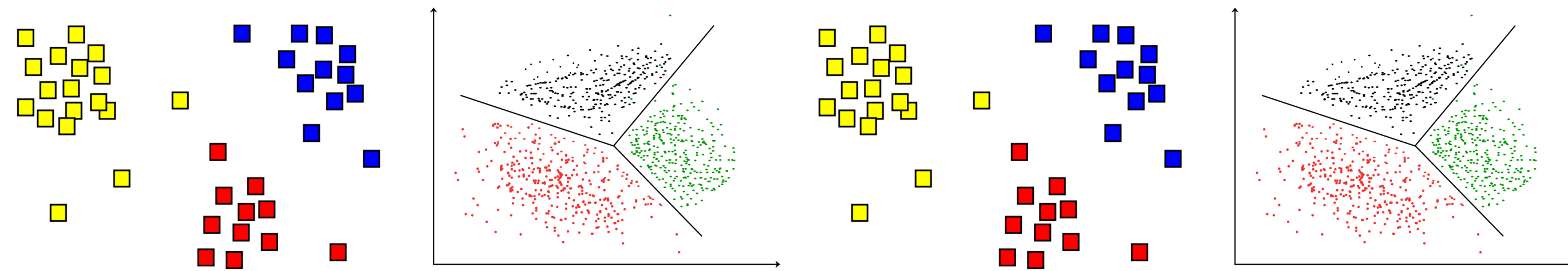




CPSC 425: Computer Vision



Lecture 25: Classification

Menu for Today (November 9, 2020)

Topics:

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

Readings:

- **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 **Wednesday**

Today's “**fun**” Example:

Audio-Visual Scene Analysis with Self-Supervised Multisensory Features

Andrew Owens Alexei A. Efros
UC Berkeley



Today's “**fun**” Example:

Audio-Visual Scene Analysis with Self-Supervised Multisensory Features

Andrew Owens Alexei A. Efros
UC Berkeley



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, ...}



cat

Classification



| | | | | | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 08 | 02 | 22 | 97 | 38 | 15 | 00 | 40 | 00 | 75 | 04 | 05 | 07 | 78 | 52 | 12 | 50 | 77 | 91 | 88 |
| 49 | 49 | 99 | 40 | 17 | 81 | 18 | 57 | 60 | 87 | 17 | 40 | 98 | 43 | 69 | 48 | 04 | 56 | 62 | 00 |
| 81 | 49 | 31 | 73 | 55 | 79 | 14 | 29 | 93 | 71 | 40 | 67 | 52 | 88 | 30 | 03 | 49 | 13 | 36 | 65 |
| 52 | 70 | 95 | 23 | 04 | 60 | 11 | 42 | 69 | 21 | 68 | 56 | 01 | 32 | 56 | 71 | 37 | 02 | 36 | 91 |
| 22 | 31 | 16 | 71 | 51 | 67 | 03 | 59 | 41 | 92 | 36 | 54 | 22 | 40 | 40 | 28 | 66 | 33 | 13 | 80 |
| 24 | 47 | 39 | 80 | 99 | 03 | 45 | 02 | 44 | 75 | 33 | 53 | 78 | 36 | 84 | 20 | 35 | 17 | 12 | 50 |
| 32 | 98 | 81 | 28 | 64 | 23 | 67 | 10 | 26 | 38 | 40 | 67 | 59 | 54 | 70 | 66 | 18 | 38 | 64 | 70 |
| 67 | 26 | 20 | 68 | 02 | 62 | 12 | 20 | 95 | 63 | 94 | 39 | 63 | 08 | 40 | 91 | 66 | 49 | 94 | 21 |
| 24 | 55 | 58 | 05 | 66 | 73 | 99 | 26 | 97 | 17 | 78 | 78 | 96 | 83 | 14 | 88 | 34 | 89 | 63 | 72 |
| 21 | 36 | 23 | 09 | 75 | 00 | 76 | 44 | 20 | 45 | 35 | 14 | 00 | 61 | 33 | 97 | 34 | 31 | 33 | 95 |
| 78 | 17 | 53 | 28 | 22 | 75 | 31 | 67 | 15 | 94 | 03 | 80 | 04 | 62 | 16 | 14 | 09 | 53 | 56 | 92 |
| 16 | 39 | 05 | 42 | 96 | 35 | 31 | 47 | 55 | 58 | 88 | 24 | 00 | 17 | 54 | 24 | 36 | 29 | 85 | 57 |
| 86 | 56 | 00 | 48 | 35 | 71 | 89 | 07 | 05 | 44 | 44 | 37 | 44 | 60 | 21 | 58 | 51 | 54 | 17 | 58 |
| 19 | 80 | 81 | 68 | 05 | 94 | 47 | 69 | 28 | 73 | 92 | 13 | 86 | 52 | 17 | 77 | 04 | 89 | 55 | 40 |
| 04 | 52 | 08 | 83 | 97 | 35 | 99 | 16 | 07 | 97 | 57 | 32 | 16 | 26 | 26 | 79 | 33 | 27 | 98 | 66 |
| 89 | 36 | 68 | 87 | 57 | 62 | 20 | 72 | 03 | 46 | 33 | 67 | 46 | 55 | 12 | 32 | 63 | 93 | 53 | 69 |
| 04 | 42 | 16 | 73 | 35 | 35 | 39 | 11 | 24 | 94 | 72 | 18 | 08 | 46 | 29 | 32 | 40 | 62 | 76 | 36 |
| 20 | 69 | 36 | 41 | 72 | 30 | 23 | 88 | 34 | 83 | 99 | 69 | 82 | 67 | 59 | 85 | 74 | 04 | 36 | 16 |
| 20 | 73 | 35 | 29 | 78 | 31 | 90 | 01 | 74 | 31 | 49 | 71 | 48 | 86 | 81 | 16 | 23 | 57 | 05 | 54 |
| 01 | 70 | 54 | 71 | 83 | 51 | 54 | 69 | 16 | 92 | 33 | 48 | 61 | 43 | 52 | 01 | 89 | 13 | 67 | 48 |

What the computer sees

image classification

- 82% cat
- 15% dog
- 2% hat
- 1% mug

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: $[1, 0, 0, 0, \dots]$ (one-hot)

Classification

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

Example training set

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

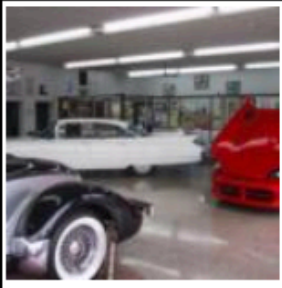
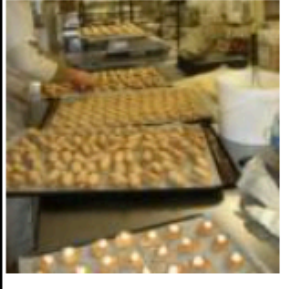

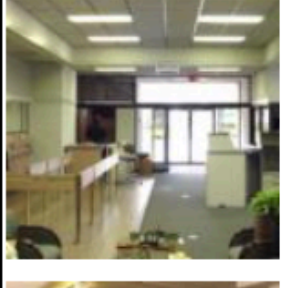
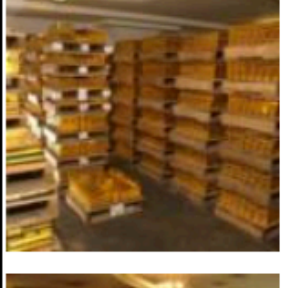





Figure credit: Duda & Hart

Example 2: Real Classification Problem

SUN Dataset

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

| | | | |
|---------------------|---|---|----------------|
| indoor | shopping and dining |  | auto showroom |
| outdoor natural | workplace (office building, factory, lab, etc.) |  | bakery kitchen |
| outdoor man-made | home or hotel |  | bakery shop |
| | transportation (vehicle interiors, stations, etc.) |  | bank indoor |
| | sports and leisure |  | bank vault |
| | cultural (art, education, religion, military, law, politics, etc.) |  | banquet hall |
| | |  | bar |
| | |  | |

Example 3: Real Classification Problem

ImageNet Dataset

- 14 Million images
- 21K **object** categories

Natural object

An object occurring naturally; not made by man

0 pictures
82.76% Popularity Percentile
Wordnet IDs

Numbers in brackets: (the number of synsets in the subtree).

- ImageNet 2011 Fall Release (32326)
 - plant, flora, plant life (4486)
 - geological formation, formation (1)
 - aquifer (0)
 - beach (1)
 - cave (3)
 - cliff, drop, drop-off (2)
 - delta (0)
 - diapir (0)
 - folium (0)
 - foreshore (0)
 - ice mass (10)
 - lakefront (0)
 - massif (0)
 - monocline (0)
 - mouth (0)
 - natural depression, depression (0)
 - natural elevation, elevation (41)
 - oceanfront (0)
 - range, mountain range, range of mountains (0)
 - relict (0)
 - ridge, ridgeline (2)
 - ridge (0)
 - shore (7)
 - slope, incline, side (17)
 - spring, fountain, outflow, outpouring, waterfall (0)
 - talus, scree (0)
 - vein, mineral vein (1)
 - volcanic crater, crater (2)
 - wall (0)
 - water table, water level, ground

Treemap Visualization Images of the Synset Downloads

ImageNet 2011 Fall Release > Natural object

Plant

Covering

Sample Extraterrestrial Body

Asterism Mechanism Celestial

Radiator Body Rock

Tangle Nest

Example 3: Real Classification Problem

ImageNet Dataset

- 14 Million images
- 21K **object** categories

Natural object

An object occurring naturally; not made by man

0 pictures
82.76% Popularity Percentile
Wordnet IDs

Numbers in brackets: (the number of synsets in the subtree).

- ImageNet 2011 Fall Release (32326)
 - plant, flora, plant life (4486)
 - geological formation, formation (1)
 - aquifer (0)
 - beach (1)
 - cave (3)
 - cliff, drop, drop-off (2)
 - delta (0)
 - diapir (0)
 - folium (0)
 - foreshore (0)
 - ice mass (10)
 - lakefront (0)
 - massif (0)
 - monocline (0)
 - mouth (0)
 - natural depression, depression (0)
 - natural elevation, elevation (41)
 - oceanfront (0)
 - range, mountain range, range of mountains (0)
 - relict (0)
 - ridge, ridgeline (2)
 - ridge (0)
 - shore (7)
 - slope, incline, side (17)
 - spring, fountain, outflow, outpouring, talus, scree (0)
 - vein, mineral vein (1)
 - volcanic crater, crater (2)
 - wall (0)
 - water table, water level, ground

Treemap Visualization Images of the Synset Downloads

ImageNet 2011 Fall Release > Natural object

Plant

Covering

Sample Extraterrestrial Body

Asterism Mechanism Celestial

Radiator Body Rock

Tangle Nest

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)

The diagram illustrates the Bayes Rule equation with color-coded labels for each component:

- class-conditional probability (a.k.a. likelihood)**: $P(x|c)$ (blue box)
- prior probability**: $p(c)$ (green box)
- posterior probability**: $P(c|x)$ (purple box)
- unconditional probability (a.k.a. marginal likelihood)**: $P(x)$ (cyan box)

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

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., $x \in \mathbb{R}$

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

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 \mathbf{x} as

1 if $p(1|\mathbf{x}) > p(2|\mathbf{x})$

2 if $p(1|\mathbf{x}) < p(2|\mathbf{x})$

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., $c \in \{1, \dots, 1000\}$
- features are high-dimensional; i.e., $x \in \mathbb{R}^{2,000+}$

Example: Discrete Bayes Classifier

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

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

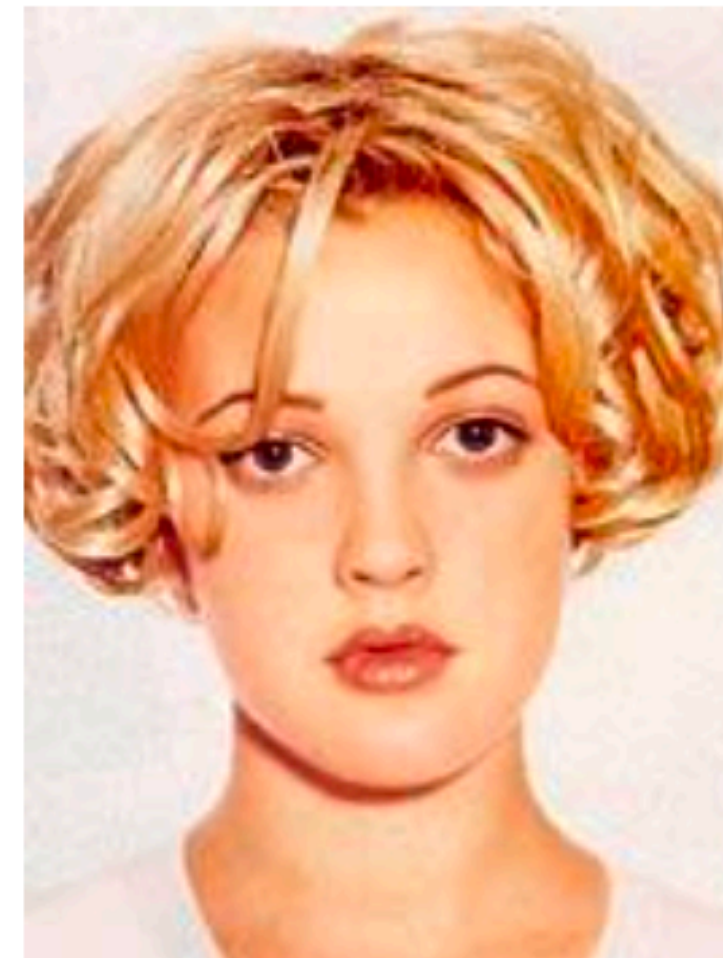
Example: Discrete Bayes Classifier

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

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



Drew Carey



Drew Barrymore

Example: Discrete Bayes Classifier

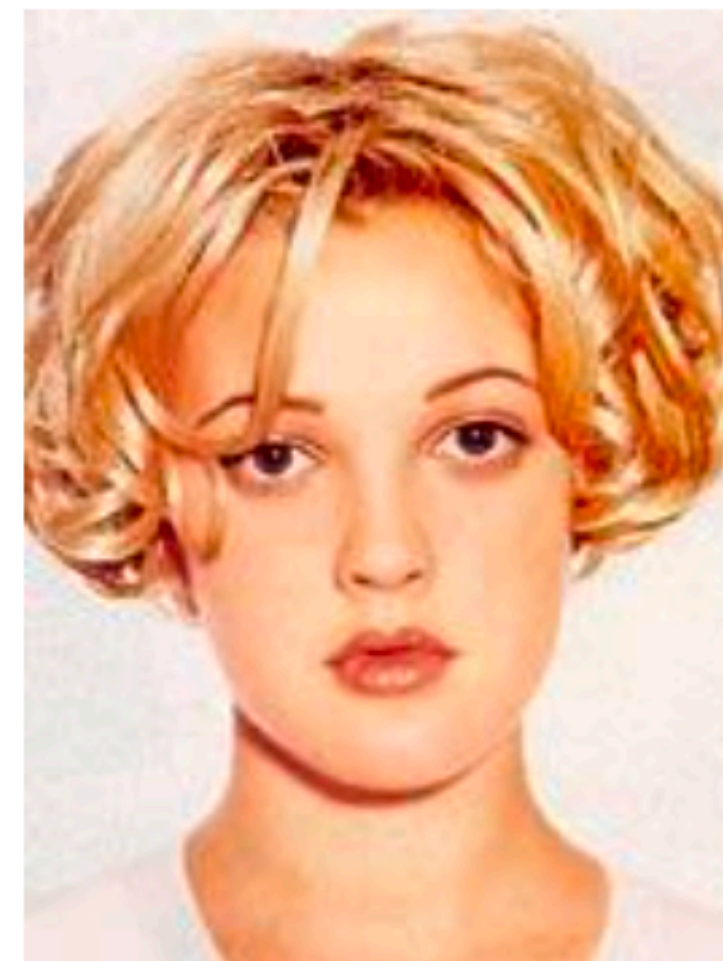
Assume we have two classes: $c_1 = \mathbf{male}$ $c_2 = \mathbf{female}$

We have a person whose gender we don't know, whose 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 $p(\mathbf{male}|drew)$
 $p(\mathbf{female}|drew)$



Drew Carey



Drew Barrymore

Example: Discrete Bayes Classifier

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

We have a person whose gender we don't know, whose 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 $p(\mathbf{male}|drew)$
 $p(\mathbf{female}|drew)$

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

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: Discrete Bayes Classifier

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

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

$$p(drew) =$$

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

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

Example: Discrete Bayes Classifier

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

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

$$p(drew) =$$

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

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

Example: Discrete Bayes Classifier

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

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

$$p(drew) =$$

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

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

Example: Discrete Bayes Classifier

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

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

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

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

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

Example: Discrete Bayes Classifier

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

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

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

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

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

Example: Discrete Bayes Classifier

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

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

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

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

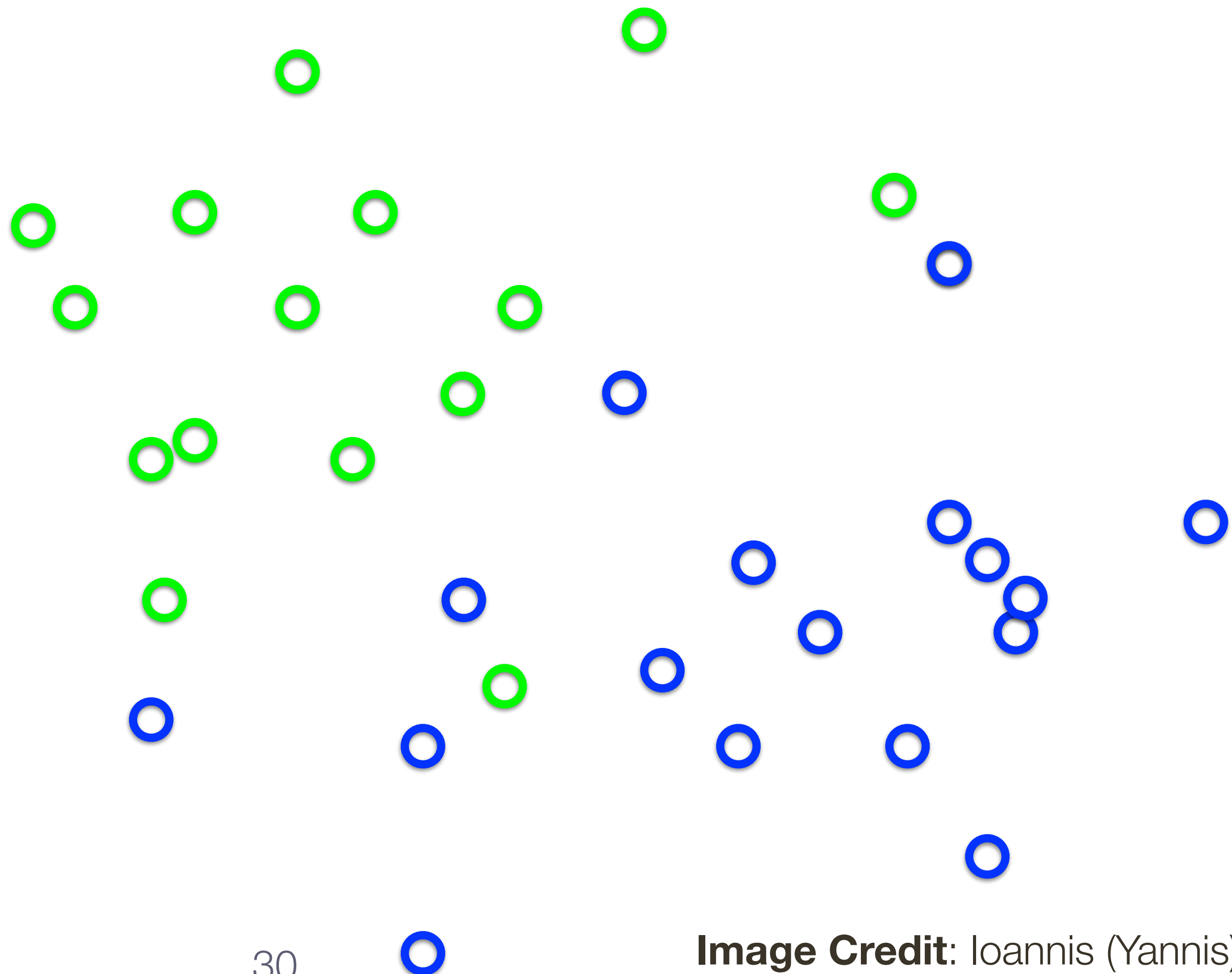
$$p(\mathbf{male}|drew) = \frac{p(drew|\mathbf{male})p(\mathbf{male})}{p(drew)} = 0.125$$

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

Example: 2D Bayes Classifier

○ 17 samples

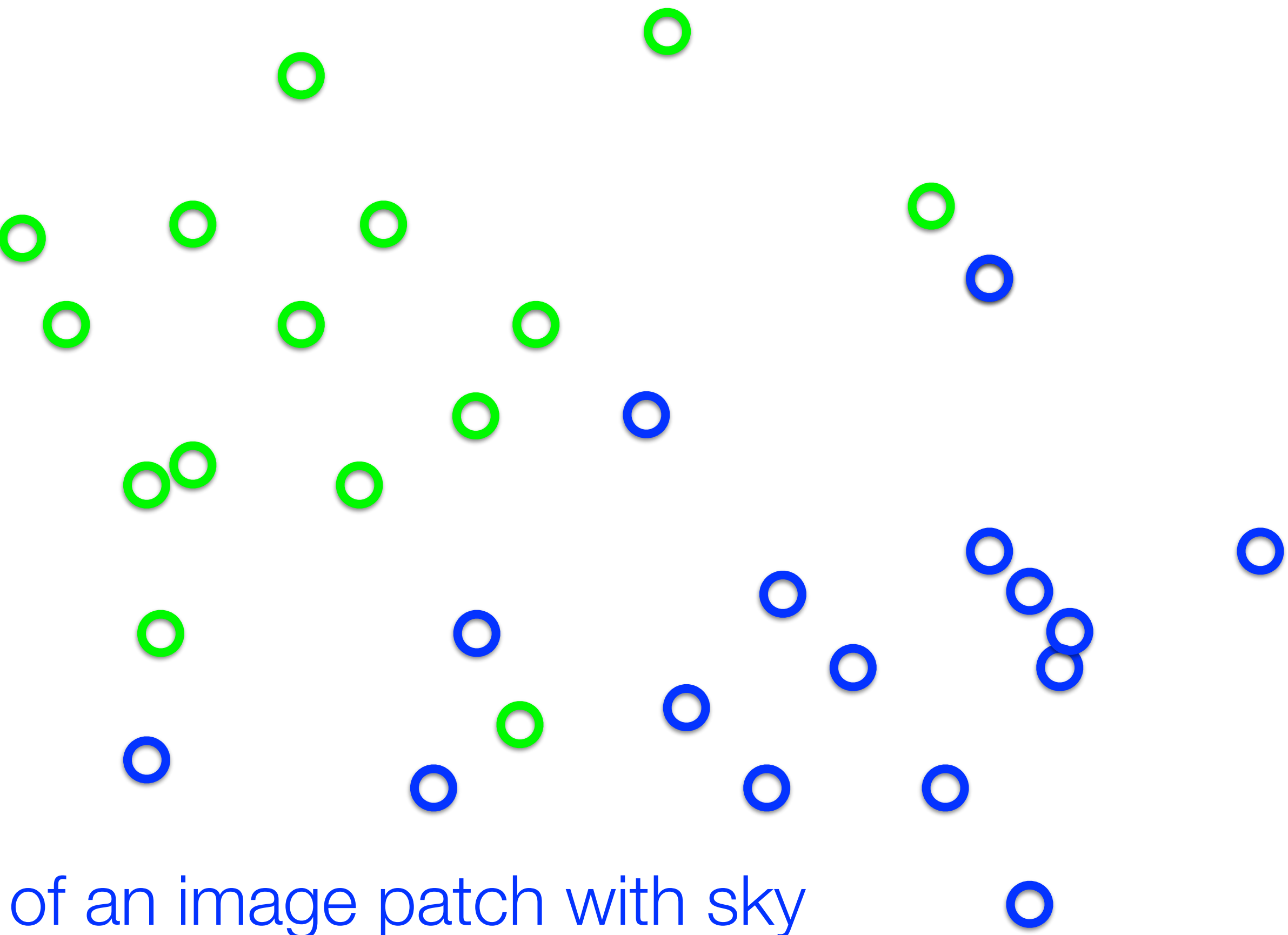
○ 15 samples



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 grass or sky?

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

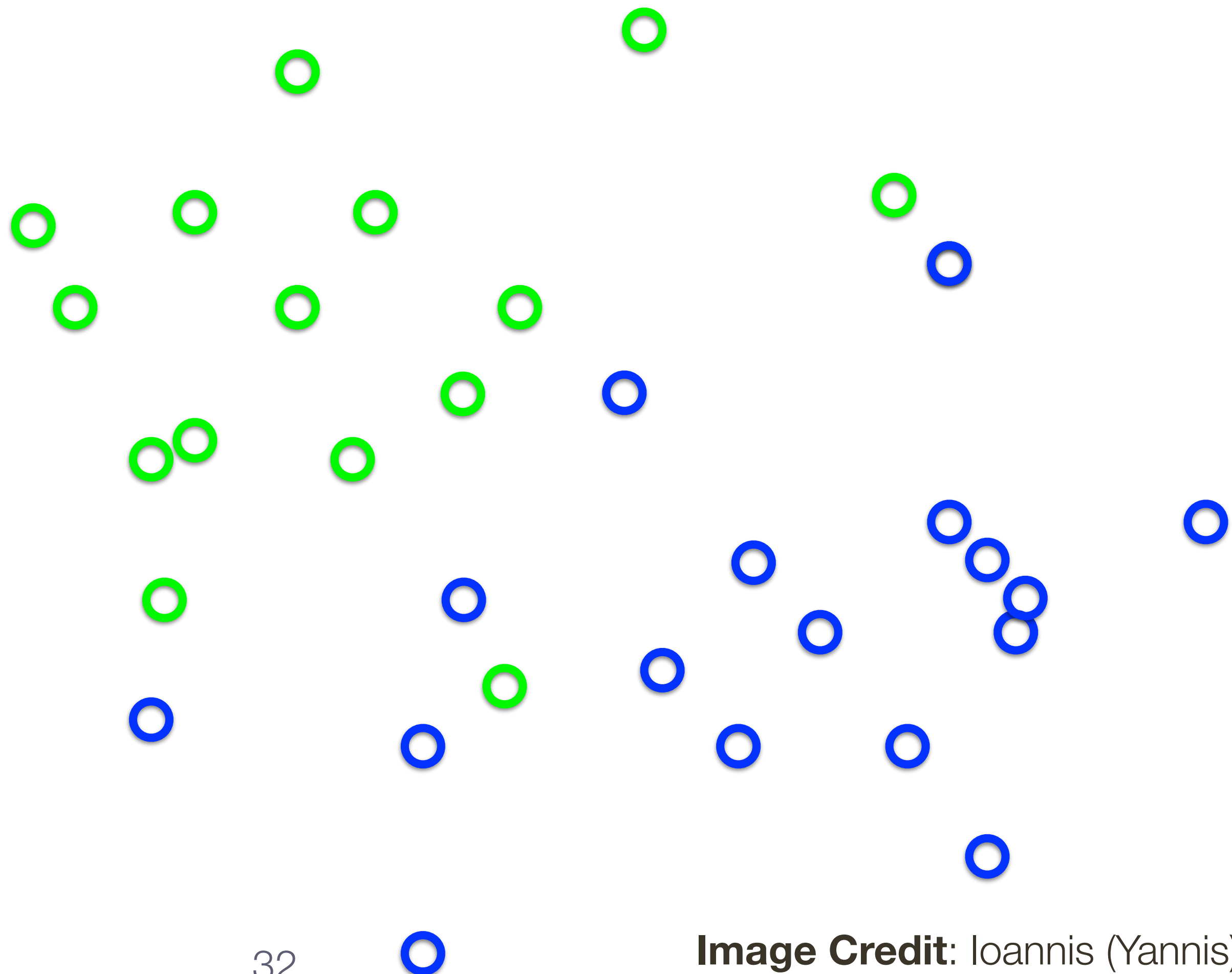
Example: 2D Bayes Classifier

○ 17 samples

○ 15 samples

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

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

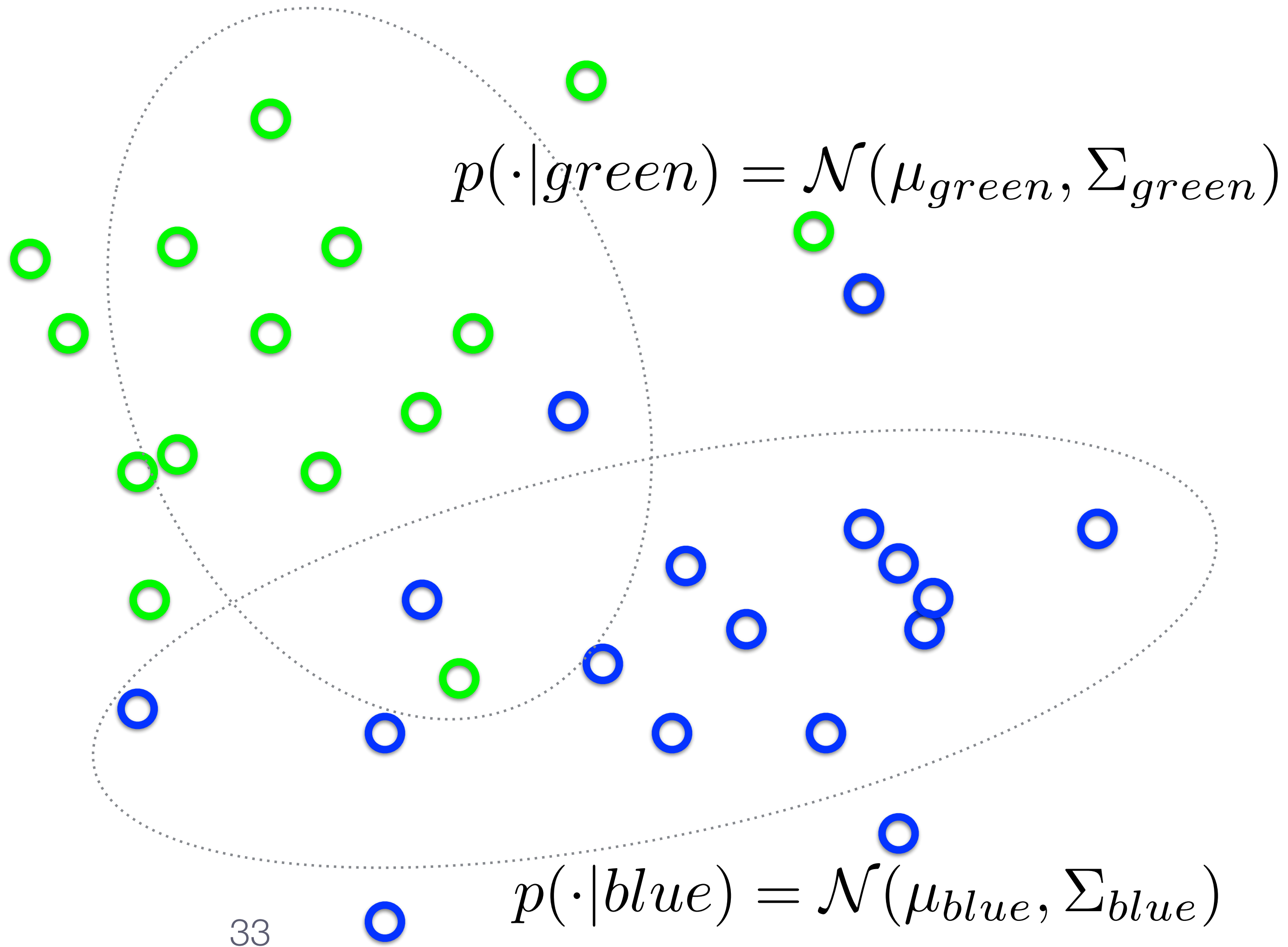


Example: 2D Bayes Classifier

- 17 samples
- 15 samples

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

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



Example: 2D Bayes Classifier

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

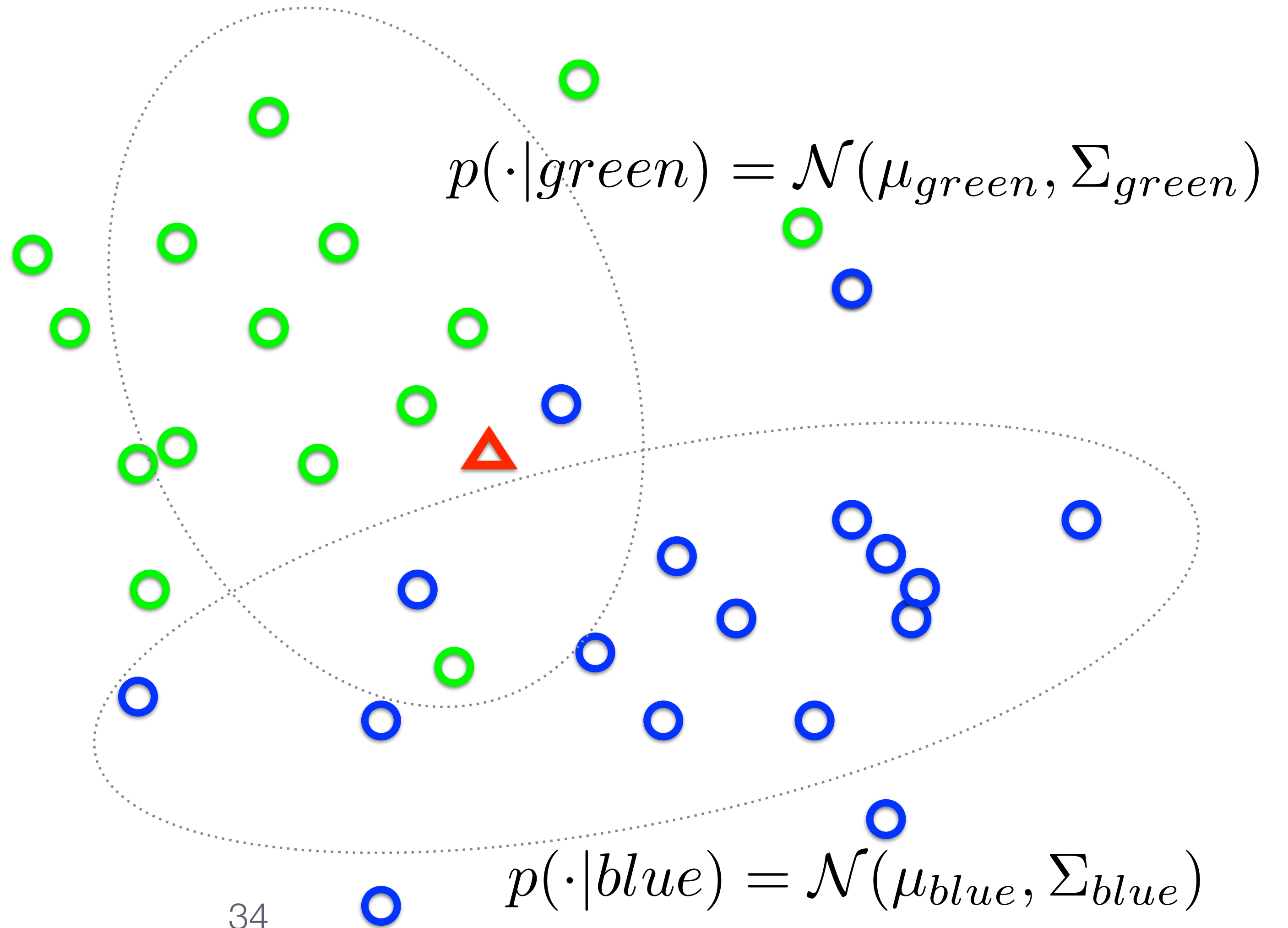
$$p(\text{blue}|\triangle) \propto \mathcal{N}(\triangle; \mu_{\text{blue}}, \Sigma_{\text{blue}})p(\text{blue})$$

○ 17 samples

○ 15 samples

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

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



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., $x \in \mathbb{R}$

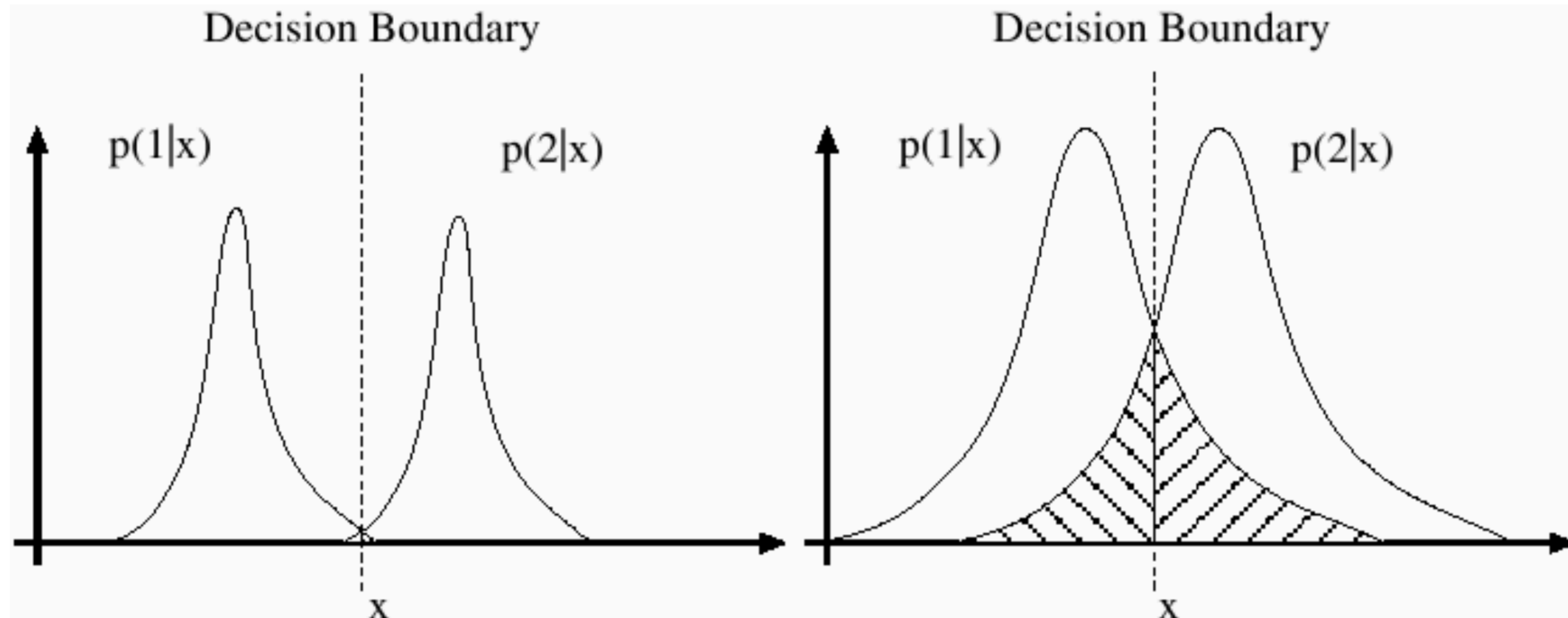
$$P(c|x) = \frac{P(x|c)p(c)}{P(x)}$$

General case:

- multi-class; i.e., $c \in \{1, \dots, 1000\}$
- features are high-dimensional; i.e., $x \in \mathbb{R}^{2,000+}$

Bayes' Risk

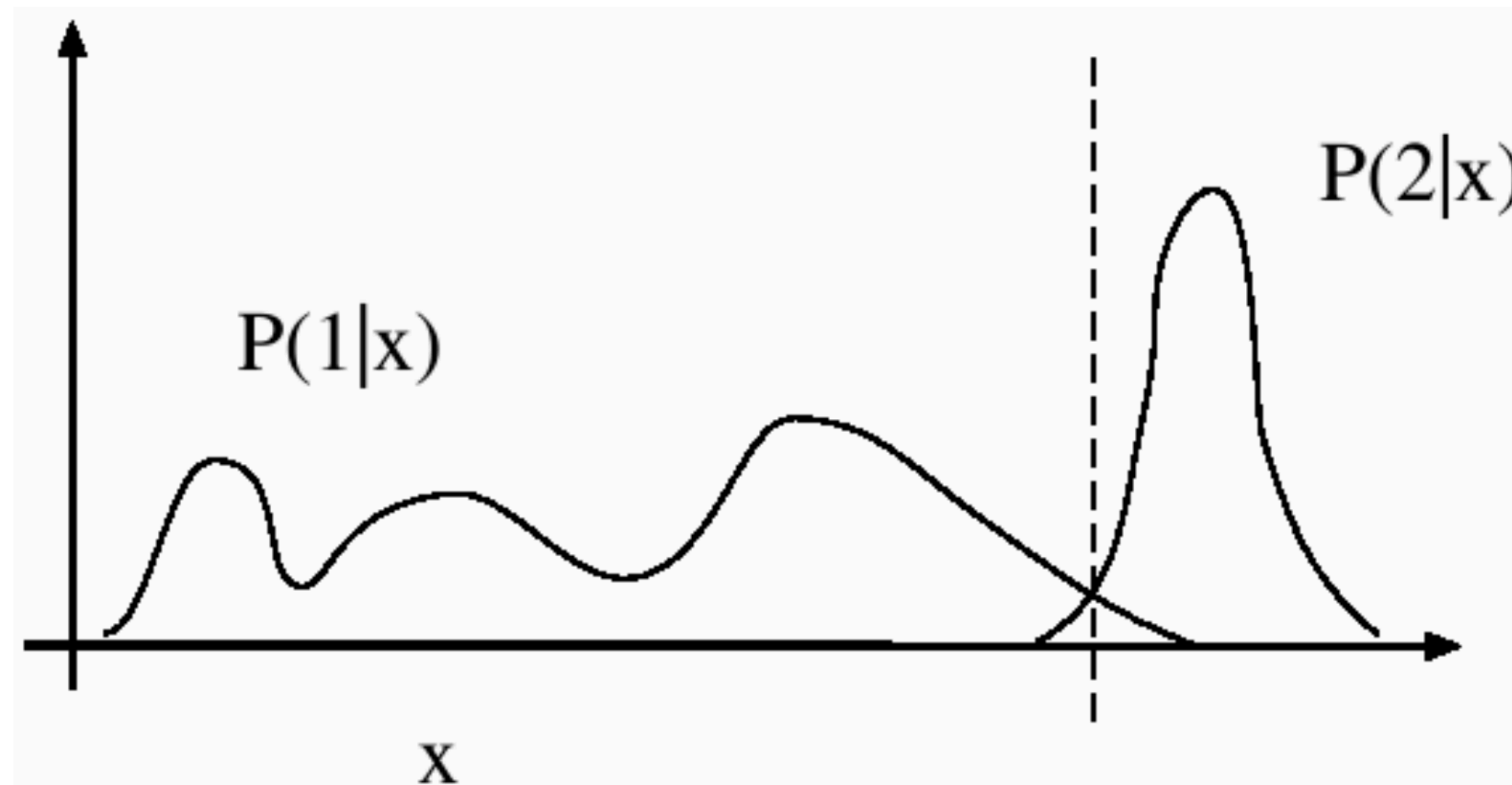
Some errors may be inevitable: the minimum risk (shaded area) is called the **Bayes' risk**



Forsyth & Ponce (2nd ed.) Figure 15.1

Discriminative vs. Generative

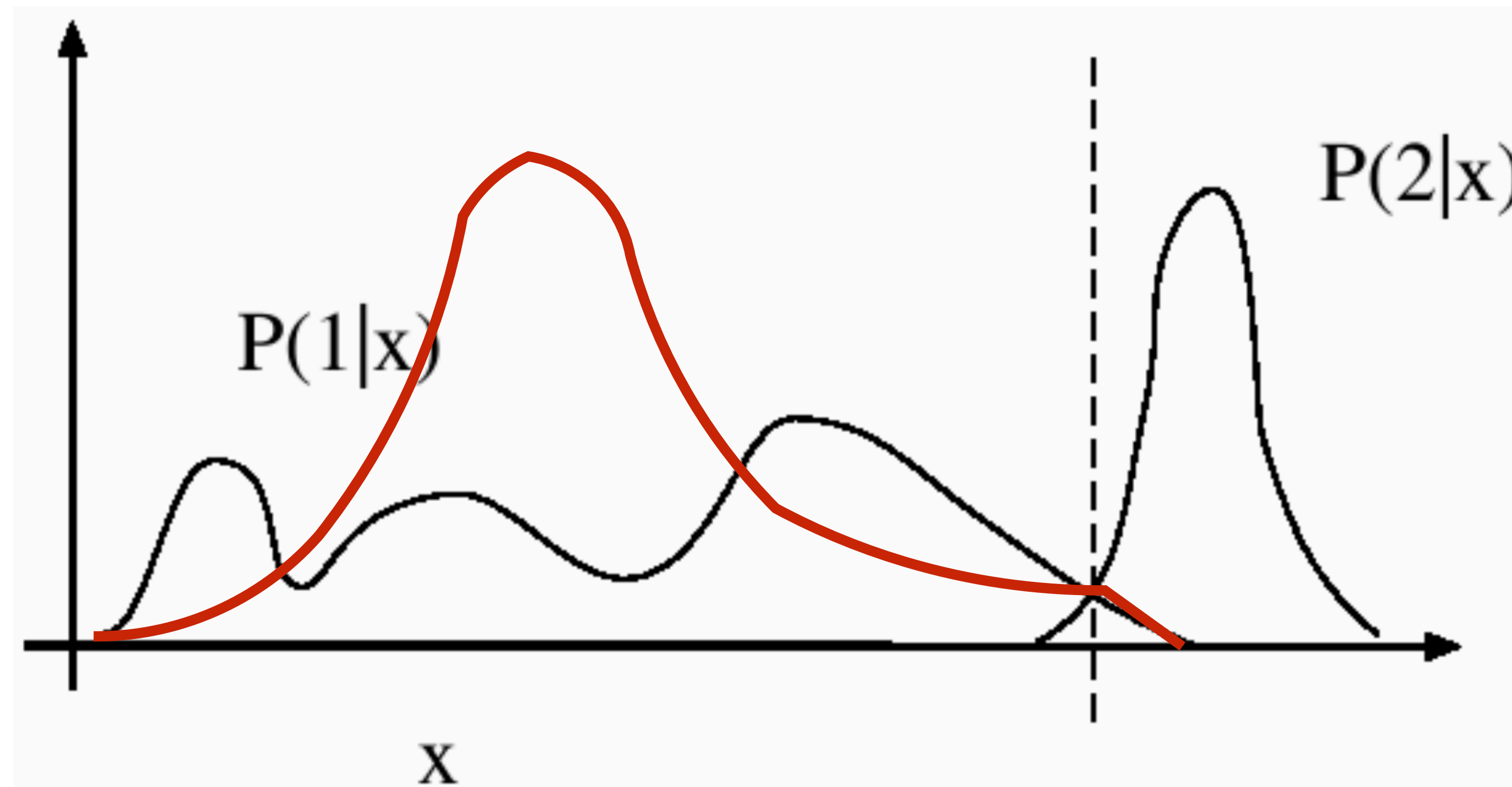
Finding a **decision boundary** is not the same as modeling a **conditional density** — while a normal density here is a poor fit to $P(1|x)$, the quality of the classifier depends only on how well the boundary is positioned



Forsyth & Ponce (2nd ed.) Figure 15.5

Discriminative vs. Generative

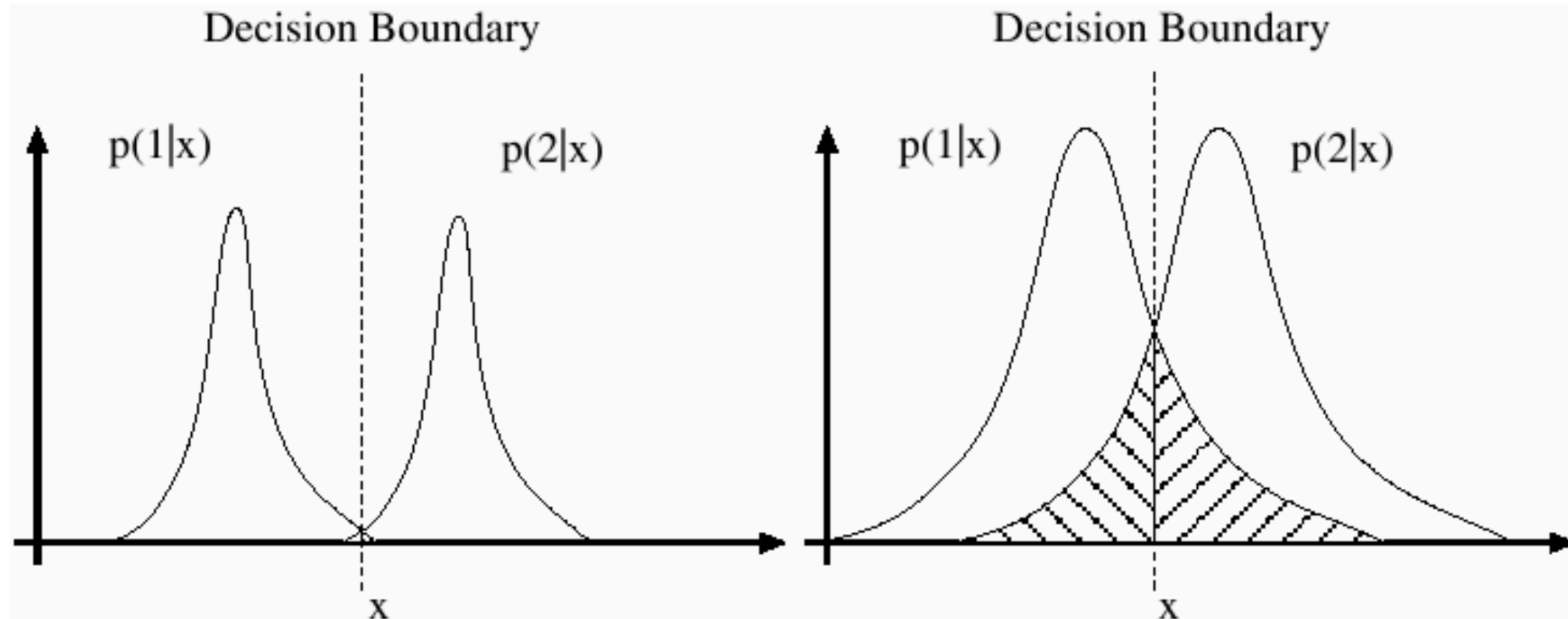
Finding a **decision boundary** is not the same as modeling a **conditional density** — while a normal density here is a poor fit to $P(1|x)$, the quality of the classifier depends only on how well the boundary is positioned



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 \mathbf{s} is

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

Two Class Classification

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

Classify \mathbf{x} as

$$1 \text{ if } p(1|\mathbf{x}) L(1 \rightarrow 2) > p(2|\mathbf{x}) L(2 \rightarrow 1)$$

$$2 \text{ if } p(1|\mathbf{x}) L(1 \rightarrow 2) < p(2|\mathbf{x}) L(2 \rightarrow 1)$$

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

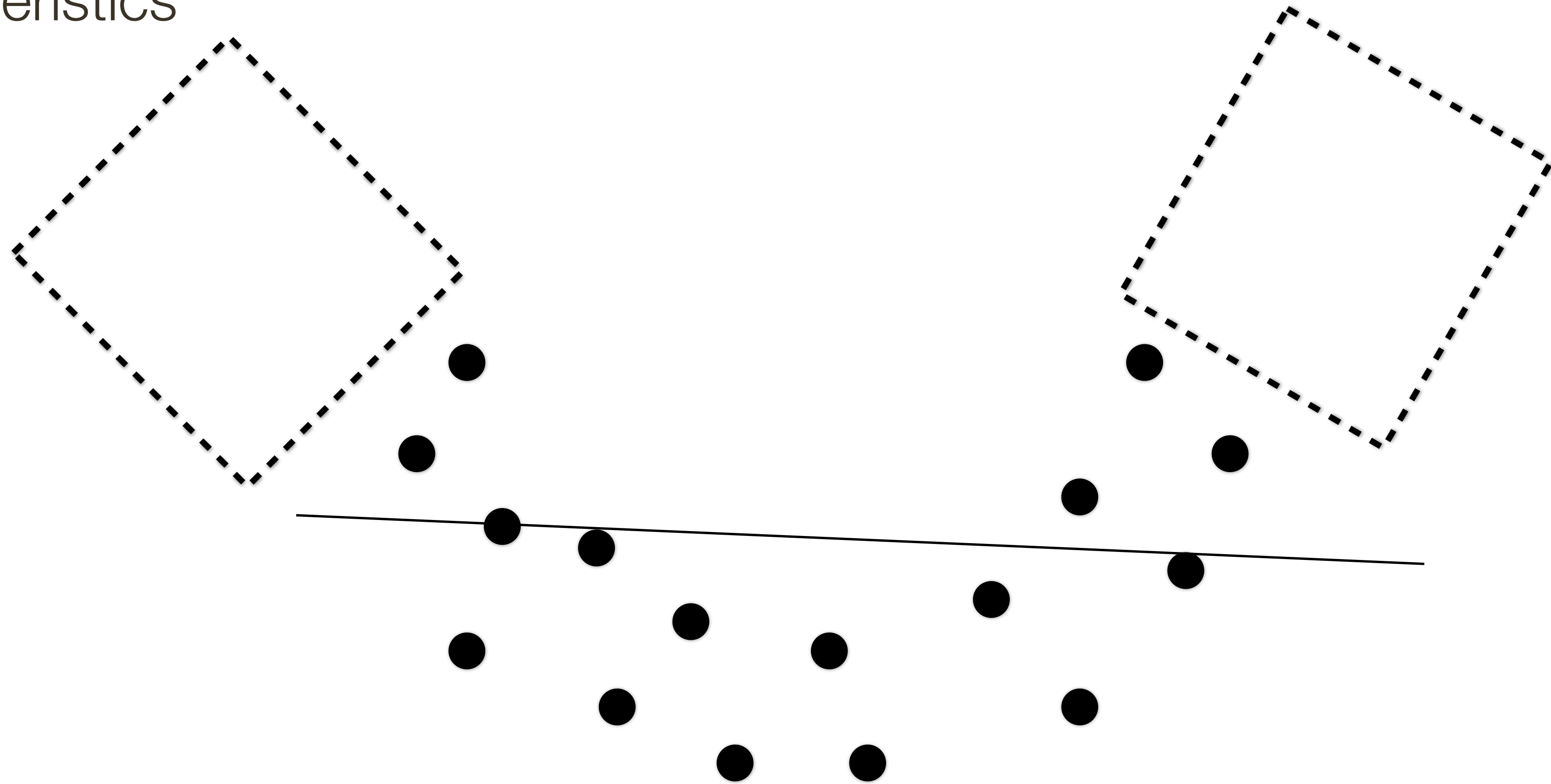
We want to minimize the **testing error** – the error the classifier makes on an unseen testing set

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**

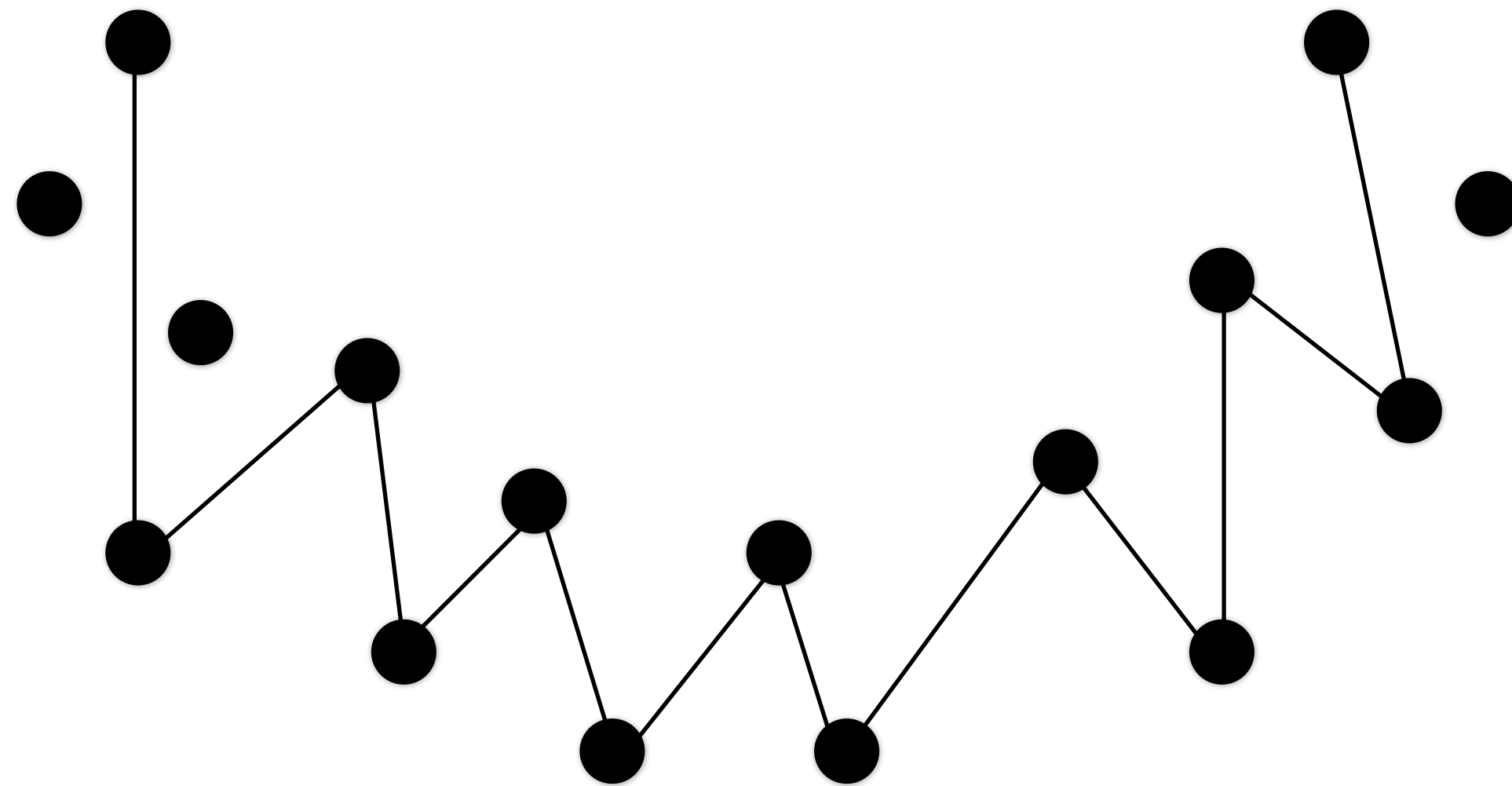
Underfitting: model is too simple to represent all the relevant class characteristics



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

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

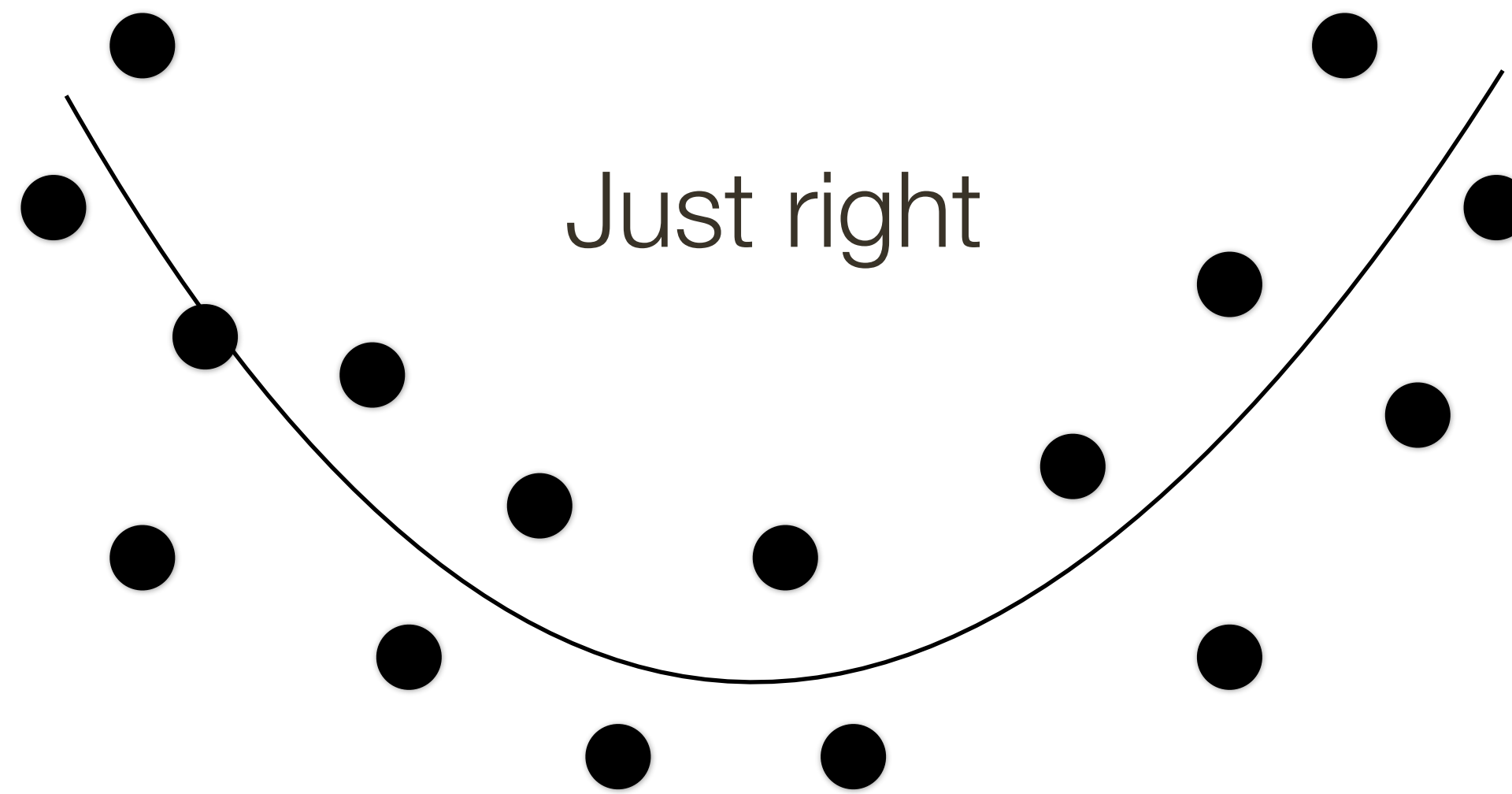
Overfitting: model is too complex and fits irrelevant characteristics (noise) in the data



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

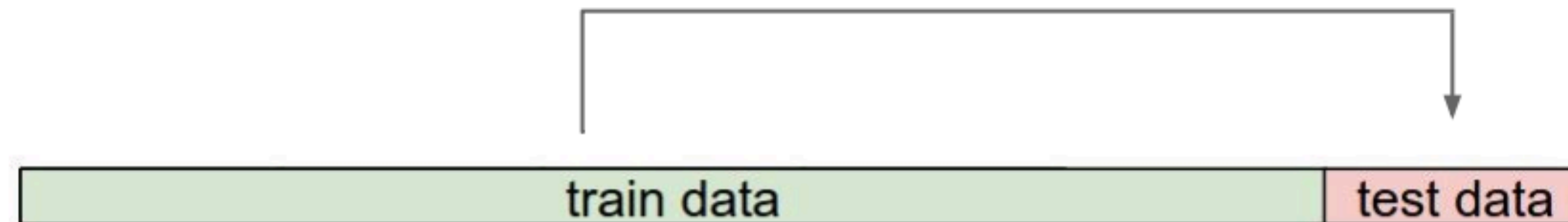


Cross-Validation

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

Try out what hyperparameters work best on test set.



Cross-Validation

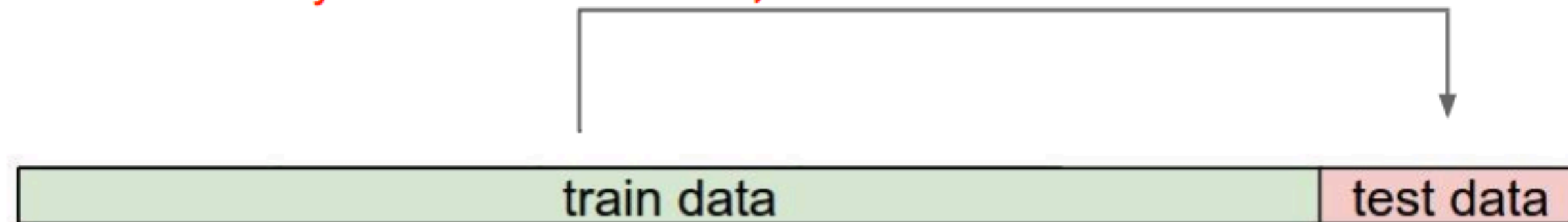
We cannot reliably estimate the error rate of the classifier using the training set

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!

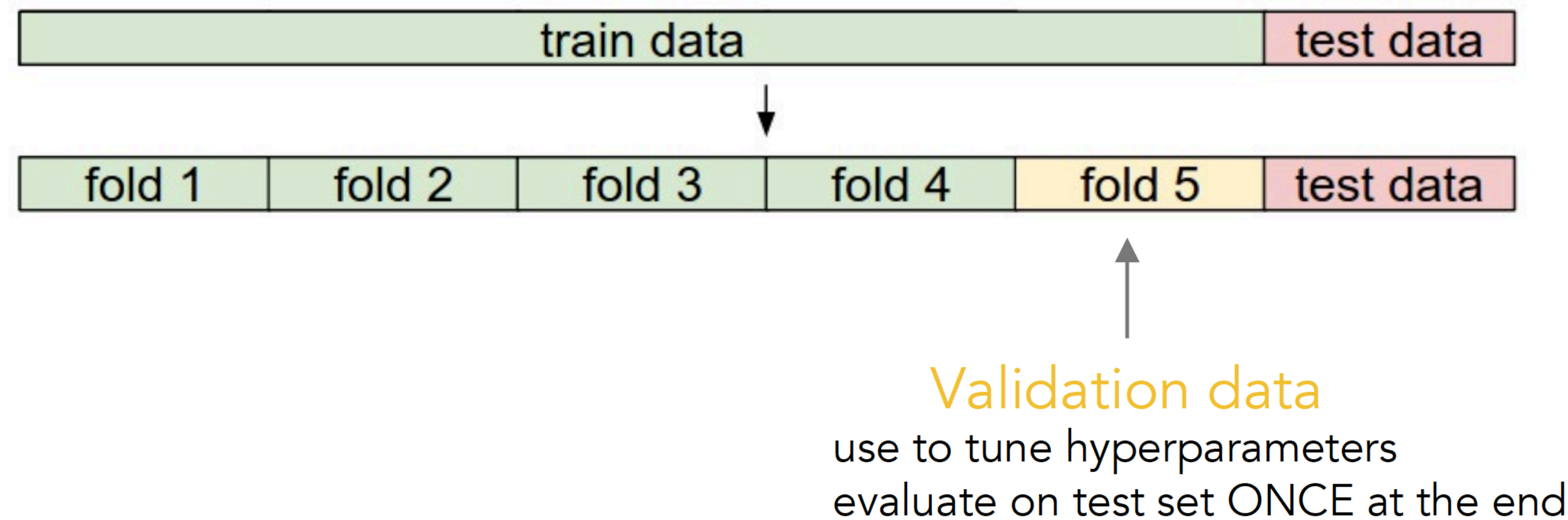
Use only **VERY SPARINGLY**, at the end.



Cross-Validation

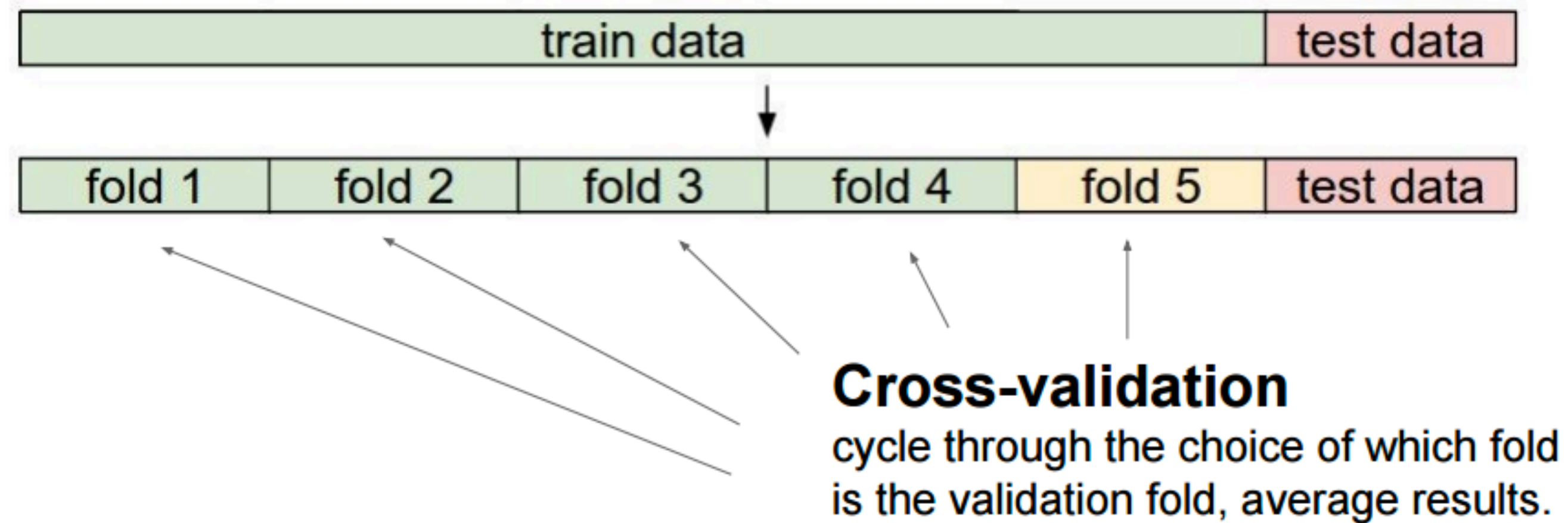
We cannot reliably estimate the error rate of the classifier using the training set

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



Cross-Validation

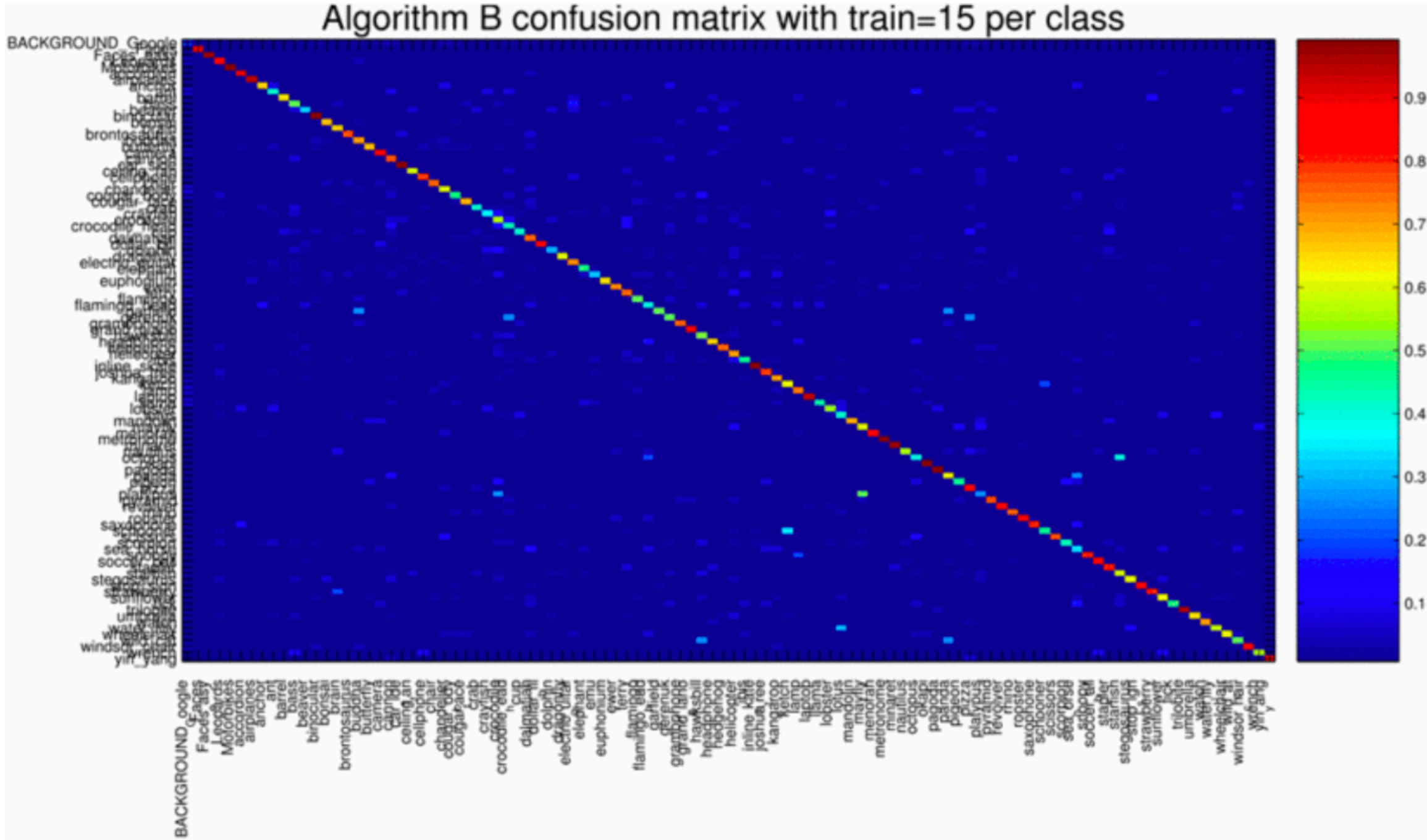
Cross-validation involves performing multiple splits and averaging the error over all splits



Confusion Matrix

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

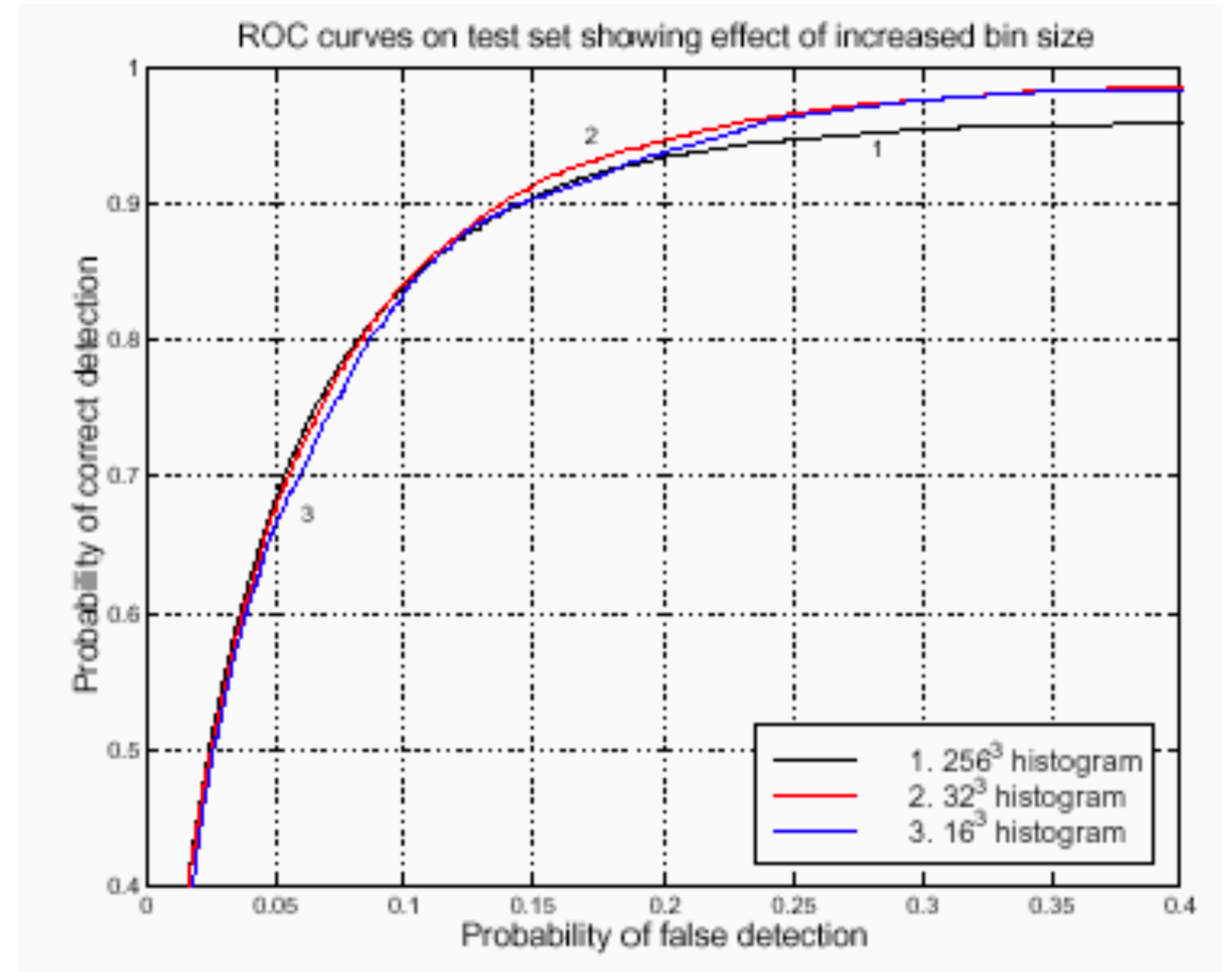
A **confusion matrix** is a table whose (i,j) th entry is the frequency (or proportion) an item of true class i was labelled as j by the classifier.



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



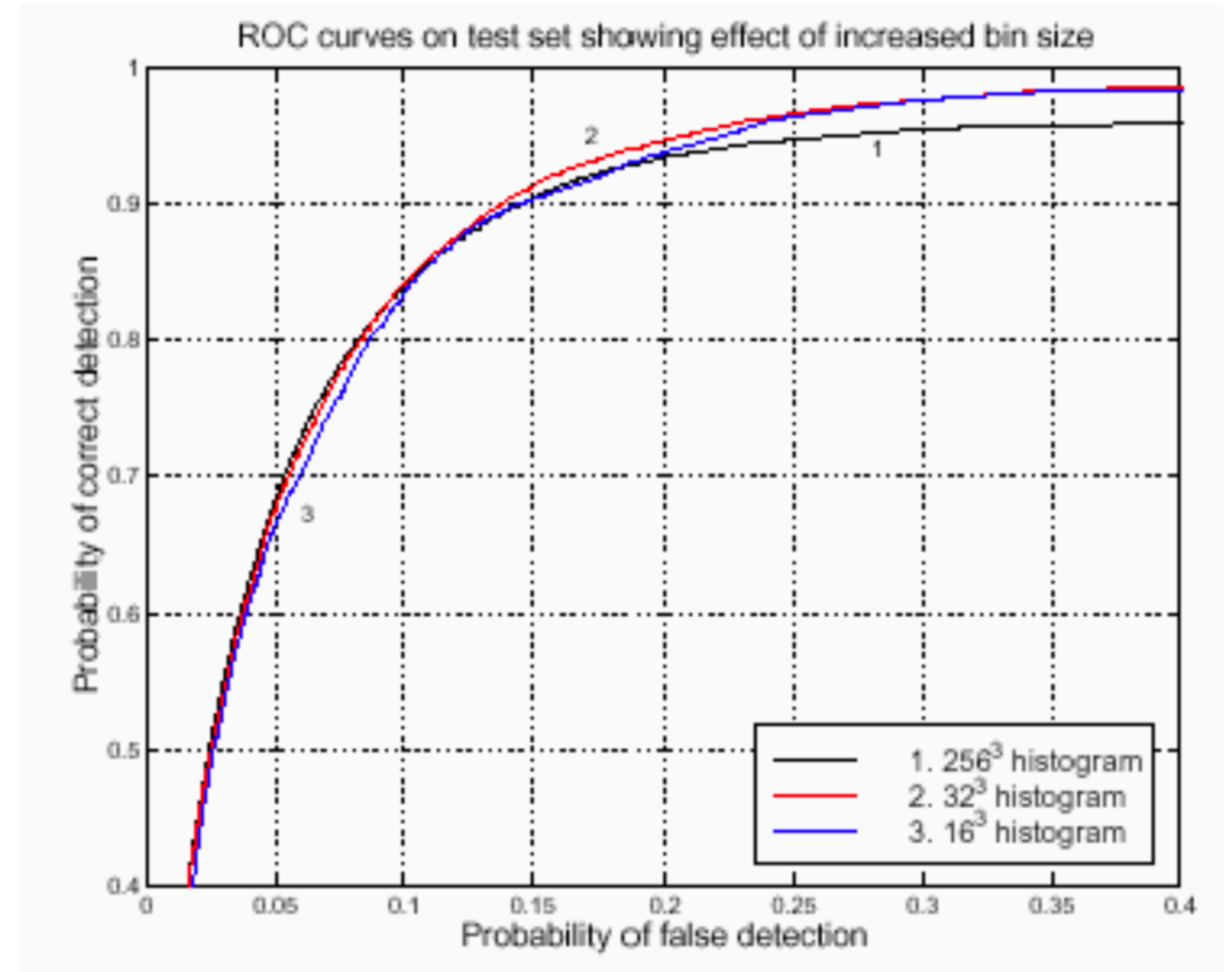
Forsyth & Ponce (2nd ed.) Figure 15.4

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



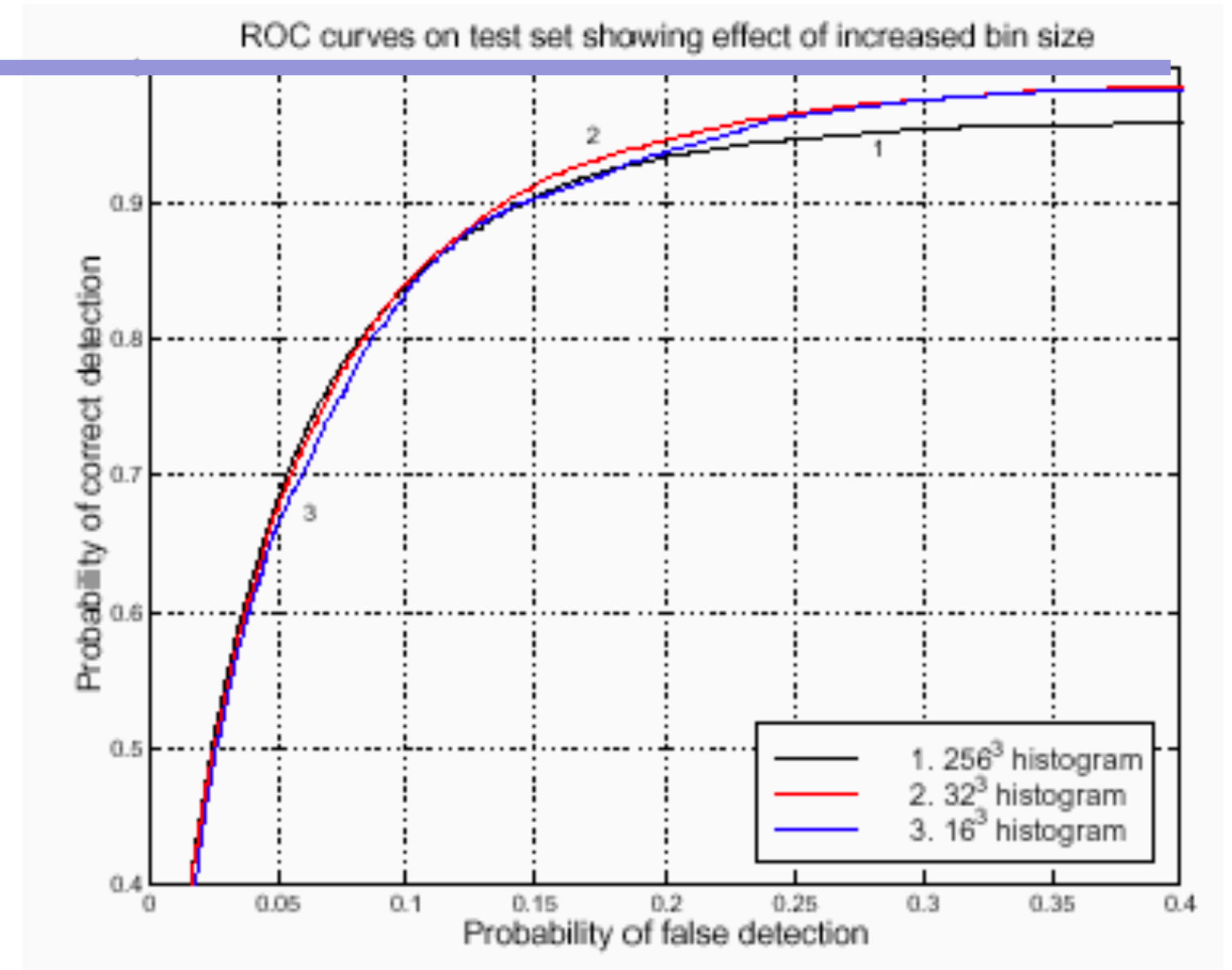
Forsyth & Ponce (2nd ed.) Figure 15.4

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.

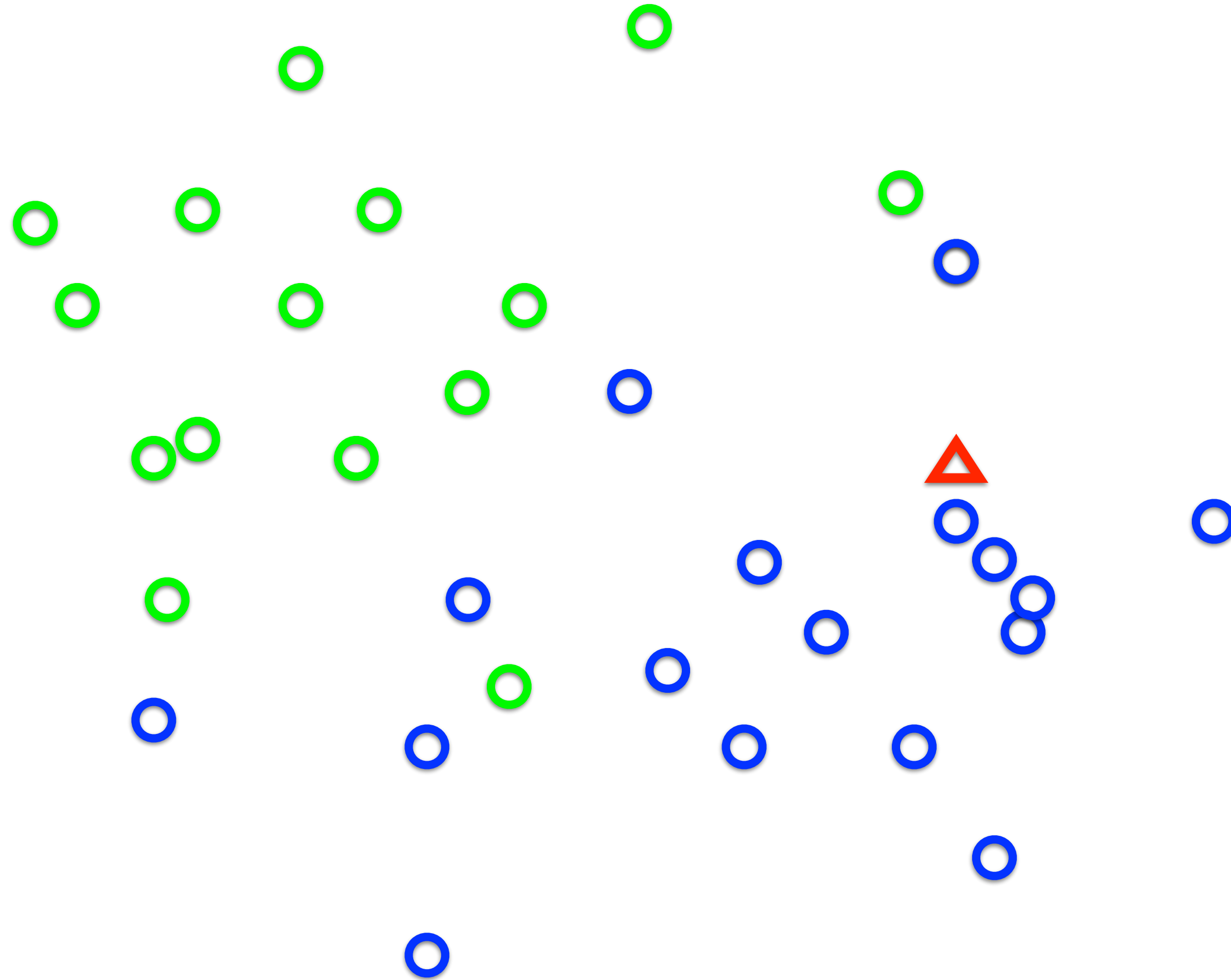
- fast, compact
- flexibility and accuracy depend on model assumptions

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

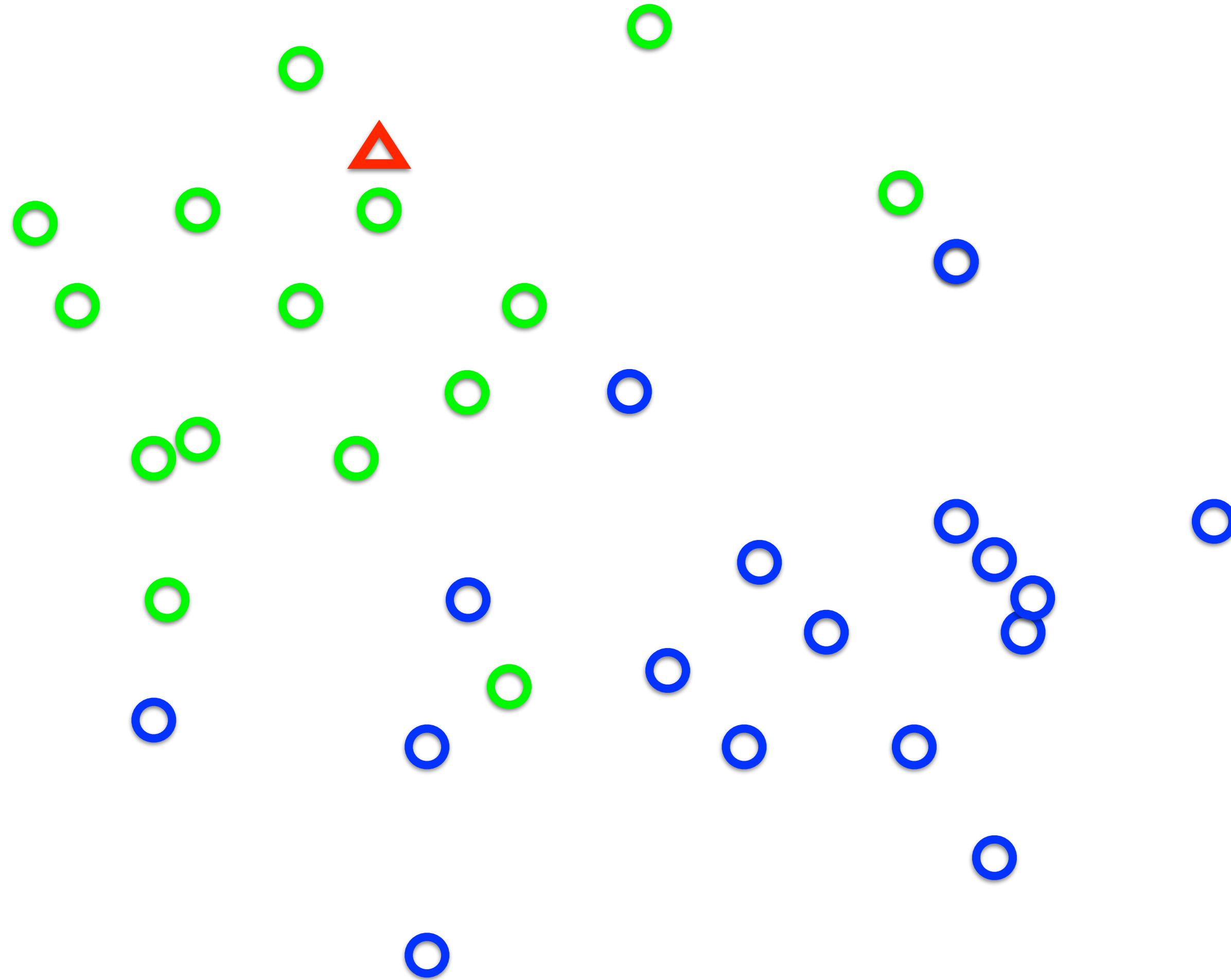
Nearest Neighbor Classifier

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



Nearest Neighbor Classifier

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



k-Nearest Neighbor (kNN) Classifier

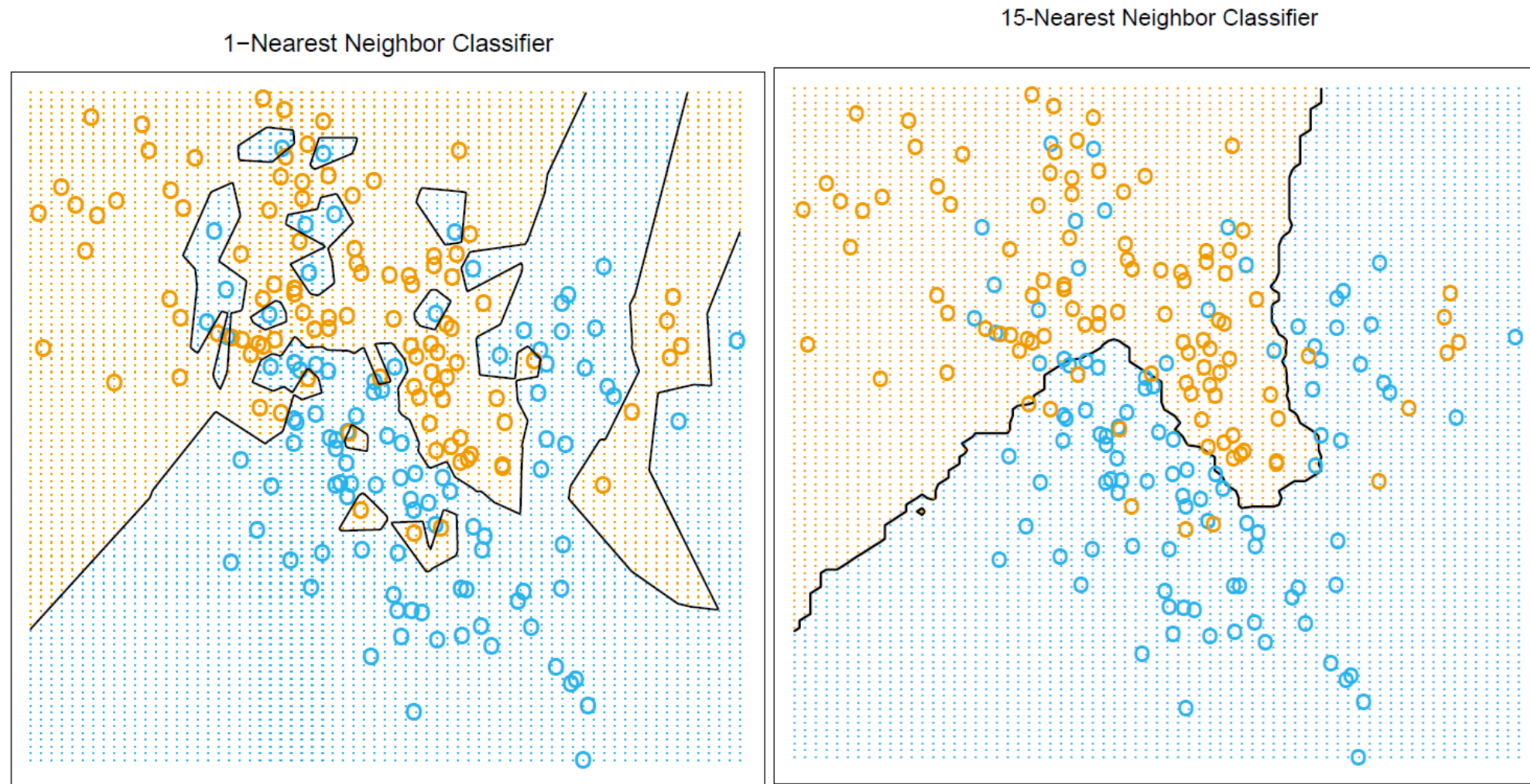
We can gain some robustness to noise by voting over **multiple** neighbours.

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.)