CS 584
Data Mining

Classification 3

Today

- Model evaluation & related concepts
- Additional classifiers
  - Naïve Bayes classifier
  - Support Vector Machine
  - Ensemble methods

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Metrics for Performance Evaluation

Focus on the predictive capability of a model, rather than how fast it takes to classify or build models, scalability, etc.

$$\text{Accuracy} = \frac{\text{Number of correct classifications}}{\text{Number of instances in our database}}$$

Accuracy is a single number, we may be better off looking at a **confusion matrix**. This gives us additional useful information…

Classified as...

True label is…

	Cat	Dog	Pig
Cat	100	0	0
Dog	9	90	1
Pig	45	45	10

Metrics for Performance Evaluation

- Confusion Matrix

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

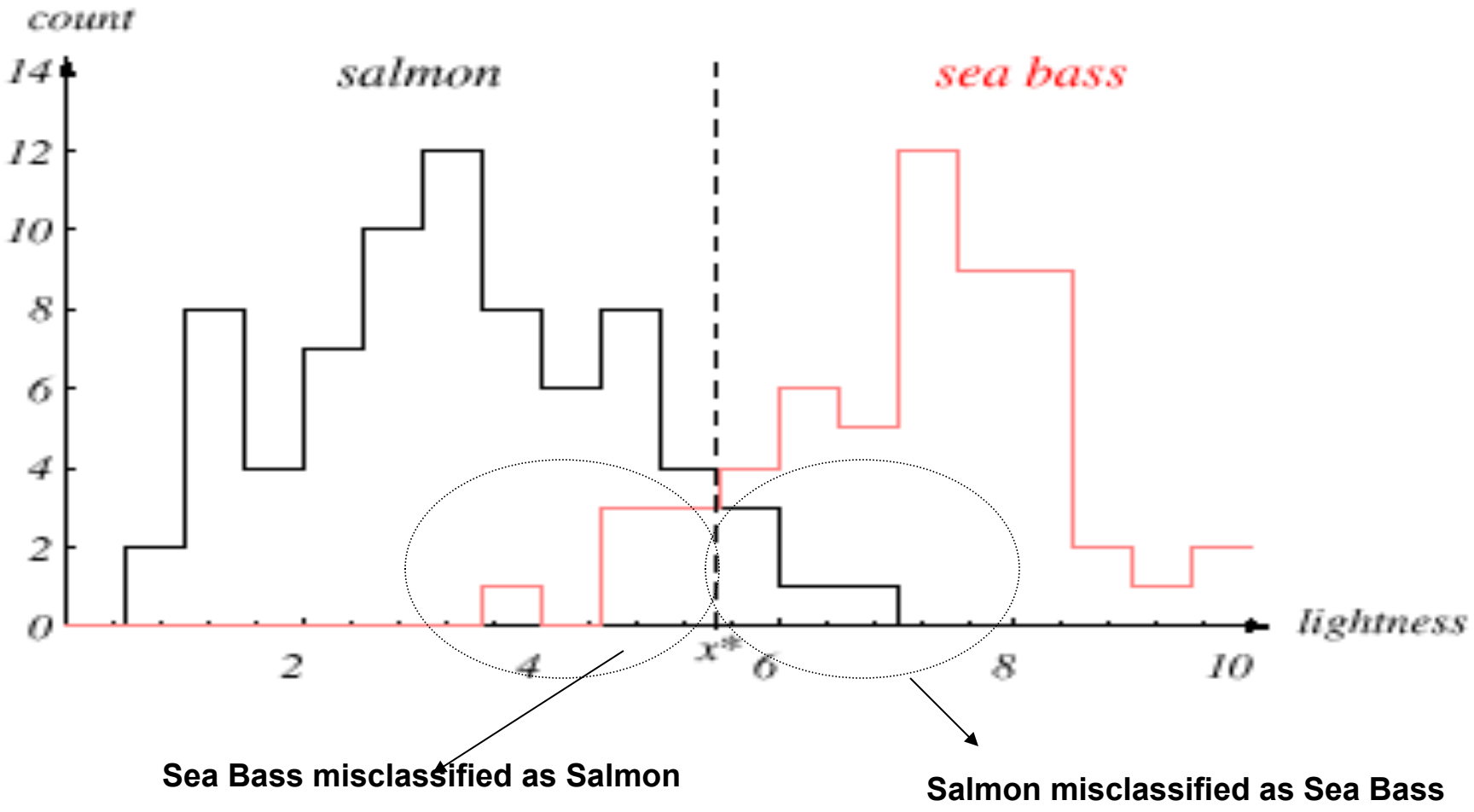
a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

Remember this Example?



Metrics for Performance Evaluation

- Confusion Matrix:

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

Predicted as...

True label is…	Salmon	Sea Bass
Salmon		
Sea Bass		

Metrics for Performance Evaluation…

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

(% of correctly classified items)

(same as the previous equation, just rewriting it)

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Methods for Performance Evaluation

- How to obtain a reliable estimate of performance?

- Performance of a model may depend on other factors besides the learning algorithm:
  - Class distribution
  - Cost of misclassification
  - Size of training and test sets

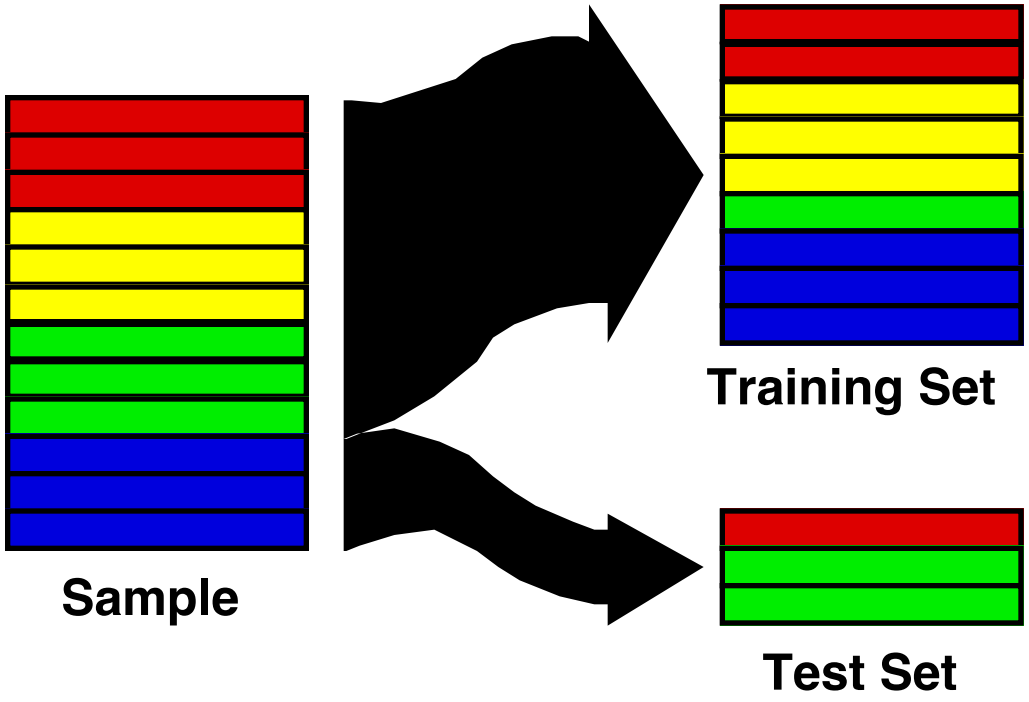
Methods of Estimation

- Holdout
  - e.g. Reserve 2/3 for training and 1/3 for testing
- Random subsampling
  - Repeated holdout
- Cross validation
  - Partition data into k disjoint subsets
  - k-fold: train on k-1 partitions, test on the remaining one
  - Leave-one-out: k=n
- Stratified sampling
  - oversampling vs undersampling
- Bootstrap
  - Sampling with replacement

Hold-out validation: simple holdout set

Partition data into training set and test set

In some domains it makes sense to partition temporally (training set before t, test set after t)



challenges: 1) what if by accident you selected a particularly easy/hard test set?
2) do you have an idea of the variation in model accuracy due to training?

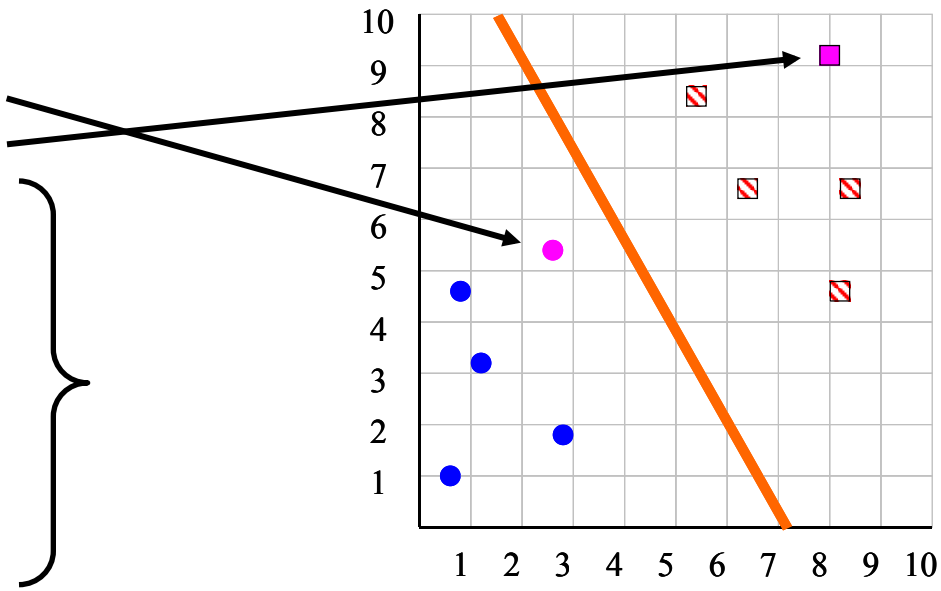
K-Fold Cross Validation

We divide the dataset into K equal sized sections. The algorithm is tested K times, each time leaving out one of the K section from building the classifier, but using it to *test* the classifier instead

$$\text{Accuracy} = \frac{\text{Number of correct classifications}}{\text{Number of instances in our database}}$$

$K = 5$

Insect ID	Abdomen Length	Antennae Length	Insect Class
1	2.7	5.5	Grasshopper
2	8.0	9.1	Katydid
3	0.9	4.7	Grasshopper
4	1.1	3.1	Grasshopper
5	5.4	8.5	Katydid
6	2.9	1.9	Grasshopper
7	6.1	6.6	Katydid
8	0.5	1.0	Grasshopper
9	8.3	6.6	Katydid
10	8.1	4.7	Katydids



Limitation of Accuracy

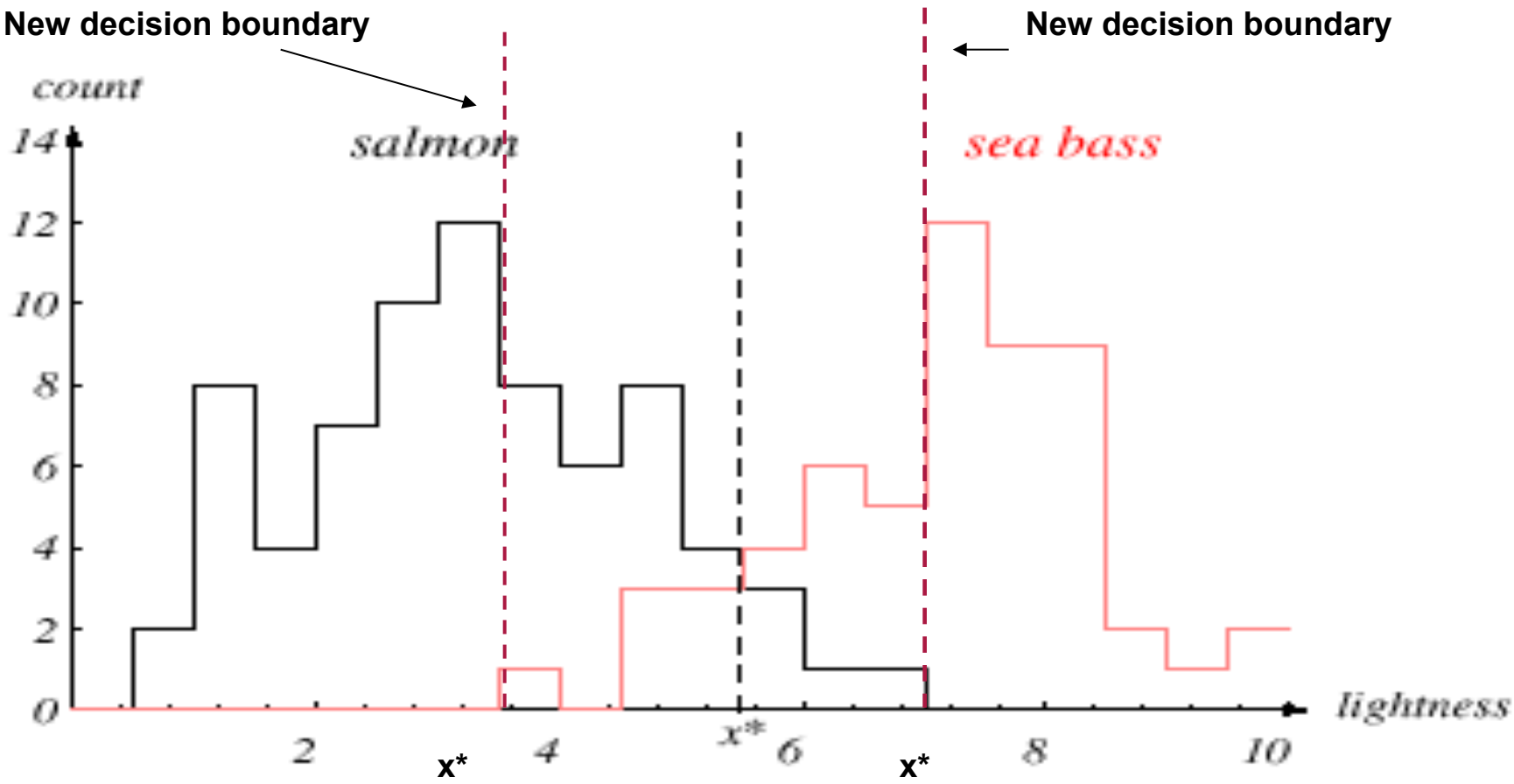
- Consider a 2-class problem
  - Number of Class 0 examples = 9990
  - Number of Class 1 examples = 10

- If model predicts everything to be class 0, accuracy is 9990/10000 = 99.9 %
  - Accuracy is misleading because model does not detect any class 1 example

What if?

Salmon is more expensive than Bass?

Bass is more expensive than Salmon?



Cost sensitive classification

- Penalize misclassifications of one class more than the other
- Changes decision boundaries

Cost Matrix

		PREDICTED CLASS	
ACTUAL CLASS	C(i, j)	**Class=Yes**	**Class=No**
	Class=Yes	C(Yes, Yes)	C(Yes, No)
	Class=No	C(No, Yes)	C(No, No)

C(i, j): Cost of misclassifying class i example as class j

Computing Cost of Classification

Cost Matrix	PREDICTED CLASS		
	C(i, j)	+	-
ACTUAL CLASS	+	-1	100
	-	1	0

False negative error cost

False positive error cost

Model M_1	PREDICTED CLASS		
		+	-
ACTUAL CLASS	+	150	40
	-	60	250

Accuracy = 80%

Cost = 3910

Model M_2	PREDICTED CLASS		
		+	-
ACTUAL CLASS	+	250	45
	-	5	200

Accuracy = 90%

Cost = 4255

Cost vs Accuracy

Count	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

Cost	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	p	q
	Class=No	q	p

Accuracy is proportional to cost if
1. C(Yes, No)=C(No, Yes) = q
2. C(Yes, Yes)=C(No, No) = p

$N = a + b + c + d$

$\text{Accuracy} = (a + d)/N$

$$\begin{aligned}\text{Cost} &= p\,(a + d) + q\,(b + c)\\ &= p\,(a + d) + q\,(N - a - d)\\ &= q\,N - (q - p)(a + d)\\ &= N\,[q - (q - p) \times \text{Accuracy}]\end{aligned}$$

Cost-Sensitive Measures

$$\text{Precision (p)} = \frac{a}{a+c}$$

$$\text{Recall (r)} = \frac{a}{a+b}$$

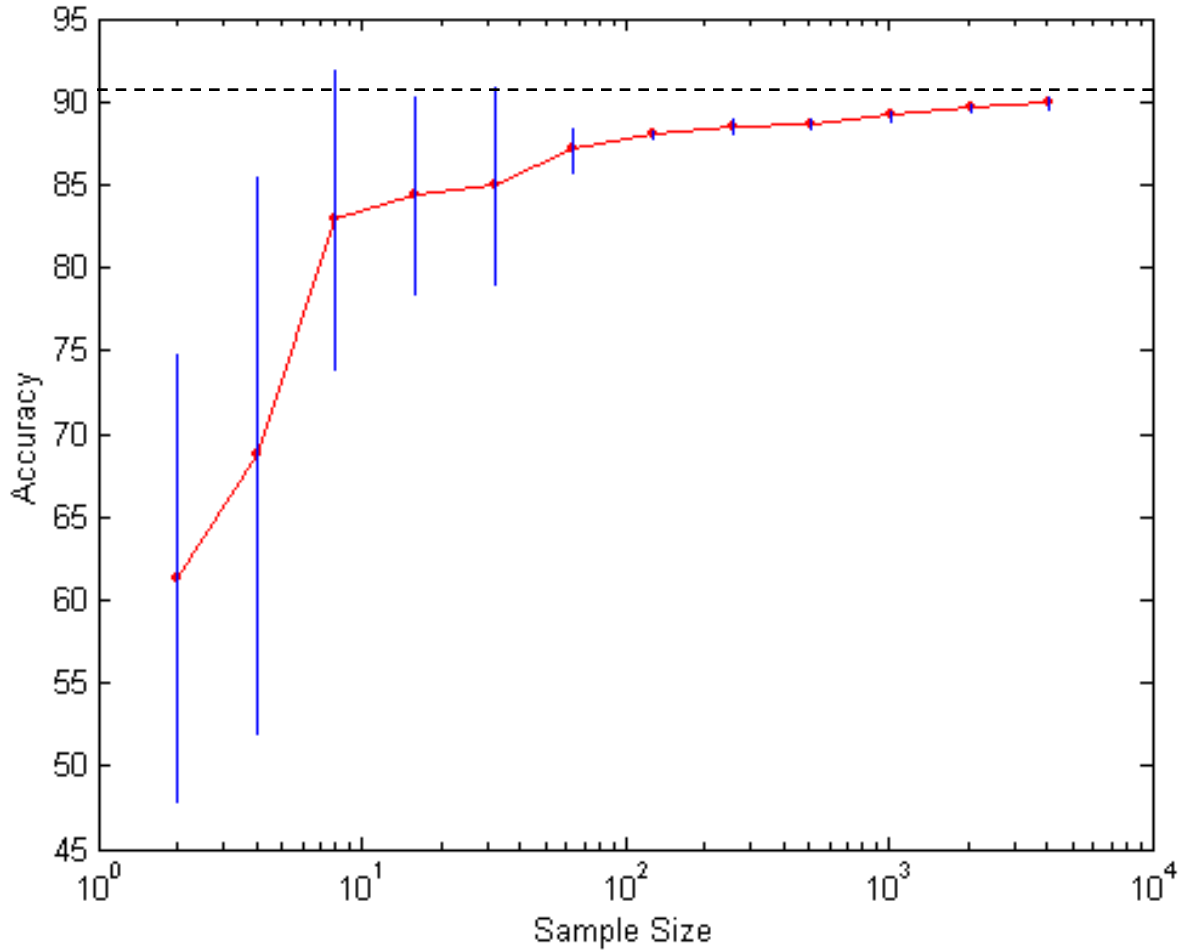
$$\text{F1-measure (F1)} = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$$

Count	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

- Precision is biased towards C(Yes, Yes) & C(No, Yes)
- Recall is biased towards C(Yes, Yes) & C(Yes, No)
- F1-measure is biased towards all except C(No, No)

$$\text{Weighted Accuracy} = \frac{w_1 a + w_4 d}{w_1 a + w_2 b + w_3 c + w_4 d}$$

Learning Curve



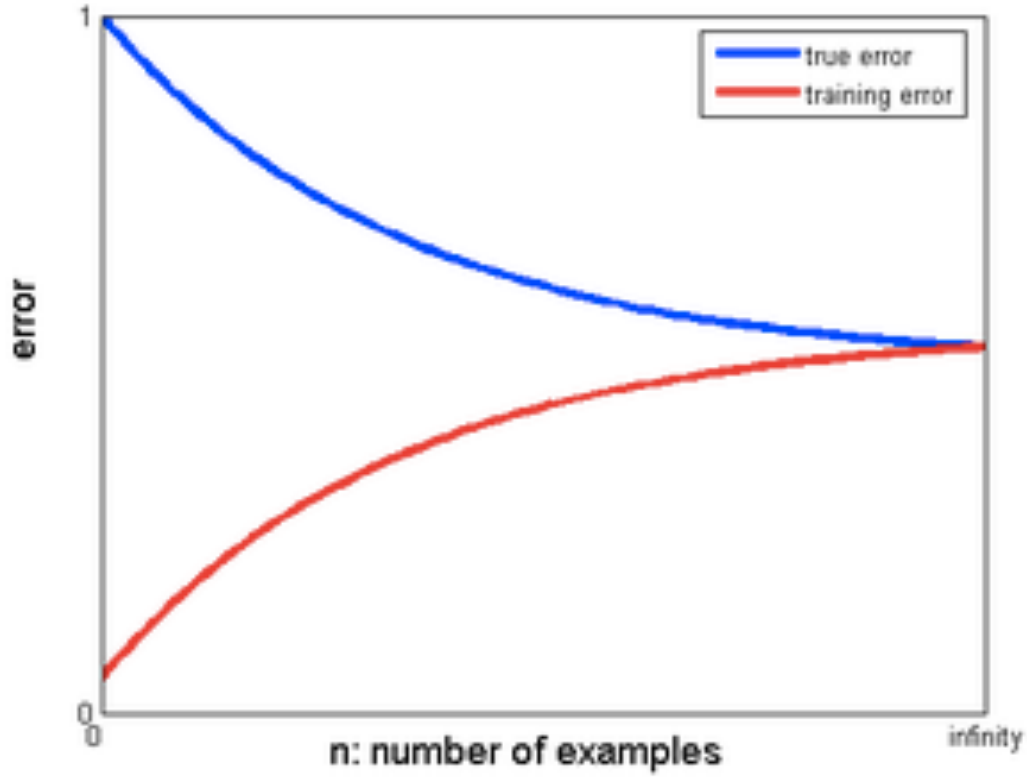
- Learning curve shows how accuracy changes with varying sample size
- Requires a sampling schedule for creating learning curve:
  - Arithmetic sampling (Langley, et al)
  - Geometric sampling (Provost et al)

Effect of small sample size:

- Bias in the estimate
- Variance of estimate

Typical Learning Curves

- For a fixed complexity penalty.
- Note the error on the y-axis (as opposed to accuracy in the previous slide)



https://alliance.seas.upenn.edu/~cis520/dynamic/2014/wiki/index.php?n=Lectures.Overfitting

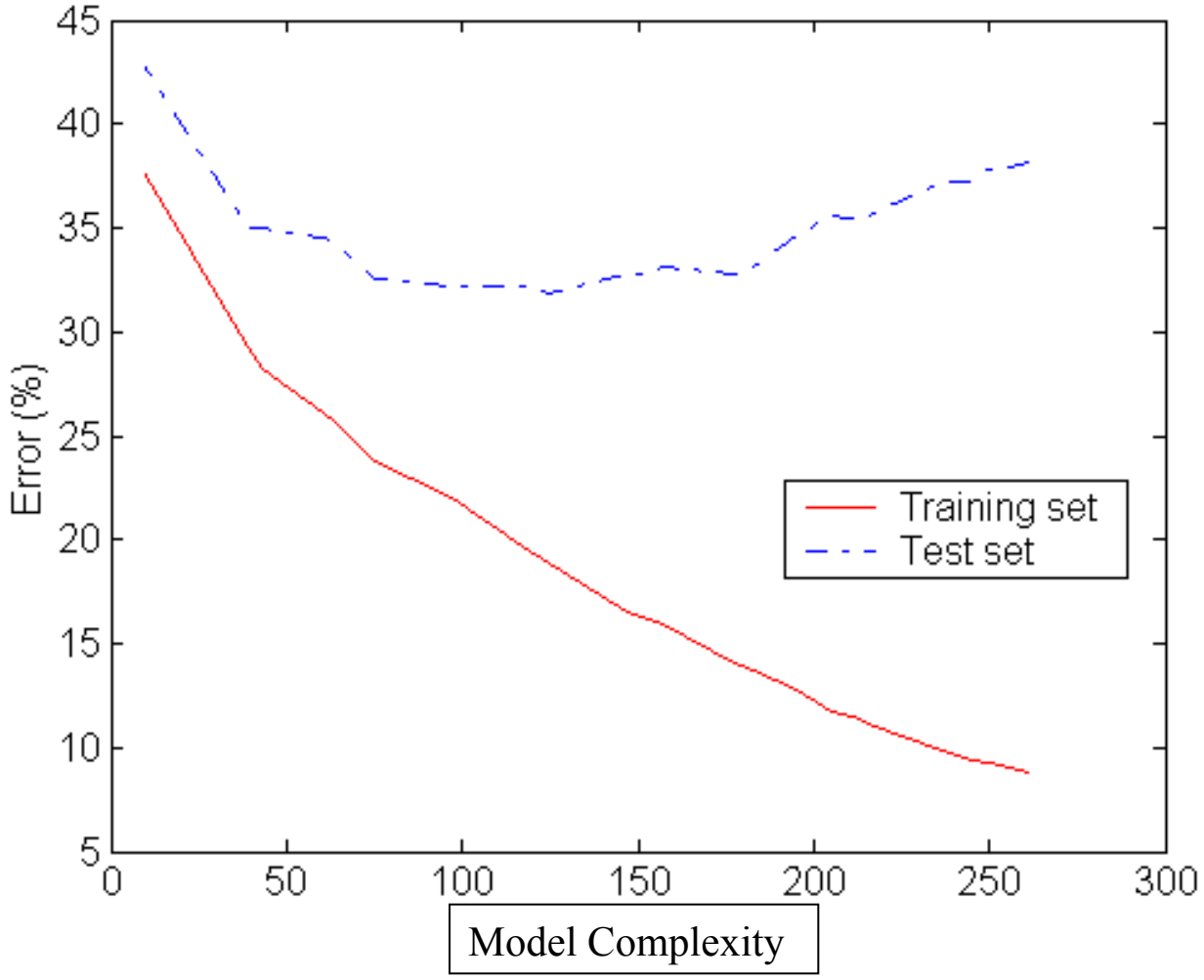
Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

revisit

Tool for model performance analytics: The fitting curve

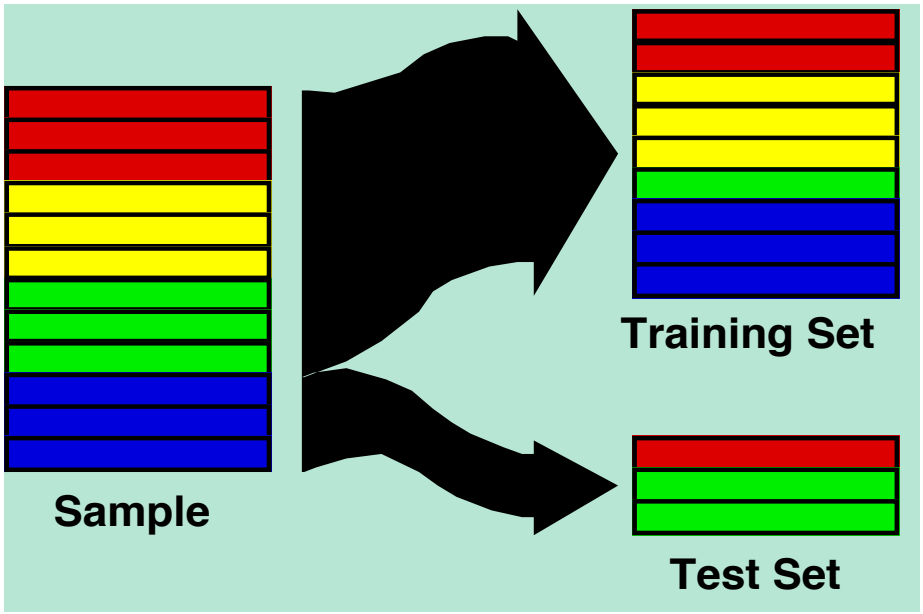
But can we find the right complexity?



Nested holdout for complexity control

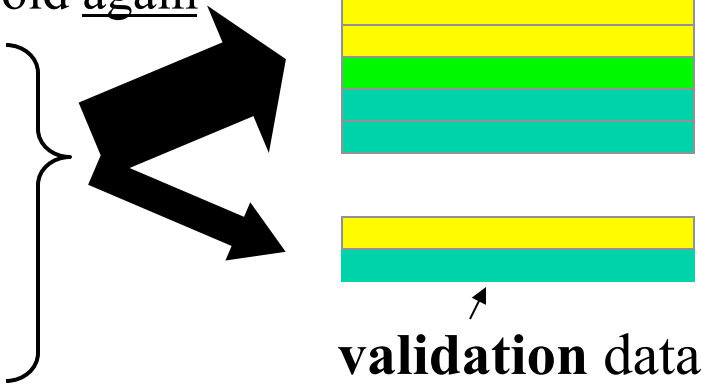
Need to be careful making data mining decisions based on testing data (or CV, cross validation)

- When choosing models, features, complexity parameters, etc.
- *Don't want to overfit the test data!*



save this for later

split training fold again



nested training data

validation data

- can run a (nested) holdout (or CV) on the training set, and make choices without examining test set.
- when choices all are made, then test on test set
- this "nested" test set is often called the "validation" set (to differentiate from the final test set).

Slide from
Foster Provost and Tom Fawcett
Data Mining for Business: What You Need to Know about Data Mining and Data-analytic Thinking

ROC (Receiver Operating Characteristic)

- Developed in 1950s for signal detection theory to analyze noisy signals
  - Characterize the trade-off between positive hits and false alarms
- ROC curve plots TPR (on the y-axis) against FPR (on the x-axis)
- Performance of each classifier represented as a point on the ROC curve
  - changing the threshold of algorithm, sample distribution or cost matrix changes the location of the point

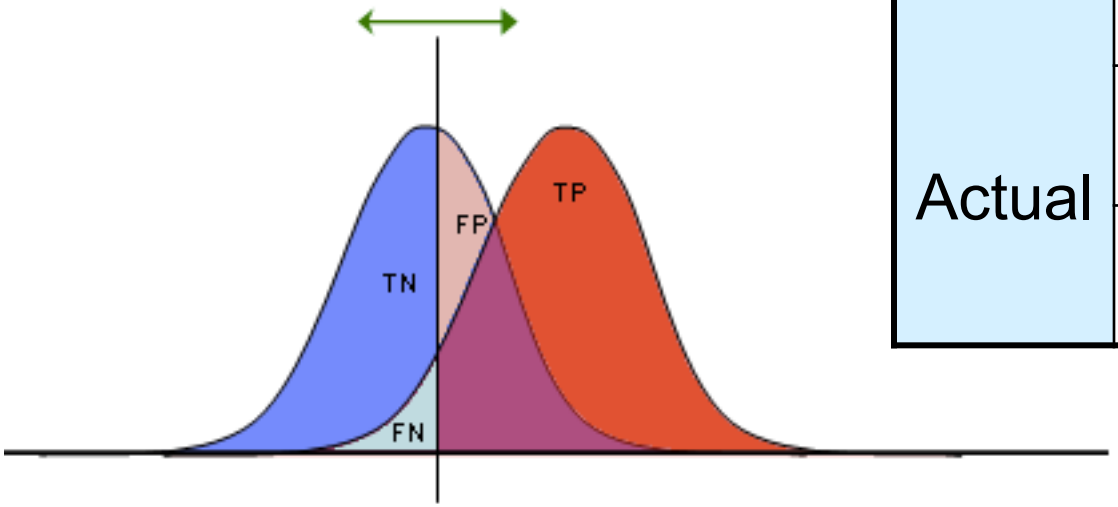
ROC (Receiver Operating Characteristic)

- Developed in 1950s for signal detection theory to analyze noisy signals
  - Characterize the trade-off between positive hits and false alarms
- **ROC** curve plots **TPR** (on the **y**-axis) against **FPR** (on the **x**-axis)

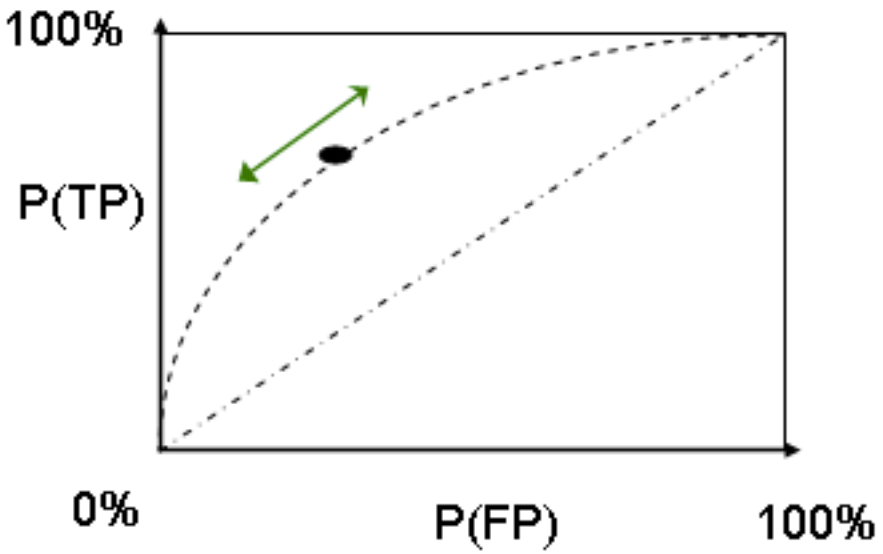
$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

		PREDICTED CLASS	
		Yes	No
Actual	Yes	a (TP)	b (FN)
	No	c (FP)	d (TN)



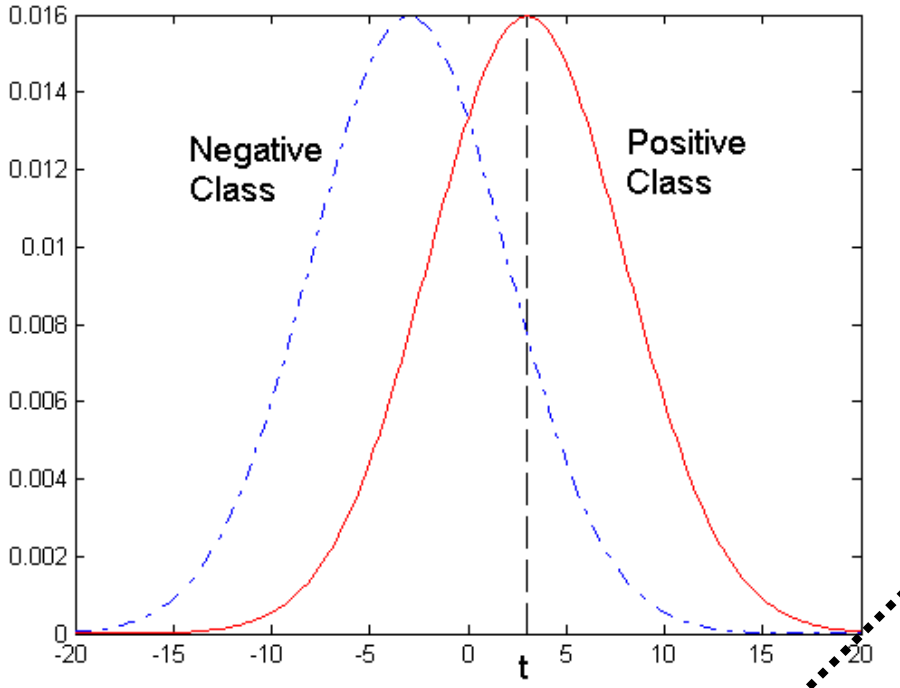
	PREDICTED CLASS		
		Yes	No
Actual	Yes	TP	FN
	No	FP	TN

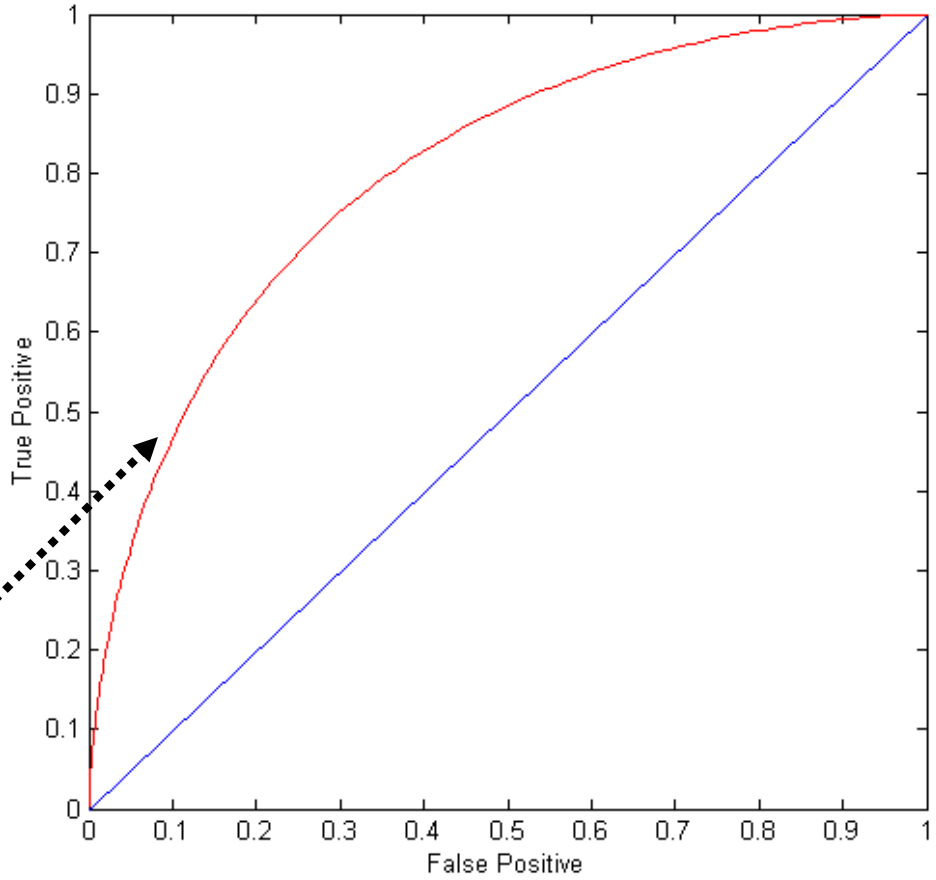


ROC Curve

- 1-dimensional data set containing 2 classes (positive and negative)

- any points located at x > t is classified as positive





At threshold t:

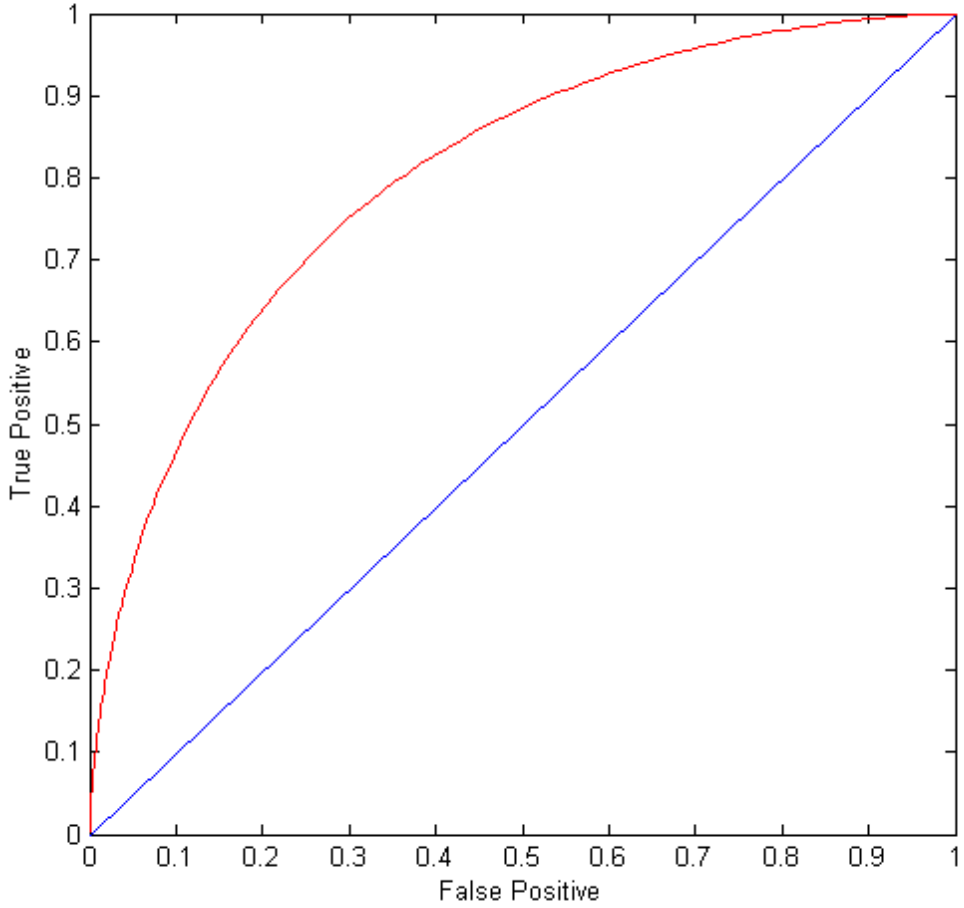
TP=0.5, FN=0.5, FP=0.12, TN=0.88

ROC Curve

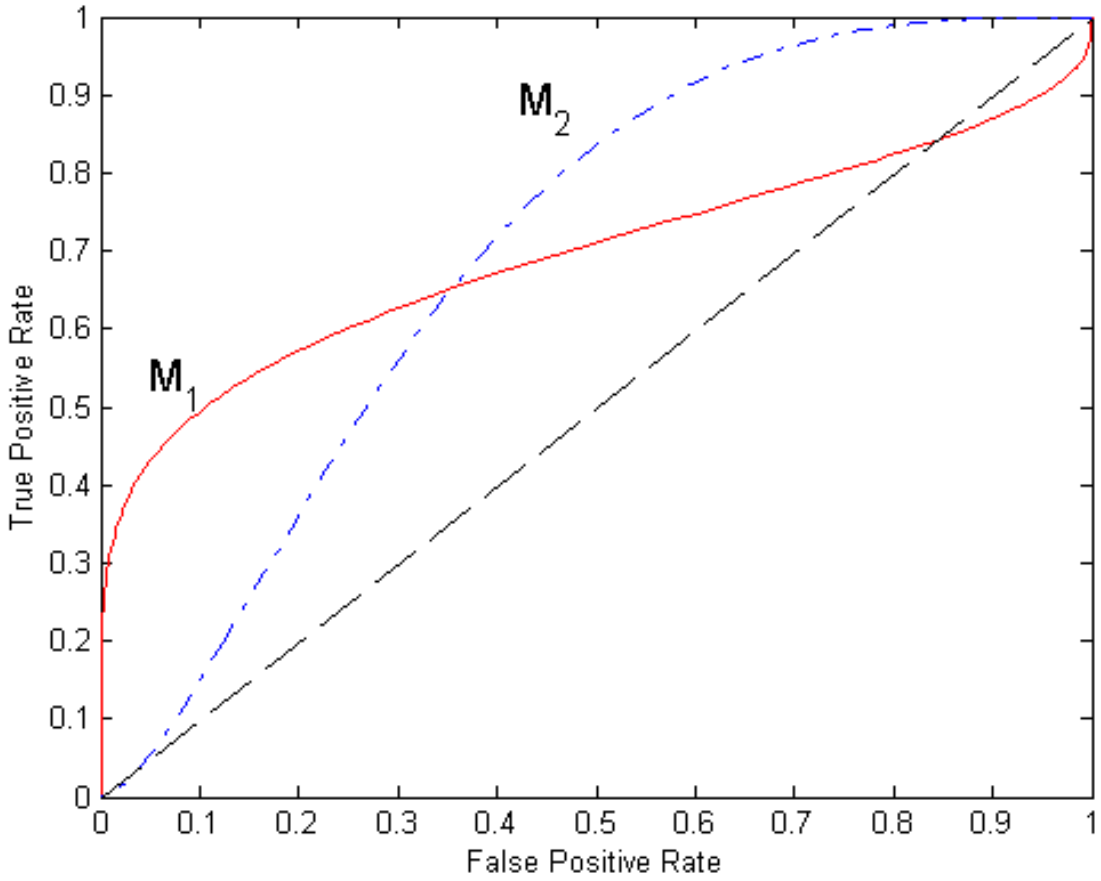
(TP,FP):

- (0,0): declare everything to be negative class
- (1,1): declare everything to be positive class
- (1,0): ideal

- Diagonal line:
  - Random guessing
  - Below diagonal line:
    - prediction is opposite of the true class



Using ROC for Model Comparison



- No model consistently outperform the other
  - M_1 is better for small FPR
  - M_2 is better for large FPR
- Area Under the ROC curve
  - Ideal:
    - Area = 1
  - Random guess:
    - Area = 0.5

How to Construct an ROC curve

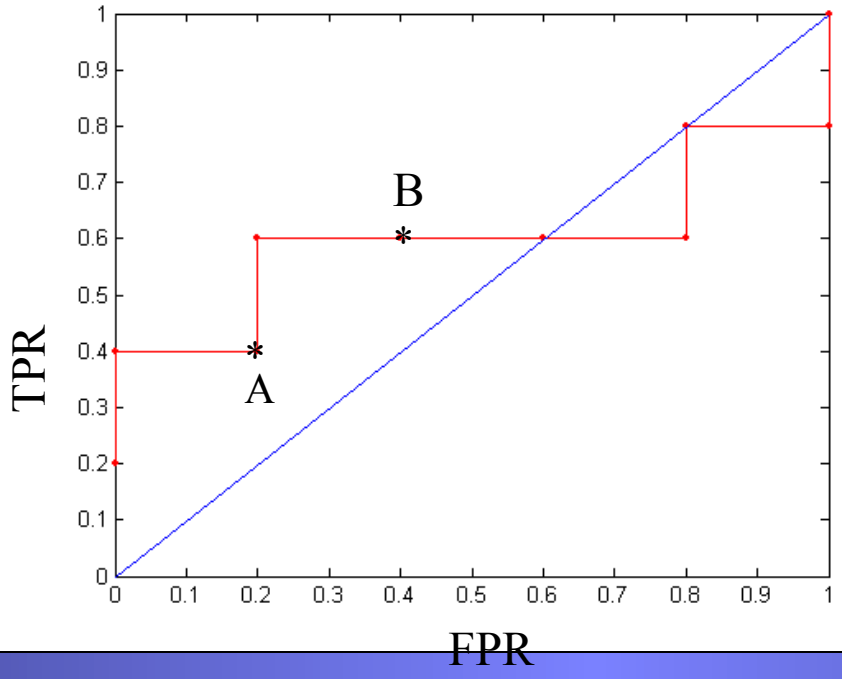
Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

- Use classifier that produces posterior probability for each test instance P(+|A)
- Sort the instances according to P(+|A) in decreasing order
- Apply threshold at each unique value of P(+|A)
- Classify the selected record and those above it to the + class.
- Count the number of TP, FP, TN, FN at each threshold
- TP rate, TPR = TP/(TP+FN)
- FP rate, FPR = FP/(FP + TN)

How to construct an ROC curve

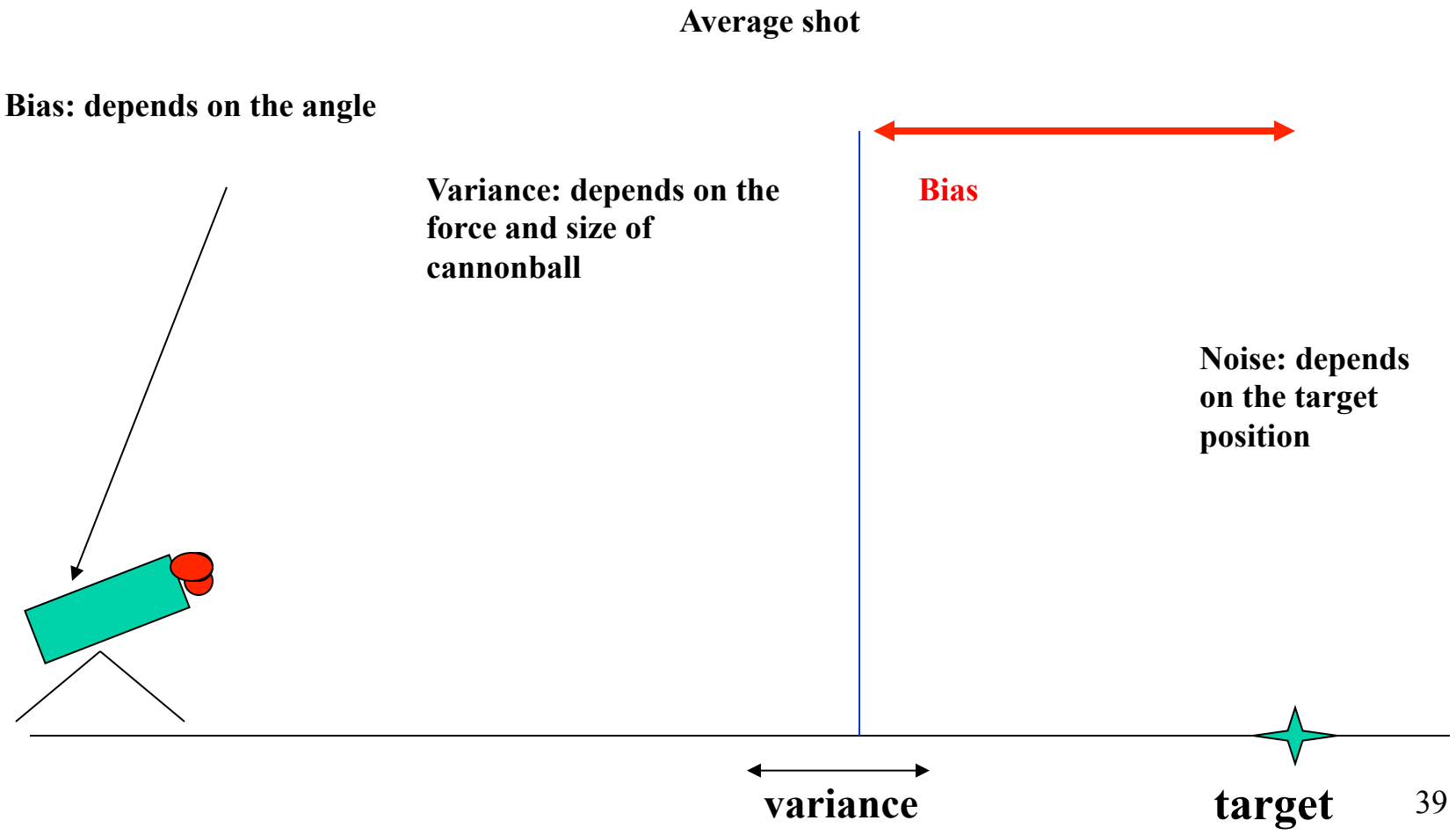
	Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=		0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
	TP	5	4	4	3	3	3	3	2	2	1	0
	FP	5	5	4	4	3	2	1	1	0	0	0
	TN	0	0	1	1	2	3	4	4	5	5	5
	FN	0	1	1	2	2	2	2	3	3	4	5
→	TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
→	FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

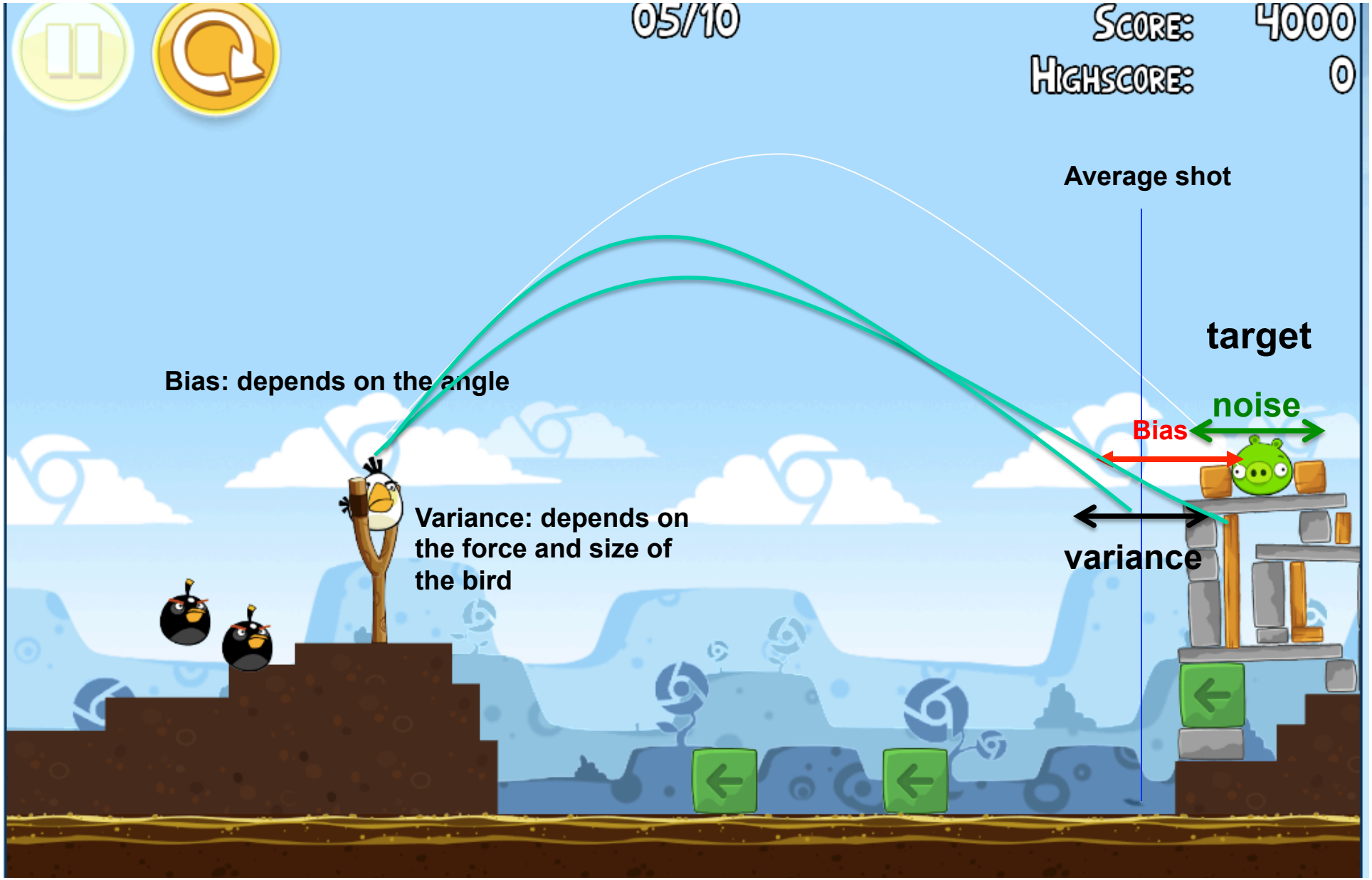
ROC Curve:



A is more conservative than B (classifying positive only if it has strong evidence)

Loss, bias, variance and noise




05/10
Score: 4000
Highscore: 0
Average shot
target
noise
Bias
Bias: depends on the angle
Variance: depends on the force and size of the bird
variance

Example: Bias

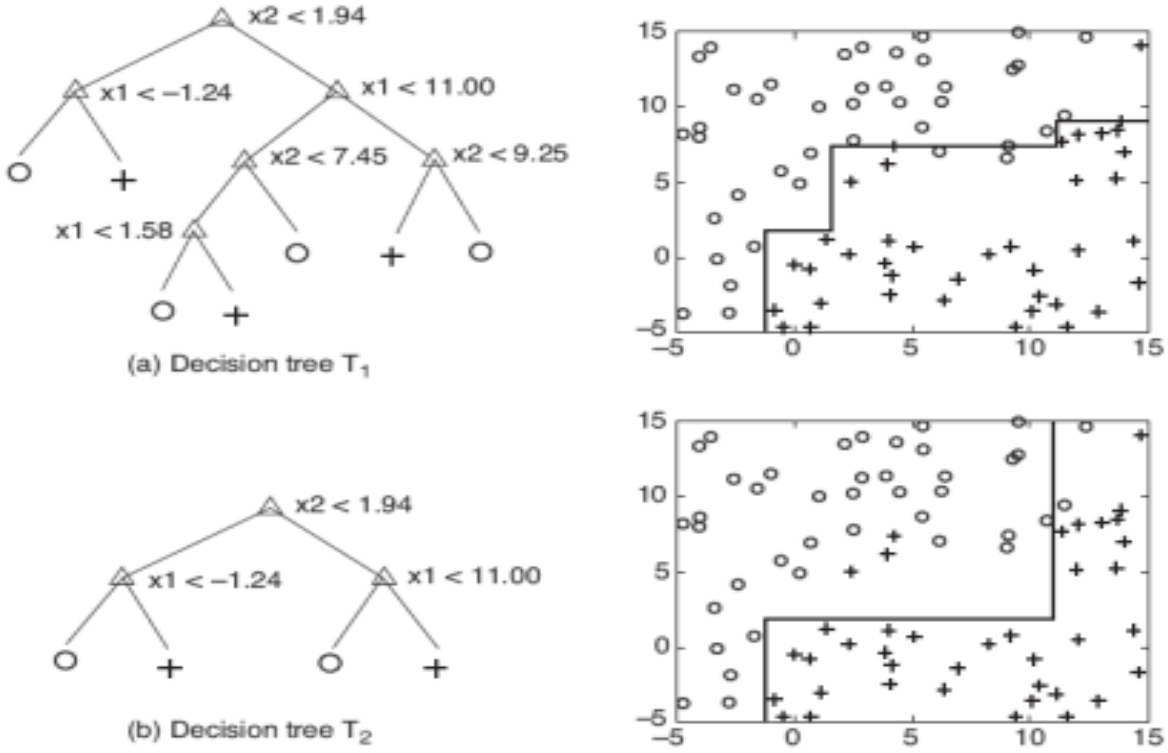
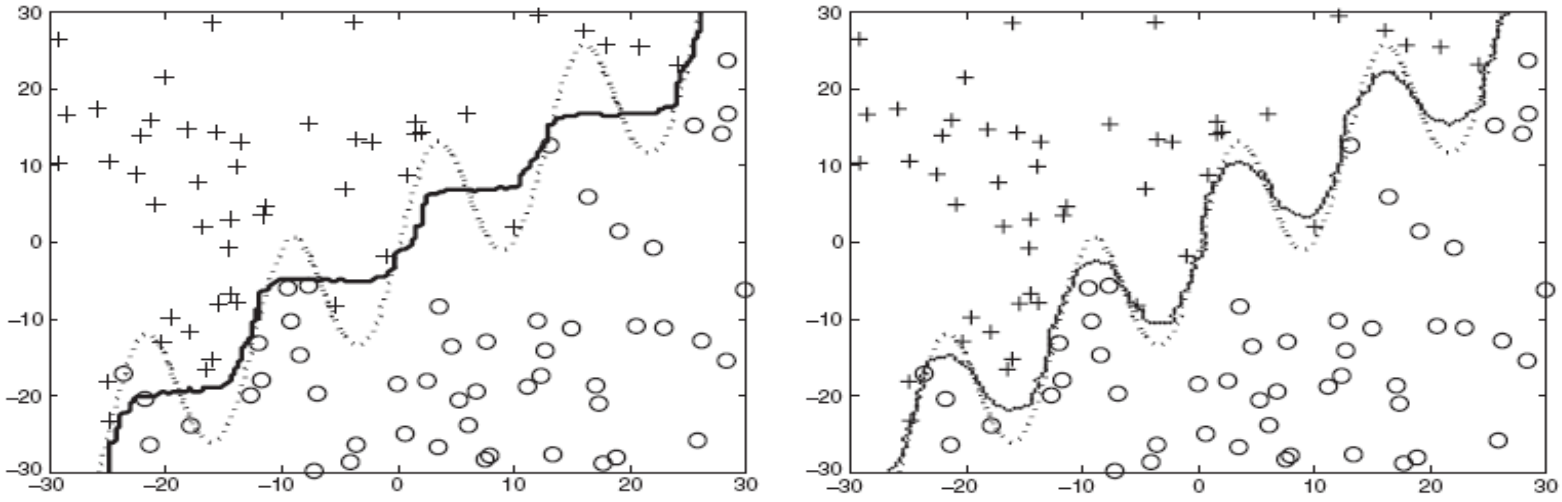


Figure 5.33. Two decision trees with different complexities induced from the same training data.

Bias-Variance (Generalize)



(a) Decision boundary for decision tree.

(b) Decision boundary for 1-nearest neighbor.

Figure 5.34. Bias of decision tree and 1-nearest neighbor classifiers.

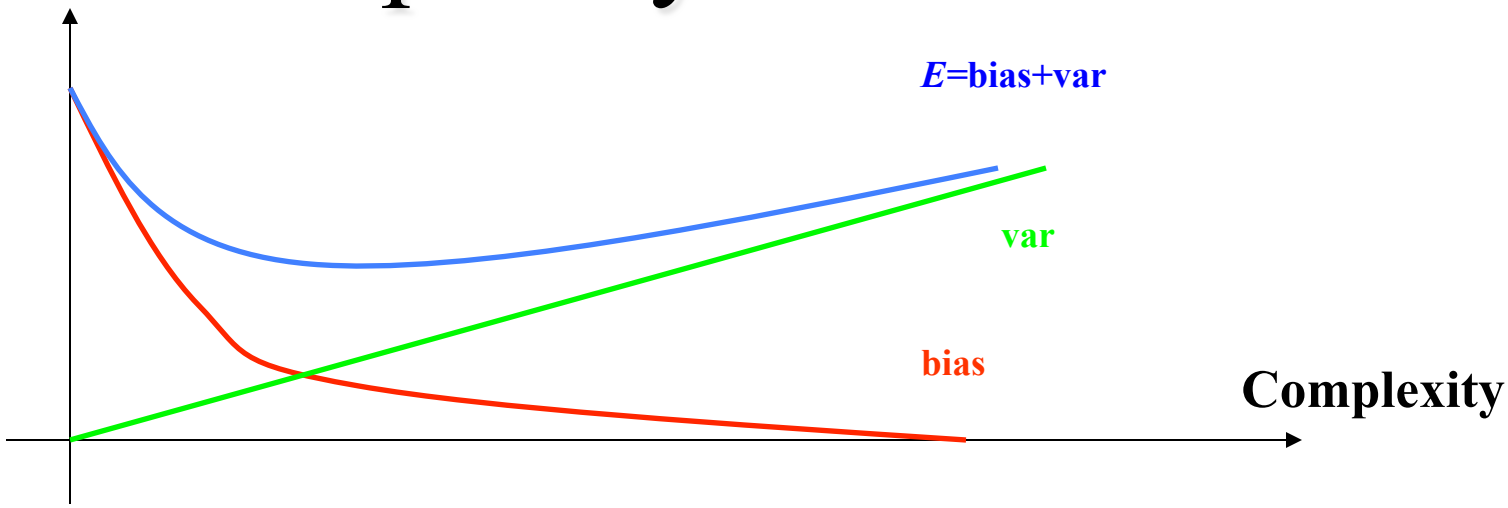
For better generalizable model

- Minimize both bias and variance
- However,
  - Neglect the input data and predict the output to be a constant value gives “zero” variance but high bias.
  - On the other hand, perfectly interpolate the given data to produce f=f* - implies zero bias but high variance.

Model Complexity

- Simple models of low complexity
  - high bias, small variance
  - potentially rubbish, but stable predictions
- Flexible models of high complexity
  - small bias, high variance
  - over-complex models can be always massaged to exactly explain the observed training data
- What is the right level of model complexity?
  - The problem of model selection

Complexity of the model



Usually, the bias is a decreasing function of the complexity, while variance is an increasing function of the complexity.

Bayesian Methods

- Learning and classification methods based on probability theory.
- Bayes theorem plays a critical role in probabilistic learning and classification.
- Builds a *generative model* that approximates how data is produced
- Uses *prior* probability of each category given no information about an item.
- Categorization produces a *posterior* probability distribution over the possible categories given a description of an item.

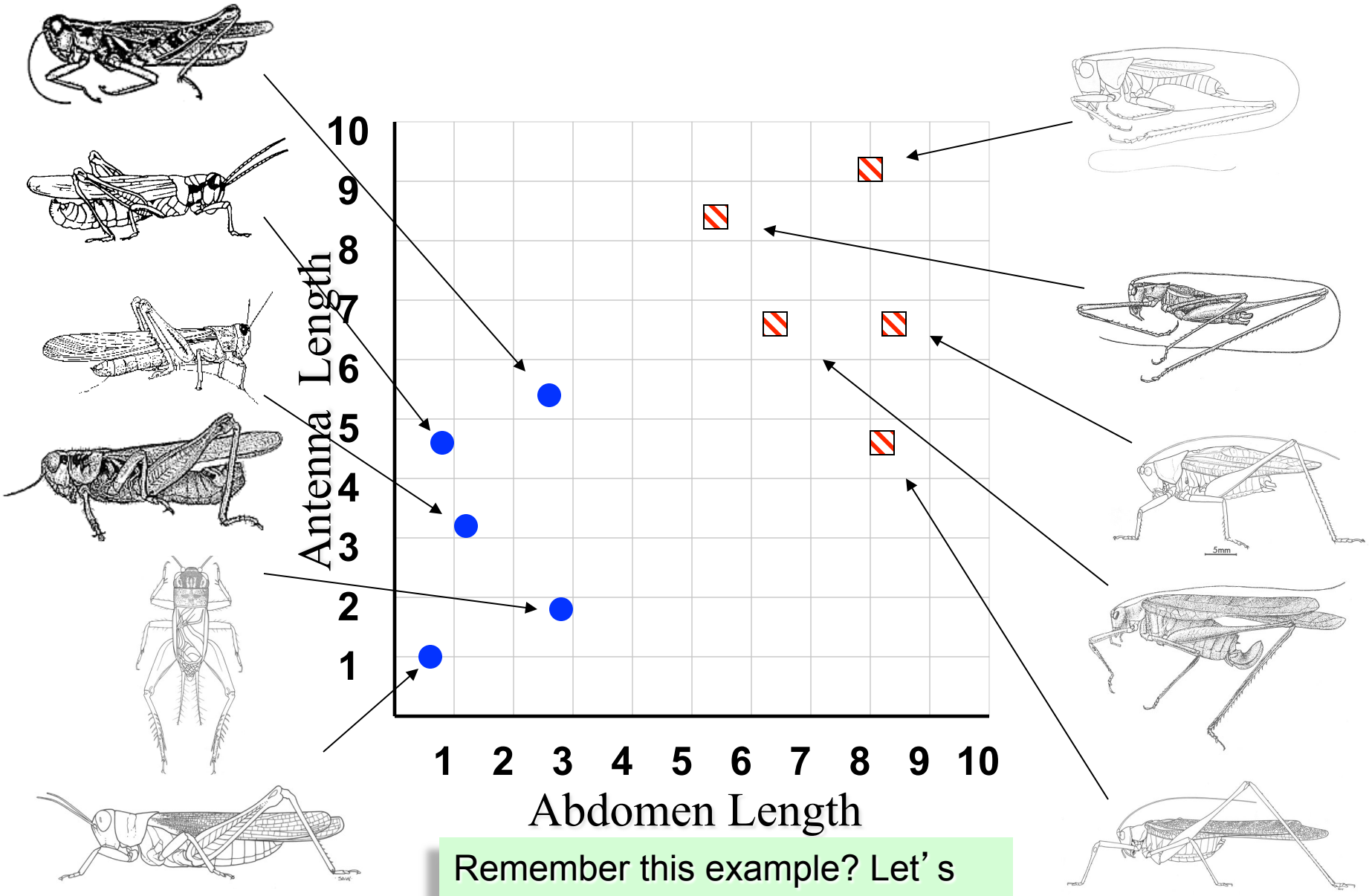
Naïve Bayes Classifier

Thomas Bayes
1702 - 1761

We will start off with a visual intuition, before looking at the math…

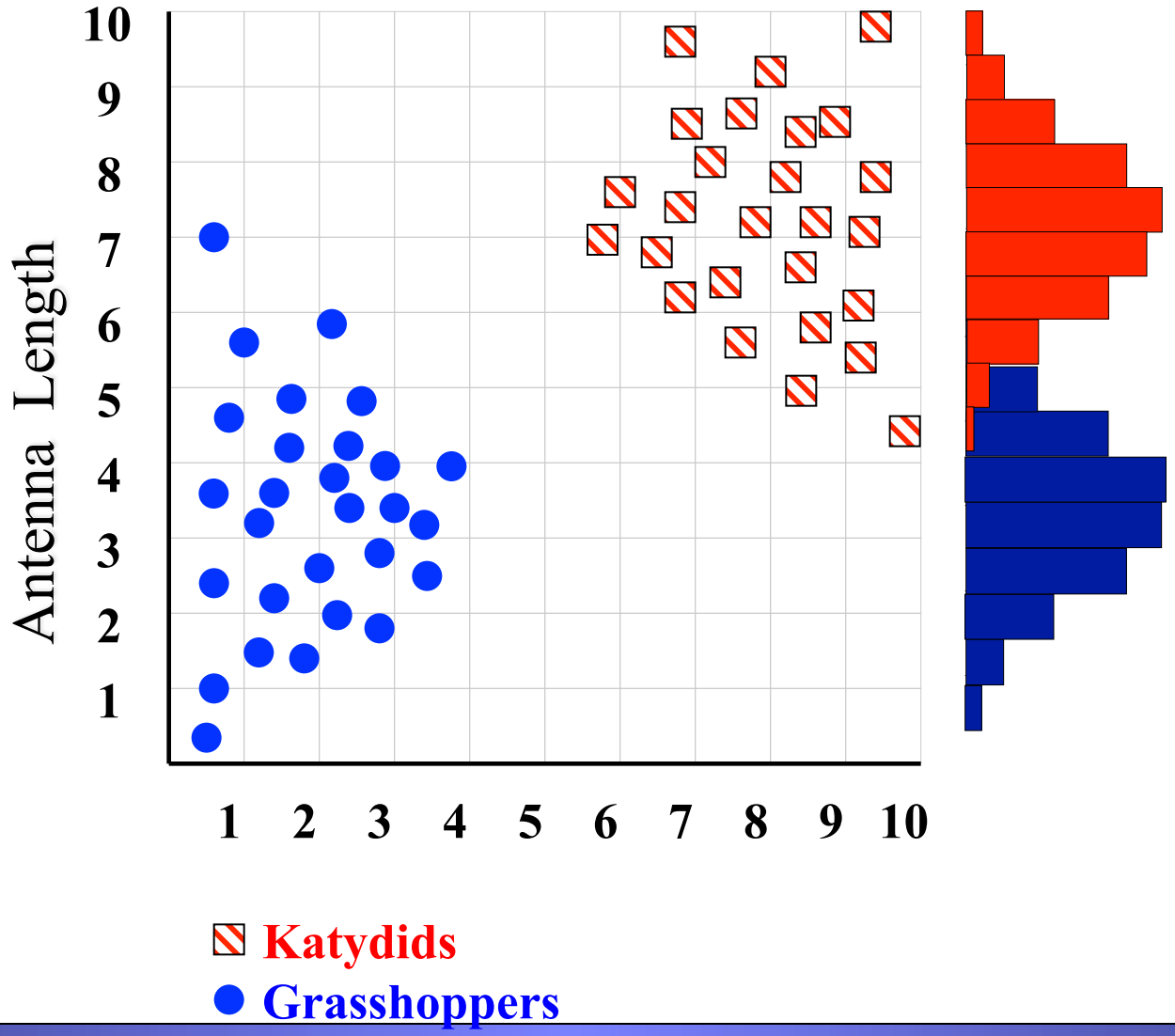
Grasshoppers

Katydids

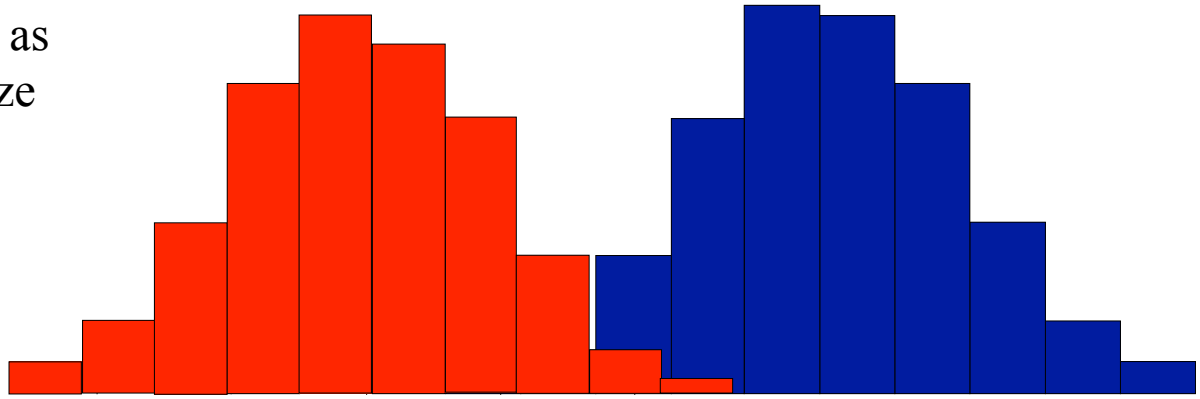


Remember this example? Let's get lots more data…

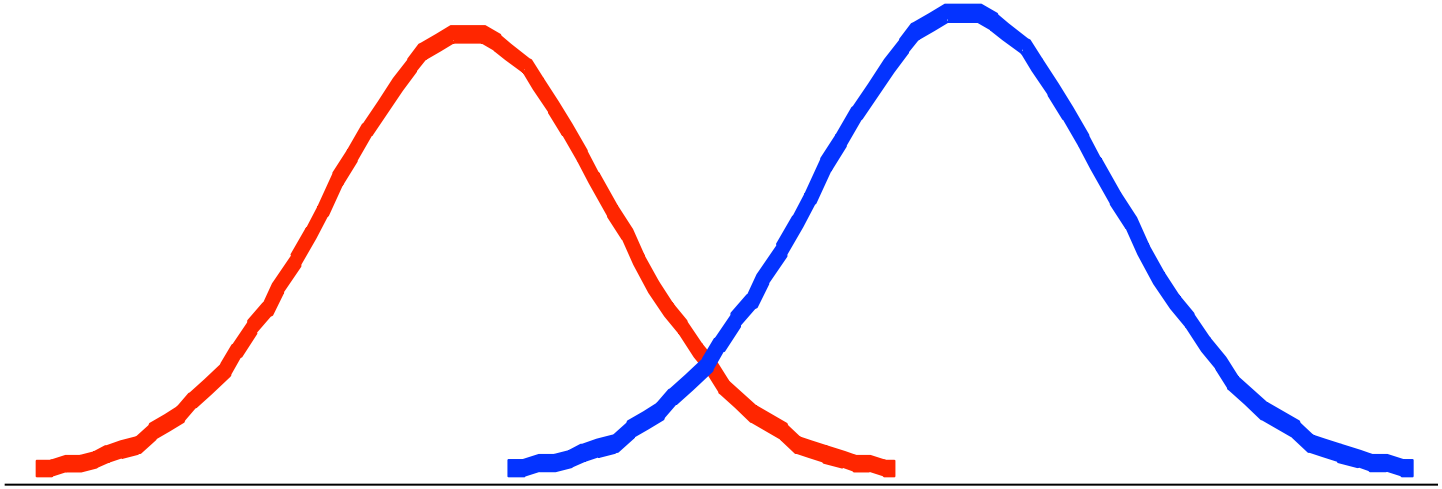
With a lot of data, we can build a histogram. Let us just build one for "Antenna Length" for now…



We can leave the histograms as they are, or we can summarize them with two normal distributions.

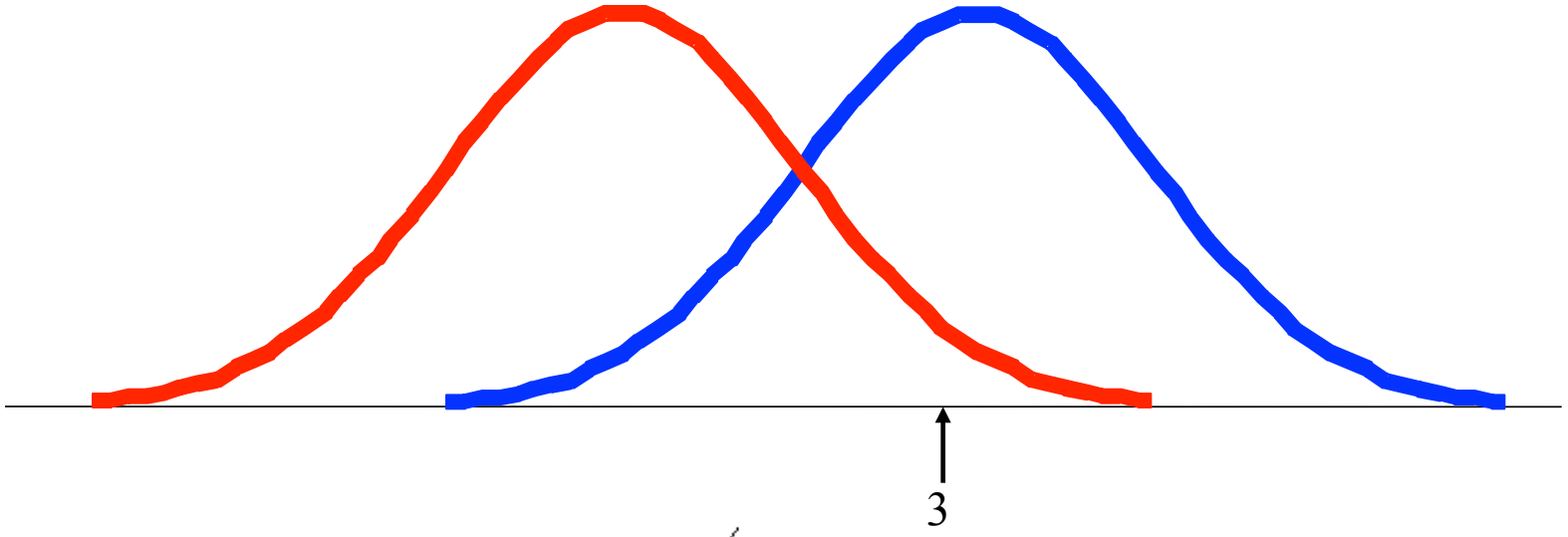


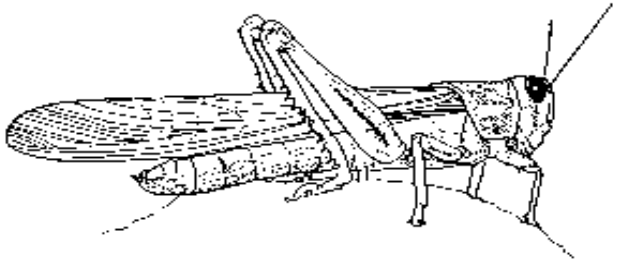
Let us use two normal distributions for ease of visualization in the following slides…



• We want to classify an insect we have found. Its antennae are 3 units long. How can we classify it?
• We can just ask ourselves, give the distributions of antennae lengths we have seen, is it more *probable* that our insect is a Grasshopper or a Katydid.
• There is a formal way to discuss the most *probable* classification…

P*(*C* | *A*) = probability of class *C*, *given* that we have observed *A



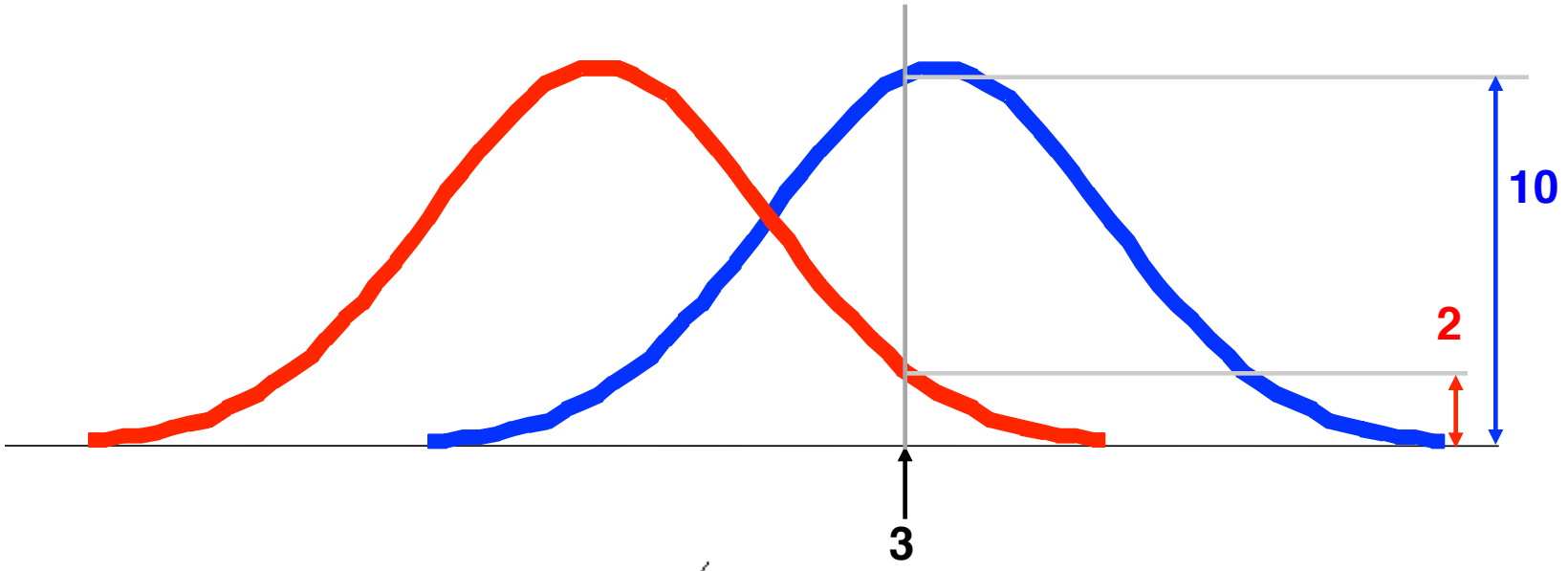


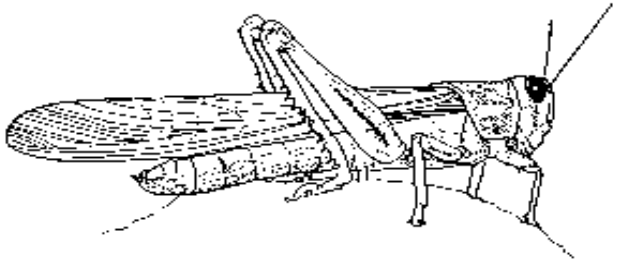
Antennae length is 3

P*(*C* | *A*) = probability of class *C*, given that we have observed *A

P(Grasshopper | 3) = 10 / (10 + 2) = 0.833

P(Katydid | 3) = 2 / (10 + 2)= 0.166



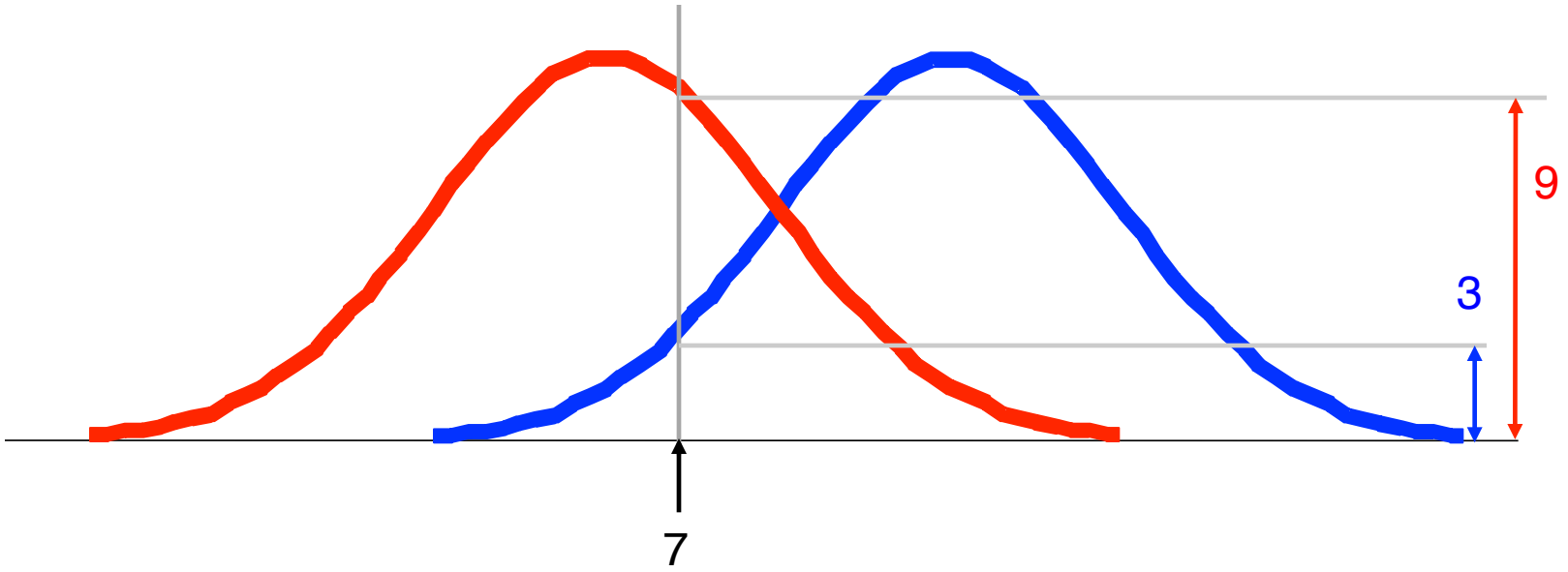


Antennae length is 3

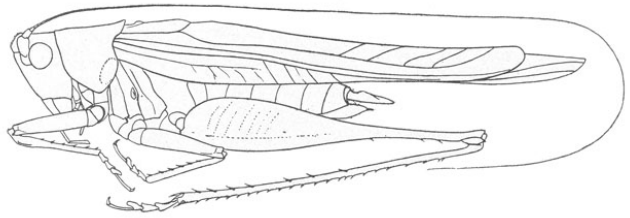
P*(*C* | *A*) = probability of class *C*, given that we have observed *A

P(Grasshopper | 7) = 3 / (3 + 9) = 0.250

P(Katydid | 7) = 9 / (3 + 9) = 0.750



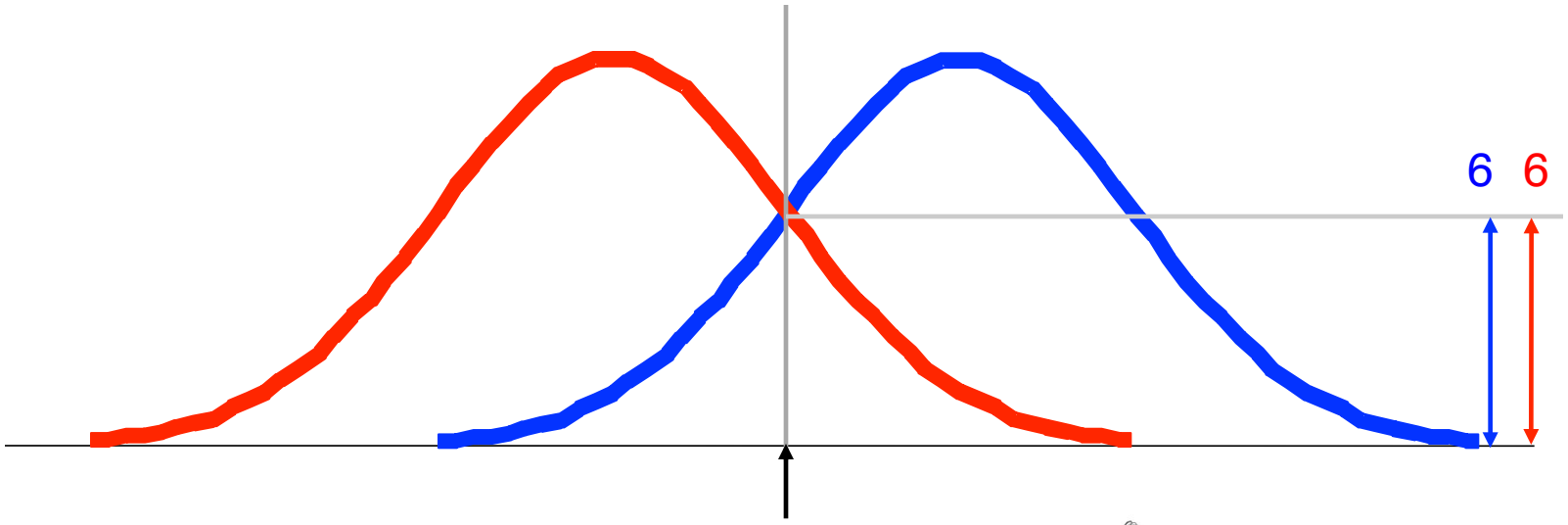
Antennae length is 7



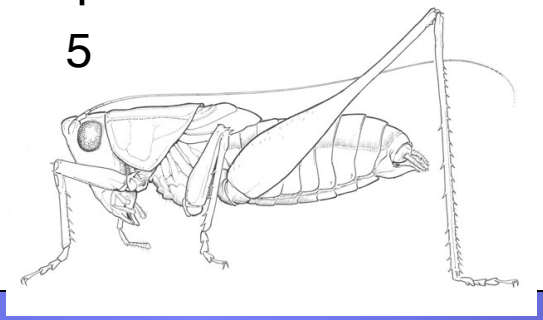
P*(*C* | *A*) = probability of class *C*, given that we have observed *A

P(Grasshopper | 5) = 6 / (6 + 6) = 0.500

P(Katydid | 5) = 6 / (6 + 6) = 0.500



Antennae length is 5



Bayes Classifiers

That was a visual intuition for a simple case of the Bayes classifier, also called the Naïve Bayes classifier.

We are about to see some of the mathematical formalisms, and more examples, but keep in mind the basic idea.

Find out the probability of the previously unseen instance belonging to each class, then simply pick the most probable class.

Bayes Classifier

- A probabilistic framework for solving classification problems
- Conditional Probability:

$$P(C \mid A) = \frac{P(A,C)}{P(A)}$$

$$P(A \mid C) = \frac{P(A,C)}{P(C)}$$

- Bayes theorem:

$$P(C \mid A) = \frac{P(A \mid C)P(C)}{P(A)}$$

Example of Bayes Theorem

- Given:
  - A doctor knows that meningitis causes stiff neck 50% of the time
  - Prior probability of any patient having meningitis is 1/50,000
  - Prior probability of any patient having stiff neck is 1/20

- If a patient has stiff neck, what's the probability he/she has meningitis?

$$P(M \mid S) = \frac{P(S \mid M)P(M)}{P(S)} = \frac{0.5 \times 1/50000}{1/20} = 0.0002$$

Bayesian Classifiers

- Consider each attribute and class label as random variables
- Given a record with attributes (A_1, A_2,…, A_n)
  - Goal is to predict class C
  - Specifically, we want to find the value of C that maximizes P(C | A_1, A_2,…, A_n)
- Can we estimate P(C | A_1, A_2,…, A_n) directly from data?

Bayesian Classifiers

- Approach:
  - compute the posterior probability $P(C \mid A_1, A_2, \ldots, A_n)$ for all values of C using the Bayes theorem

$$P(C \mid A_1 A_2 \ldots A_n) = \frac{P(A_1 A_2 \ldots A_n \mid C) P(C)}{P(A_1 A_2 \ldots A_n)}$$

  - Choose value of C that maximizes
    $P(C \mid A_1, A_2, \ldots, A_n)$

  - Equivalent to choosing value of C that maximizes
    $P(A_1, A_2, \ldots, A_n \mid C)\, P(C)$

- How to estimate $P(A_1, A_2, \ldots, A_n \mid C)$?

Closer Look At Bayes Theorem

$$P(C \mid A) = \frac{P(A \mid C)\, P(C)}{P(A)}$$

- $P(C \mid A)$ = probability of instance A being in class C,
  This is what we are trying to compute
- $P(A \mid C)$ = probability of generating instance A given class C,
  We can imagine that being in class C, causes you to have feature A with some probability
- $P(C)$ = probability of occurrence of class C,
  This is just how frequent the class C, is in our database
- $P(A)$ = probability of instance A occurring
  This can actually be ignored, since it is the same for all classes

How to Estimate Probabilities from Data?

Tid	Home Owner	Marital Status	Annual Income	Defaulted
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- Class: $P(C) = N_c/N$
  - i.e., P(No) = 7/10,
    P(Yes) = 3/10
- For discrete attributes:

$$P(A_i \mid C_k) = |A_{ik}| / N_c$$

  - where $|A_{ik}|$ is number of instances having attribute A_i and belongs to class C_k
  - Examples:

    P(MaritalStatus=Married|No) = 4/7
    P(HomeOwner=Yes|Yes)=0

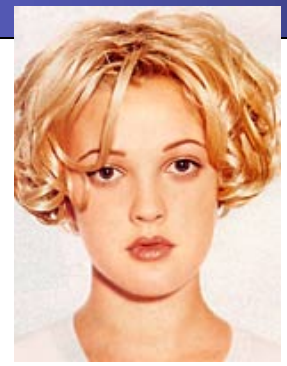
Assume that we have two classes

c_1 = male, and c_2 = female.

We have a person whose sex we do not know, say "*drew*" or *A*.

Classifying *drew* as male or female is equivalent to asking is it more probable that *drew* is male or female, i.e. which is greater *p*(male | *drew*) or *p*(female | *drew*)

(Note: "Drew can be a male or female name")



Drew Barrymore

Drew Carey

What is the probability of being called "*drew*" given that you are a male?

What is the probability of being a male?

$$P(\text{male} \mid drew) = \frac{P(drew \mid \text{male})\, P(\text{male})}{P(drew)}$$

What is the probability of being named "*drew*"? (actually irrelevant, since it is the same for all classes)

Officer Drew

This is Officer Drew. Is Officer Drew a Male or Female?

Luckily, we have a small database with names and sex.

We can use it to apply Bayes rule…

Name	**Sex**
Drew	Male
Claudia	Female
Drew	Female
Drew	Female
Alberto	Male
Karin	Female
Nina	Female
Sergio	Male

$$P(C \mid A) = \frac{P(A \mid C)\, P(C)}{P(A)}$$

Officer Drew

$$P(C \mid A) = \frac{P(A \mid C)\, P(C)}{p(A)}$$

Name	Sex
Drew	Male
Claudia	Female
Drew	Female
Drew	Female
Alberto	Male
Karin	Female
Nina	Female
Sergio	Male

$$P(\text{male} \mid drew) = \frac{1/3 \;*\; 3/8}{3/8} = \frac{0.125}{3/8}$$

$$p(\text{female} \mid drew) = \frac{2/5 \;*\; 5/8}{3/8} = \frac{0.250}{3/8}$$

Officer Drew is more likely to be a Female.

So far we have only considered Bayes Classification when we have one attribute (the "*antennae length*", or the "*name*"). But we may have many features. How do we use all the features?

$$P(C \mid A) = \frac{P(A \mid \mathrm{C})\, P(C)}{p(A)}$$

Name	**Over 170CM**	**Eye**	**Hair length**	**Sex**
Drew	No	Blue	Short	Male
Claudia	Yes	Brown	Long	Female
Drew	No	Blue	Long	Female
Drew	No	Blue	Long	Female
Alberto	Yes	Brown	Short	Male
Karin	No	Blue	Long	Female
Nina	Yes	Brown	Short	Female
Sergio	Yes	Blue	Long	Male

- To simplify the task, **naïve Bayesian classifiers** assume attributes have independent distributions, and thereby estimate

$$P(A|C) = P(A_1|C) * P(A_2|C) * \ldots * P(\mathrm{A_n}|C)$$

The probability of class *C* generating instance *A*, equals….

The probability of class *C* generating the observed value for feature 1, multiplied by..

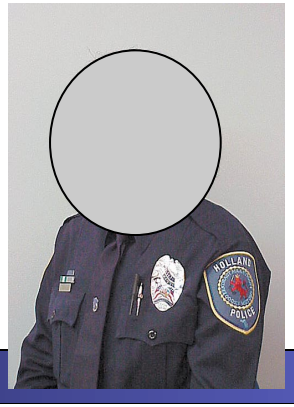
The probability of class *C* generating the observed value for feature 2, multiplied by..

- To simplify the task, **naïve Bayesian classifiers** assume attributes have independent distributions, and thereby estimate

$$P(A|C) = P(A_1|C) * P(A_2|C) * \ldots * P(A_n|C)$$

New point is classified to C if $P(C) \prod P(A_i|\, C)$ is *maximal*.

$$P(\text{officer drew}|C) = p(\text{over\_170}_{\text{cm}} = \text{yes}|C) * p(\text{eye} = \textit{blue}|C) * \ldots.$$



Officer Drew is blue-eyed, over 170_{cm} tall, and has long hair

$$p(\text{officer drew}|\ \text{Female}) = 2/5 * 3/5 * \ldots.$$

$$p(\text{officer drew}|\ \text{Male}) = 2/3 * 2/3 * \ldots.$$

How to Estimate Probabilities from Data?

- For continuous attributes:
  - Discretize the range into bins
    - one ordinal attribute per bin
    - violates independence assumption
  - Two-way split: $(A < v)$ or $(A > v)$
    - choose only one of the two splits as new attribute
  - Probability density estimation:
    - Assume attribute follows a normal distribution
    - Use data to estimate parameters of distribution (e.g., mean and standard deviation)
    - Once probability distribution is known, can use it to estimate the conditional probability $P(A_i|C)$

k