variance

High variance

Each dot is a model that is learned from a a different dataset

Bias variance tradeoffs

Error = bias + variance (+ noise)

High bias → both training and test error can be high

- Arises when the classifier cannot represent the data

High variance → training error can be low, but test error will be high

- Arises when the learner overfits the training set

Evaluation

BIAS AND VARIANCE FORMALLY

Questions

1. Given a hypothesis h and a data sample containing n examples drawn at random according to the distribution D, what is the best estimate of the accuracy of h over future instances drawn from the same distribution?
    - A robust model would give us the same prediction whatever data we used for training our model

2. What is the probable error in this accuracy estimate?

Some definitions

Expected value or mean of a random variable Y:

$$\mu_y \equiv E[Y] = \sum_i y_i \Pr(Y = y_i)$$

Variance of a random variable Y characterizes the width or dispersion of the distribution around its mean:

$$E[(Y - \mu_y)^2] = \sum_i (y_i - \mu_y)^2 \Pr(Y = y_i)$$

$$Var(Y) = E[(Y - \mu_y)^2] = E[Y^2] - E[Y]^2 = E[Y^2] - \mu_y^2$$

Standard deviation of Y:

$$\sigma_Y \equiv \sqrt{Var(Y)}$$

Some definitions

An estimator is a random variable Y used to estimate some parameter p of an underlying population.

The estimation bias of Y as an estimator for p is the quantity $(E[Y] - p)$. An unbiased estimator is one for which the bias is zero.

A $N\%$ confidence interval estimate for parameter p is an interval that includes p with probability $N\%$

Two definitions of error

The **true error** of hypothesis h with respect to target function f and distribution D is the probability that h will misclassify an instance drawn at random according to D

$$error_D(h) \equiv \Pr_{x \in D}[f(x) \neq h(x)]$$

The notation $\Pr_{x \in D}$ denotes that the probability is taken over distribution D

The **sample error** of hypothesis h with respect to target function f and data sample S is the proportion of examples h misclassifies

$$error_S(h) \equiv \frac{1}{n} \sum_{x \in S} \delta(f(x), h(x))$$

where n is # of examples in S and $\delta(f(x), h(x))$ is 1 if $f(x) \neq h(x)$ and 0 otherwise

Sample error vs. true error

What we'd like to know: true error, $error_D(h)$

What we are able to measure: sample error, $error_S(h)$

- Every time we collect a sample S' containing new randomly drawn examples, we might expect the sample error errors to vary slightly from the sample error $error_S(h)$. We expect a difference due to the random differences in S and S'

Questions:

- How good an estimate of $error_D(h)$ is provided by $error_S(h)$?
- How does the deviation between $error_S(h)$ and$error_D(h)$ depend on the size of the data sample?

Problems estimating error

1. **Bias:** if S is training set, $error_S(h)$ is optimistically biased

$$bias \equiv E[error_S(h)] - error_D(h)$$

For unbiased estimate, h and S must be chosen independently.

2. **Variance**: even with unbiased S, $error_S(h)$ may still vary from $error_D(h)$

Example

Hypothesis h misclassifies 12 of the 40 examples in S

$$error_S(h) = \frac{12}{40} = .30$$

What is $error_D(h)$?

Estimators

Experiment

1. Choose sample S of size n according to distribution D
2. Measure $error_S(h)$

$error_S(h)$ is a random variable (i.e., result of an experiment)

$error_S(h)$ is an unbiased *estimator* for $error_D(h)$

Given observed $error_S(h)$ what can we conclude about $error_D(h)$?

Confidence intervals

If

- S contains n examples, drawn independently of h according to probability distribution D
- $n \geq 30$
- hypothesis h commits r errors over these n examples (i.e., $error_S(h) = r/n$)

Then statistical theory says

- Given no other information, the most probable value of $error_D(h)$ is $error_S(h)$
- With approximately $95\,\%$ probability, the true $error_D(h)$ lies in the interval

$$error_S(h) \pm 1.96\sqrt{\frac{error_S(h)(1 - error_S(h))}{n}}$$

where

$N\,\%$:	$50\,\%$	$68\,\%$	$80\,\%$	$90\,\%$	$95\,\%$	$98\,\%$	$99\,\%$
z_N :	0.67	1.00	1.28	1.64	1.96	2.33	2.58

Binomial random variable

Suppose that n independent trials, each of which results in a "success" with probability p and in a "failure" with probability $1 - p$, are to be performed. If X represents the number of successes that occur in the n trials, then X is said to be a binomial random variable with parameters (n, p).

The probability mass function of a binomial random variable having parameters (n, p) is given by

$$p(i) = \binom{n}{i} p^i (1-p)^{n-i}, \qquad i = 0,1,\ldots,n$$

where

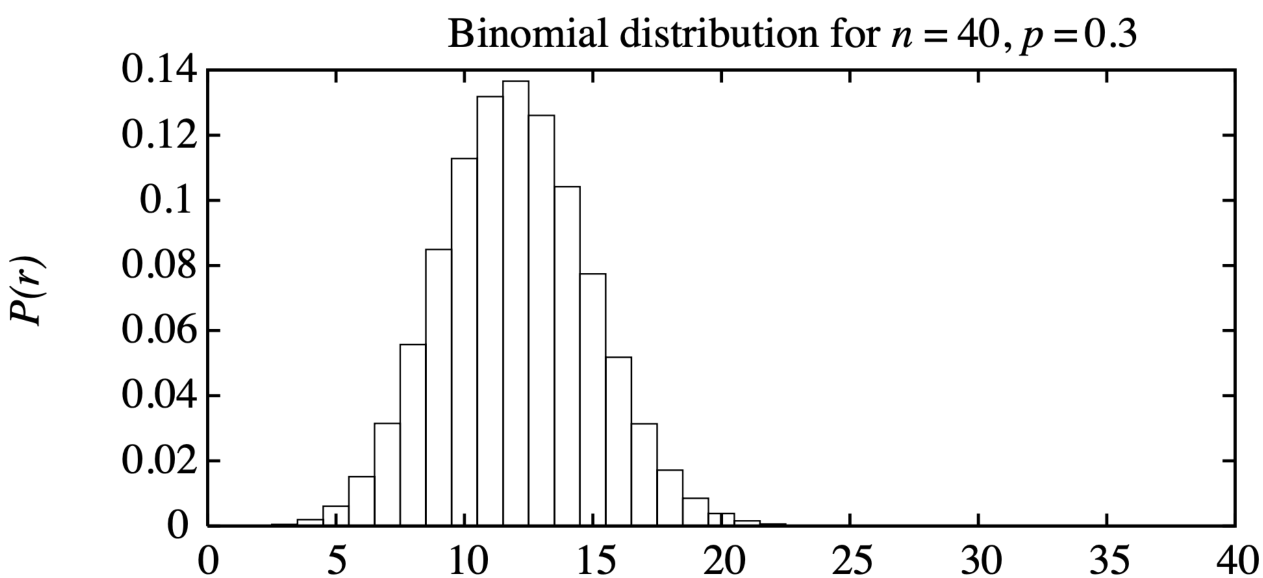
$$\binom{n}{i} = \frac{n!}{(n-i)!i!}$$

equals the number of different groups of i objects that can be chosen from a set of n objects. The validity of this equation may be verified by first noting that the probability of any particular sequence of the n outcomes containing i successes and $n - i$ failures is, by the assumed independence of trials, $p^i(1-p)^{n-i}$.

$error_S(h)$ is a binomial random variable

Rerun the experiment with different randomly drawn S (of size n)

Probability of observing r misclassified examples:



Histogram of frequency with which we observe each possible sample error

$$P(r) = \frac{n!}{r!(n-r)!} error_D(h)^r (1 - error_D(h))^{n-r}$$

$error_S(h)$ is a binomial random variable

$$P(r) = \frac{n!}{r!(n-r)!} error_D(h)^r (1 - error_D(h))^{n-r}$$

Probability $P(r)$ of r heads in n coin flips, if $p = \Pr(heads)$

- Expected, or mean value of X, $E[X]$, is

$$E[X] \equiv \sum_{i=0}^{n} iP(i) = np$$

- Variance of X is

$$Var(X) \equiv E[(X - E[X])^2] = np(1-p)$$

- Standard deviation of X, σ_X, is

$$\sigma_X \equiv \sqrt{E[(X - E[X])^2]} = \sqrt{np(1-p)}$$

Normal distribution approximates binomial

$error_S(h)$ follows a binomial distribution with

mean $\mu_{error_S}(h) = error_D(h)$

standard deviation $\sigma_{error_S}(h) = \sqrt{\frac{error_D(h)(1 - error_D(h)}{n}}$

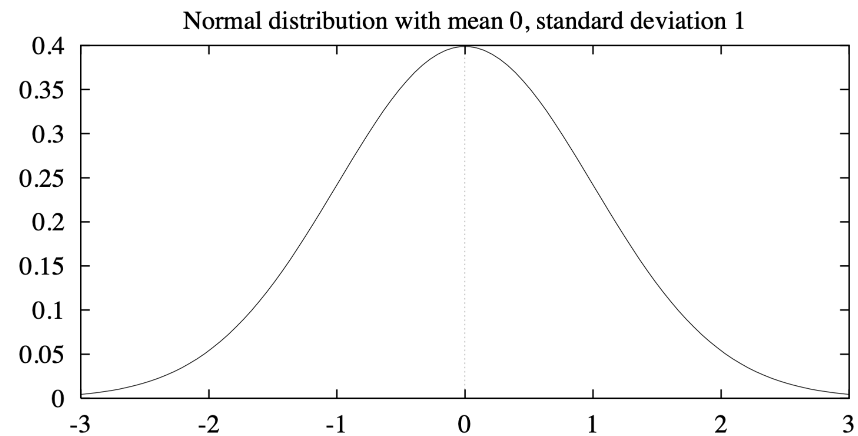
Approximate this by a normal distribution with

mean $\mu_{error_S}(h) = error_D(h)$

standard deviation $\sigma_{error_S}(h) \approx \sqrt{\frac{error_S(h)(1 - error_S(h)}{n}}$

Normal probability distribution

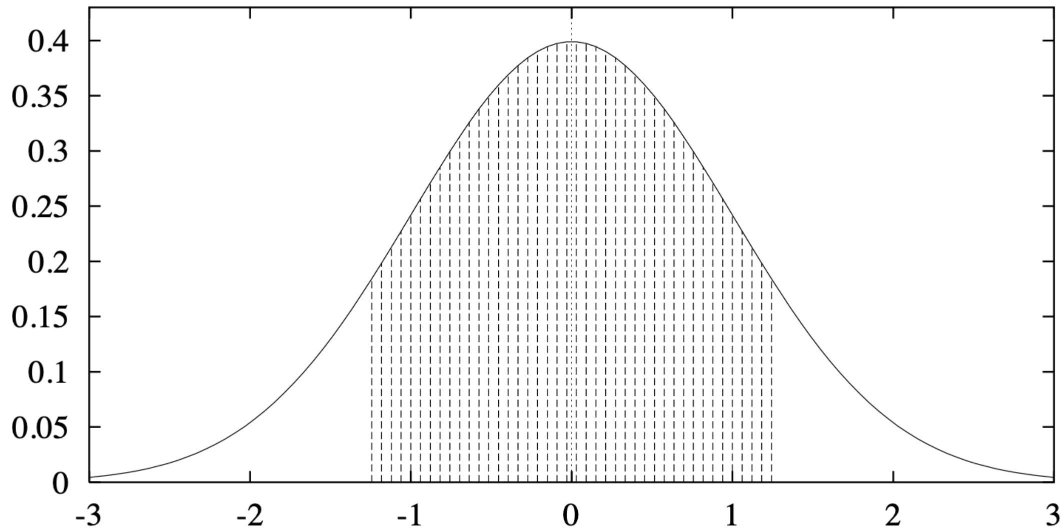
$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



The probability that X will fall into the interval (a, b) is given by $\int_a^b p(x)dx$

- Expected, or mean value of X, $E[X]$, is $E[X] = \mu$
- Variance of X is $Var(X) = \sigma^2$
- Standard deviation of X, σ_X, is $\sigma_X = \sigma$

Normal probability distribution



$80\,\%$ of area (probability) lies in $\mu \pm 1.28\sigma$

$N\,\%$ of area (probability) lies in $\mu \pm z_N\sigma$

where

$N\,\%$:	$50\,\%$	$68\,\%$	$80\,\%$	$90\,\%$	$95\,\%$	$98\,\%$	$99\,\%$
z_N :	0.67	1.00	1.28	1.64	1.96	2.33	2.58

Confidence intervals more correctly

If

- S contains n examples, drawn independently of h according to probability distribution D
- $n \geq 30$
- hypothesis h commits r errors over these n examples (i.e., $error_S(h) = r/n$)

Then statistical theory says

- With approximately $95\,\%$ probability, the sample error, $error_S(h)$ lies in the interval

$$error_D(h) \pm 1.96\sqrt{\frac{error_D(h)(1 - error_D(h))}{n}}$$

Use normal standard deviation in confidence interval not
binomial standard deviation, since approximating with normal.

equivalently, the true error, $error_D(h)$ lies in the interval

$$error_S(h) \pm 1.96\sqrt{\frac{error_D(h)(1 - error_D(h))}{n}}$$

which is approximately

$$error_S(h) \pm 1.96\sqrt{\frac{error_S(h)(1 - error_S(h))}{n}}$$

Central Limit Theorem

Consider a set of independent, identically distributed random variables $Y_1 \ldots Y_n$ all governed by an arbitrary probability distribution with mean μ and finite variance σ^2. Define the sample mean,

$$\bar{Y} \equiv \frac{1}{n} \sum_{i=1}^{n} Y_i$$

Central Limit Theorem. As $n \to \infty$, the distribution governing $\bar{Y}$ approaches a normal distribution, with mean μ and variance $\frac{\sigma^2}{n}$

Calculating confidence intervals

1. Pick parameter p to estimate: $error_D(h)$
2. Choose an estimator: $error_S(h)$
3. Determine probability distribution that governs estimator: $error_S(h)$ governed by Binomial distribution approximated by Normal when $n \geq 30$
4. Find interval (L, U) such that $N\%$ of probability mass falls in the interval: use table of z_N values

Difference between hypotheses

Test h_1 on sample S_1, test h_2 on S_2

1. Pick parameter to estimate:

$$d \equiv error_D(h_1) - error_D(h_2)$$

2. Choose an estimator

$$\hat{d} \equiv error_{S_1}(h_1) - error_{S_2}(h_2)$$

Since the sum of two independent normal distributed random variables is normal: mean is the sum of the two means, variance is the sum of the two variances

3. Determine probability distribution that governs estimator

$$\sigma_{\hat{d}}^2 \approx \sqrt{\frac{error_{S_1}(h_1)(1 - error_{S_1}(h_1))}{n_1} + \frac{error_{S_2}(h_2)(1 - error_{S_2}(h_2))}{n_2}}$$

4. Find interval (L, U) such that $N\%$ of probability mass falls in the interval

$$\hat{d} \pm z_n \sqrt{\frac{error_{S_1}(h_1)(1 - error_{S_1}(h_1))}{n_1} + \frac{error_{S_2}(h_2)(1 - error_{S_2}(h_2))}{n_2}}$$

Aside

It can be shown that the difference between the sample errors, $\hat{d}$, gives an unbiased estimate of d, that is $E[\hat{d}] = d$

What is the probability distribution governing the random variable $\hat{d}$? For large n_1 and n_2 (e.g., both ≥ 30, both $error_{S_1}(h_1)$ and $error_{S_2}(h_2)$ follow distributions that are approximately Normal.

Because the difference of two Normal distributions is also a Normal distribution, $\hat{d}$ will also follow a distribution that is approximately Normal, with mean d. It can also be shown that the variance of this distribution is the sum of the variances of $error_{S_1}(h_1)$ and $error_{S_2}(h_2)$

Causes of estimation error

Bias: If Y is an estimator for some parameter p, the estimation bias of Y is the difference between p and the expected value of Y. For example, if S is the training data used to formulate hypothesis h, then $error_S(h)$ gives an optimistically biased estimate of the true error $error_D(h)$

Variance: Even with an unbiased estimator, the observed value of the estimator is likely to vary from one experiment to another. The variance σ^2 of the distribution governing the estimator characterizes how widely this estimate is likely to vary form the correct value. This variance decreases as the size of the data sample is increased.

Evaluation

BIAS AND VARIANCE INTUITION

The i.i.d. assumption

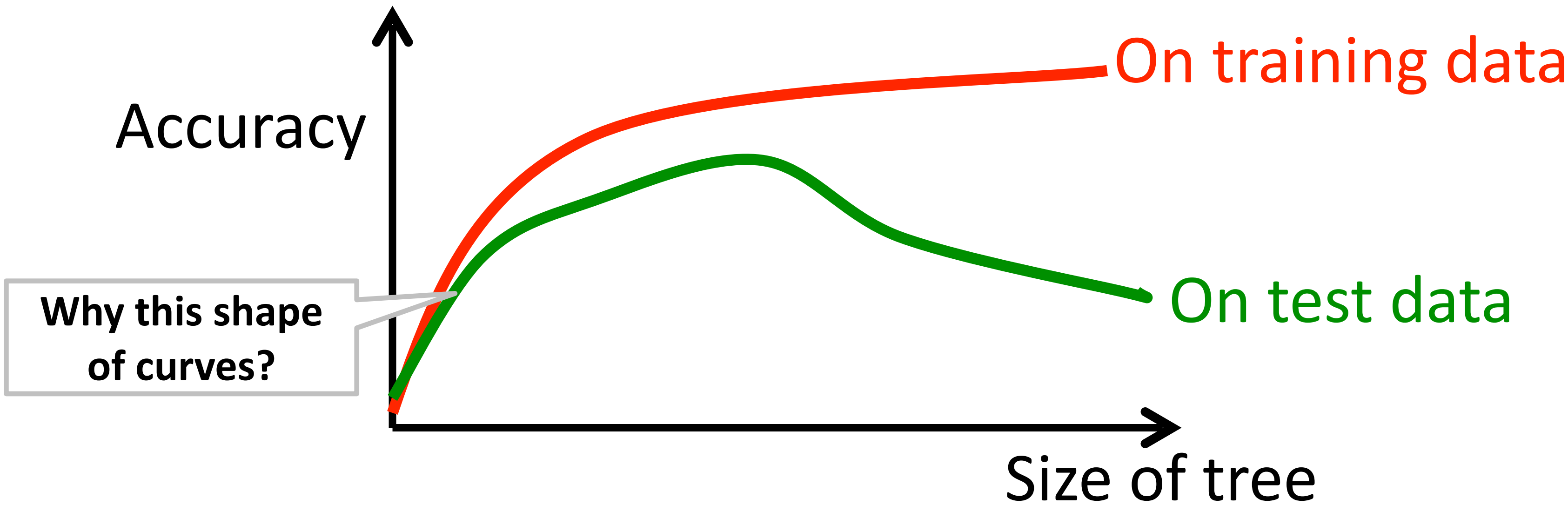
Training and test items are independently and identically distributed (i.i.d.):

- There is a distribution $P(\mathbf{X}, Y)$ from which the data $D = \{(\mathbf{x}, y)\}$ is generated. Sometimes it's useful to rewrite $P(\mathbf{X}, Y)$ as $P(\mathbf{X})P(Y \mid \mathbf{X})$. Usually $P(\mathbf{X}, Y)$ is unknown to us (we just know it exists)

- Training and test data are samples drawn from the *same* $P(\mathbf{X}, Y)$: they are identically distributed

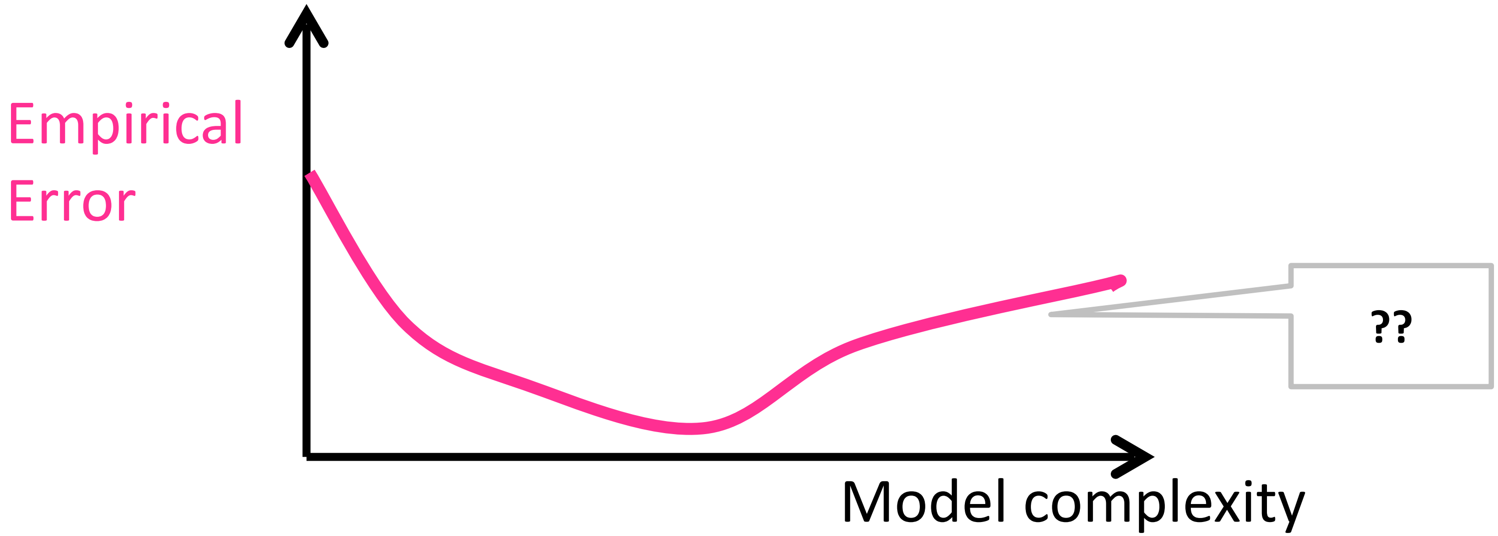
- Each $(\mathbf{x}, y)$ is drawn independently from $P(\mathbf{X}, Y)$

Overfitting



A decision tree overfits the training data when its accuracy on the training data goes up but its accuracy on unseen data goes down

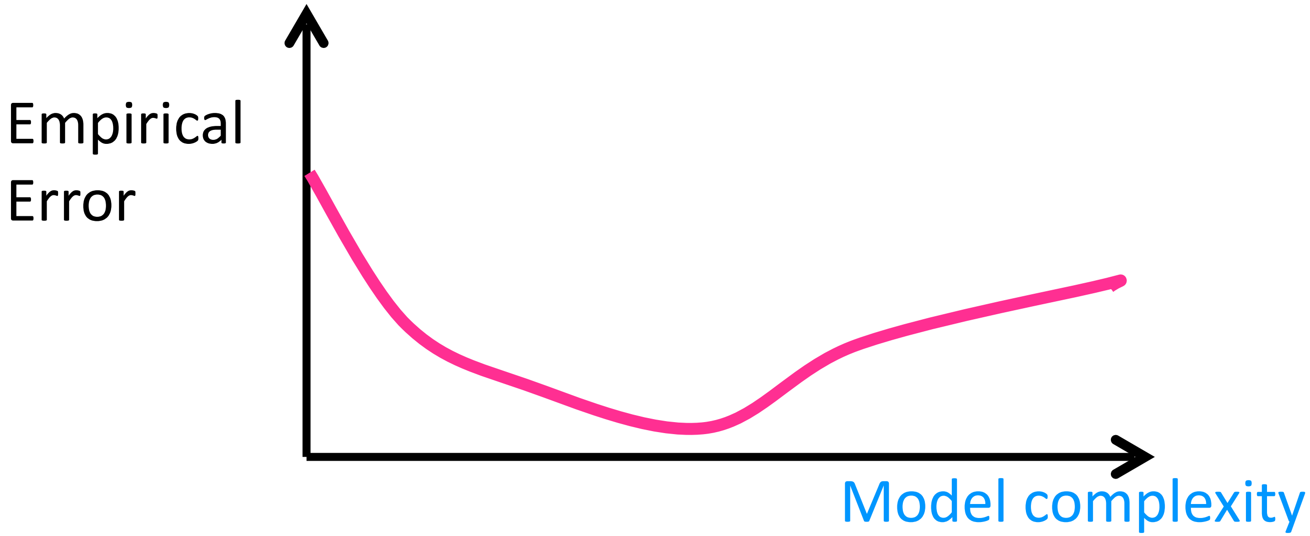
Overfitting



Empirical error (= on a given data set):

The percentage of items in this data set are misclassified by the classifier *f*.

Overfitting

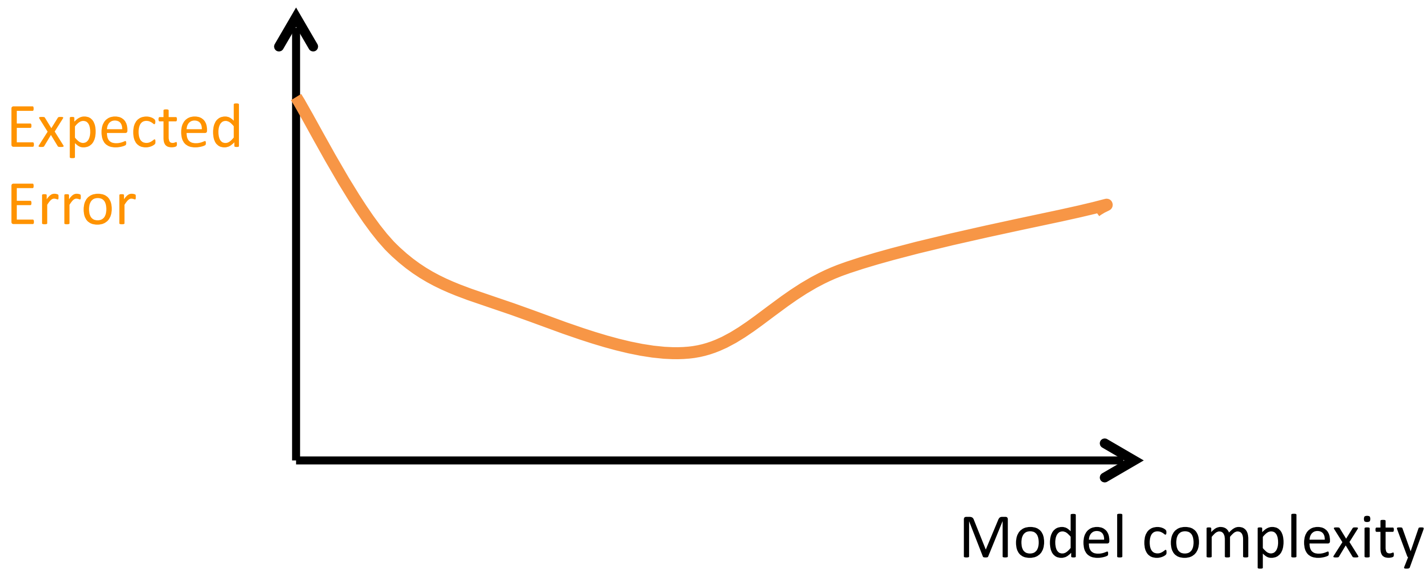


Model complexity (informally):

How many parameters do we have to learn?

Decision trees: complexity = # of nodes

Overfitting

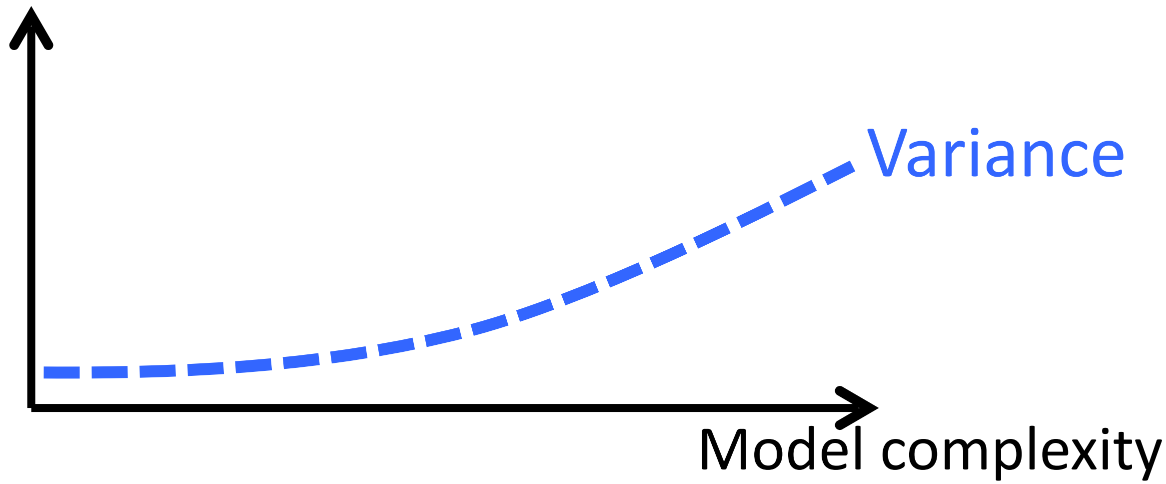


Expected error:

What percentage of items drawn from $P(\mathbf{x}, y)$ do we expect to be misclassified by f?

(That's what we really care about – **generalization**)

Variance of a learner (informally)



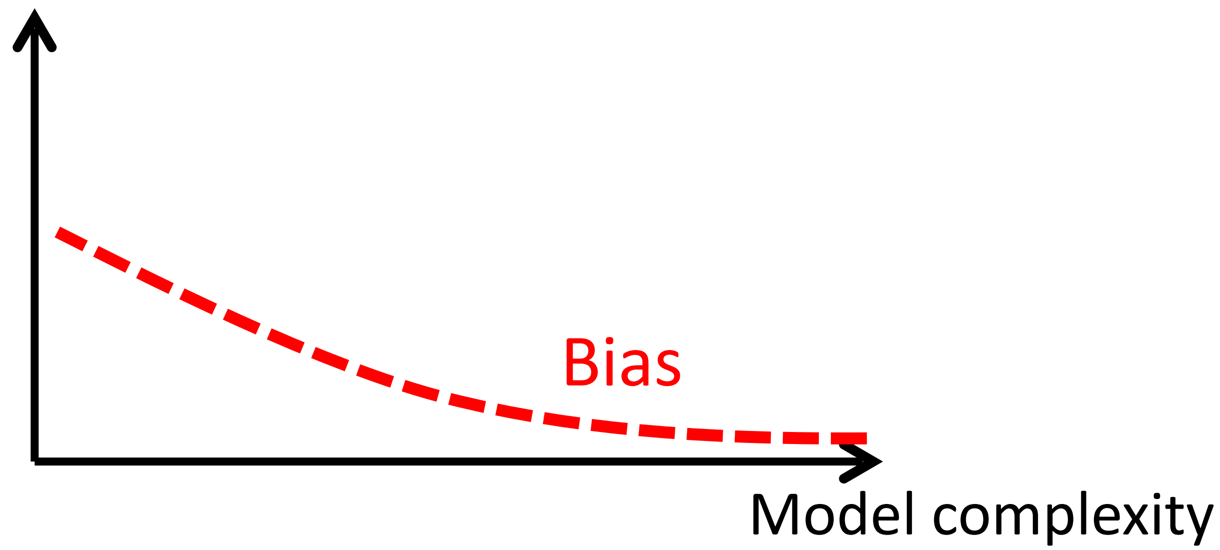
How susceptible is the learner to minor changes in the training data?

- (i.e. to different samples from $P(\mathbf{x}, y)$)

Variance increases with model complexity

- Think about extreme cases: a hypothesis space with one function vs. all functions.
- Or, adding the “wind” feature in the decision tree earlier.
- The larger the hypothesis space is, the more flexible the selection of the chosen hypothesis is as a function of the data.
- More accurately: for each sample data set D, you will learn a different hypothesis $h(D)$, that will have a different sample error $e(h)$; we are looking here at the variance of this random variable from the true error.

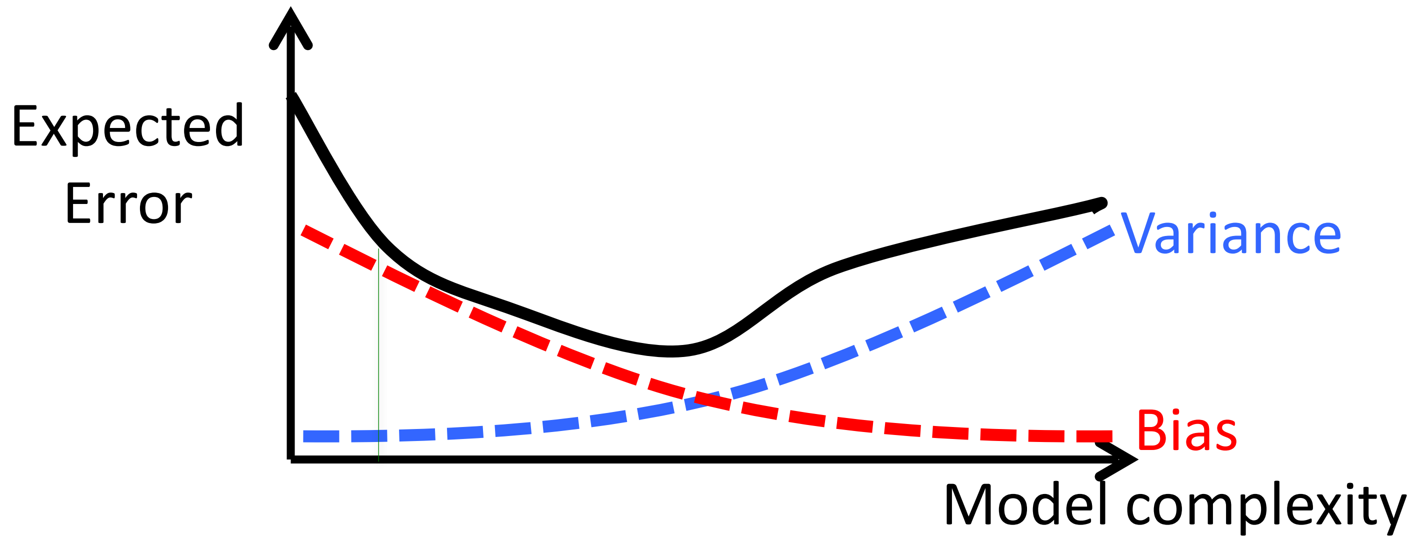
Bias of a learner (informally)



How likely is the learner to identify the **target** hypothesis?

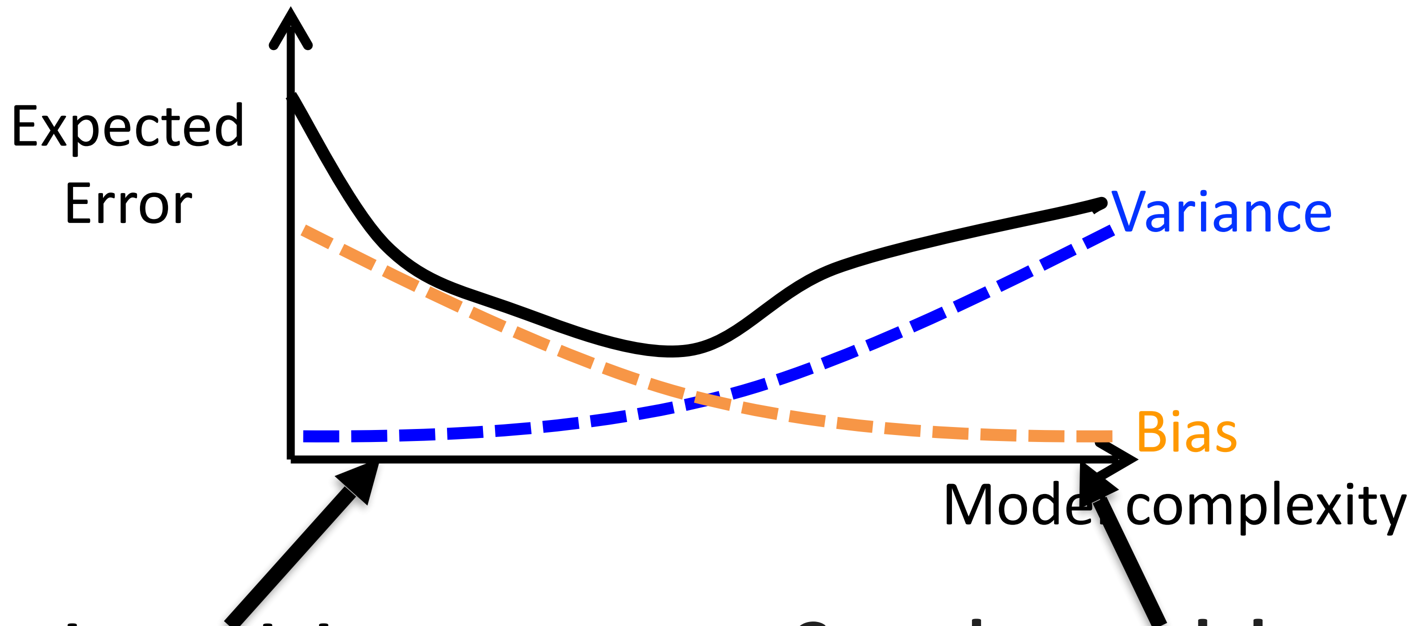
- Bias is low when the model is expressive (low empirical error)
- Bias is high when the model is (too) simple
- The larger the hypothesis space is, the easiest it is to be close to the true hypothesis.
- More accurately: for each data set D, you learn a different hypothesis $h(D)$, that has a different true error $e(h)$; we are looking here at the difference of the mean of this random variable from the true error.

Impact of bias and variance



Expected error ≈ bias + variance

Model complexity



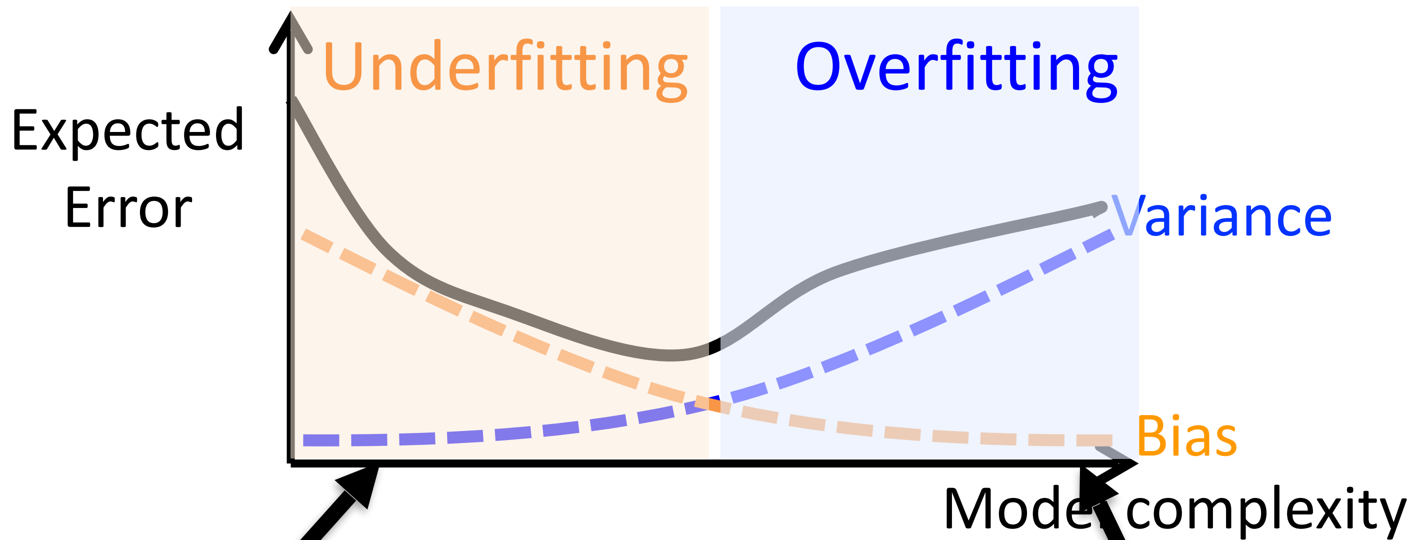
Simple models:

High bias and low variance

Complex models:

High variance and low bias

Model complexity



Simple models:
High bias and low variance

Complex models:
High variance and low bias

This can be made more accurate for some loss functions.

We will discuss a more precise and general theory that trades **expressivity of models** with **empirical error**

Managing of bias and variance

Ensemble methods reduce variance

- Multiple classifiers are combined
- E.g., bagging, boosting

Decision trees of a given depth

- Increasing depth decreases bias, increases variance

Neural networks

- Deeper models can increase variance, but decrease bias