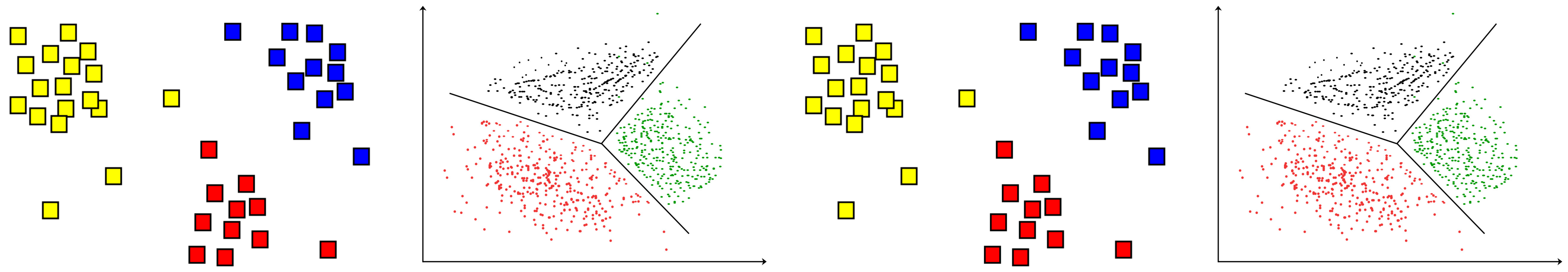
CPSC 425: Computer Vision



Lecture 25: Classification

Menu for Today (**November 9, 2020**)

Topics:

- Classification
- kNN, SVMs
- Bag of Words Representation
- Scene Classification

Redings:

- **Today's** Lecture: Forsyth & Ponce (2nd ed.) 15
- **Next** Lecture: Forsyth & Ponce (2nd ed.) 16.1.3, 16.1.4, 16.1.9

Reminders:

- **Assignment 5**: Scene Recognition with Bag of Words is **out**
- **Quiz 4** is out and due at the end of day **today**
- **No class** on **Wednsday**

Today's "**fun**" Example:

Audio-Visual Scene Analysis with
Self-Supervised Multisensory Features

Andrew Owens   Alexei A. Efros
UC Berkeley

BAIR
BERKELEY ARTIFICIAL INTELLIGENCE RESEARCH

Today's "**fun**" Example:

Audio-Visual Scene Analysis with Self-Supervised Multisensory Features

Andrew Owens   Alexei A. Efros

UC Berkeley

BAIR

BERKELEY ARTIFICIAL INTELLIGENCE RESEARCH

Classification

Problem:

Assign new observations into one of a fixed set of categories (classes)

Key Idea(s):

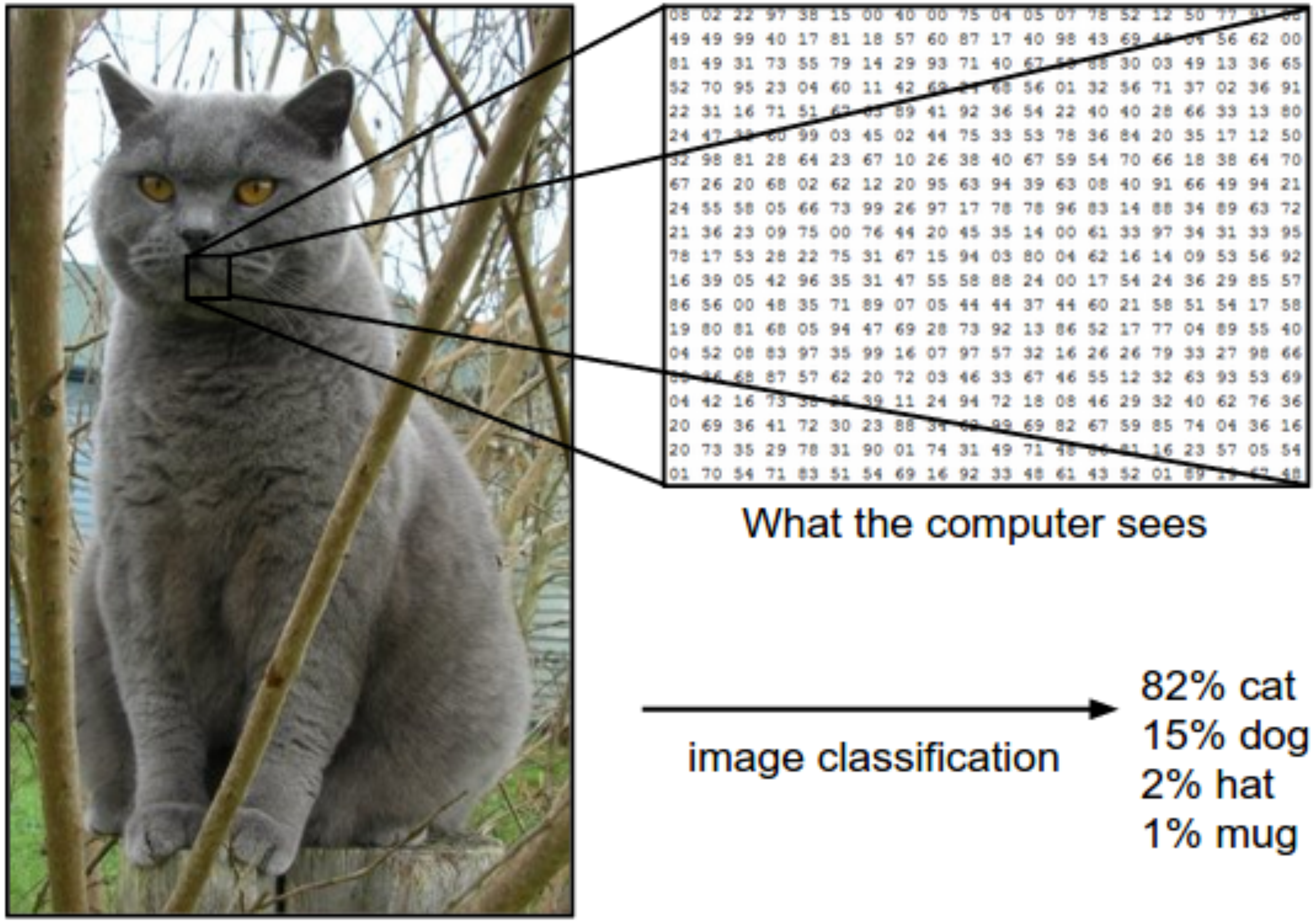
Build a model of data in a given category based on observations of instances in that category

Classification

(assume given set of discrete labels)
{dog, cat, truck, plane, ...}

$\longrightarrow$ cat

Classification



Classification

A **classifier** is a procedure that accepts as input a set of features and outputs a class **label** (probability over class labels)

Classifiers can be **binary** (face vs. not-face) or **multi-class** (cat, dog, horse, ...).

We build a classifier using a **training set** of labelled examples $\{(\mathbf{x}_i, y_i)\}$, where each $\mathbf{x}_i$ is a feature vector and each y_i is a class label.

Given a previously unseen observation, we use the classifier to predict its class label.

Binary: $[0]/[1]$ **Multi-class**: $[1, 0, 0, 0, ...]$ (one-hot)

Classification

- Collect a database of images with labels
- Use ML to train an image classifier
- Evaluate the classifier on test images

Label

Feature vector computed from the image

Example 1: A Classification Problem

Categorize images of fish
— “Atlantic salmon” vs “Pacific salmon”

Use **features** such as length, width, lightness, fin shape & number, mouth position, etc.

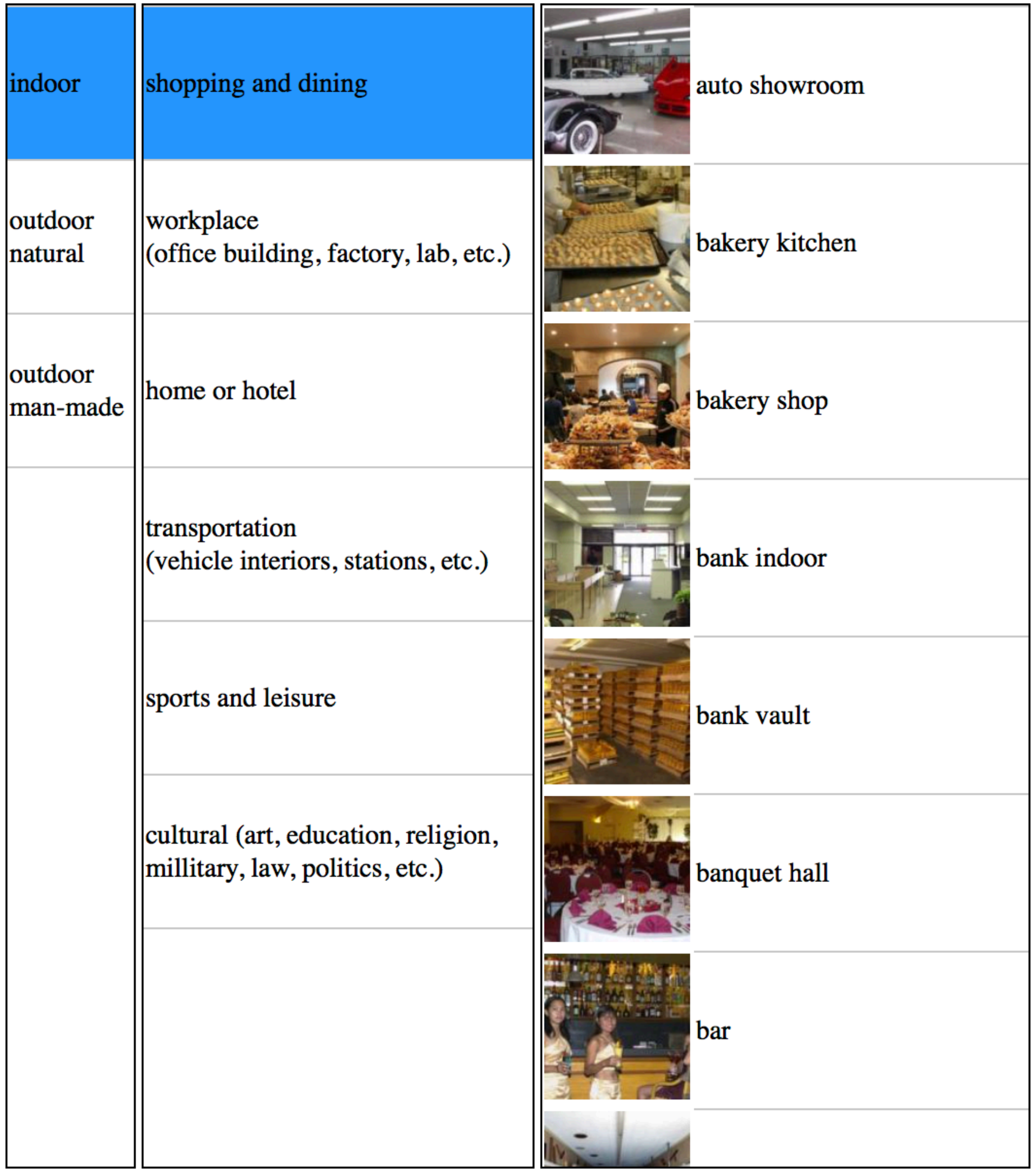
Given a previously unobserved image of a salmon, use the learned classifier to guess whether it is an Atlantic or Pacific salmon

Figure credit: Duda & Hart

Example 2: Real Classification Problem

SUN Dataset

- 131K images
- 908 **scene** categories

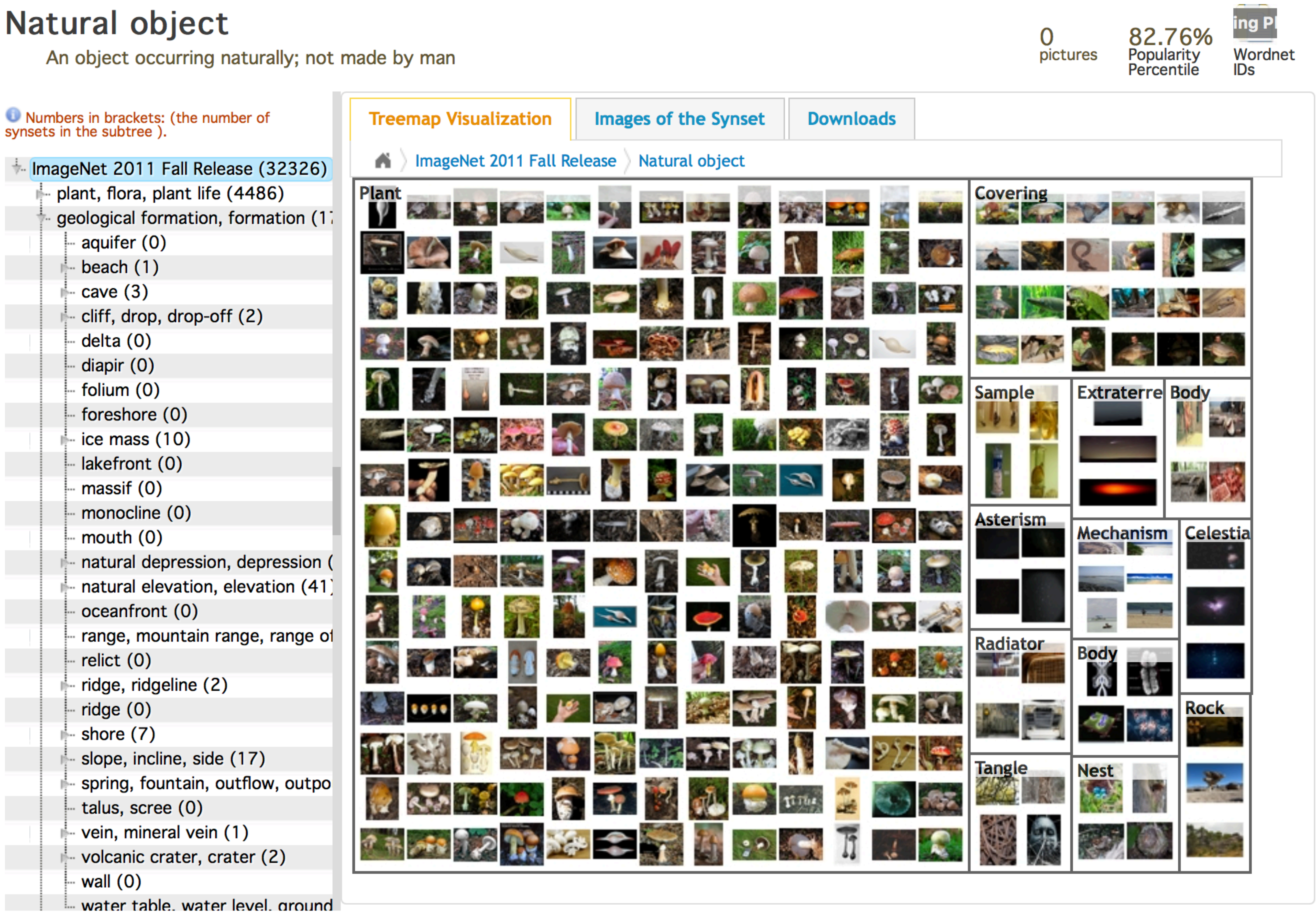


Example 3: Real Classification Problem

ImageNet Dataset

- 14 Million images

- 21K **object** categories

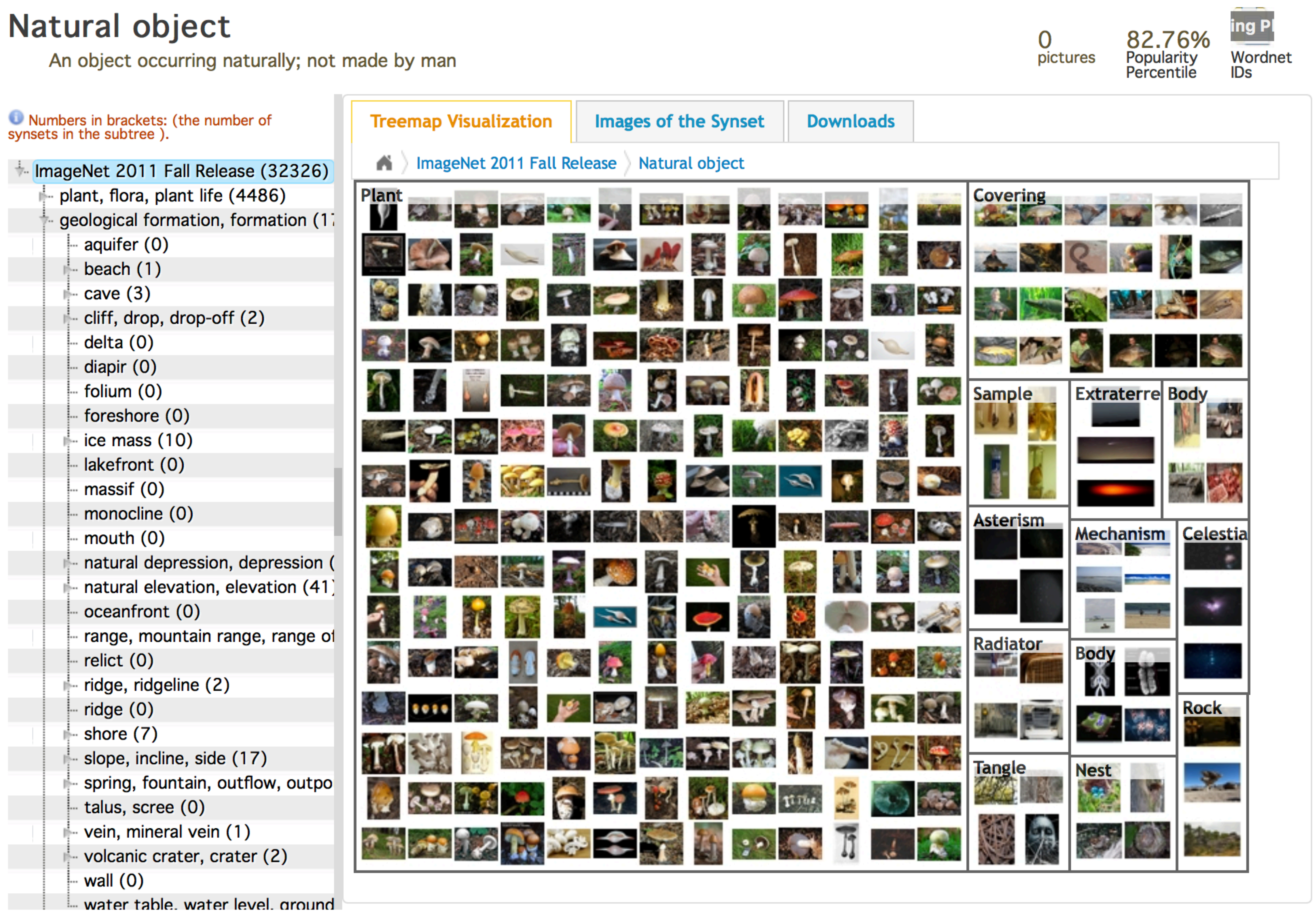


Example 3: Real Classification Problem

ImageNet Dataset

- 14 Million images

- 21K **object** categories



Closed-world problem

Issue: Classification assumes that incoming image belongs to one of k classes. However, in practice it is impossible to enumerate all relevant classes in the world, nor would doing so be useful. So how do we deal with images which don't belong?

Solution: Create an "unknown" or "irrelevant" class.

Bayes Rule (Review and Definitions)

Let c be the **class label** and let x be the **measurement** (i.e., evidence)

$$P(c|x) = \frac{P(x|c)p(c)}{P(x)}$$

posterior probability

Bayes Rule (Review and Definitions)

Let c be the **class label** and let x be the **measurement** (i.e., evidence)

class–conditional probability
(a.k.a. likelihood)

prior probability

$$P(c|x) = \frac{P(x|c)p(c)}{P(x)}$$

posterior probability

unconditional probability
(a.k.a. marginal likelihood)

Bayes Rule (Review and Definitions)

Let c be the **class label** and let x be the **measurement** (i.e., evidence)

Simple case:

— binary classification; i.e., $c \in \{1, 2\}$

— features are 1D; i.e., $x \in \mathbb{R}$

$$P(c|x) = \frac{P(x|c)p(c)}{P(x)}$$

Bayes Rule (Review and Definitions)

Let c be the **class label** and let x be the **measurement** (i.e., evidence)

Simple case:

— binary classification; i.e., $c \in \{1, 2\}$

— features are 1D; i.e., $x \in \mathbb{R}$

$$P(c|x) = \frac{P(x|c)p(c)}{P(x)}$$

Classify ***x*** as

1 if p(1|***x***) > p(2|***x***)

2 if p(1|***x***) < p(2|***x***)

Bayes Rule (Review and Definitions)

Let c be the **class label** and let x be the **measurement** (i.e., evidence)

Simple case:

— binary classification; i.e., $c \in \{1, 2\}$

— features are 1D; i.e., $x \in \mathbb{R}$

$$P(c|x) = \frac{P(x|c)p(c)}{P(x)}$$

General case:

— multi-class; i.e., $c \in \{1, ..., 1000\}$

— features are high-dimensional; i.e., $x \in \mathbb{R}^{2,000+}$

Example: Discrete Bayes Classifier

Assume we have two classes: $c_1 = \textbf{male}$ $c_2 = \textbf{female}$

We have a person who's gender we don't know, who's name is *drew*

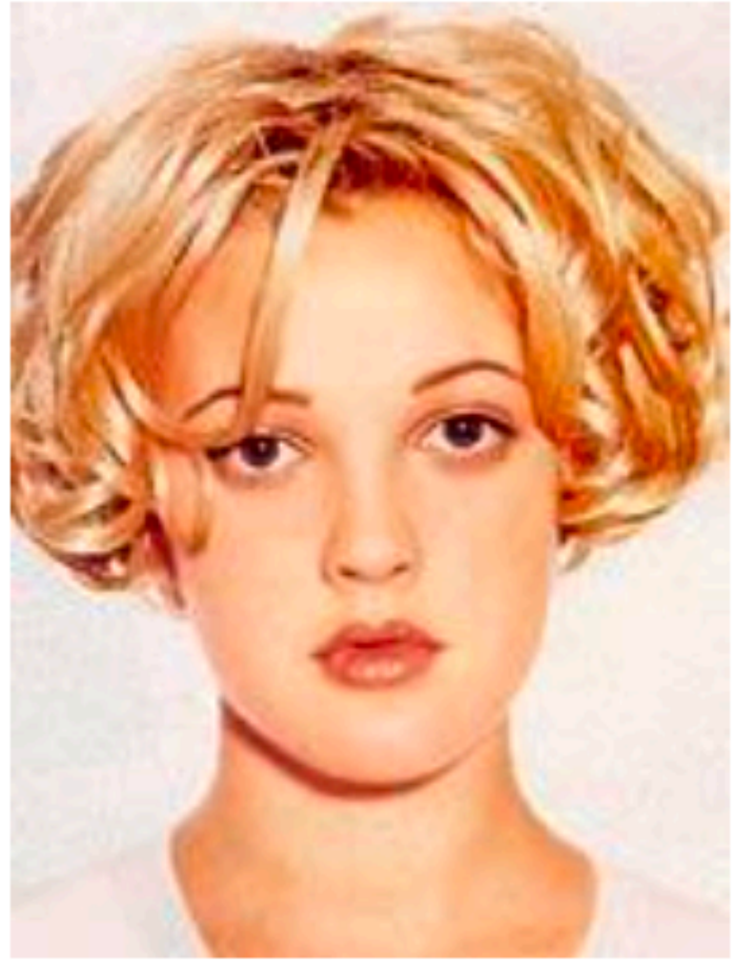
Example from: Eamonn Keogh

Example: Discrete Bayes Classifier

Assume we have two classes: $c_1 = \textbf{male}$ $c_2 = \textbf{female}$

We have a person who's gender we don't know, who's name is *drew*

Drew Carey



Drew Barrymore

Example from: Eamonn Keogh

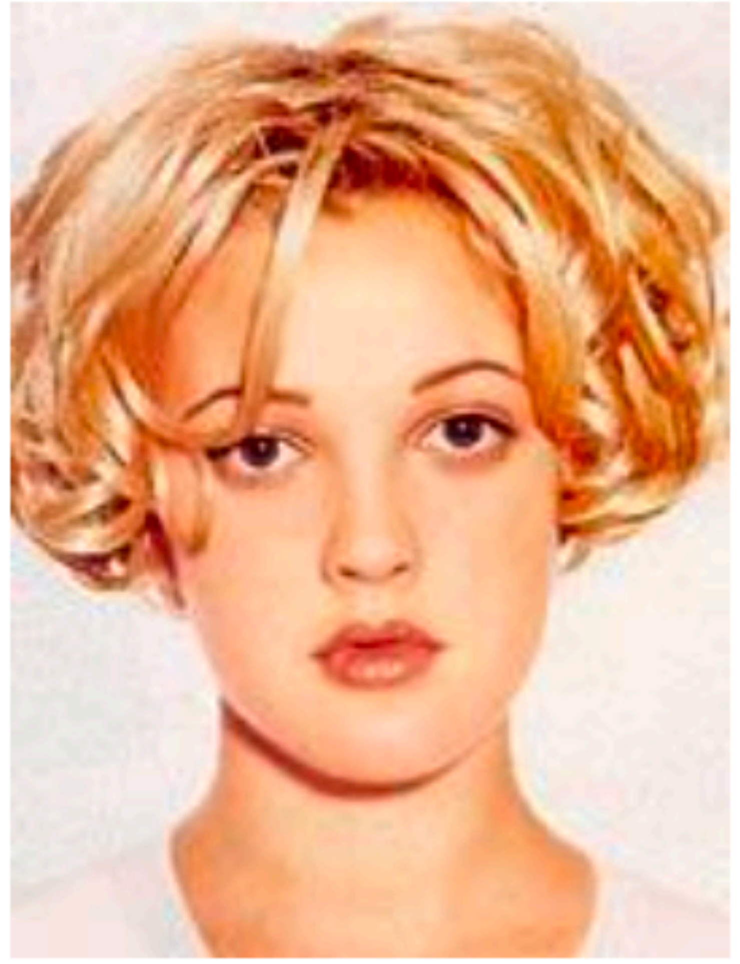
Example: Discrete Bayes Classifier

Assume we have two classes: $c_1 = \textbf{male}$ $c_2 = \textbf{female}$

We have a person who's gender we don't know, who's name is *drew*

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

Drew Carey



Drew Barrymore

Example from: Eamonn Keogh

Example: Discrete Bayes Classifier

Assume we have two classes: $c_1 = \textbf{male}$ $c_2 = \textbf{female}$

We have a person who's gender we don't know, who's name is *drew*

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

$p(\textbf{female}|drew)$

$$p(\textbf{male}|drew) = \frac{p(drew|\textbf{male})p(\textbf{male})}{p(drew)}$$

Example from: Eamonn Keogh

Example: Discrete Bayes Classifier

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

$$p(\mathbf{male}|drew) = \frac{p(drew|\mathbf{male})p(\mathbf{male})}{p(drew)}$$

Example from: Eamonn Keogh

Example: Discrete Bayes Classifier

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

$$p(\mathbf{male}) =$$

$$p(drew|\mathbf{male}) =$$

$$p(drew) =$$

$$p(\mathbf{male}|drew) = \frac{p(drew|\mathbf{male})p(\mathbf{male})}{p(drew)}$$

Example from: Eamonn Keogh

Example: Discrete Bayes Classifier

$$p(\textbf{male}) = \frac{3}{8}$$

$$p(drew|\textbf{male}) =$$

$$p(drew) =$$

$$p(\textbf{male}|drew) = \frac{p(drew|\textbf{male})p(\textbf{male})}{p(drew)}$$

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

Example from: Eamonn Keogh

Example: Discrete Bayes Classifier

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

$$p(\mathbf{male}) = \frac{3}{8}$$

$$p(drew|\mathbf{male}) = \frac{1}{3}$$

$$p(drew) =$$

$$p(\mathbf{male}|drew) = \frac{p(drew|\mathbf{male})p(\mathbf{male})}{p(drew)}$$

Example from: Eamonn Keogh

Example: Discrete Bayes Classifier

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

$$p(\mathbf{male}) = \frac{3}{8}$$

$$p(drew|\mathbf{male}) = \frac{1}{3}$$

$$p(drew) = \frac{3}{8}$$

$$p(\mathbf{male}|drew) = \frac{p(drew|\mathbf{male})p(\mathbf{male})}{p(drew)}$$

Example from: Eamonn Keogh

Example: Discrete Bayes Classifier

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

$$p(\mathbf{male}) = \frac{3}{8}$$

$$p(drew|\mathbf{male}) = \frac{1}{3}$$

$$\cancel{p(drew) = \frac{3}{8}}$$

$$p(\mathbf{male}|drew) = \frac{p(drew|\mathbf{male})p(\mathbf{male})}{\cancel{p(drew)}} = 0.125$$

Example from: Eamonn Keogh

Example: Discrete Bayes Classifier

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

$$p(\mathbf{male}) = \frac{3}{8} \qquad p(\mathbf{female}) = \frac{5}{8}$$

$$p(drew|\mathbf{male}) = \frac{1}{3} \qquad p(drew|\mathbf{female}) = \frac{2}{5}$$

$$\cancel{p(drew) = \frac{3}{8}}$$

$$p(\mathbf{male}|drew) = \frac{p(drew|\mathbf{male})p(\mathbf{male})}{\cancel{p(drew)}} = 0.125$$

$$p(\mathbf{female}|drew) = \frac{p(drew|\mathbf{female})p(\mathbf{female})}{\cancel{p(drew)}} = 0.25$$

Example from: Eamonn Keogh

Example: 2D Bayes Classifier

- 17 samples
- 15 samples

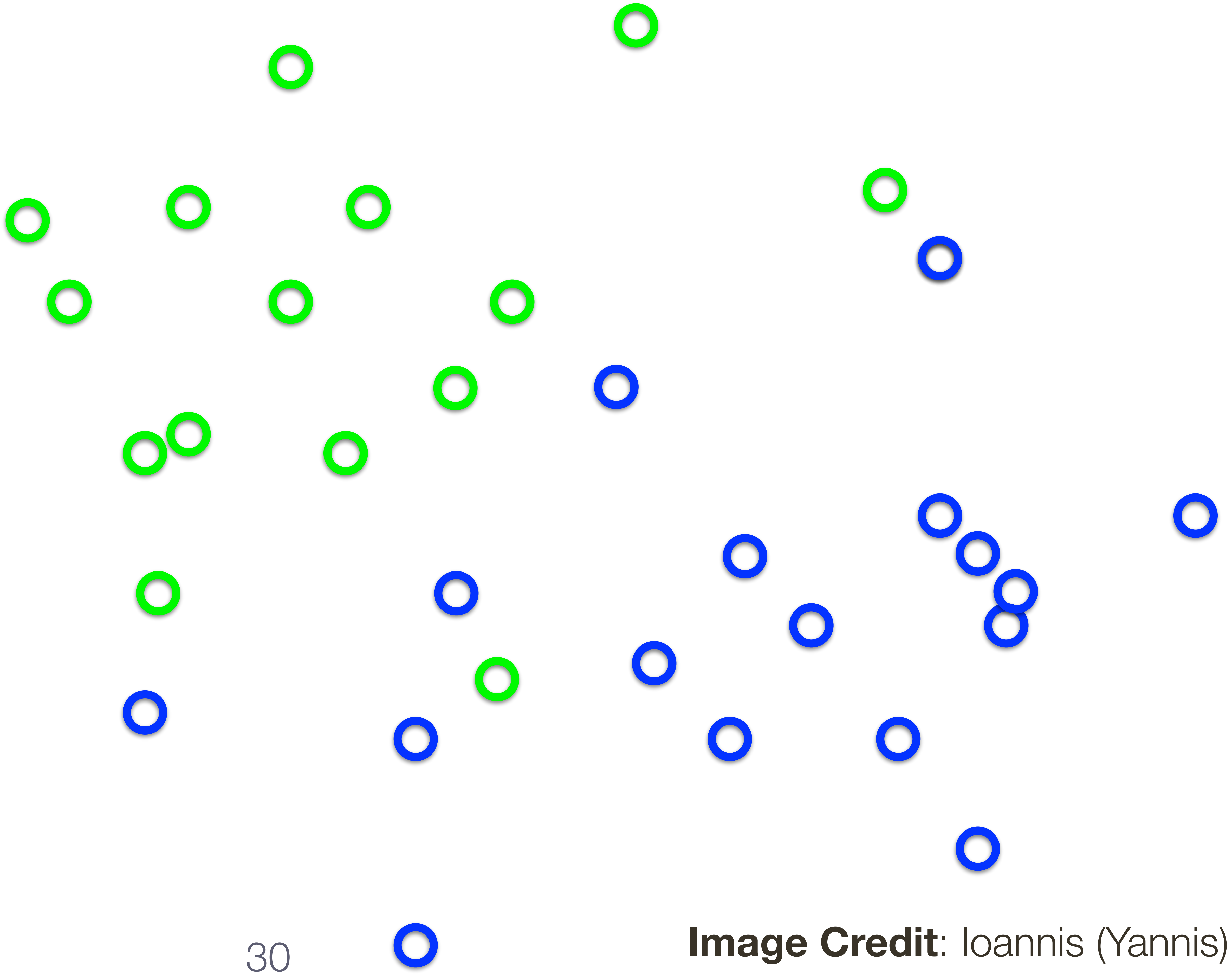


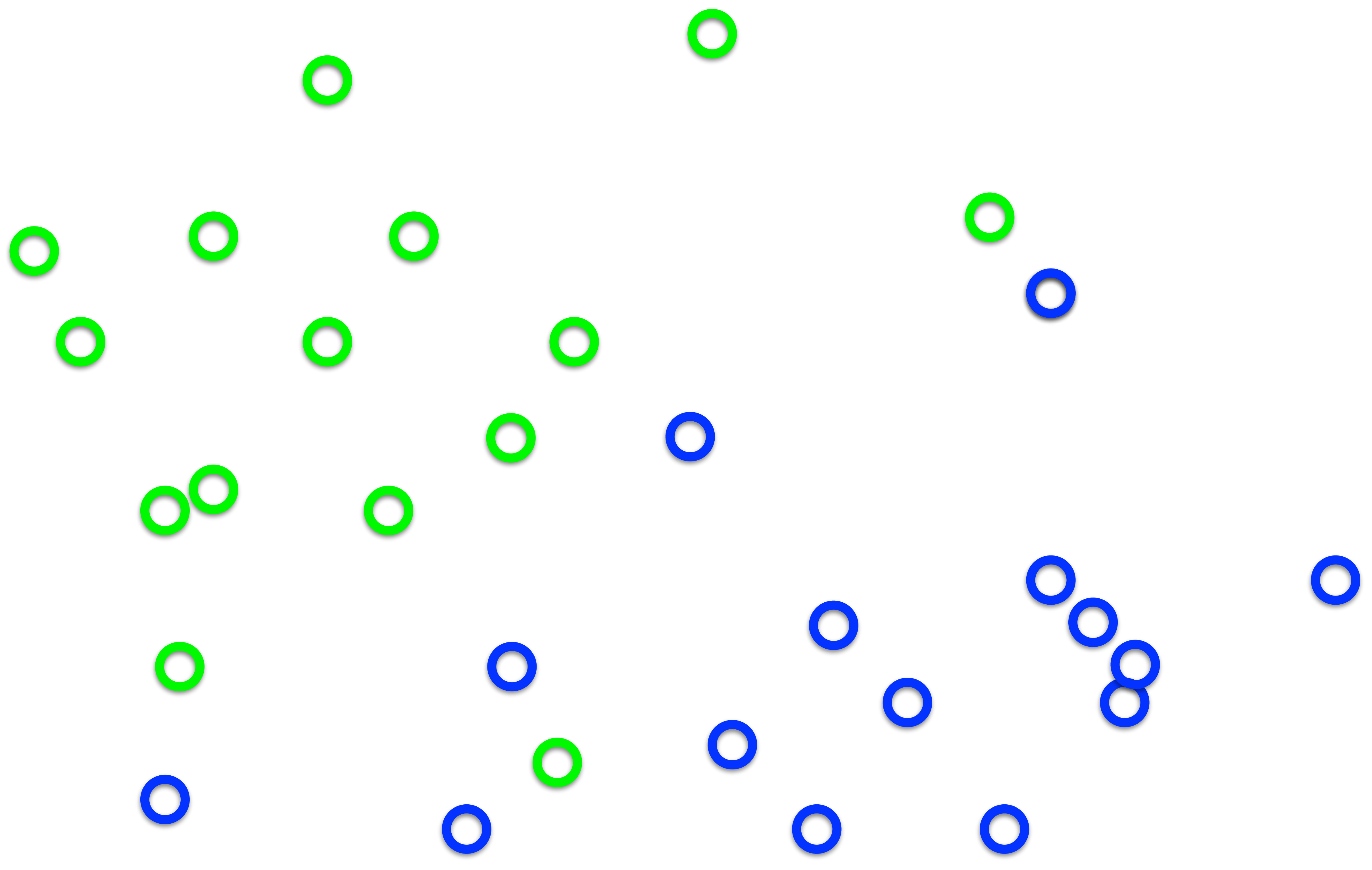
Image Credit: Ioannis (Yannis) Gkioulekas (CMU)

Example: 2D Bayes Classifier

- 17 samples
- 15 samples

These could be (g,b) pixel value of an image patch with grass

Given a (g,b) pixel value from a new patch is it more likely to be be grass or sky?



These could be (g,b) pixel value of an image patch with sky

Image Credit: Ioannis (Yannis) Gkioulekas (CMU)

Example: 2D Bayes Classifier

- 17 samples
- 15 samples

$$p(blue) = \frac{17}{17 + 15}$$

$$p(green) = \frac{15}{17 + 15}$$

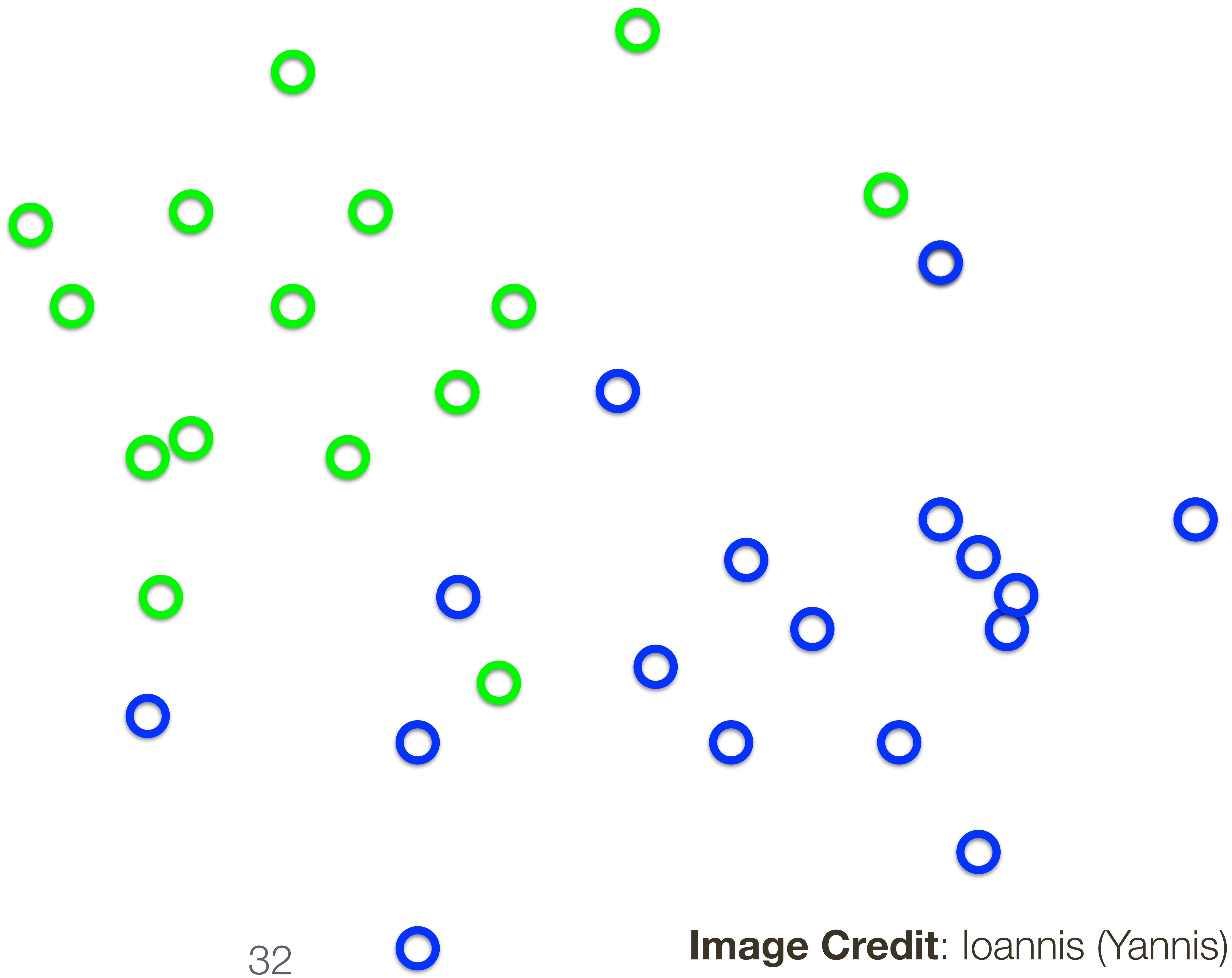


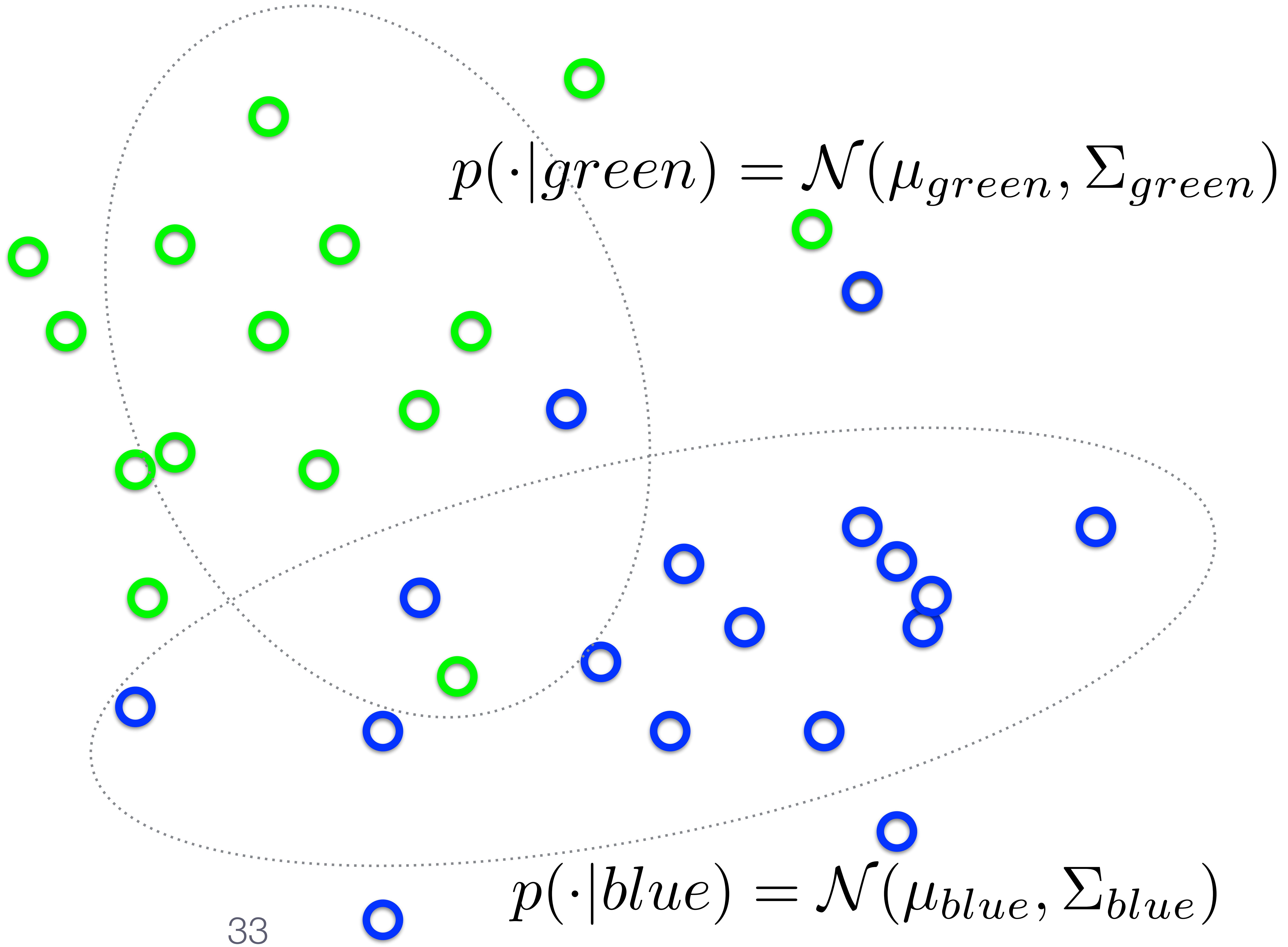
Image Credit: Ioannis (Yannis) Gkioulekas (CMU)

Example: 2D Bayes Classifier

- 17 samples (blue)
- 15 samples (green)

$$p(blue) = \frac{17}{17+15}$$

$$p(green) = \frac{15}{17+15}$$



Example: 2D Bayes Classifier

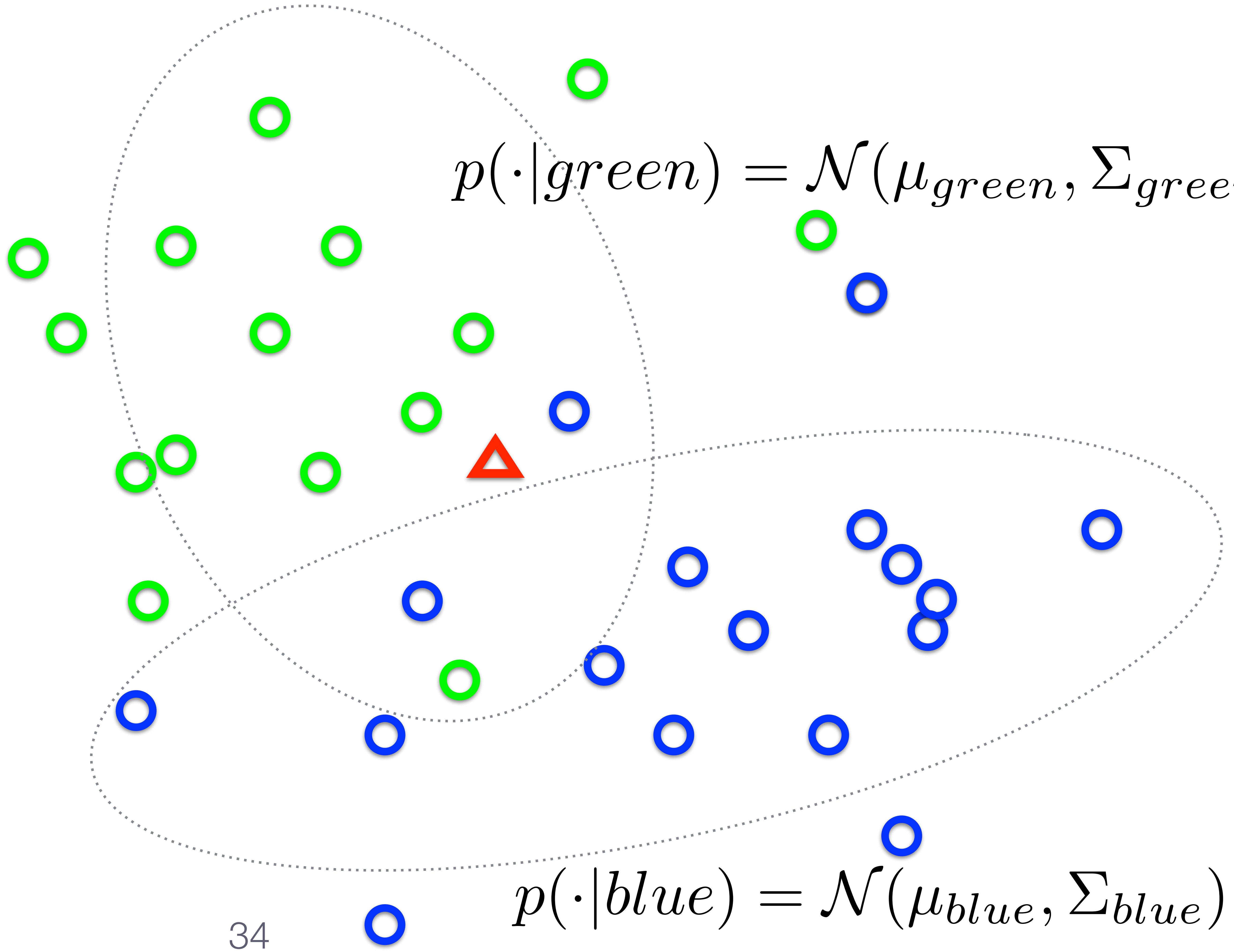
$$p(green|\triangle) \propto \mathcal{N}(\triangle; \mu_{green}, \Sigma_{green})p(green)$$

$$p(blue|\triangle) \propto \mathcal{N}(\triangle; \mu_{blue}, \Sigma_{blue})p(blue)$$

- 17 samples
- 15 samples

$$p(blue) = \frac{17}{17+15}$$

$$p(green) = \frac{15}{17+15}$$



$$p(\cdot|green) = \mathcal{N}(\mu_{green}, \Sigma_{green})$$

$$p(\cdot|blue) = \mathcal{N}(\mu_{blue}, \Sigma_{blue})$$

Bayes Rule (Review and Definitions)

Let c be the **class label** and let x be the **measurement** (i.e., evidence)

Simple case:

— binary classification; i.e., $c \in \{1, 2\}$

— features are 1D; i.e., $x \in \mathbb{R}$

$$P(c|x) = \frac{P(x|c)p(c)}{P(x)}$$

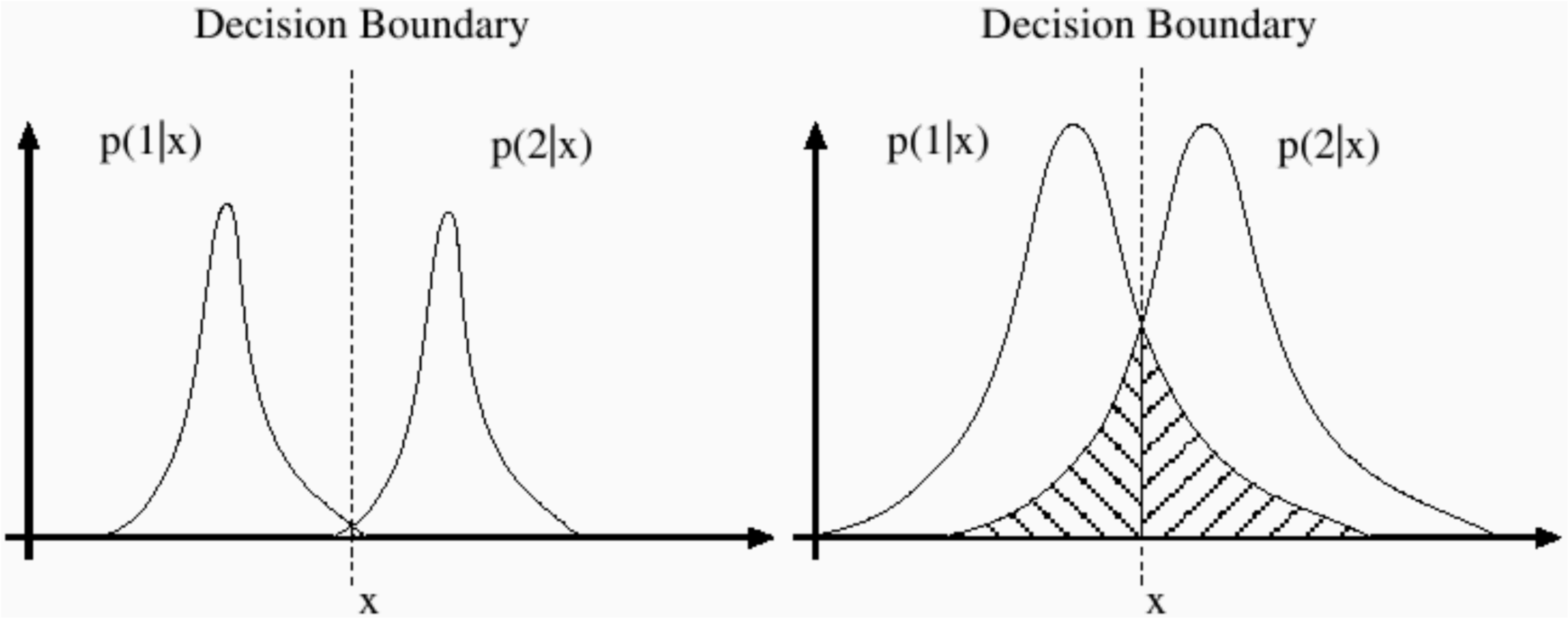
General case:

— multi-class; i.e., $c \in \{1, ..., 1000\}$

— features are high-dimensional; i.e., $x \in \mathbb{R}^{2,000+}$

Bayes' Risk

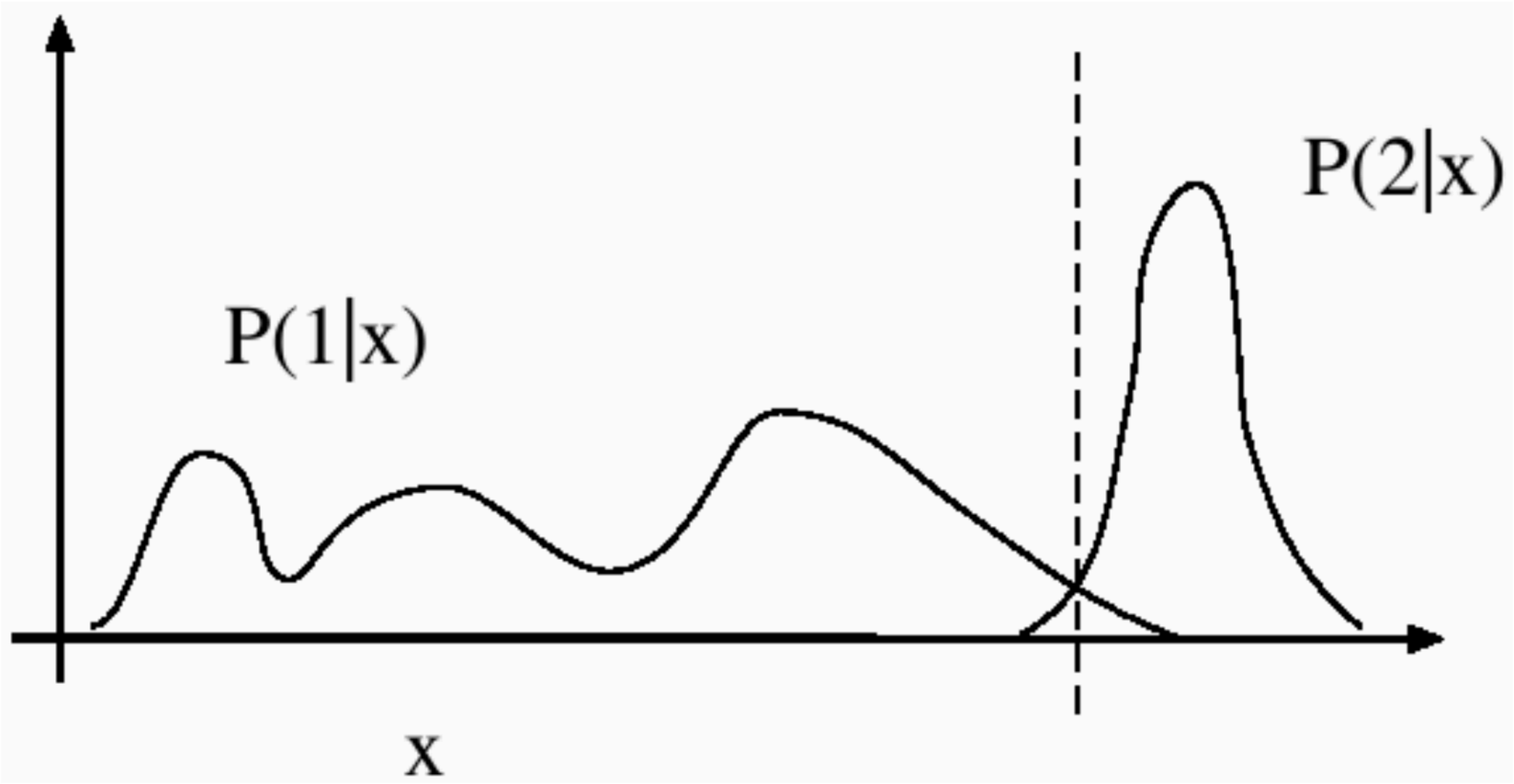
Some errors may be inevitable: the minimum risk (shaded area) is called the **Bayes' risk**



Forsyth & Ponce (2nd ed.) Figure 15.1

Discriminative vs. **Generative**

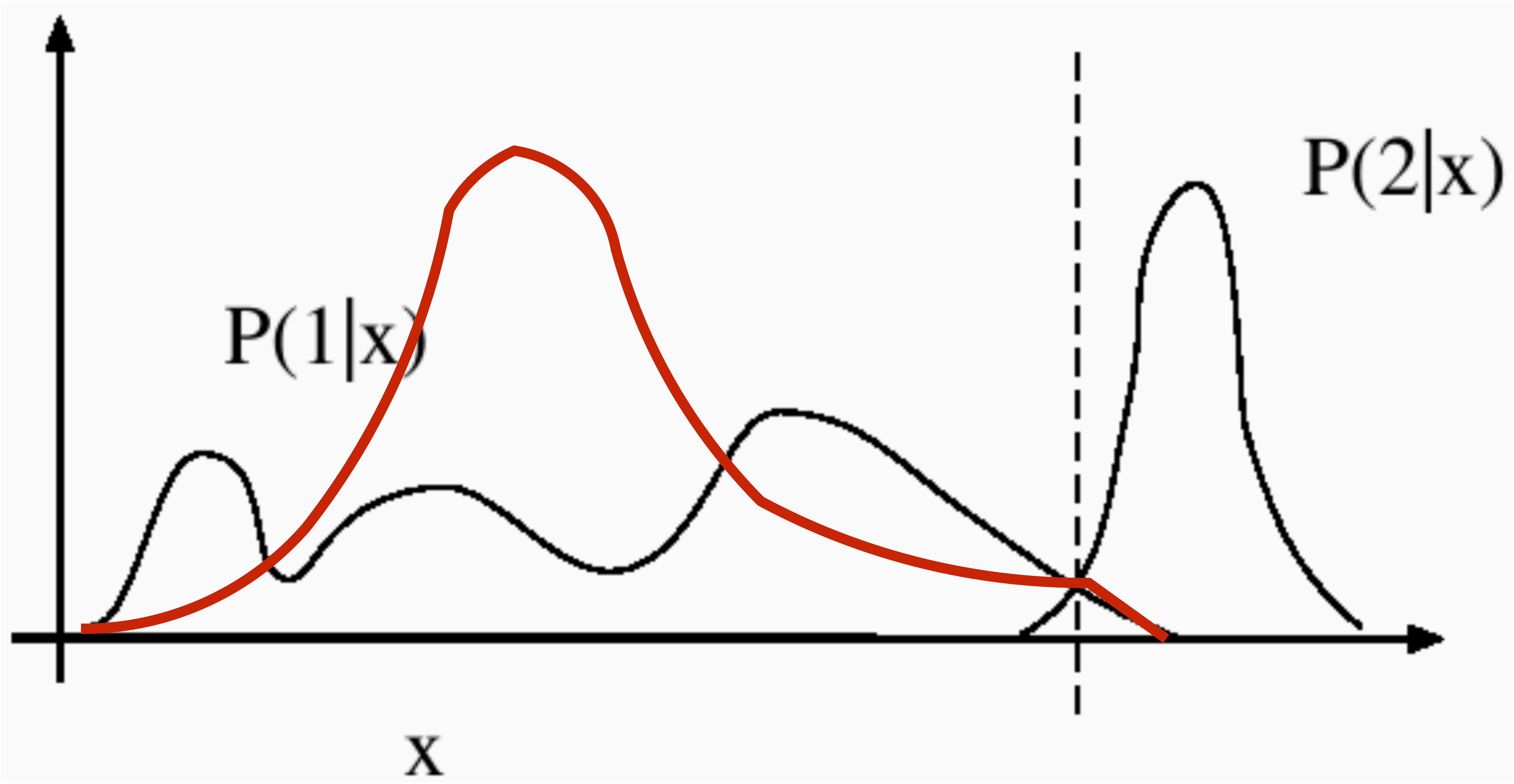
Finding a **decision boundary** is not the same as modeling a **conditional density** — while a normal density here is a poor fit to P(1|x), the quality of the classifier depends only on how well the boundary is positioned



Forsyth & Ponce (2nd ed.) Figure 15.5

Discriminative vs. **Generative**

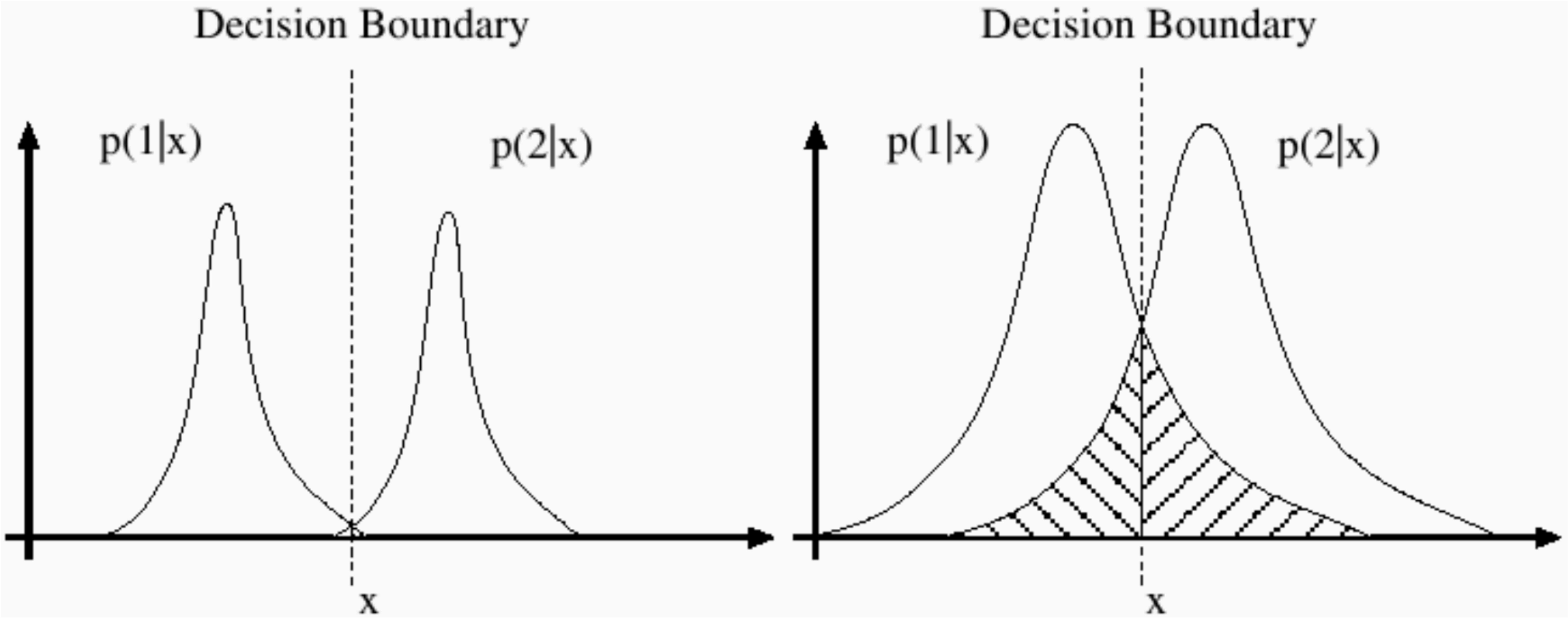
Finding a **decision boundary** is not the same as modeling a **conditional density** — while a normal density here is a poor fit to P(1|x), the quality of the classifier depends only on how well the boundary is positioned



Forsyth & Ponce (2nd ed.) Figure 15.5

Bayes’ Risk

Some errors may be inevitable: the minimum risk (shaded area) is called the **Bayes’ risk**



Forsyth & Ponce (2nd ed.) Figure 15.1

Loss Functions and Classifiers

Loss

- Some errors may be more expensive than others

  Example: A fatal disease that is easily cured by a cheap medicine with no side-effects. Here, false positives in diagnosis are better than false negatives

- We discuss two class classification:
  $L(1 \rightarrow 2)$ is the loss caused by calling 1 a 2

Total risk of using classifier ***s*** is

$$R(s) = \Pr\{1 \rightarrow 2 \mid \text{using } \boldsymbol{s}\}\, L(1 \rightarrow 2) + \Pr\{2 \rightarrow 1 \mid \text{using } \boldsymbol{s}\}\, L(2 \rightarrow 1)$$

Two Class Classification

Generally, we should classify as 1 if the expected loss of classifying as 1 is less than for 2

Classify $\boldsymbol{x}$ as

$$1 \quad \text{if } p(1|\boldsymbol{x})\, L(1 \to 2) > p(2|\boldsymbol{x})\, L(2 \to 1)$$

$$2 \quad \text{if } p(1|\boldsymbol{x})\, L(1 \to 2) < p(2|\boldsymbol{x})\, L(2 \to 1)$$

Decision boundary: points where the loss is the same for either class.

Training Error, **Testing** Error, and Overfitting

Training error is the error a classifier makes on the training set

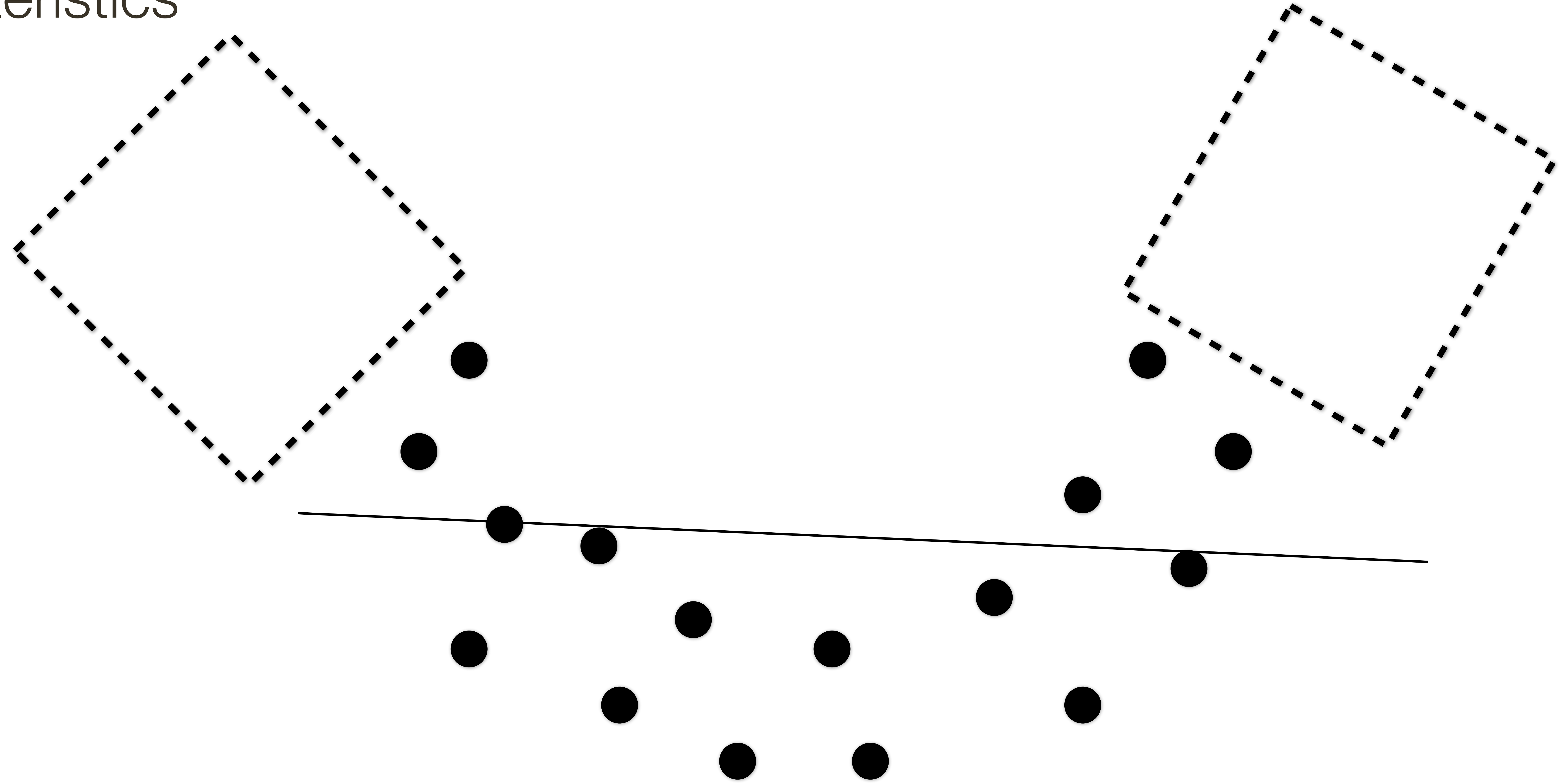
We want to minimize the **testing error** – the error the classifier makes on an unseen testing set

Classifiers that have small training error may not necessarily have small testing error

The phenomenon that causes testing error to be worse than training error is called **overfitting**

Training Error, Testing Error, and **Overfitting**

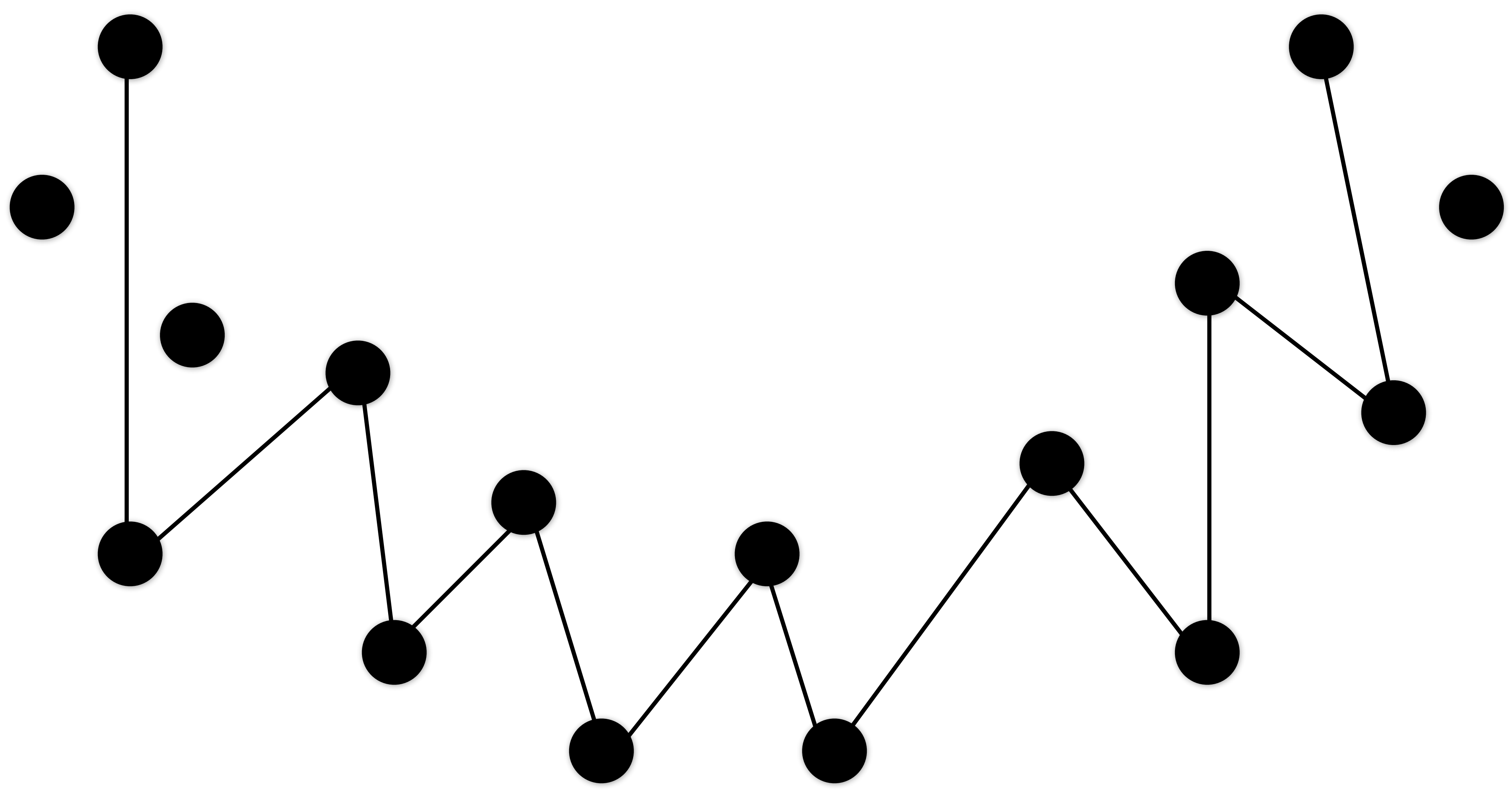
Underfitting: model is too simple to represent all the relevant class characteristics



Training Error, Testing Error, and **Overfitting**

Underfitting: model is too simple to represent all the relevant class characteristics

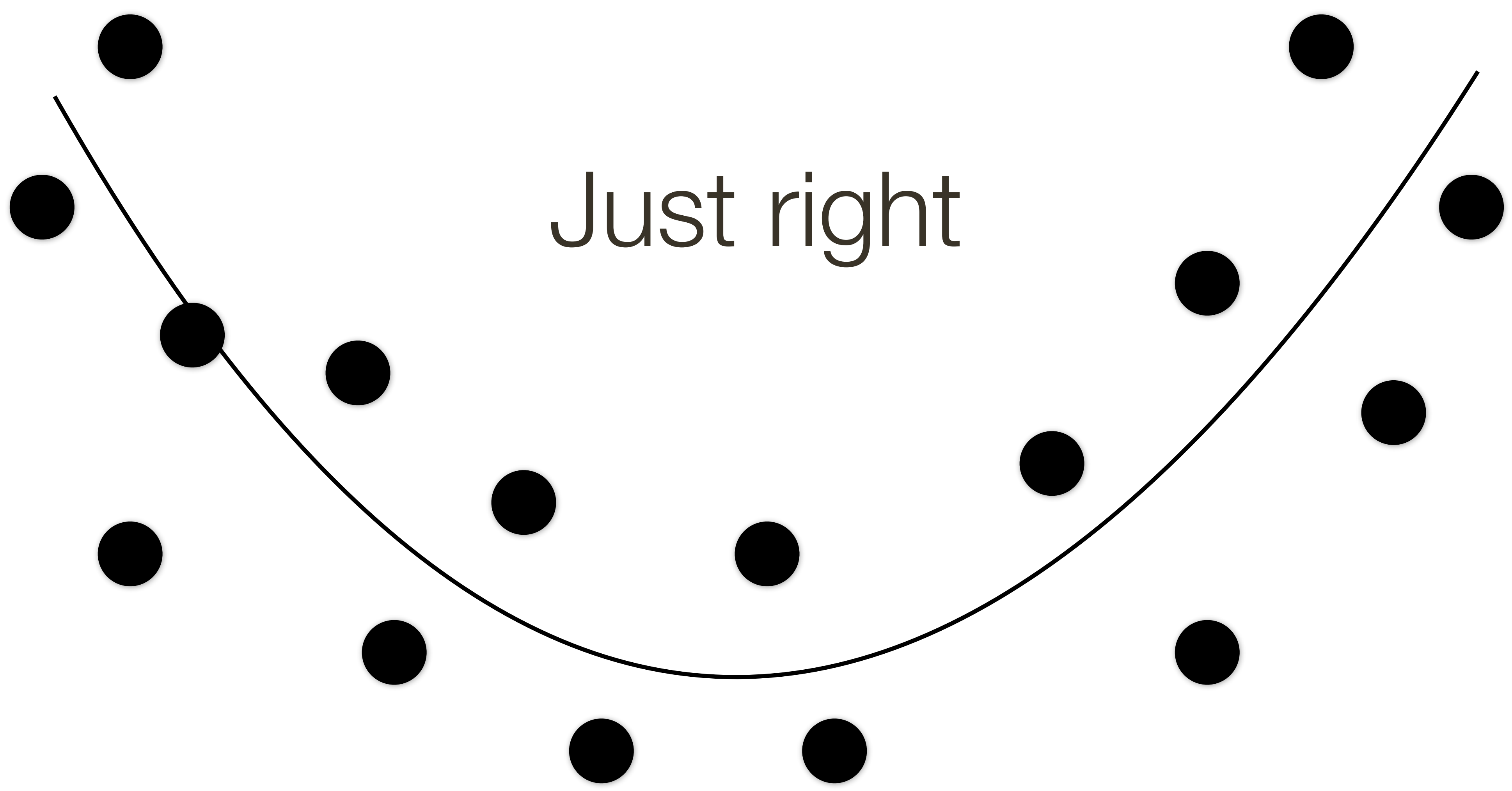
Overfitting: model is too complex and fits irrelevant characteristics (noise) in the data



Training Error, Testing Error, and **Overfitting**

Underfitting: model is too simple to represent all the relevant class characteristics

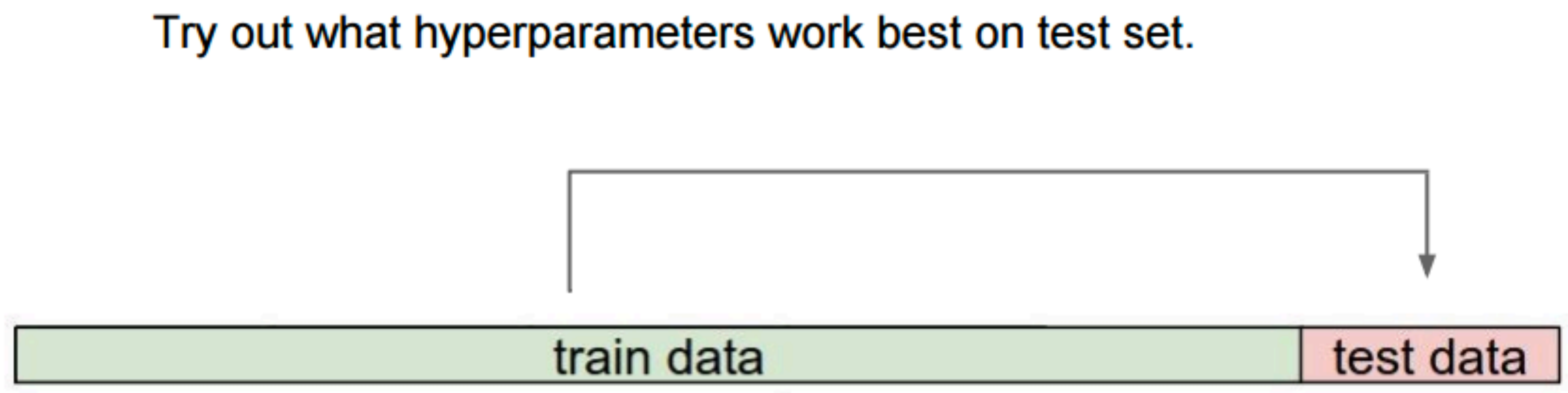
Overfitting: model is too complex and fits irrelevant characteristics (noise) in the data



Cross-Validation

We cannot reliably estimate the error rate of the classifier using the training set

An alternative is to split some training data to form a **validation** set, then train the classifier on the rest of the data and evaluate on the validation set

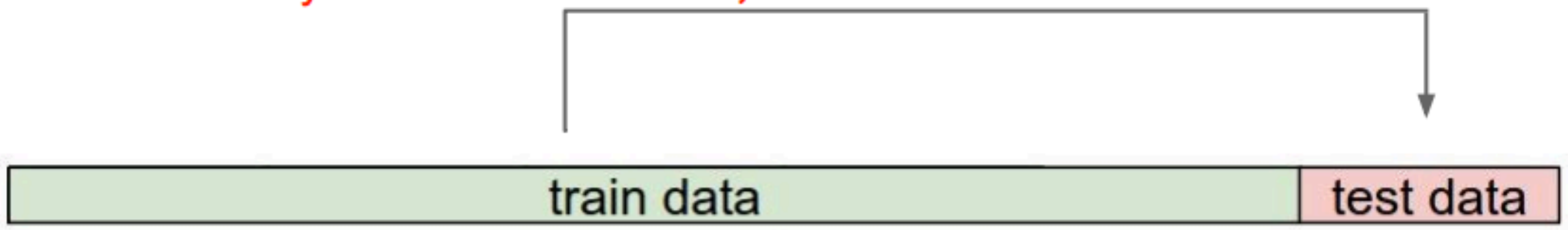


Cross-Validation

We cannot reliably estimate the error rate of the classifier using the training set

An alternative is to split some training data to form a **validation** set, then train the classifier on the rest of the data and evaluate on the validation set

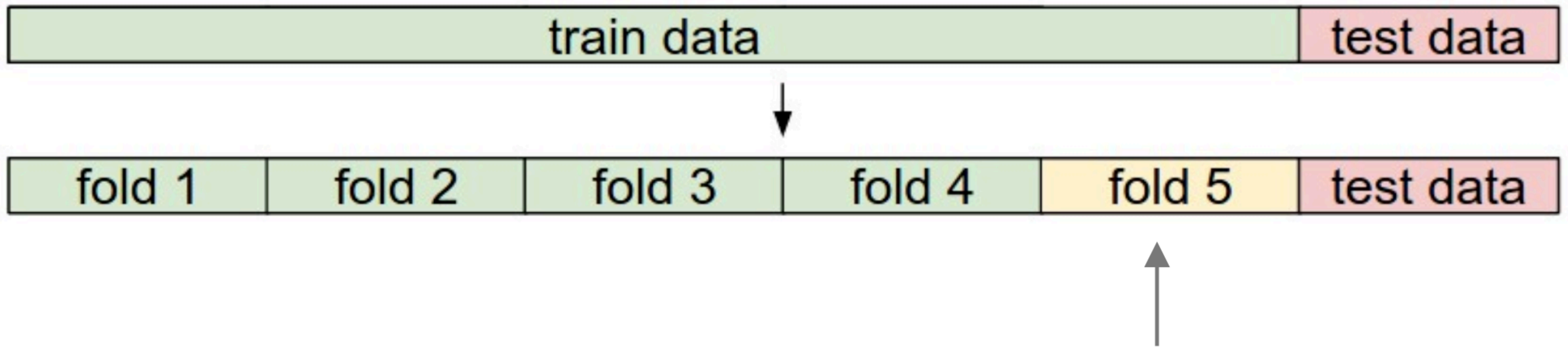
Trying out what hyperparameters work best on test set:
Very bad idea. The test set is a proxy for the generalization performance!
Use only **VERY SPARINGLY,** at the end.



Cross-Validation

We cannot reliably estimate the error rate of the classifier using the training set

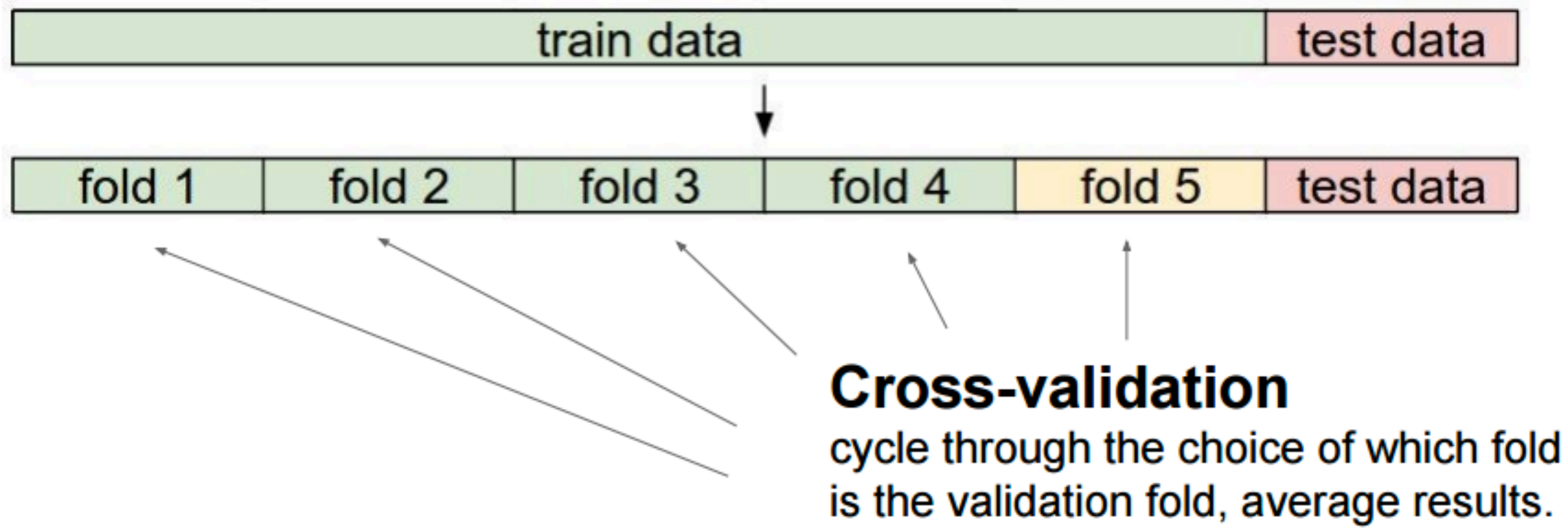
An alternative is to split some training data to form a **validation** set, then train the classifier on the rest of the data and evaluate on the validation set



Validation data
use to tune hyperparameters
evaluate on test set ONCE at the end

Cross-Validation

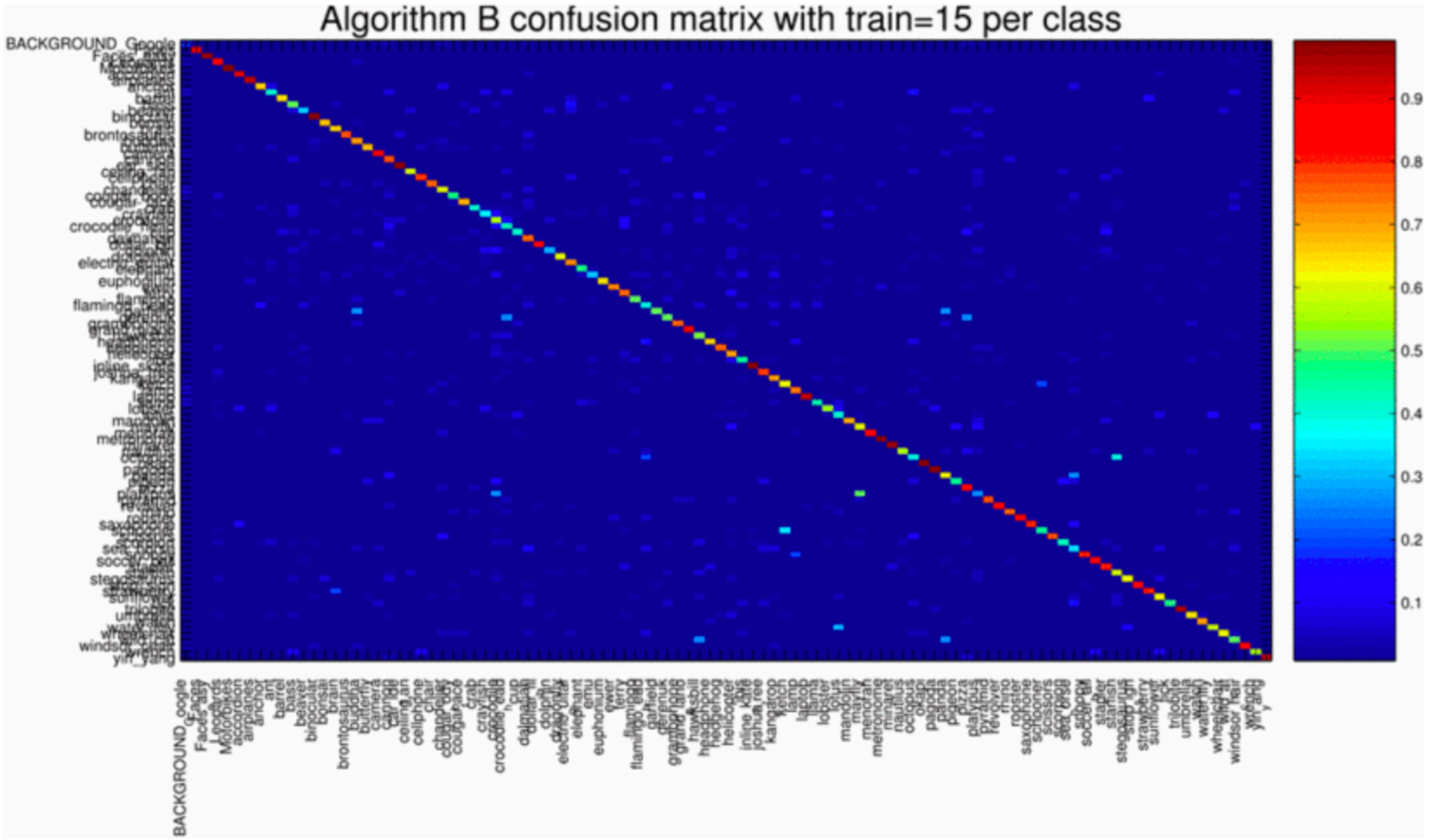
Cross-validation involves performing multiple splits and averaging the error over all splits



Confusion Matrix

When evaluating a multi-class classifier, it may be useful to know how often certain classes are often misclassified as others.

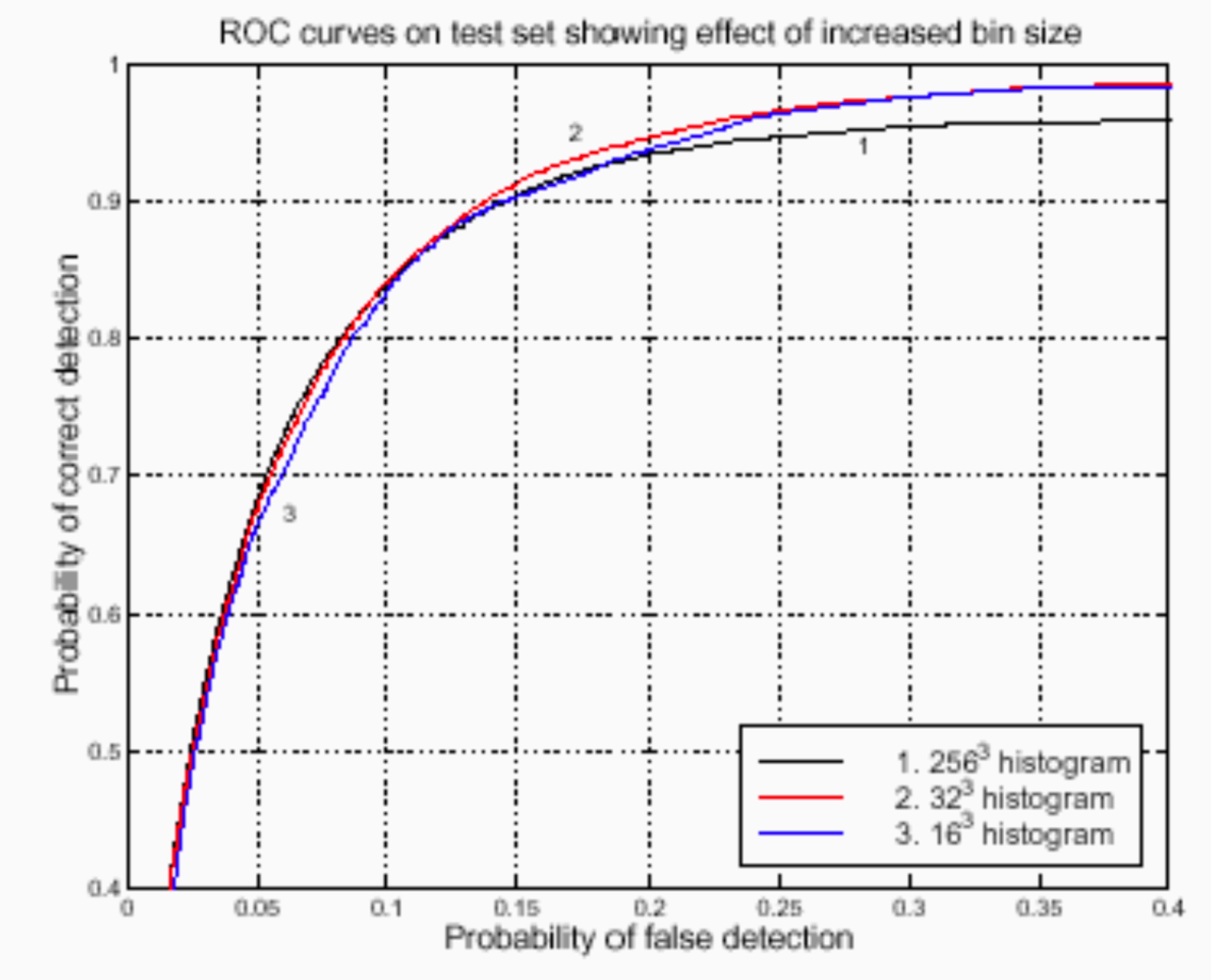
A **confusion matrix** is a table whose (i,j)th entry is the frequency (or proportion) an item of true class i was labelled as j by the classifier.



Forsyth & Ponce (2nd ed.) Figure 15.3. Original credit: H. Zhang et al., 2006.

Receiver Operating Characteristics (ROC)

ROC curves plot trade-off between false positives and false negatives



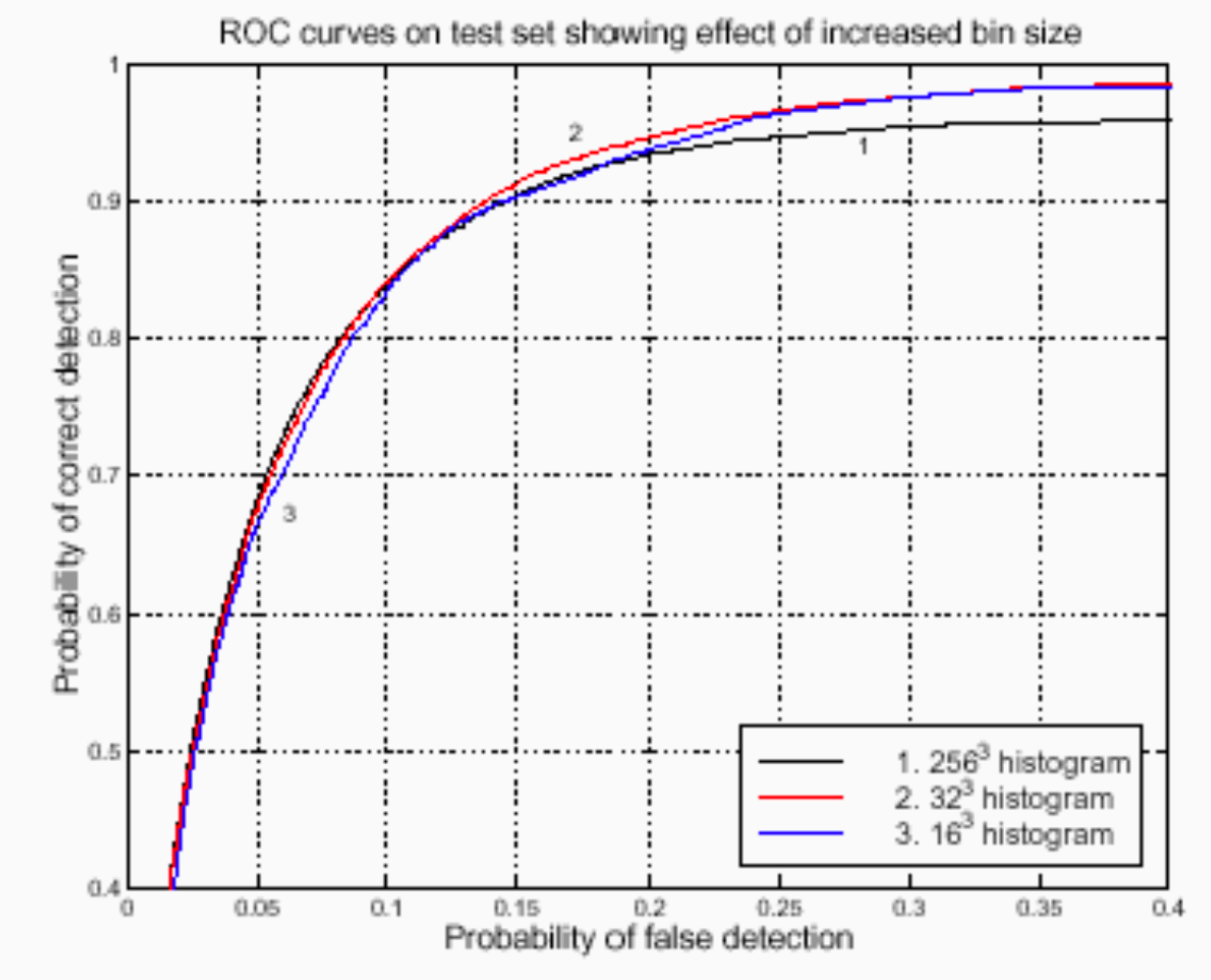
Forsyth & Ponce (2nd ed.) Figure 15.4

Figure from M. J. Jones and J. Rehg, “Statistical color models with application to skin detection,” Proc. CVPR, 1999, IEEE

Receiver Operating Characteristics (ROC)

What is a ROC curve for a perfect classifier?

ROC curves plot trade-off between false positives and false negatives



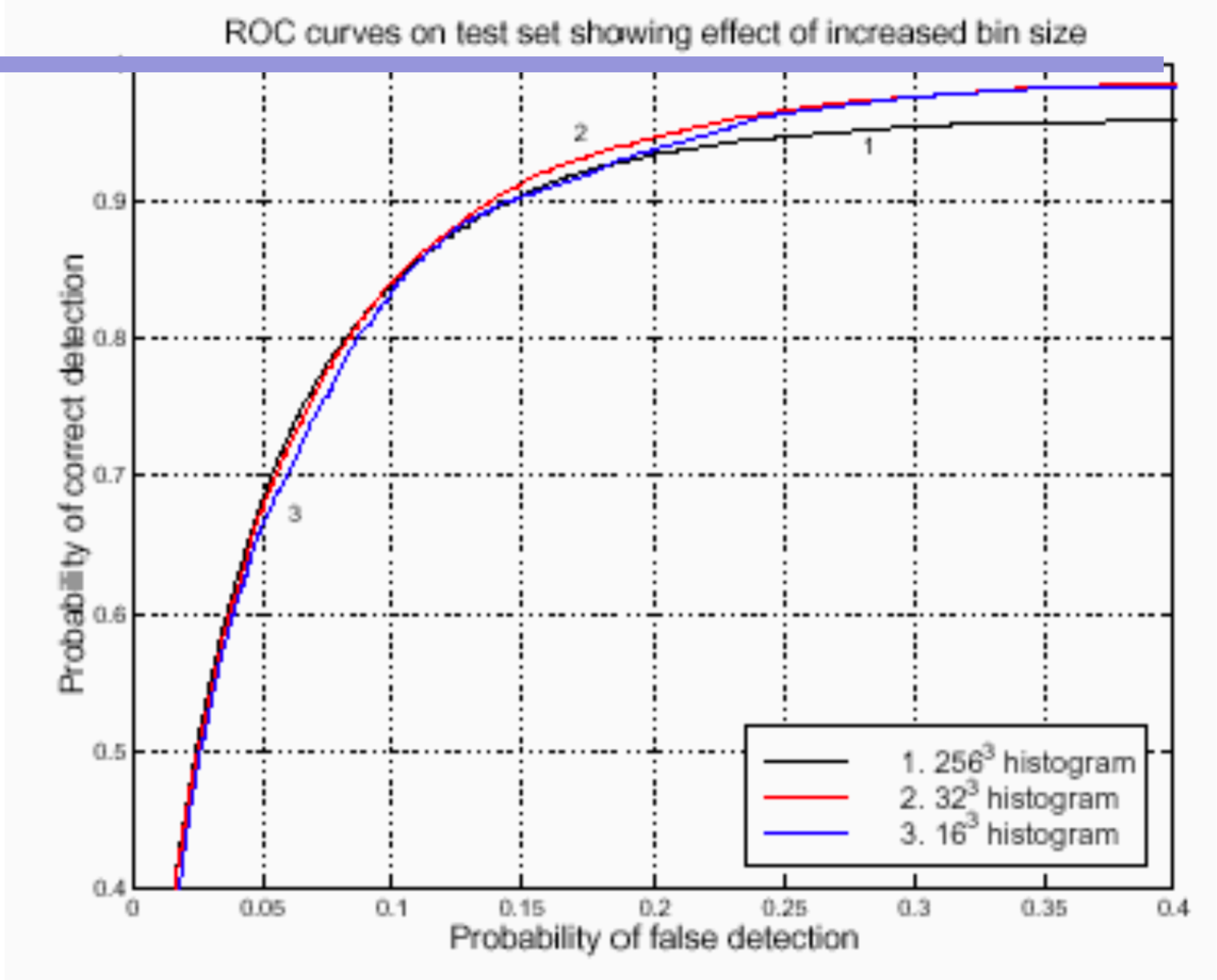
Forsyth & Ponce (2nd ed.) Figure 15.4

Figure from M. J. Jones and J. Rehg, “Statistical color models with application to skin detection,” Proc. CVPR, 1999, IEEE

Receiver Operating Characteristics (ROC)

What is a ROC curve for a perfect classifier?

ROC curves plot trade-off between false positives and false negatives



Forsyth & Ponce (2nd ed.) Figure 15.4

Figure from M. J. Jones and J. Rehg, "Statistical color models with application to skin detection," Proc. CVPR, 1999, IEEE

Classifier Strategies

Classification strategies fall under two broad types: **parametric** and **non-parametric**.

Classifier Strategies

Classification strategies fall under two broad types: **parametric** and **non-parametric**.

Parametric classifiers are **model driven**. The parameters of the model are learned from training examples. New data points are classified by the learned model.

— fast, compact

— flexibility and accuracy depend on model assumptions

Classifier Strategies

Classification strategies fall under two broad types: **parametric** and **non-parametric**.

Parametric classifiers are **model driven**. The parameters of the model are learned from training examples. New data points are classified by the learned model.

— fast, compact

— flexibility and accuracy depend on model assumptions

Non-parametric classifiers are **data driven**. New data points are classified by comparing to the training examples directly. "The data is the model".

— slow

— highly flexible decision boundaries

Nearest Neighbor Classifier

Given a new data point, assign the label of nearest training example in feature space.

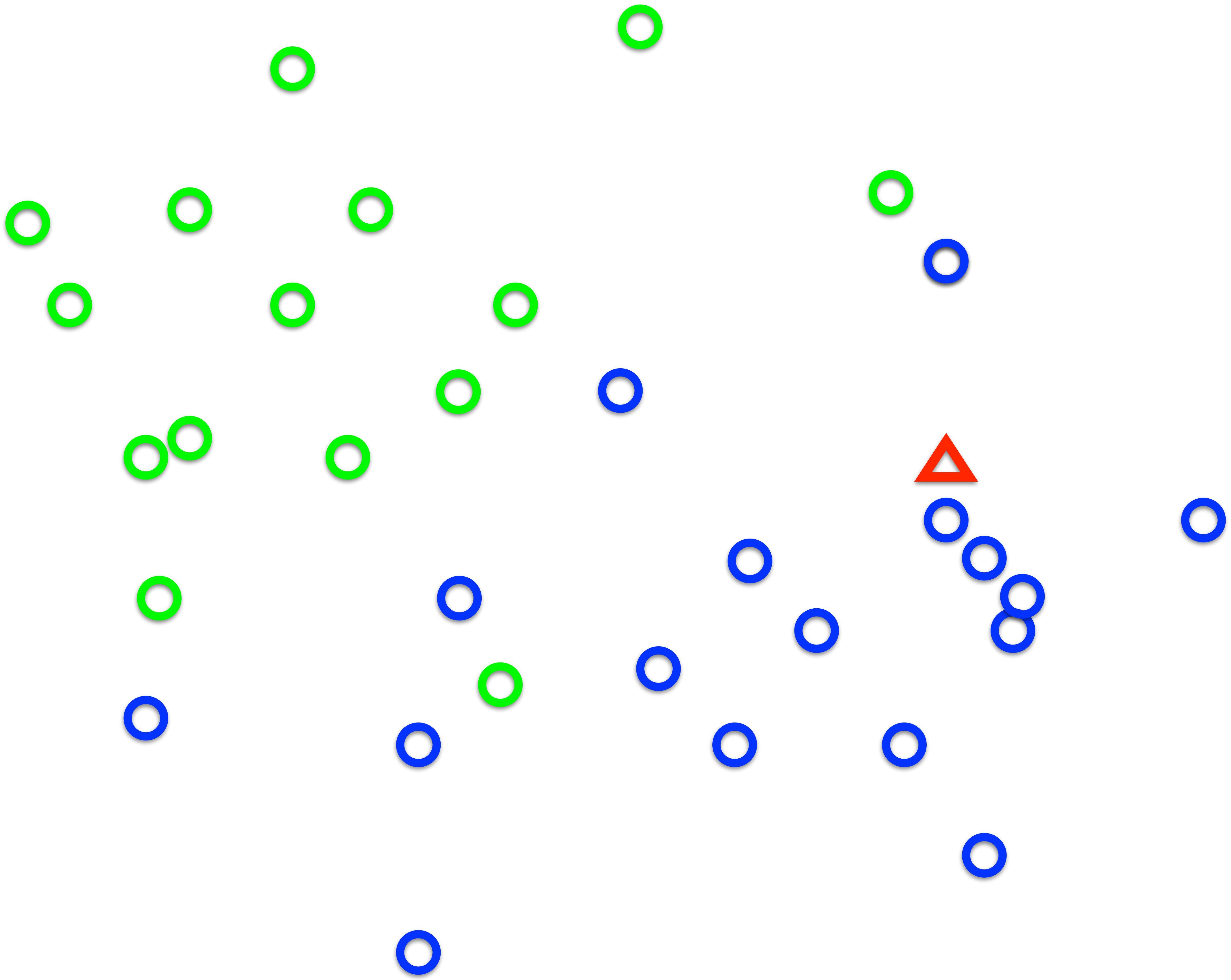


Image Credit: Ioannis (Yannis) Gkioulekas (CMU)

Nearest Neighbor Classifier

Given a new data point, assign the label of nearest training example in feature space.

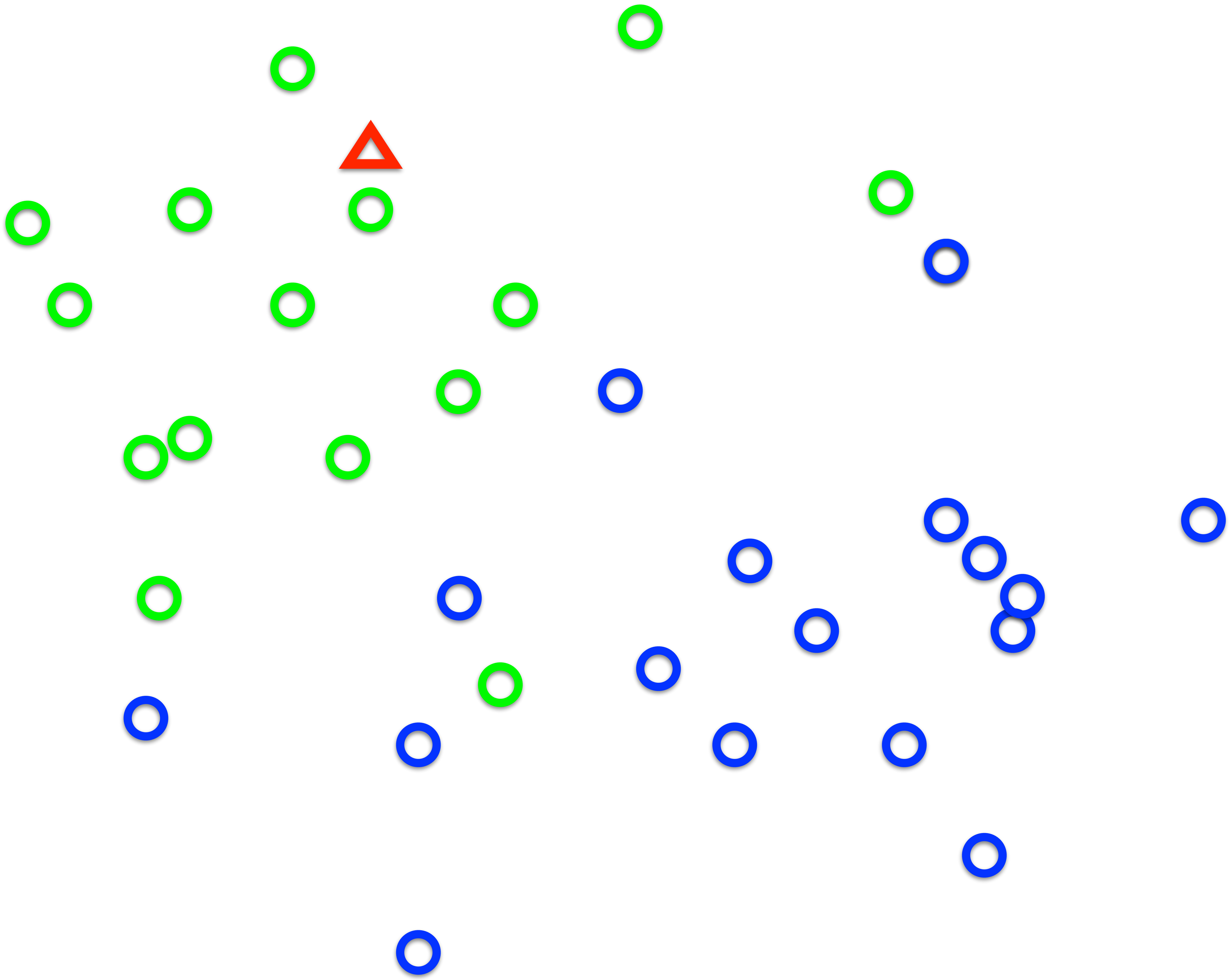


Image Credit: Ioannis (Yannis) Gkioulekas (CMU)

k-**Nearest Neighbor** (kNN) Classifier

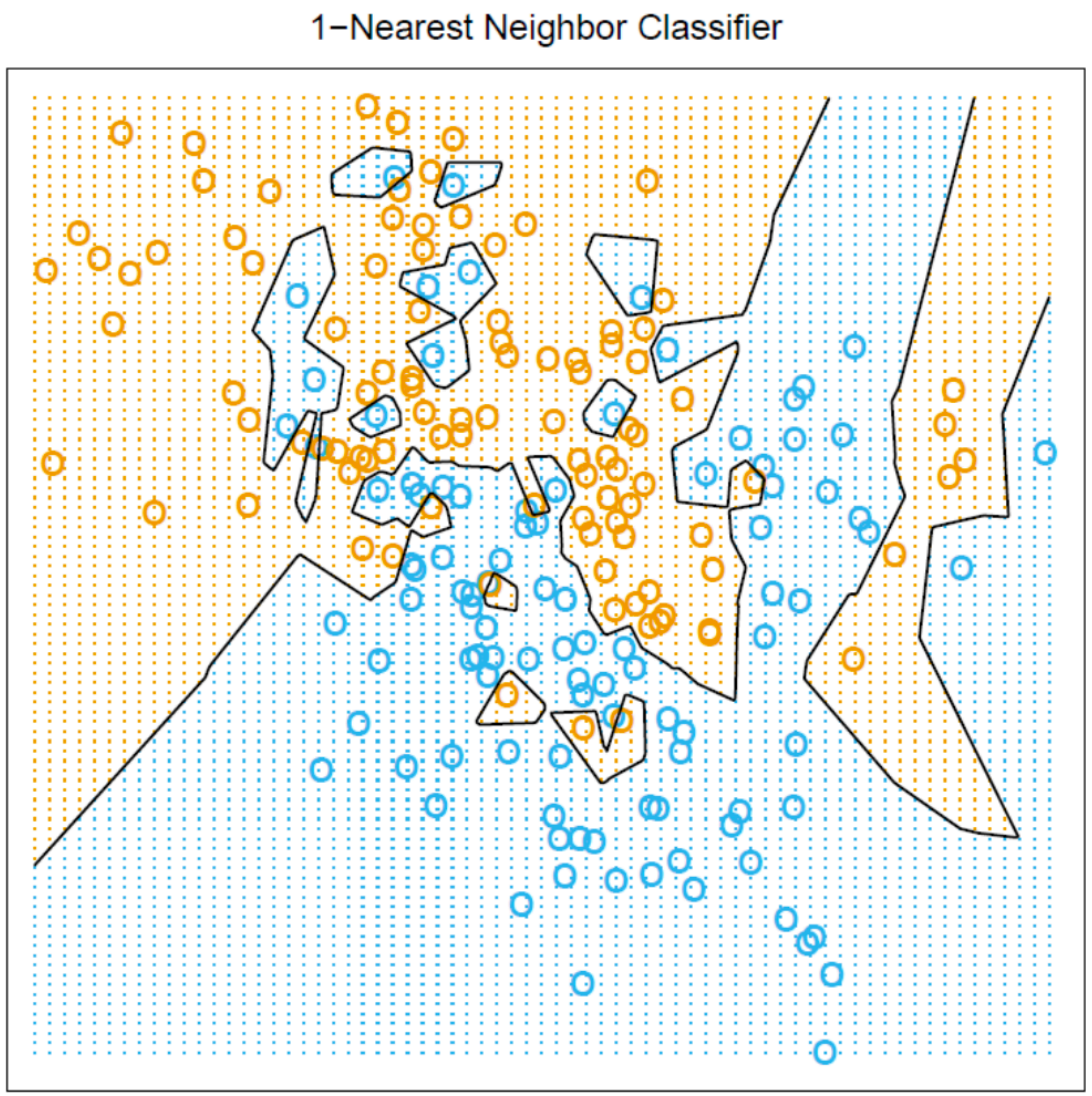
We can gain some robustness to noise by voting over **multiple** neighbours.

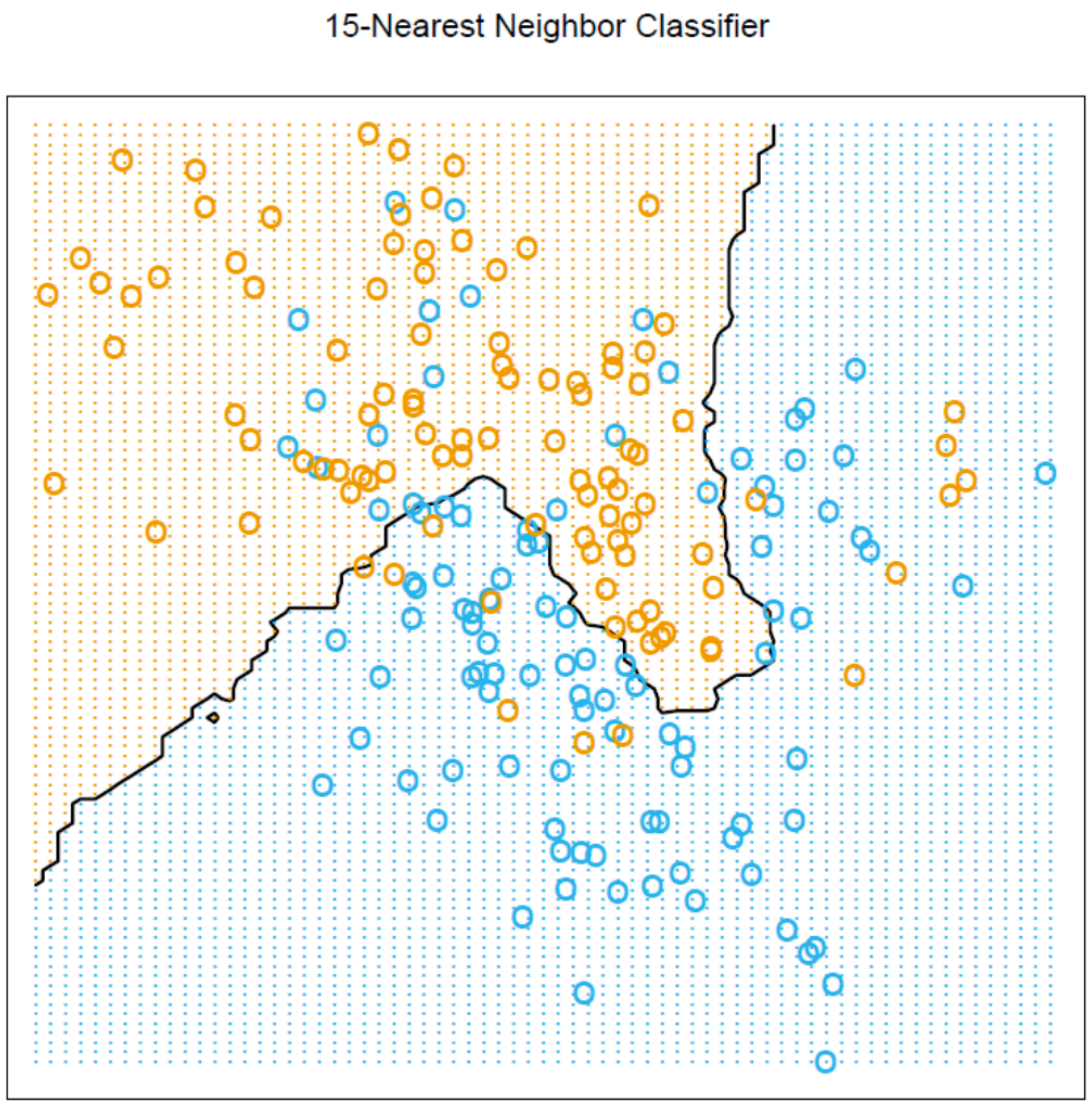
Given a **new** data point, find the k nearest training examples. Assign the label by **majority vote**.

Simple method that works well if the **distance measure** correctly weights the various dimensions

For **large data sets**, as k increases kNN approaches optimality in terms of minimizing probability of error

k-**Nearest Neighbor** (kNN) Classifier





kNN decision boundaries respond to local clusters where one class dominates

Figure credit: Hastie, Tibshirani & Friedman (2nd ed.)