Statistical Pattern Recognition

Bayesian Decision Theory

Johan DEBAYLE, Yann GAVET, Jean-Charles PINOLI

Ecole Nationale Supérieure des Mines de Saint-Etienne
Laboratoire Georges Friedel, UMR CNRS 5307

Section 1

Introduction

Human perception

- **Humans** have developed highly sophisticated skills for sensing their environment and taking actions according to what they observe, e.g.
  - Recognizing a face
  - Understanding spoken words
  - Reading handwriting
  - Distinguishing fresh food from its smell
- We would like to give similar capabilities to **machines**

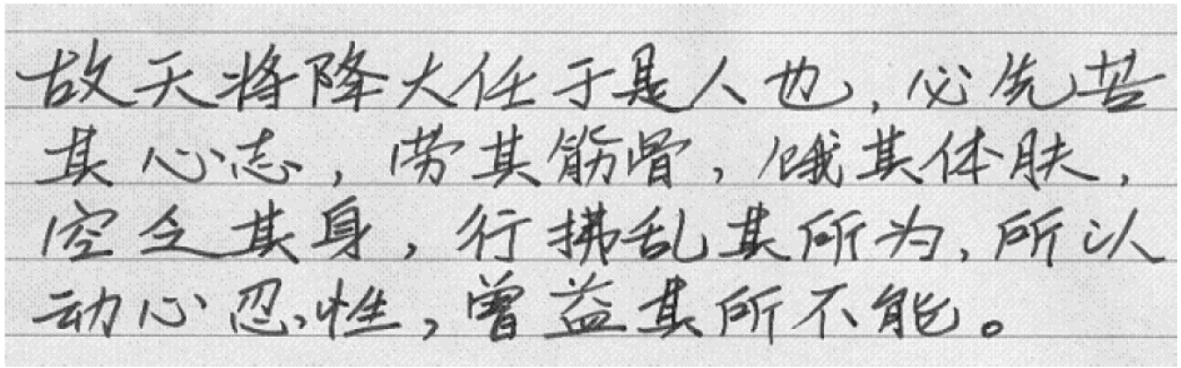
What is pattern recognition

- A **pattern** is an entity, vaguely defined, that could be given a name, e.g.,
  - Fingerprint image
  - Handwritten word
  - Human face
  - Speech signal
  - DNA sequence
  - ...
- **Pattern recognition** is the study of how machines can
  - Observe the environment
  - Learn to distinguish patterns of interest
  - Make sound and reasonable decisions about the categories of the patterns

Human and machine perception

- We are often influenced by the knowledge of how patterns are modeled and recognized in nature when we develop pattern recognition algorithms
- Research on machine perception also helps us gain deeper understanding and appreciation for pattern recognition systems in nature
- Yet, we also apply many techniques that are purely numerical and do not have any correspondence in natural systems

Pattern recognition applications



(a) Handwriting

故天将降大任于是人也，必先苦
其心志，劳其筋骨，饿其体肤，
空乏其身，行拂乱其所为，所以
动心忍性，曾益其所不能。

(b) Corresponding Machine Print

Figure: Chinese handwriting recognition

Pattern recognition applications

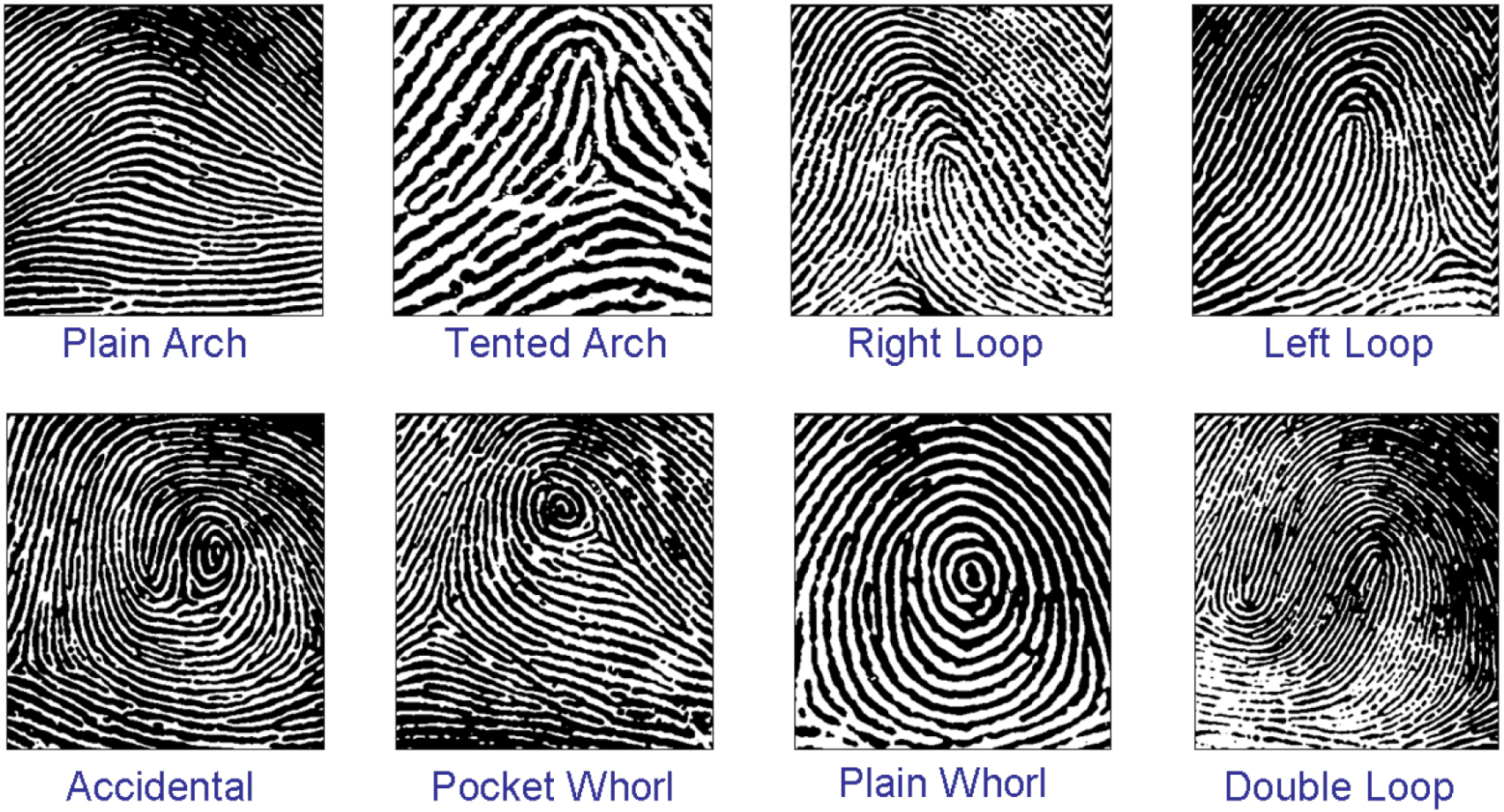


Figure: Fingerprint recognition

Pattern recognition applications

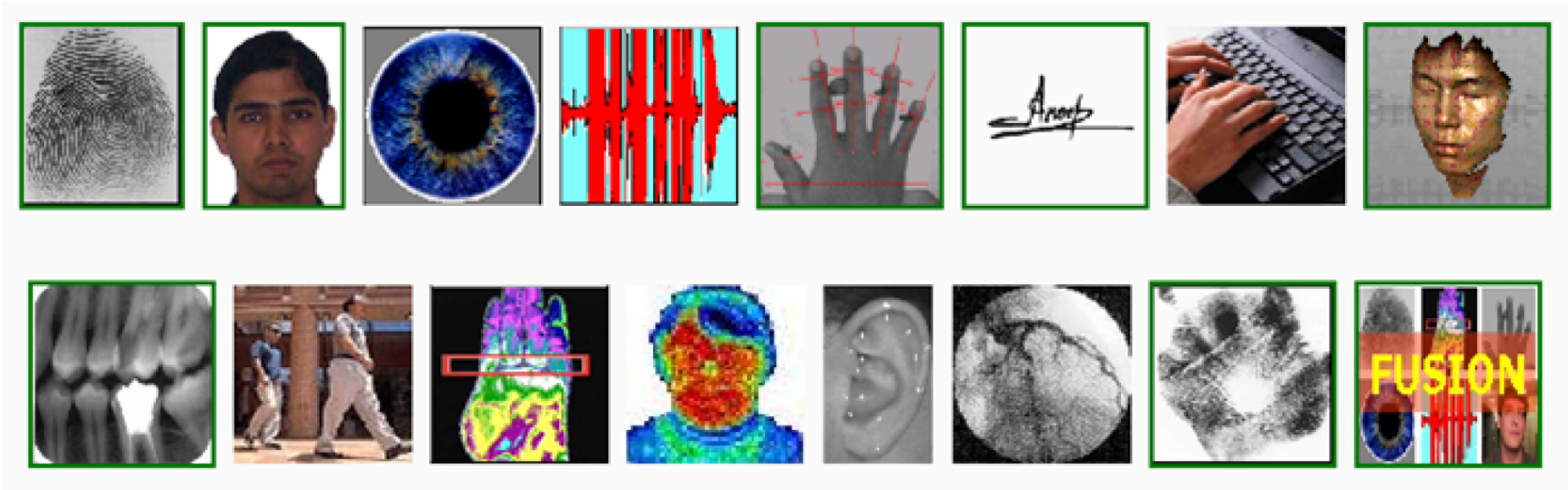


Figure: Biometric recognition

Pattern recognition applications

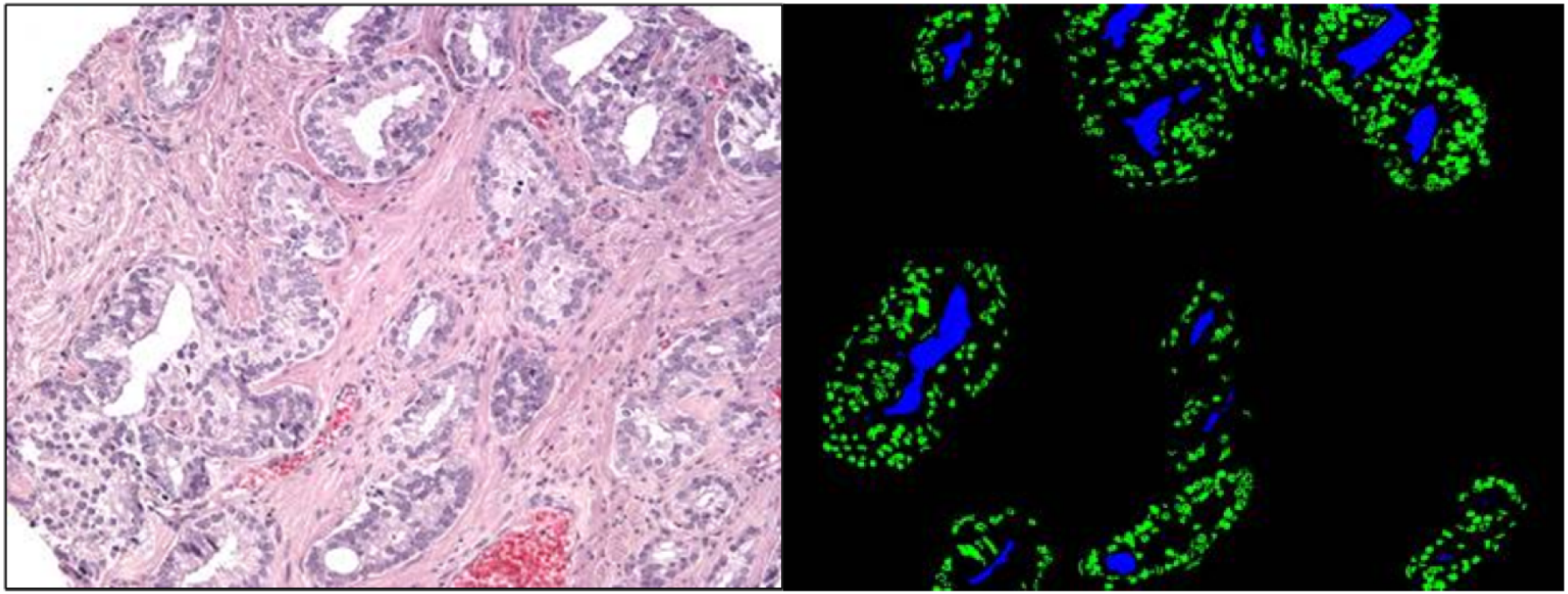


Figure: Cancer detection and grading using microscopy tissue data

Pattern recognition applications

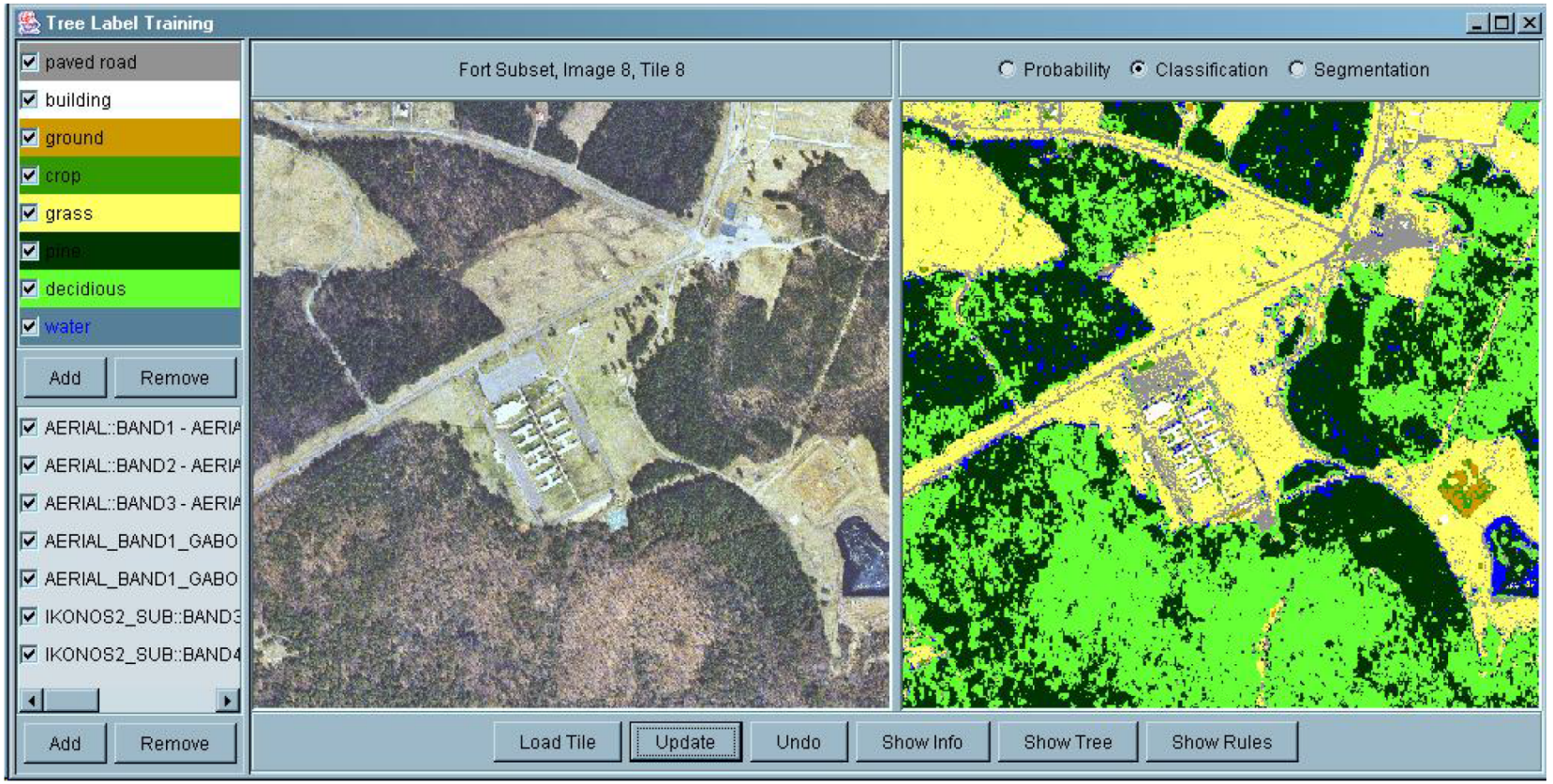


Figure: Land cover classification using satellite data

Pattern recognition applications

Figure: License plate recognition: US license plates

An example

- **Problem**: Sorting incoming fish on a conveyor belt according to species
- Assume that we have only two kinds of fish:
  - Sea bass
  - Salmon

Figure: Picture taken from a camera

An example: Decision process

- What kind of information can distinguish one species from the other?
  - Length, width, weight, number and shape of fins, tail shape...
- What can cause problems during sensing?
  - Lighting conditions, position of fish on the conveyor belt, camera noise...
- What are the steps in the process?
  - Image $\rightarrow$ isolate fish $\rightarrow$ take measurements $\rightarrow$ make decision

An example: Selecting features

- Assume a fisherman told us that a sea bass is generally longer than a salmon
- We can use length as a **feature** and decide between sea bass and salmon according to a threshold on length
- How can we choose this threshold?

An example: Selecting features

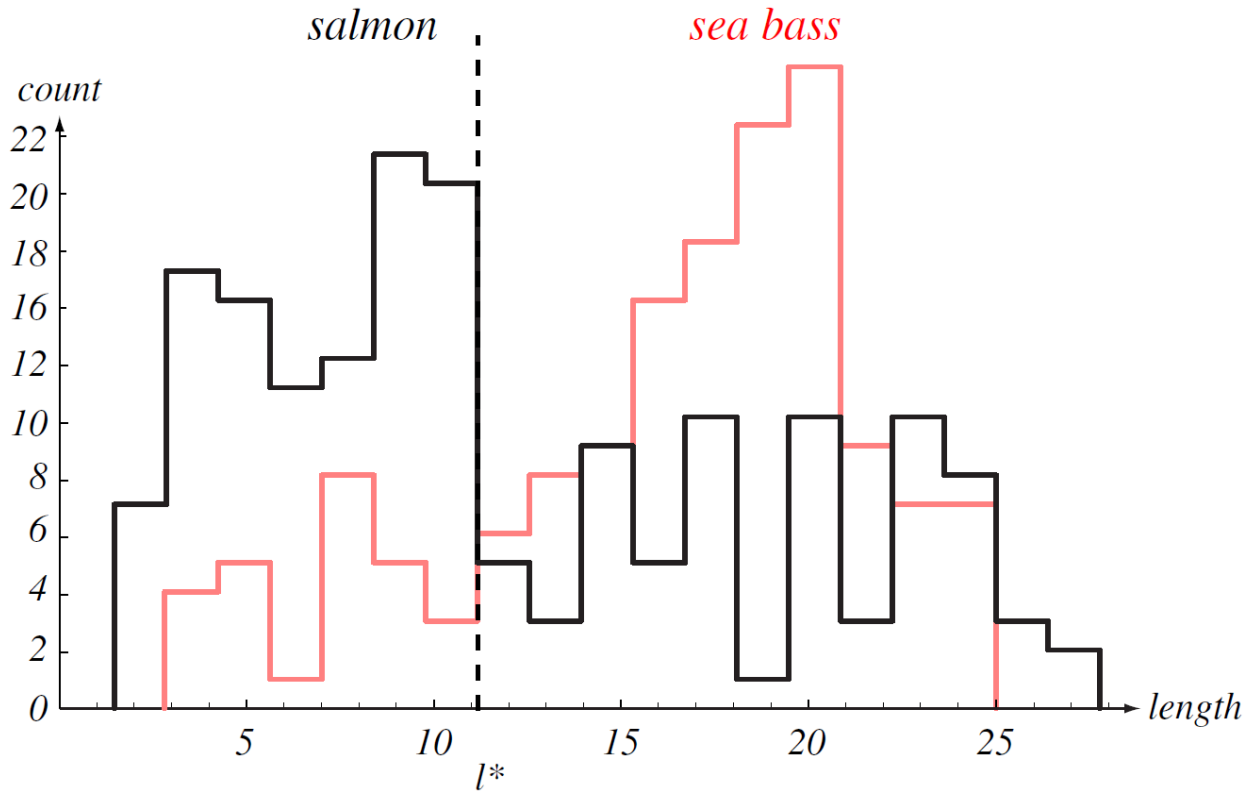


Figure: **Histograms** of the length feature for two types of fish in **training samples**. How can we choose the threshold l^* to make a reliable decision?

An example: Selecting features

- Even though sea bass is longer than salmon on the average, there are many examples of fish where this observation does not hold.
- Try **another feature**: average lightness of the fish scales.

An example: Selecting features

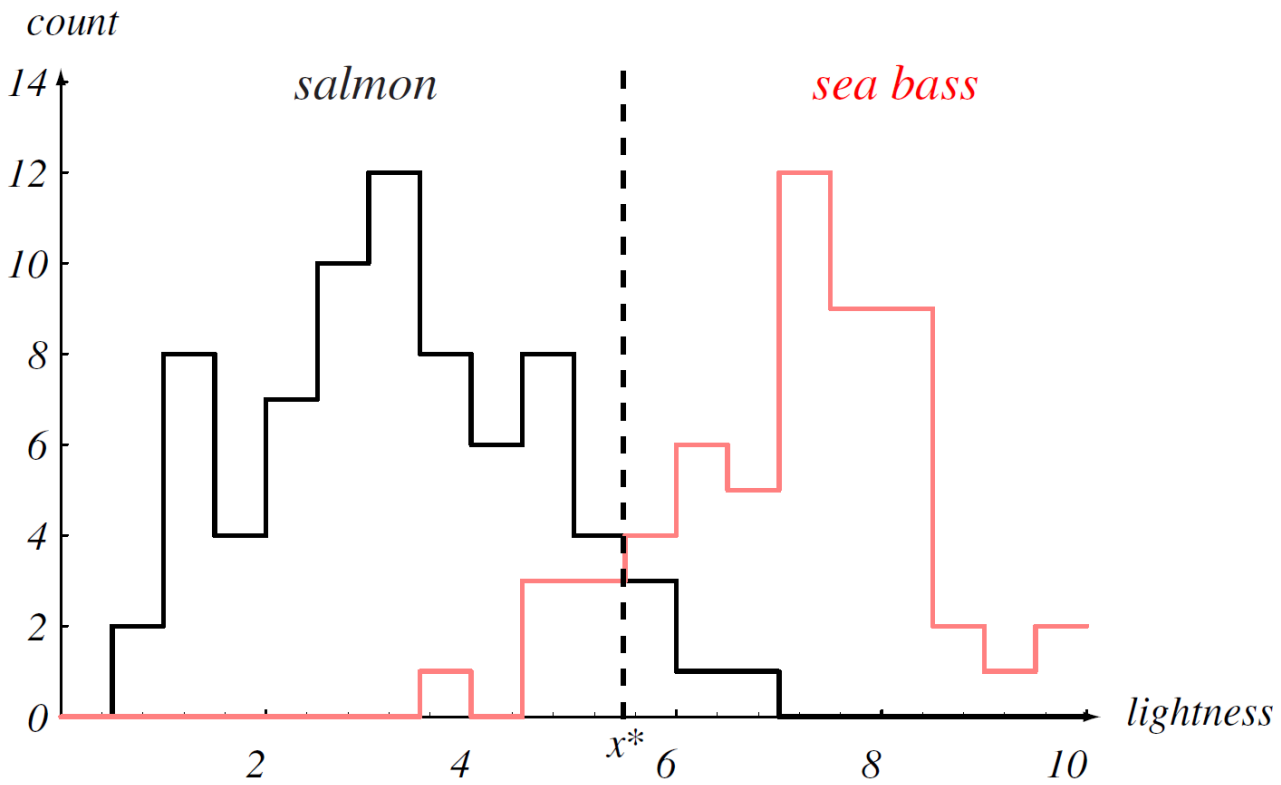


Figure: Histograms of the lightness feature for two types of fish in training samples. It looks easier to choose the threshold x^* but we still cannot make a perfect decision

An example: Multiple features

- Assume we also observed that sea bass are typically wider than salmon.
- We can use two features in our decision:
  - Lightness: x_1
  - Width: x_2
- Each fish image is now represented as a point (**feature vector**)

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

in a two-dimensional **feature space**

An example: Multiple features

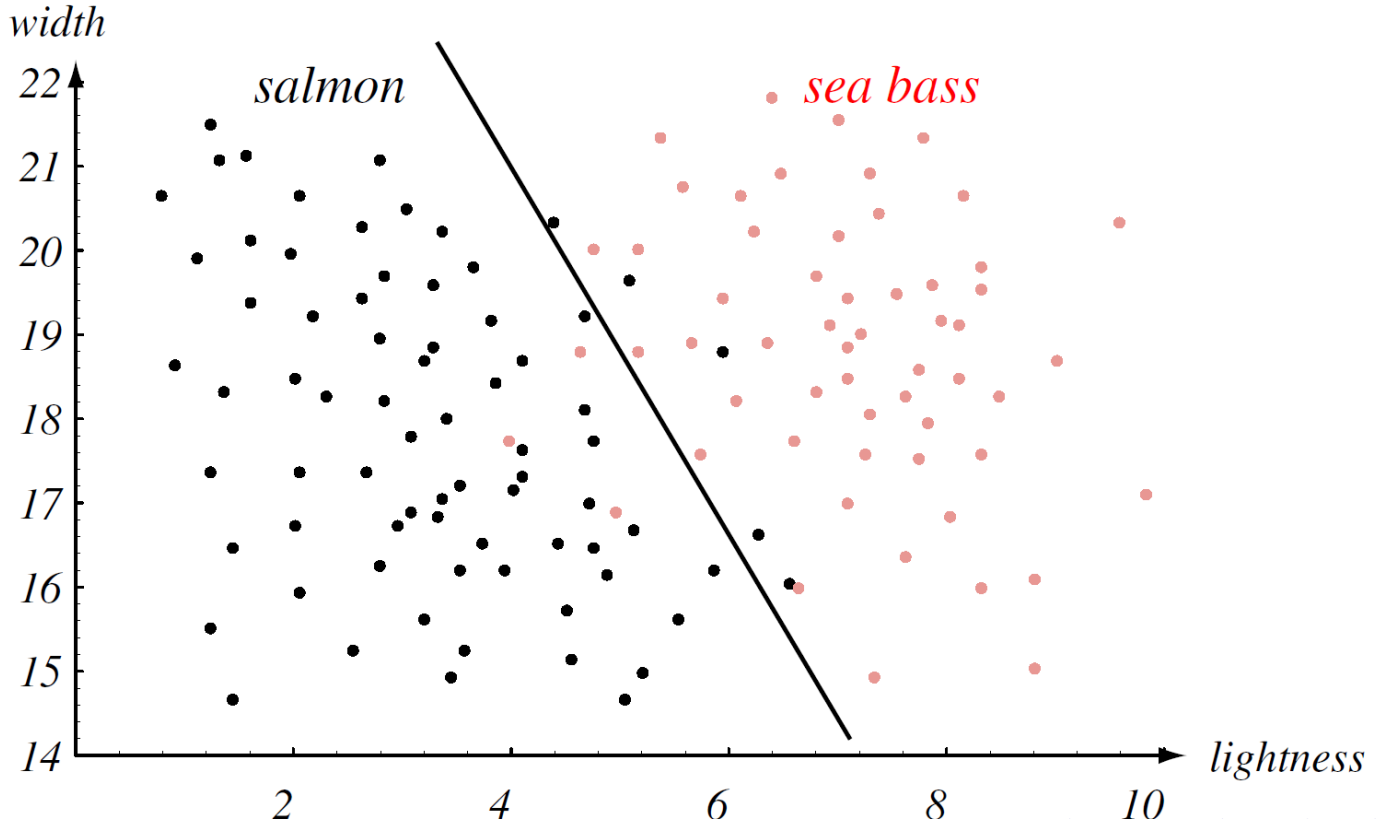


Figure: **Scatter plot** of lightness and width features for training samples. We can draw a **decision boundary** to divide the feature space into two regions. Does it look better than using only lightness?

An example: Multiple features

- Does adding **more features** always improve the results?
  - Avoid unreliable features
  - Be careful about correlations with existing features
  - Be careful about measurement costs
  - Be careful about noise in the measurements
- Is there some **curse** for working in very high dimensions?

An example: Decision boundaries

- Can we do better with another decision rule?
- More complex models result in more complex boundaries

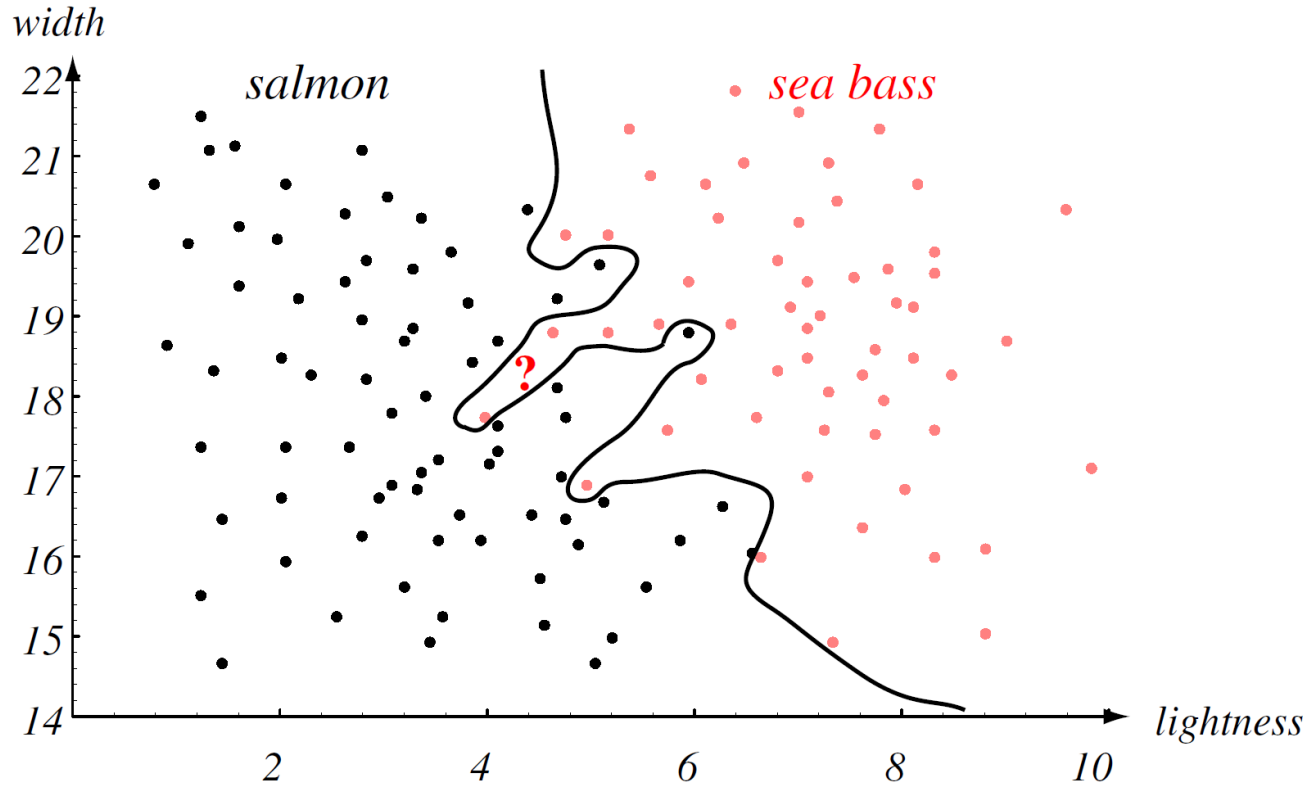


Figure: We may distinguish training samples perfectly but how can we predict how well we can **generalize** to unknown samples?

An example: Decision boundaries

- How can we manage the **tradeoff** between complexity of decision rules and their performance to unknown samples?

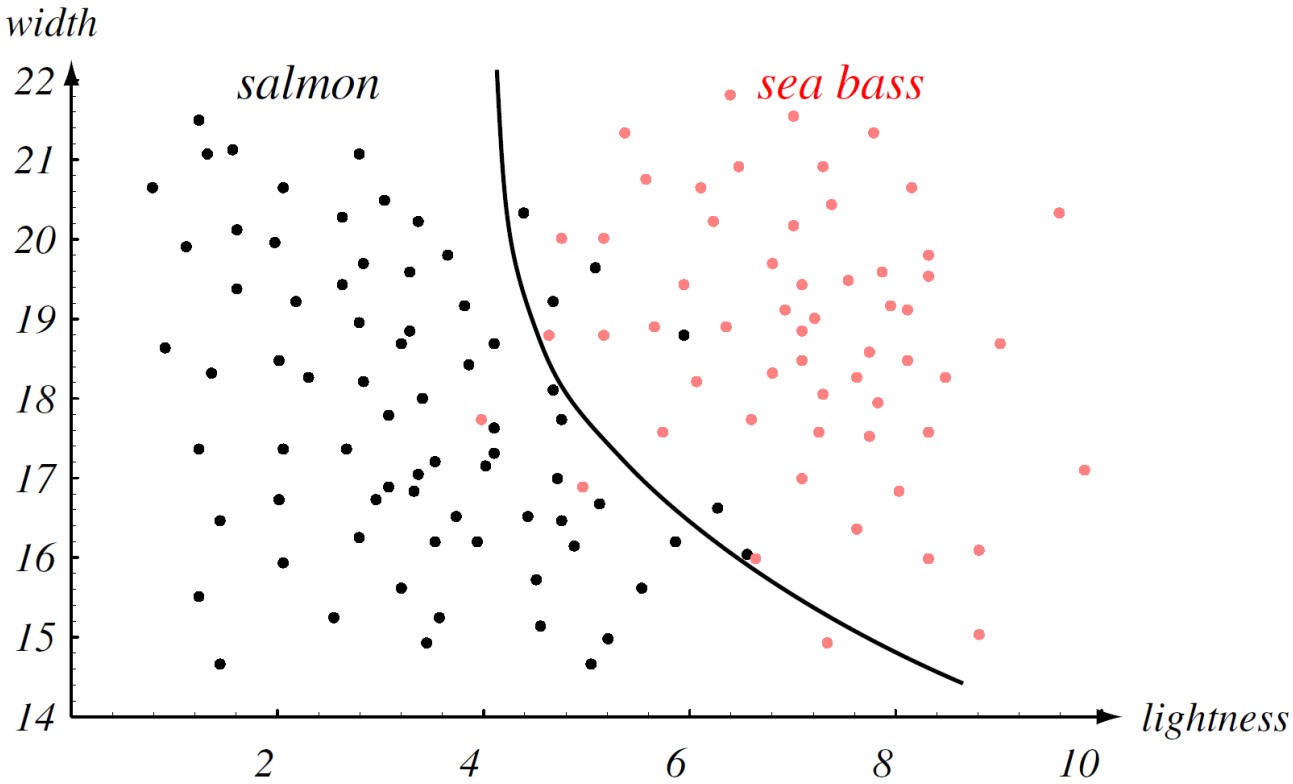


Figure: Different criteria lead to different decision boundaries.

Pattern recognition systems

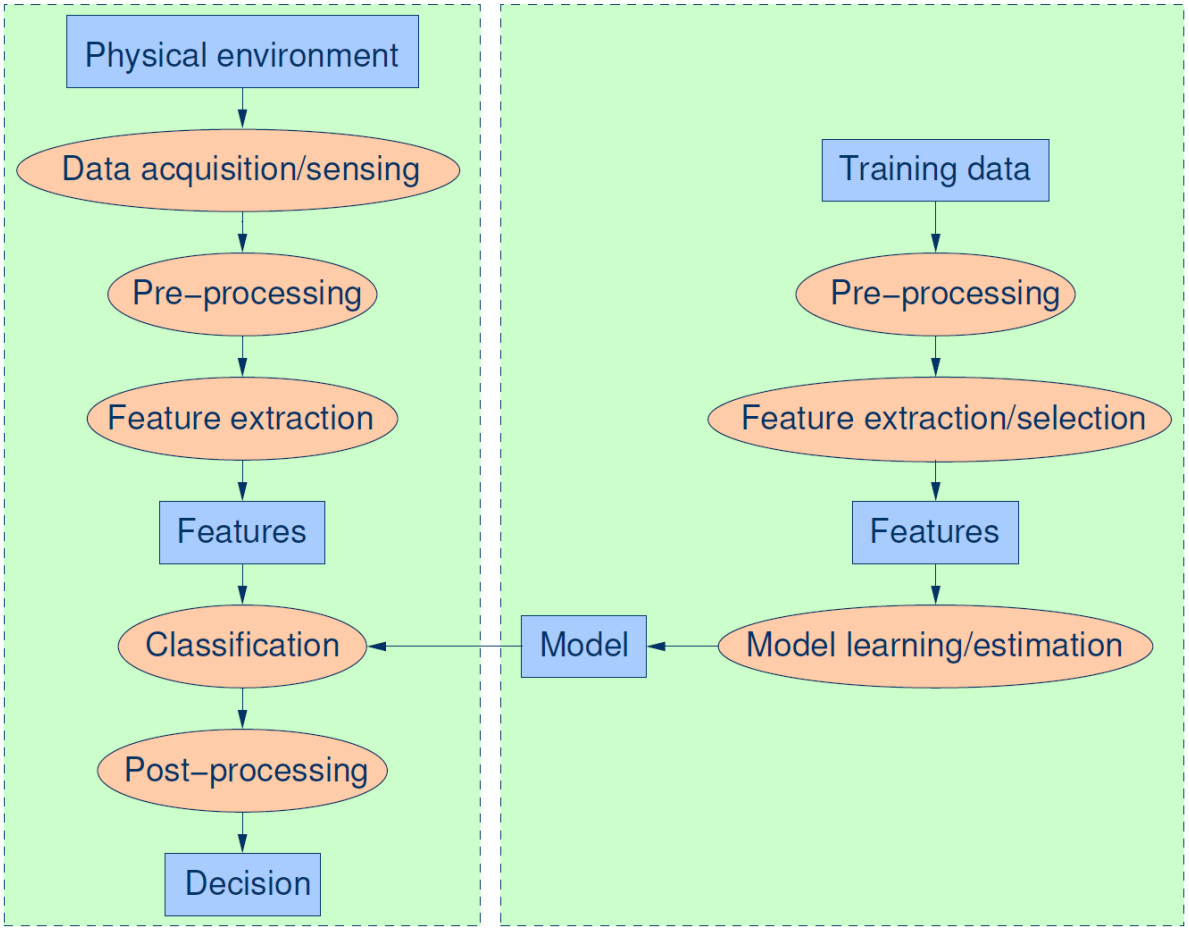


Figure: Object/process diagram of a pattern recognition system.

Pattern recognition sytems

- **Data acquisition and sensing**
  - Measurements of physical variables
  - Important issues: bandwidth, resolution, sensitivity, distortion, SNR, latency...
- **Pre-processing**
  - Removal of noise in data
  - Isolation of patterns of interest from the background
- **Feature extraction**
  - Finding a new representation in terms of features

Pattern recognition sytems

- **Model learning and estimation**
  - Learning a mapping between features and pattern groups and categories
- **Classification**
  - Using features and learned models to assign a pattern to a category
- **Post-processing**
  - Evaluation of confidence in decisions
  - Exploitation of context to improve performance
  - Combination of experts

The design cycle

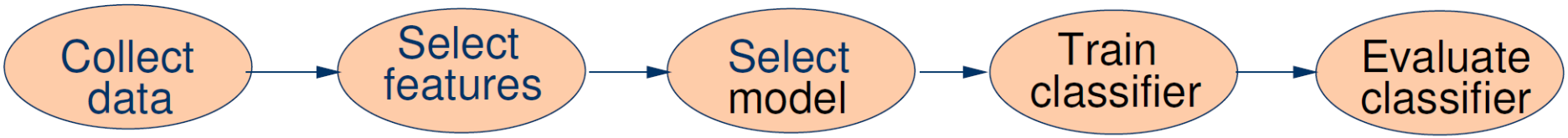


Figure: The design cycle

- **Data collection**
  - Collecting training and testing data
  - How can we know when we have adequately large and representative set of samples?

The design cycle

- **Feature selection**
  - Domain dependence and prior information
  - Computational cost and feasibility
  - Discriminative features
    - Similar values for similar patterns
    - Different values for different patterns
  - Invariant features with respect to translation, rotation and scale
  - Robust features with respect to occlusion, distortion, deformation, and variations in environment

The design cycle

- **Model selection**
  - Domain dependence and prior information
  - Definition of design criteria
  - Parametric vs. non-parametric models
  - Handling of missing features
  - Computational complexity
  - Types of models: templates, decision-theoretic or statistical, syntactic or structural, neural, and hybrid
  - How can we know how close we are to the true model underlying the patterns?

The design cycle

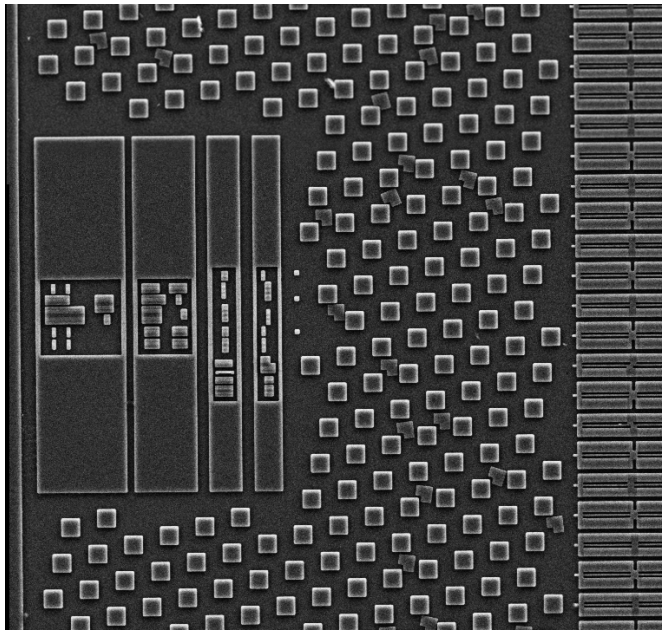
- **Training**
  - How can we learn the rule from data?
  - Supervised learning: a teacher provides a category label or cost for each pattern in the training set
  - Unsupervised learning: the system forms clusters or natural groupings of the input patterns
  - Reinforcement learning: no desired category is given but the teacher provides feedback to the system such as the decision is right or wrong

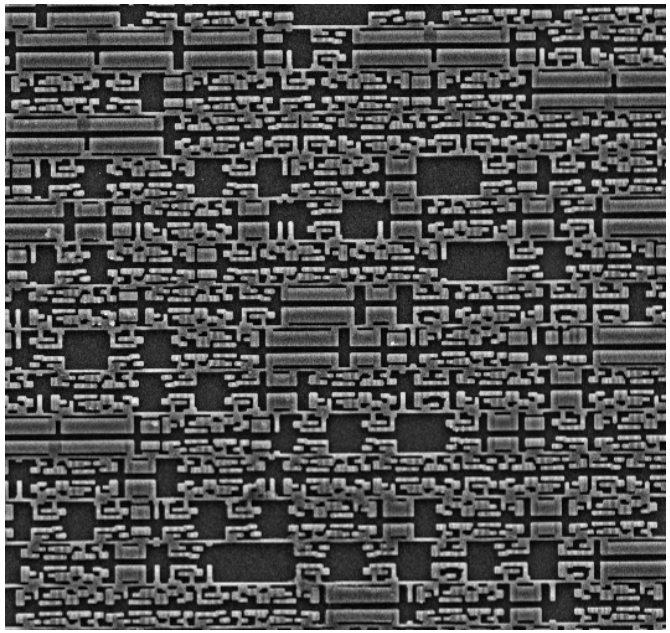
The design cycle

- **Evaluation**
  - How can we estimate the performance with training samples?
  - How can we predict the performance with future data?
  - Problems of overfitting and generalization

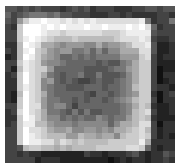
Application to integrated circuits

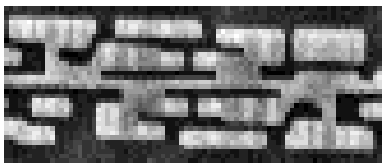
- Input images

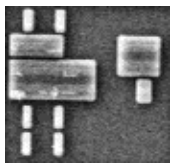




- Query patterns





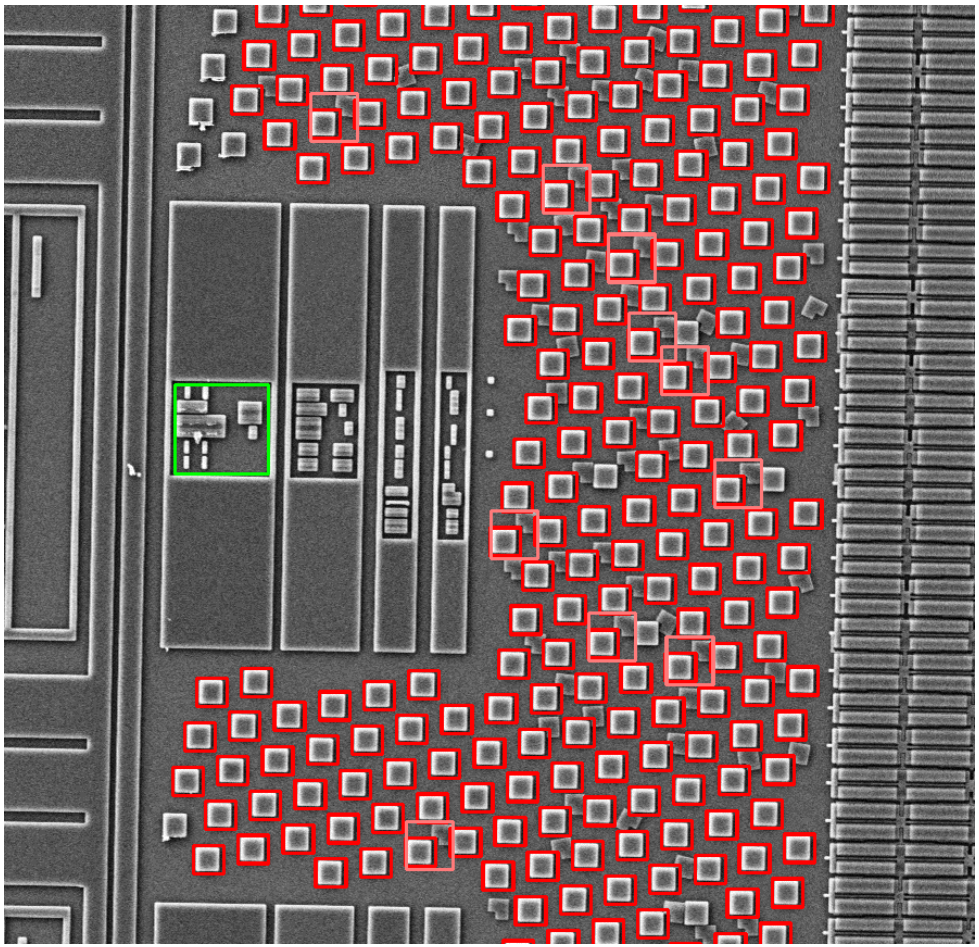


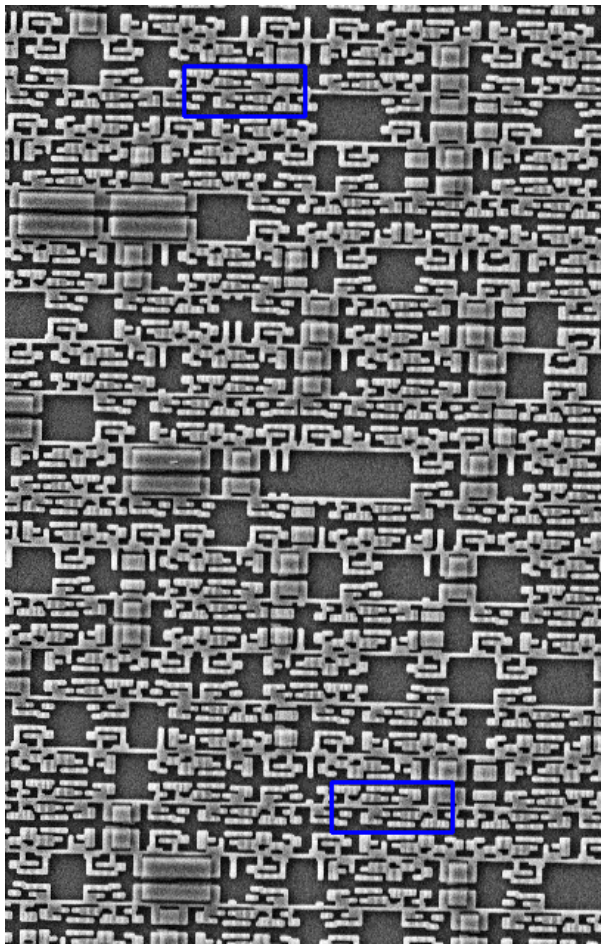
Application to integrated circuits

- Preprocessing: null
- Features: local gray-levels
- Model: $corr(W, Q) > T$
  - W: local window
  - Q: query pattern
  - T: decision threshold
- Training: null

Application to integrated circuits

- Results





Section 2

Bayesian Decision Theory

Overview

- Bayesian Decision Theory is a fundamental statistical approach to the problem of pattern classification
- Quantifies the tradeoffs between various classifications using probability and the costs that accompany such classifications.
- Assumptions:
  - Decision problem is posed in probabilistic terms.
  - All relevant probability values are known.

Recall the fish!

- Recall our example on classifying two fish as salmon or sea bass.
- And recall our agreement that any given fish is either a salmon or a sea bass; it is called the state of nature of the fish.
- Let's define a (probabilistic) variable ω that describes the state of nature.

$$\omega = \omega_1 \quad \text{for sea bass} \tag{1}$$
$$\omega = \omega_2 \quad \text{for salmon} \tag{2}$$

- Let's assume this two class case.

Salmon

Sea bass

Prior probability

- The *a priori* or prior probability reflects our knowledge of how likely we expect a certain state of nature before we can actually observe said state of nature.
- In the fish example, it is the probability that we will see either a salmon or a sea bass next on the conveyor belt.
- Note: The prior may vary depending on the situation.
  - If we get equal numbers of salmon and sea bass in a catch, then the priors are equal, or uniform.
  - Depending on the season, we may get more salmon than sea bass, for example.
- We write $P(\omega = \omega_1)$ or just $P(\omega_1)$ for the prior the next is a sea bass.
- The priors must exhibit exclusivity and exhaustivity. For c states of nature, or classes:

$$1 = \sum_{i=1}^{c} P(\omega_i) \tag{3}$$

Decision rule from only priors

- A decision rule prescribes what action to take based on observed input.
- IDEA CHECK: What is a reasonable Decision Rule if
  - the only available information is the prior, and
  - the cost of any incorrect classification is equal?
- Decide ω_1 if $P(\omega_1) > P(\omega_2)$; otherwide decide ω_2.
- What can we say about this decision rule?
  - Seems reasonable, but it will always choose the same fish.
  - If the priors are uniform, this rule will behave poorly.
  - Under the given assumptions, no other rule can do better! (We will see this later on.)

Features and feature spaces

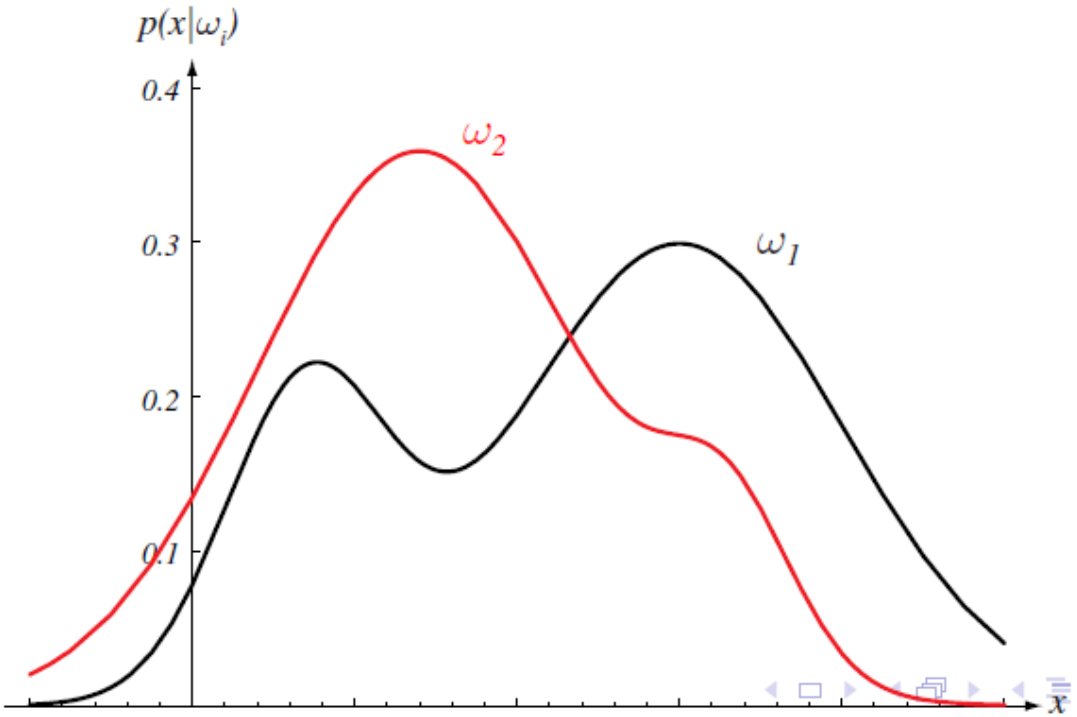
- A feature is an observable variable.
- A feature space is a set from which we can sample or observe values.
- Examples of features:
  - Length
  - Width
  - Lightness
  - Location of dorsal fin
- For simplicity, let's assume that our features are all continuous values.
- Denote a scalar feature as x and a vector feature as $\mathbf{x}$. For a d-dimensional feature space, $\mathbf{x} \in \mathbb{R}^2$.

Class-conditional density of Likelihood

- The class-conditional probability density function is the probability density function for **x**, our feature, given that the state of nature is ω:

$$p(\mathbf{x}|\omega) \tag{4}$$

- Here is the hypothetical class-conditional density $p(x|\omega)$ for lightness values of sea bass and salmon.



Posterior probability / Bayes formula

- If we know the prior distribution and the class-conditional density, how does this affect our decision rule?
- Posterior probability is the probability of a certain state of nature given our observables: $P(\omega|\mathbf{x})$.
- Use Bayes formula:

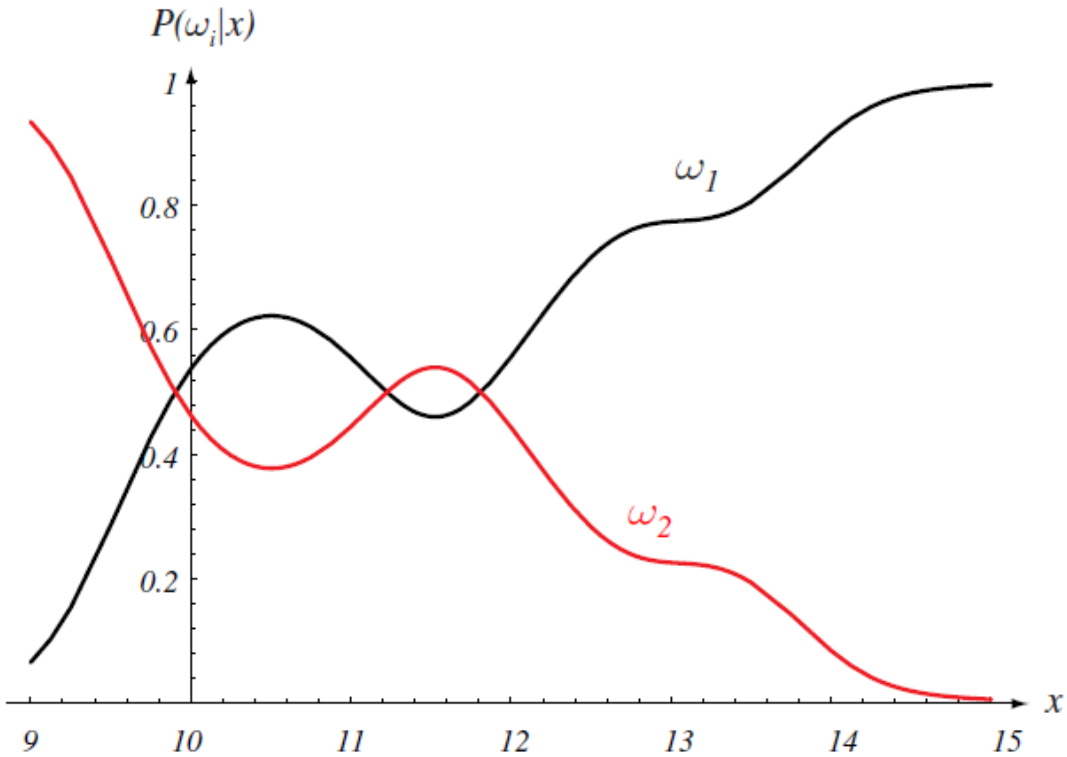
$$P(\omega, \mathbf{x}) = P(\omega|\mathbf{x})p(\mathbf{x}) = p(\mathbf{x}|\omega)P(\omega) \tag{5}$$

$$P(\omega|\mathbf{x}) = \frac{p(\mathbf{x}|\omega)P(\omega)}{p(x)} \tag{6}$$

$$= \frac{p(\mathbf{x}|\omega)P(\omega)}{\sum_i p(\mathbf{x}|\omega_i)P(\omega_i)} \tag{7}$$

Posterior probability

- Notice the likelihood and the prior govern the posterior. The $p(\mathbf{x})$ evidence term is a scale-factor to normalize the density.
- For the case of $P(\omega_1) = 2/3$ and $P(\omega_2) = 1/3$ the posterior is



Probability of error

- For a given observation x, we would be inclined to let the posterior govern our decision:

$$\omega^* = \arg\max_i P(\omega_i|\mathbf{x}) \tag{8}$$

- What is our probability of error?
- For the two class situation, we have

$$P(\text{error}|\mathbf{x}) = \begin{cases} P(\omega_1|\mathbf{x}) & \text{if we decide } \omega_2 \\ P(\omega_2|\mathbf{x}) & \text{if we decide } \omega_1 \end{cases} \tag{9}$$

Probability of error

- We can minimize the probability of error by following the posterior:

$$\text{Decide } \omega_1 \text{ if } P(\omega_1|\mathbf{x}) > P(\omega_2|\mathbf{x}) \tag{10}$$

- And, this minimizes the average probability of error too:

$$P(\text{error}) = \int_{-\infty}^{+\infty} P(\text{error}|\mathbf{x})p(\mathbf{x})d\mathbf{x} \tag{11}$$

(Because the integral will be minimized when we can ensure each $P(\text{error}|\mathbf{x})$ is as small as possible.)

Bayes decision rule (with equal costs)

- Decide ω_1 if $P(\omega_1|\mathbf{x}) > P(\omega_2|\mathbf{x})$; otherwise decide ω_2
- Probability of error becomes

$$P(\text{error}|\mathbf{x}) = \min[P(\omega_1|\mathbf{x}), P(\omega_2|\mathbf{x})] \tag{12}$$

- Equivalently, Decide ω_1 if $p(\mathbf{x}|\omega_1)P(\omega_1) > p(\mathbf{x}|\omega_2)P(\omega_2)$; otherwise decide ω_2
- i.e., the evidence term is not used in decision making.
- If we have $p(\mathbf{x}|\omega_1) = p(\mathbf{x}|\omega_2)$, then the decision will rely exclusively on the priors.
- Conversely, if we have uniform priors, then the decision will rely exclusively on the likelihoods.
- Take Home Message: Decision making relies on both the priors and the likelihoods and Bayes Decision Rule combines them to achieve the minimum probability of error.

Loss functions

- A loss function states exactly how costly each action is.
- As earlier, we have c classes $\{\omega_1, \ldots, \omega_c\}$.
- We also have a possible actions $\{\alpha_1, \ldots, \alpha_a\}$.
- The loss function $\lambda(\alpha_i|\omega_j)$ is the loss incurred for taking action α_i when the class is ω_j.
- The Zero-One Loss Function is a particularly common one:

$$\lambda(\alpha_i|\omega_j) = \begin{cases} 0 & i = j \\ 1 & i \neq j \end{cases} \qquad i, j = 1, 2, \ldots, c \tag{13}$$

It assigns no loss to a correct decision and uniform unit loss to an incorrect decision.

Expected loss a.k.a. Conditional risk

- We can consider the loss that would be incurred from taking each possible action in our set.
- The expected loss or conditional risk is by definition

$$R(\alpha_i|\mathbf{x}) = \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j) P(\omega_j|\mathbf{x}) \tag{14}$$

- The zero-one conditional risk is

$$\begin{aligned} R(\alpha_i|\mathbf{x}) &= \sum_{j \neq i} P(\omega_j|\mathbf{x}) && (15) \\ &= 1 - P(\omega_i|\mathbf{x}) && (16) \end{aligned}$$

- Hence, for an observation $\mathbf{x}$, we can minimize the expected loss by selecting the action that minimizes the conditional risk.
- (Teaser) You guessed it: this is what Bayes Decision Rule does!

Overall risk

- Let $\alpha(x)$ denote a decision rule, a mapping from the input feature space to an action, $\mathbb{R}^d \mapsto \{\alpha_1, \ldots, \alpha_a\}$.
  This is what we want to learn.
- The overall risk is the expected loss associated with a given decision rule.

$$R = \int R(\alpha(\mathbf{x})|\mathbf{x})p(\mathbf{x})d\mathbf{x} \tag{17}$$

Clearly, we want the rule $\alpha(.)$ that minimizes $R(\alpha(\mathbf{x})|\mathbf{x})$ for all $\mathbf{x}$.

Bayes risk / The minimum overall risk

- Bayes Decision Rule gives us a method for minimizing the overall risk.
- Select the action that minimizes the conditional risk:

$$\alpha^* = \arg\min_{\alpha_i} R(\alpha_i|\mathbf{x}) \tag{18}$$

$$= \arg\min_{\alpha_i} \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j) P(\omega_j|\mathbf{x}) \tag{19}$$

- The Bayes risk is the best we can do.

Two-category classification examples risk

- Consider two classes and two actions, α_1 when the true class is ω_1 and α_2 for ω_2.
- Writing out the conditional risks gives:

$$R(\alpha_1|\mathbf{x}) = \lambda_{11}P(\omega_1|\mathbf{x}) + \lambda_{12}P(\omega_2|x) \tag{20}$$

$$R(\alpha_2|\mathbf{x}) = \lambda_{21}P(\omega_1|\mathbf{x}) + \lambda_{22}P(\omega_2|x) \tag{21}$$

- Fundamental rule is decide ω_1 if

$$R(\alpha_1|\mathbf{x}) < R(\alpha_2|\mathbf{x}) \tag{22}$$

- In terms of posteriors, decide ω_1 if

$$(\lambda_{21} - \lambda_{11})P(\omega_1|\mathbf{x}) > (\lambda_{12} - \lambda_{22})P(\omega_2|\mathbf{x}) \tag{23}$$

The more likely state of nature is scaled by the differences in loss (which are generally positive).

Two-category classification examples risk

- Or, expanding via Bayes Rule, decide ω_1 if

$$(\lambda_{21} - \lambda_{11})p(\mathbf{x}|\omega_1)P(\omega_1) > (\lambda_{12} - \lambda_{22})p(\mathbf{x}|\omega_2)P(\omega_2) \tag{24}$$

- Or, assuming $\lambda_{21} > \lambda_{11}$, decide ω_1 if

$$\frac{p(\mathbf{x}|\omega_1)}{p(\mathbf{x}|\omega_2)} = \frac{\lambda_{12} - \lambda_{22}}{\lambda_{21} - \lambda_{11}} \frac{P(\omega_2)}{P(\omega_1)} \tag{25}$$

  LHS is called the likelihood ratio.

- Thus, we can say the Bayes Decision Rule says to decide ω_1 if the likelihood ratio exceeds a threshold that is independent of the observation $\mathbf{x}$.

Discriminant functions

- Discriminant Functions are a useful way of representing pattern classifiers.
- Let's say $g_i(\mathbf{x})$ is a discriminant function for the ith class.
- This classifier will assign a class ω_i to the feature vector $\mathbf{x}$ if

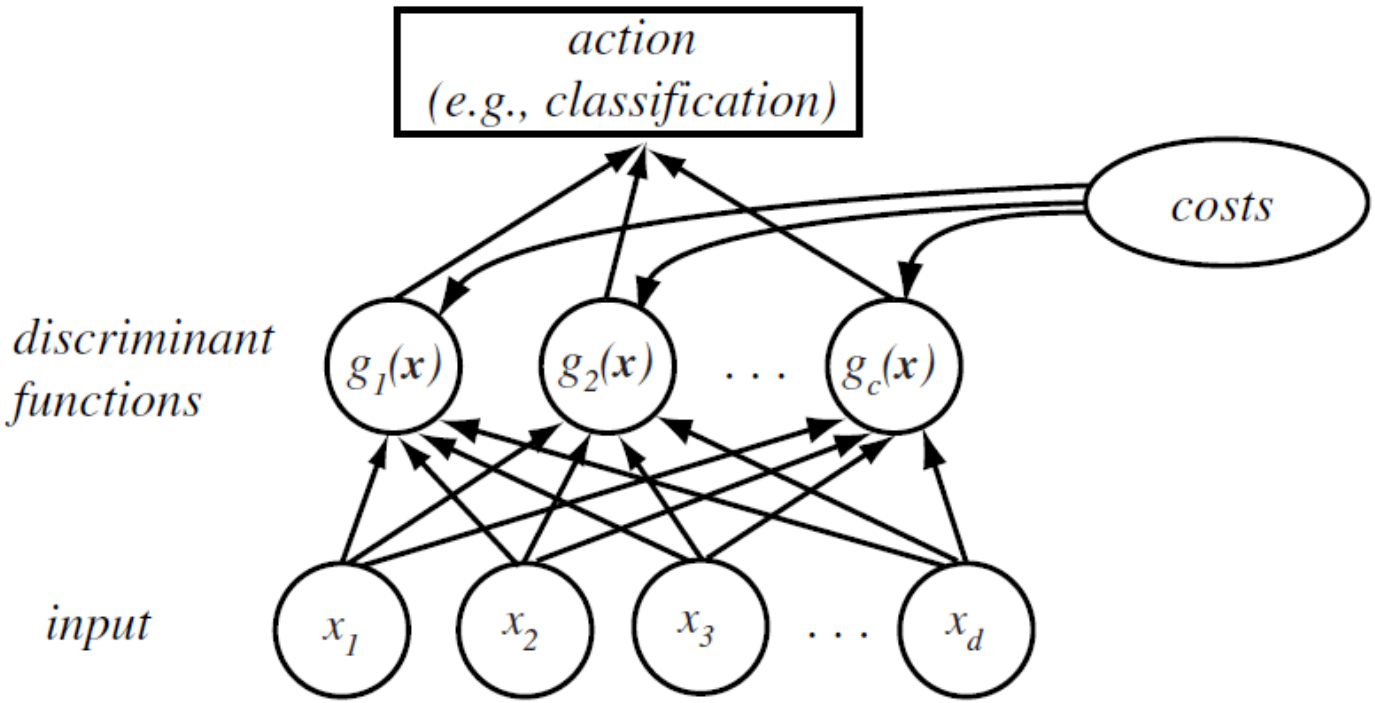
$$g_i(\mathbf{x}) > g_j(\mathbf{x}) \quad \forall j \neq i, \tag{26}$$

or, equivalently

$$i^* = \arg\max_i g_i(\mathbf{x}), \quad \text{decide } \omega_{i^*} \tag{27}$$

Discriminants as a network

- We can view the discriminant classifier as a network (for c classes and a d-dimensional input vector).



Bayes discriminants / Min. conditional risk discriminant

- General case with risks

$$g_i(\mathbf{x}) = -R(\alpha_i|\mathbf{x}) \quad (28)$$

$$= -\sum_{j=1}^{c} \lambda(\alpha_i|\omega_j) P(\omega_j|\mathbf{x}) \quad (29)$$

- Can we prove that this is correct?
- Yes! The minimum conditional risk corresponds to the maximum discriminant.

Minimum error-rate discriminant

- In the case of zero-one loss function, the Bayes Discriminant can be further simplified:

$$g_i(\mathbf{x}) = P(\omega_i|\mathbf{x}) \tag{30}$$

Uniqueness of discriminants

- Is the choice of discriminant functions unique?
- No!
- Multiply by some positive constant.
- Shift them by some additive constant.
- For monotonically increasing function $f(.)$, we can replace each $g_i(\mathbf{x})$ by $f(g_i(\mathbf{x}))$ without affecting our classification accuracy.
  - These can help for ease of understanding or computability.
  - The following all yield the same exact classification results for minimum-error-rate classification.

$$g_i(\mathbf{x}) = P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{\sum_j p(\mathbf{x}|\omega_j)P(\omega_j)} \quad (31)$$

$$g_i(\mathbf{x}) = p(\mathbf{x}|\omega_i)P(\omega_i) \quad (32)$$

$$g_i(\mathbf{x}) = \ln p(\mathbf{x}|\omega_i) + \ln P(\omega_i) \quad (33)$$

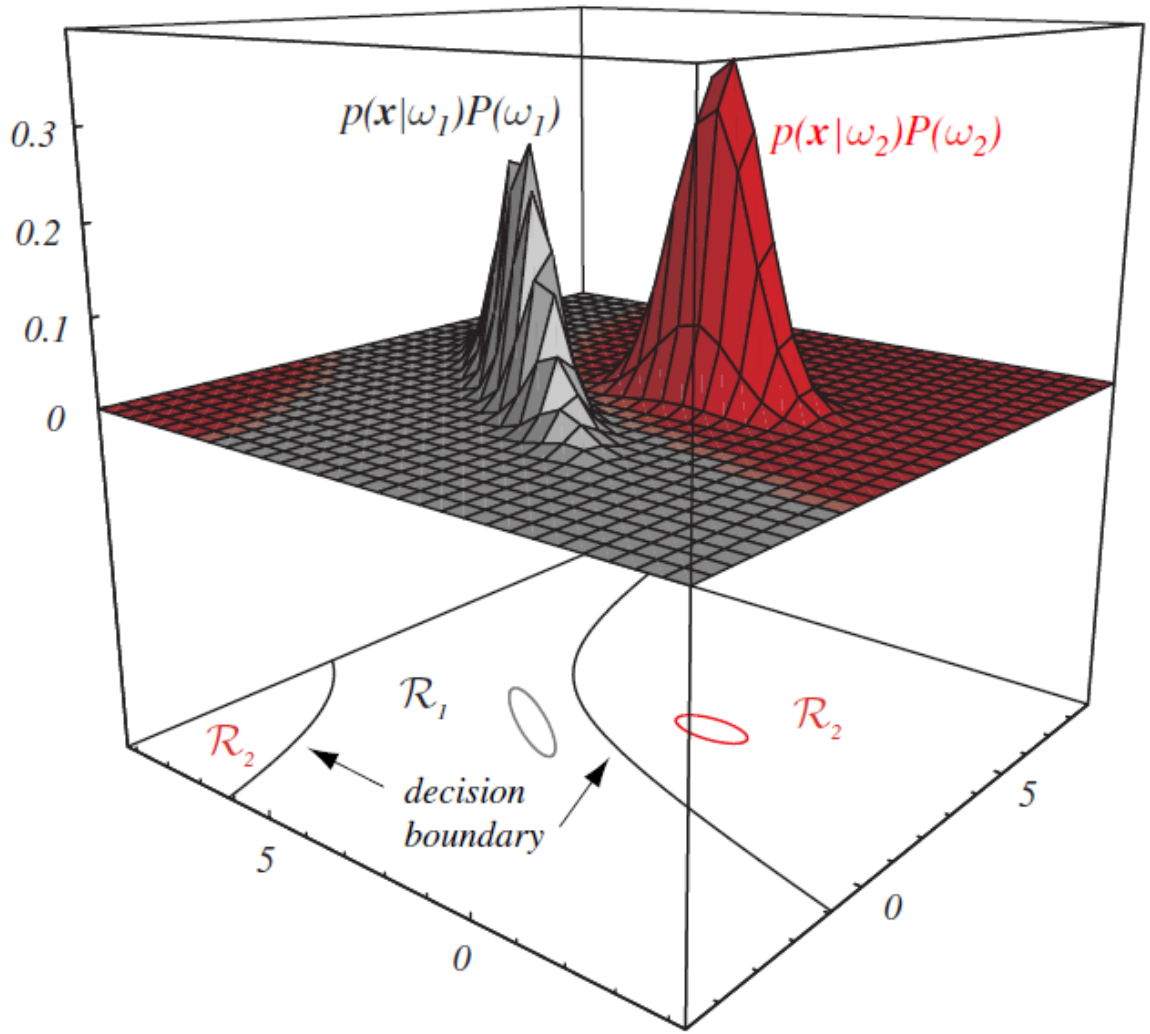
Visualizing discriminants: decision regions

- The effect of any decision rule is to divide the feature space into decision regions.
- Denote a decision region $\mathcal{R}_i$ for ω_i.
- One not necessarily connected region is created for each category and assignments is according to:

$$\text{If } g_i(\mathbf{x}) > g_j(\mathbf{x}) \quad \forall j \neq i, \text{ then } \mathbf{x} \text{ is in } \mathcal{R}_i \tag{34}$$

- Decision boundaries separate the regions; they are ties among the discriminant functions.

Visualizing discriminants: decision regions



Two-category discriminants / Dichotomizers

- In the two-category case, one considers single discriminant

$$g(\mathbf{x}) = g_1(\mathbf{x}) - g_2(\mathbf{x}) \tag{35}$$

- What is a suitable decision rule?
- The following simple rule is then used:

$$\text{Decide } \omega_1 \text{ if } g(\mathbf{x}) > 0;\ \text{otherwise decide } \omega_2 \tag{36}$$

- Various manipulations of the discriminant:

$$g(\mathbf{x}) = P(\omega_1|\mathbf{x}) - P(\omega_2|\mathbf{x}) \tag{37}$$

$$g(\mathbf{x}) = \ln \frac{p(\mathbf{x}|\omega_1)}{p(\mathbf{x}|\omega_2)} + \ln \frac{P(\omega_1)}{P(\omega_2)} \tag{38}$$

Acknowledgments

- **Jason CORSO**
  Associate Professor of Electrical Engineering and Computer Science
  University of Michigan, U.S.A.