MACHINE LEARNING

CHAPTER 1: PRELIMINARY

Learning Objectives

1、What is pattern recognition and machine learning?

2、What are curve fitting and regularization?

3、What are ML and MAP Bayesian inferences?

4、How to deal with the curse of dimensionality?

5、What is the relationship between decision theory and machine learning?

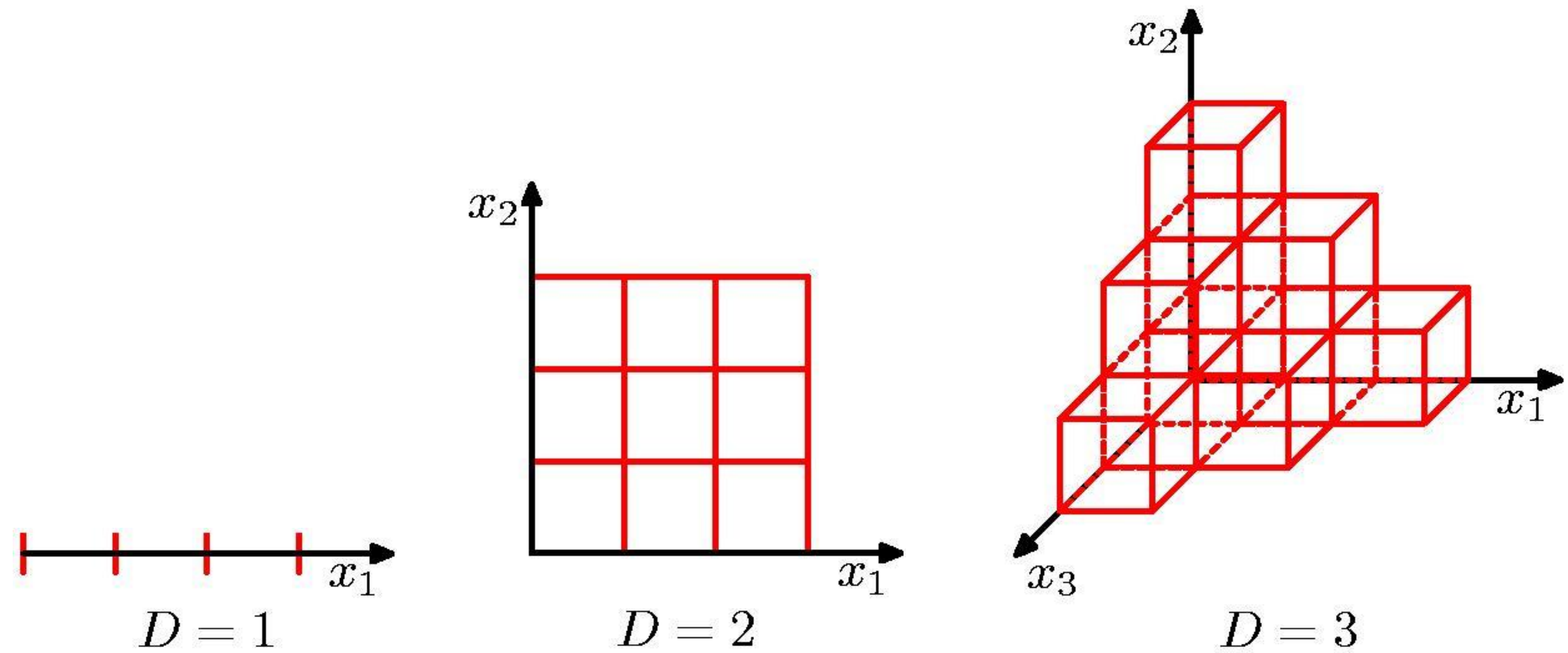
6、What are generative and discriminative models?

7、How to use entropy、KL divergence and mutual information for machine learning?

Outlines

- Pattern Recognition and Machine Learning
- Curve Fitting and Regularization
- Probabilities and Gaussian Distributions
- Bayesian Inferences (ML and MAP)
- Curse of Dimensionality
- Decision Theory
- Entropy and Information

Curse of Dimensionality

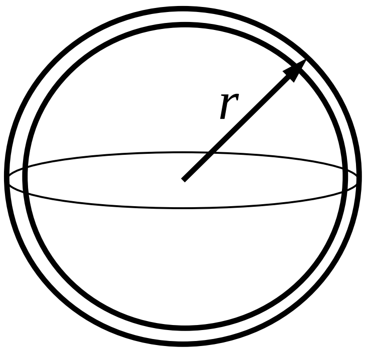


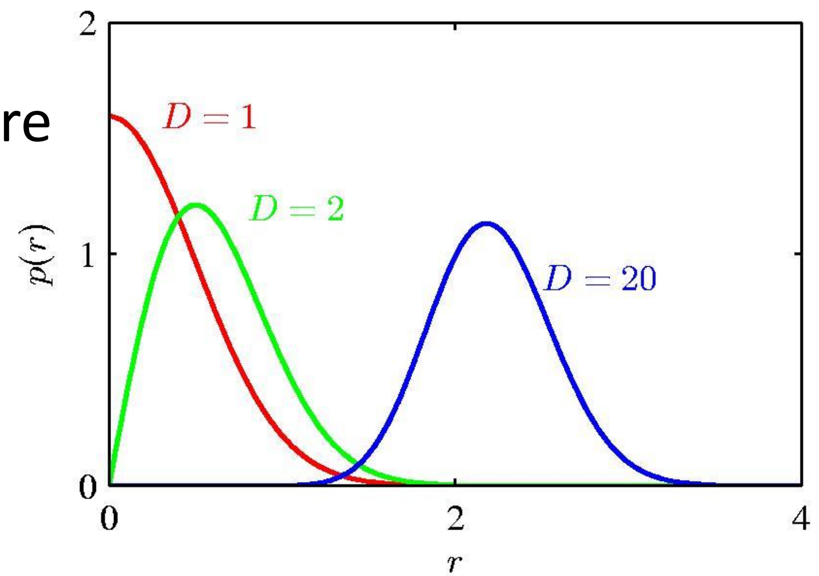
Curse of Dimensionality

Polynomial curve fitting, $M = 3$

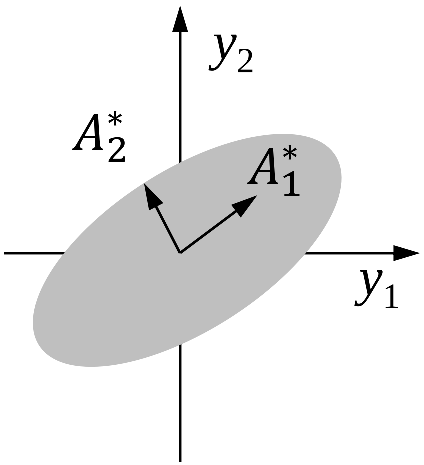
$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^{D} w_i x_i + \sum_{i=1}^{D}\sum_{j=1}^{D} w_{ij} x_i x_j + \sum_{i=1}^{D}\sum_{j=1}^{D}\sum_{k=1}^{D} w_{ijk} x_i x_j x_k$$

Gaussian Densities in
higher dimensions of a sphere





Reduction of Dimensionality (PCA)



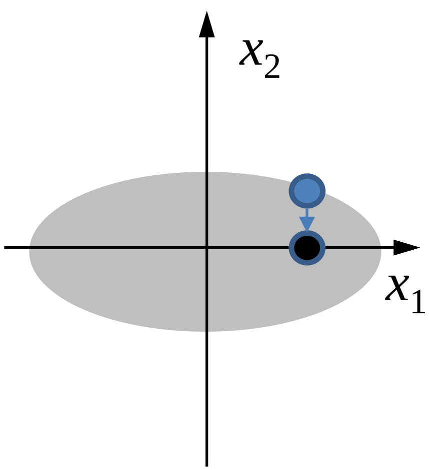
Basis

Data

Coefficients

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}$$

principal component analysis



$$\boxed{\max_{A_i} {A_i}^T \, COV(Y_n) A_i}$$

A : rotation (I x I)

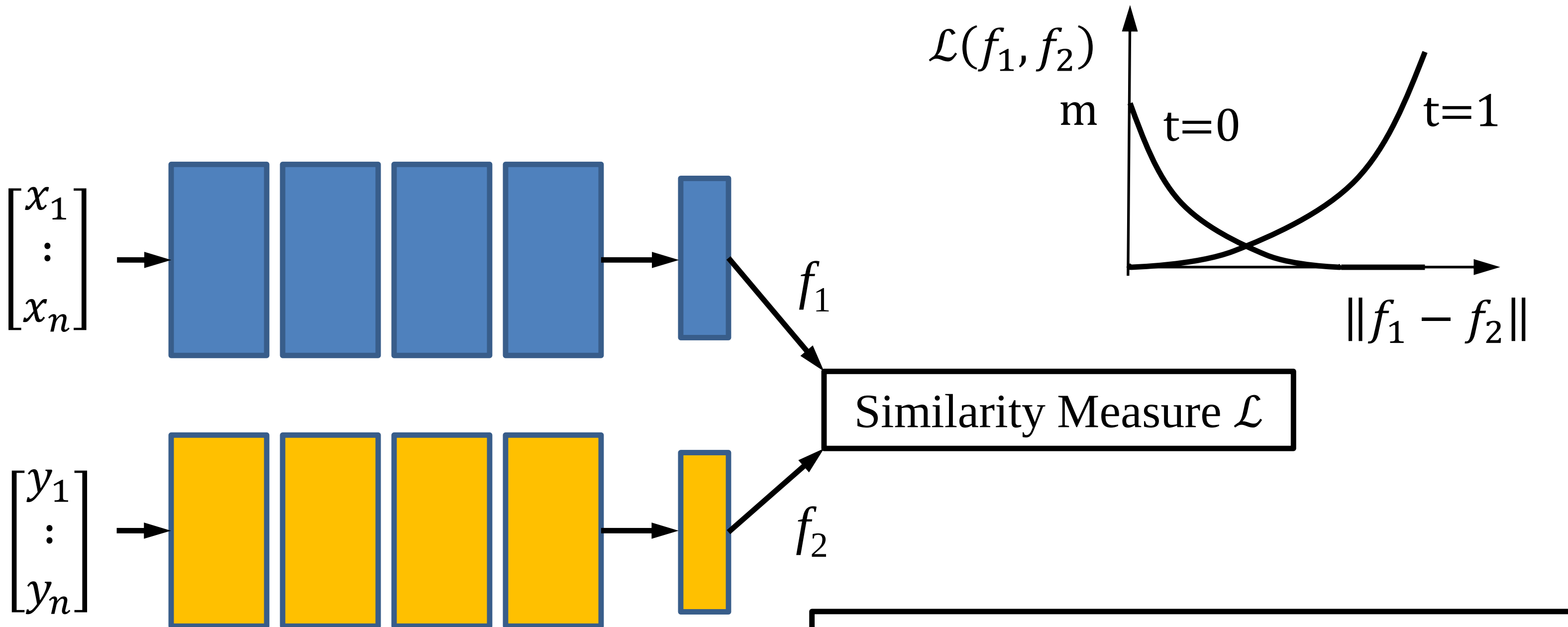
Y: data (I x N)

A_i^* : optimal solution eigenvector

$$A_i^{*T} COV(Y_n) A_i^* = \lambda_i$$

$$s.t. \quad {A_i}^T A_i = 1 \quad {A_i}^T A_j = 0 \quad E[Y_n] = \mathbf{0}$$

Feature Extraction (Contrastive Loss)

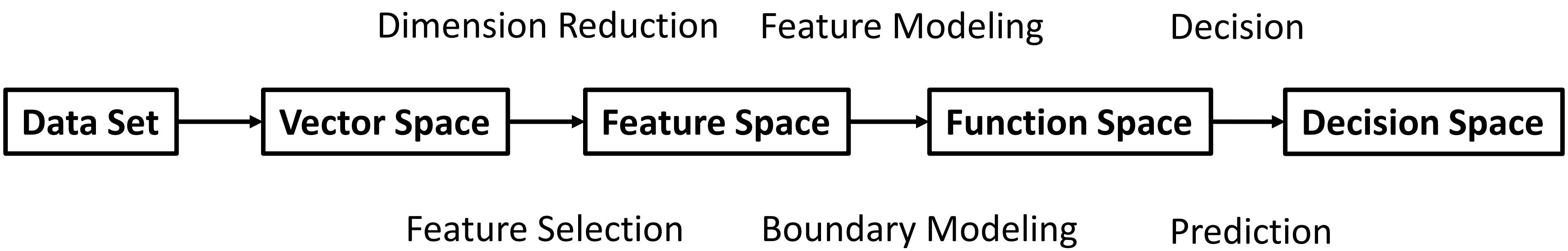


maximum distance between clusters

$$\mathcal{L}(f_1, f_2) = t\|f_1 - f_2\|^2 + (1 - t)[\mathrm{m} - \|f_1 - f_2\|^2]_+$$

t =1: two vectors belong to the same category; $[\ ]_+$: non-negative

Machine Learning Pipeline



Outlines

- Pattern Recognition and Machine Learning
- Curve Fitting and Regularization
- Probabilities and Gaussian Distributions
- Bayesian Inferences (ML and MAP)
- Curse of Dimensionality
- Decision Theory
- Entropy and Information

Decision Theory

Inference step

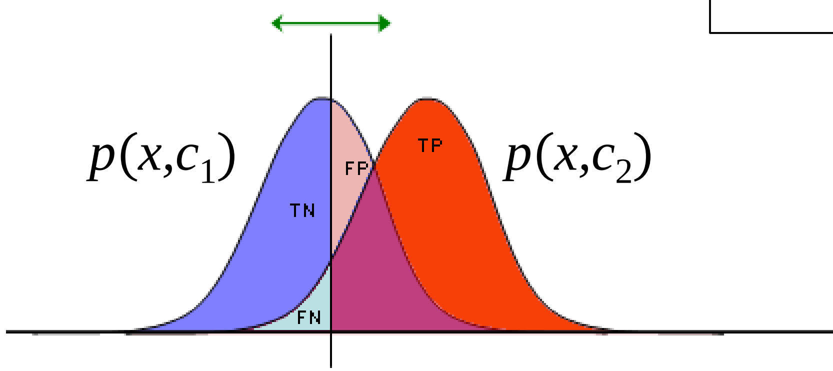
Determine either $p(t|\mathbf{x})$ or $p(\mathbf{x}, t)$.

Decision step

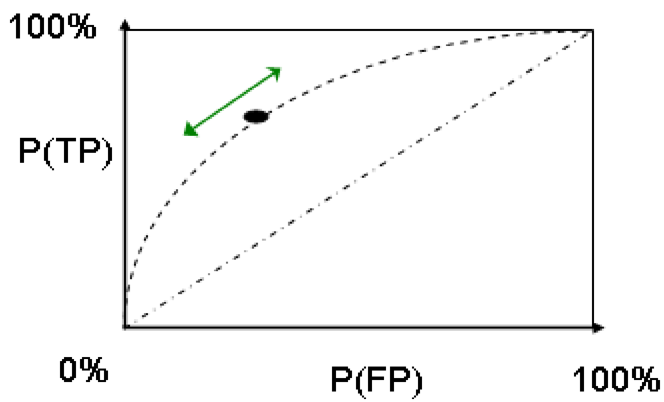
For given $\mathbf{x}$, determine optimal t.

Receiver Operating Characteristic Curve

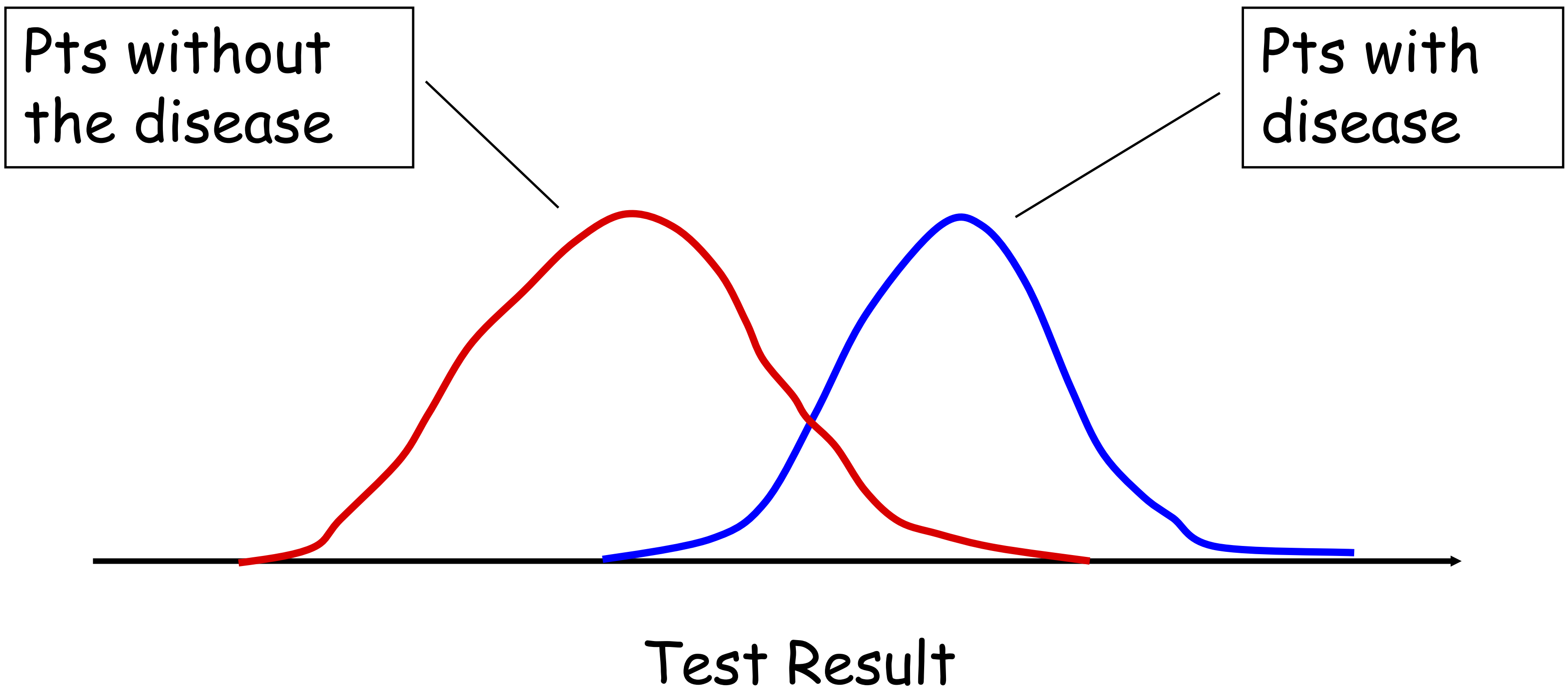
$$\begin{aligned} p(x) &= p(x,c_1) + p(x,c_2) \\ &= p(x|c_1)\, p(c_1) + p(x|c_2)\, p(c_2) \end{aligned}$$



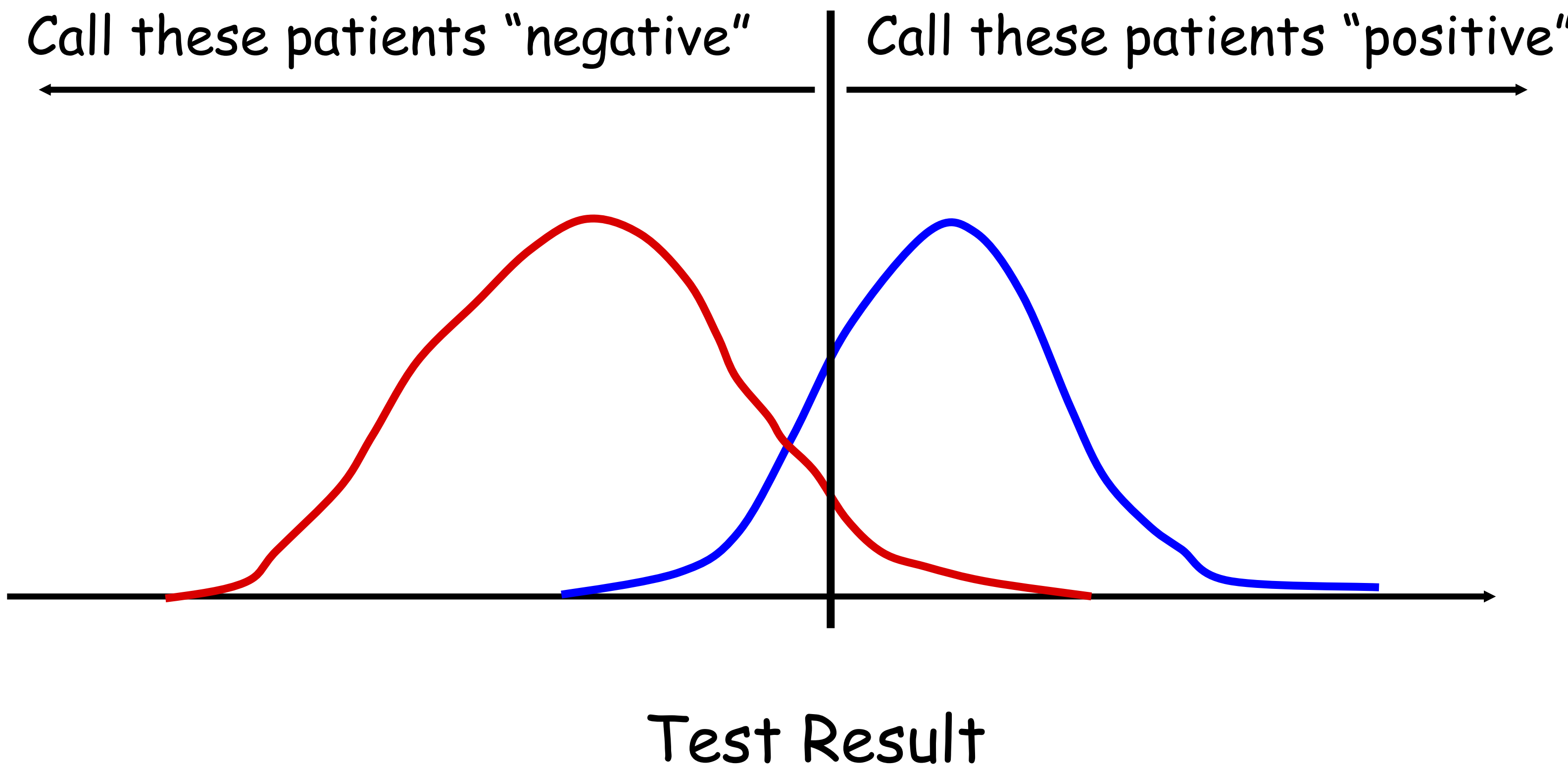
TP	FP
FN	TN
1	1



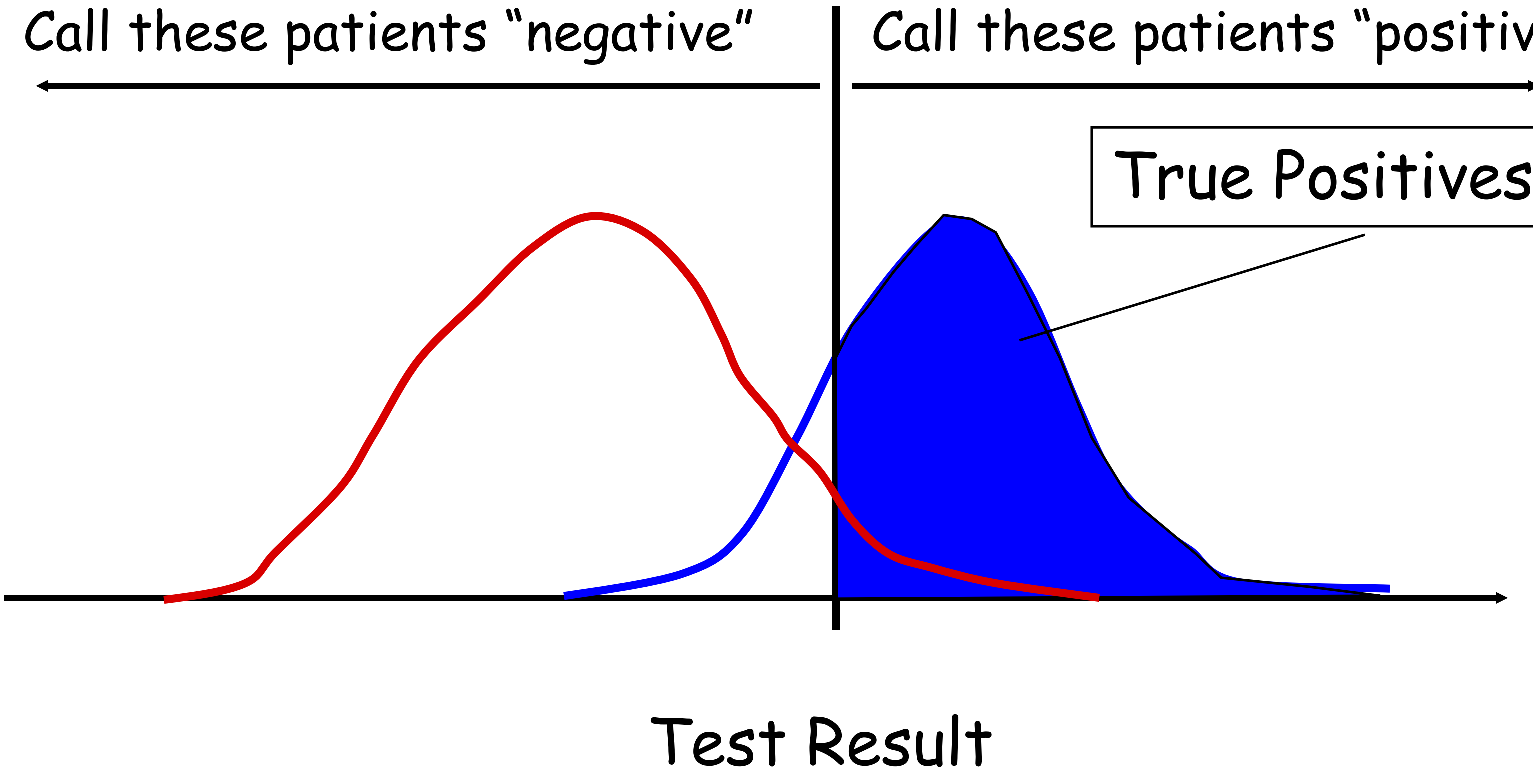
Bimodal Distribution (Data Model)



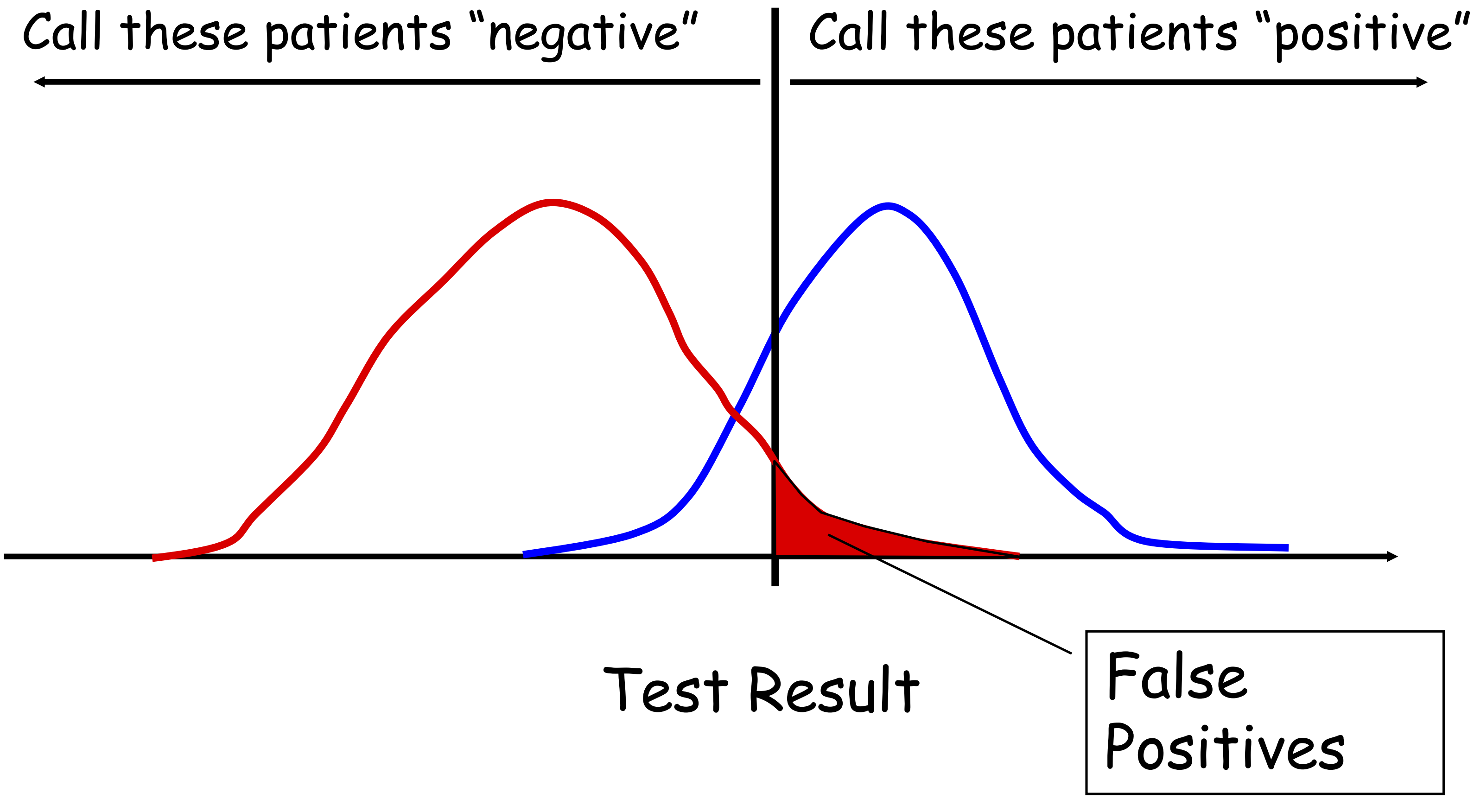
Decision Threshold (Boundary Model)



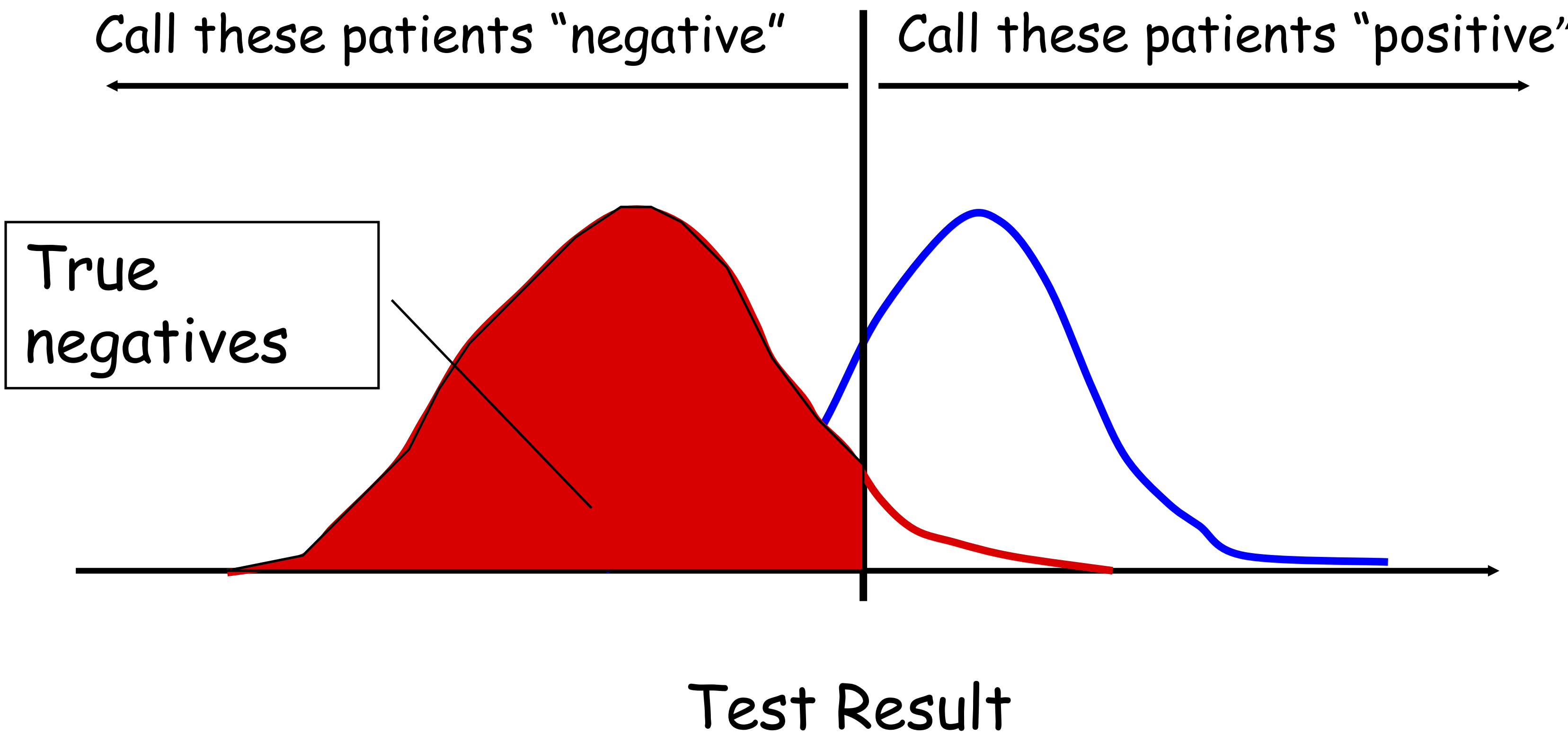
True Positive



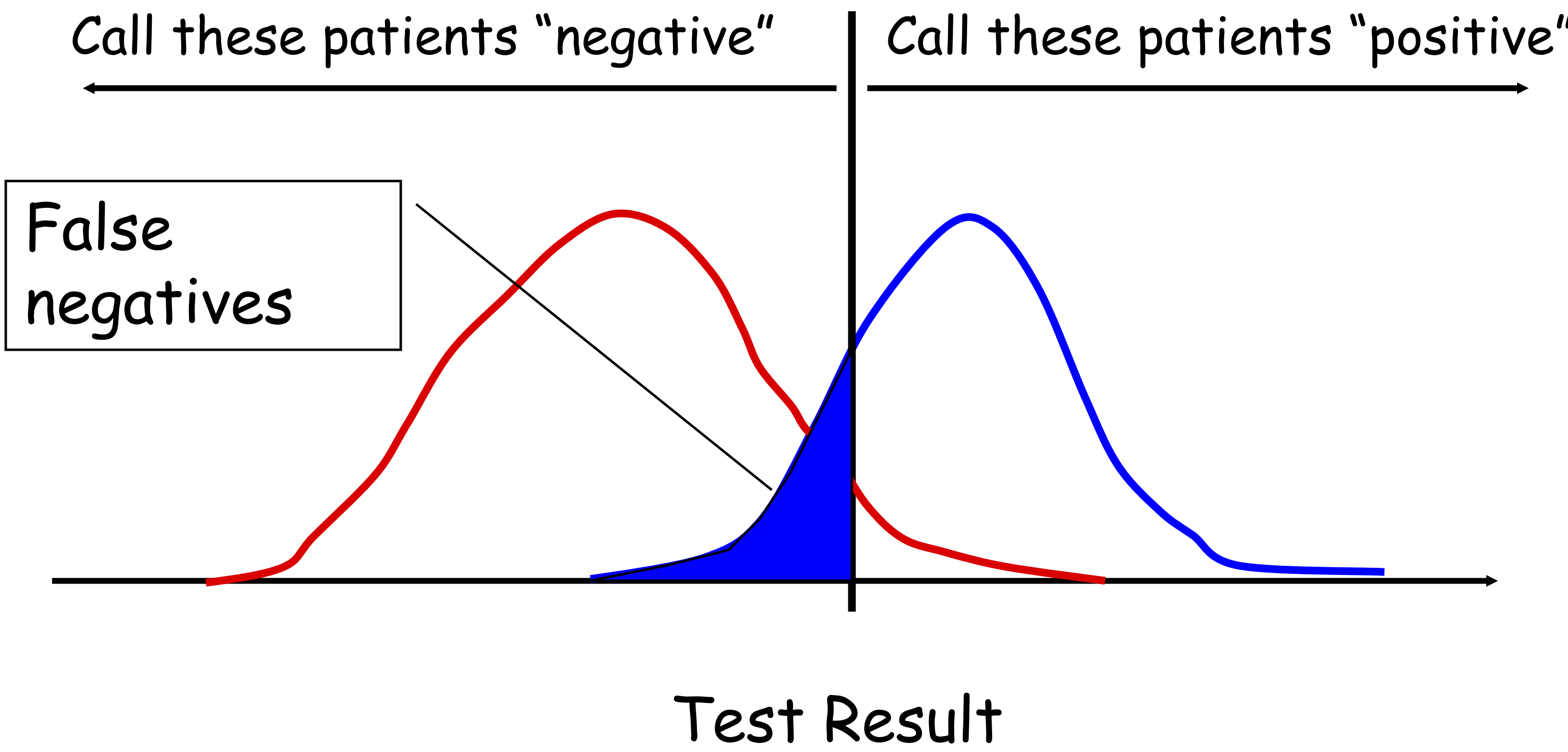
False Positive



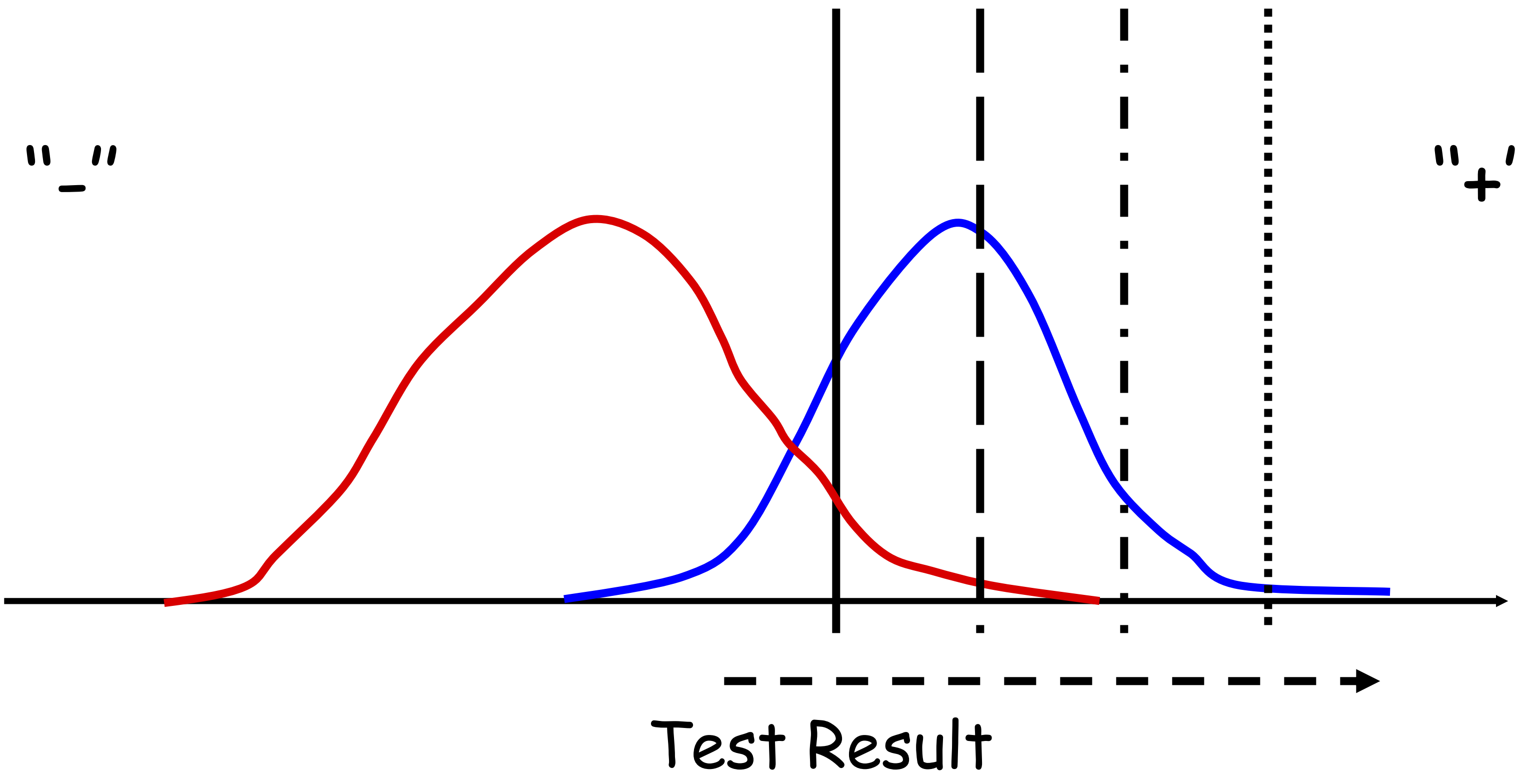
True Negative



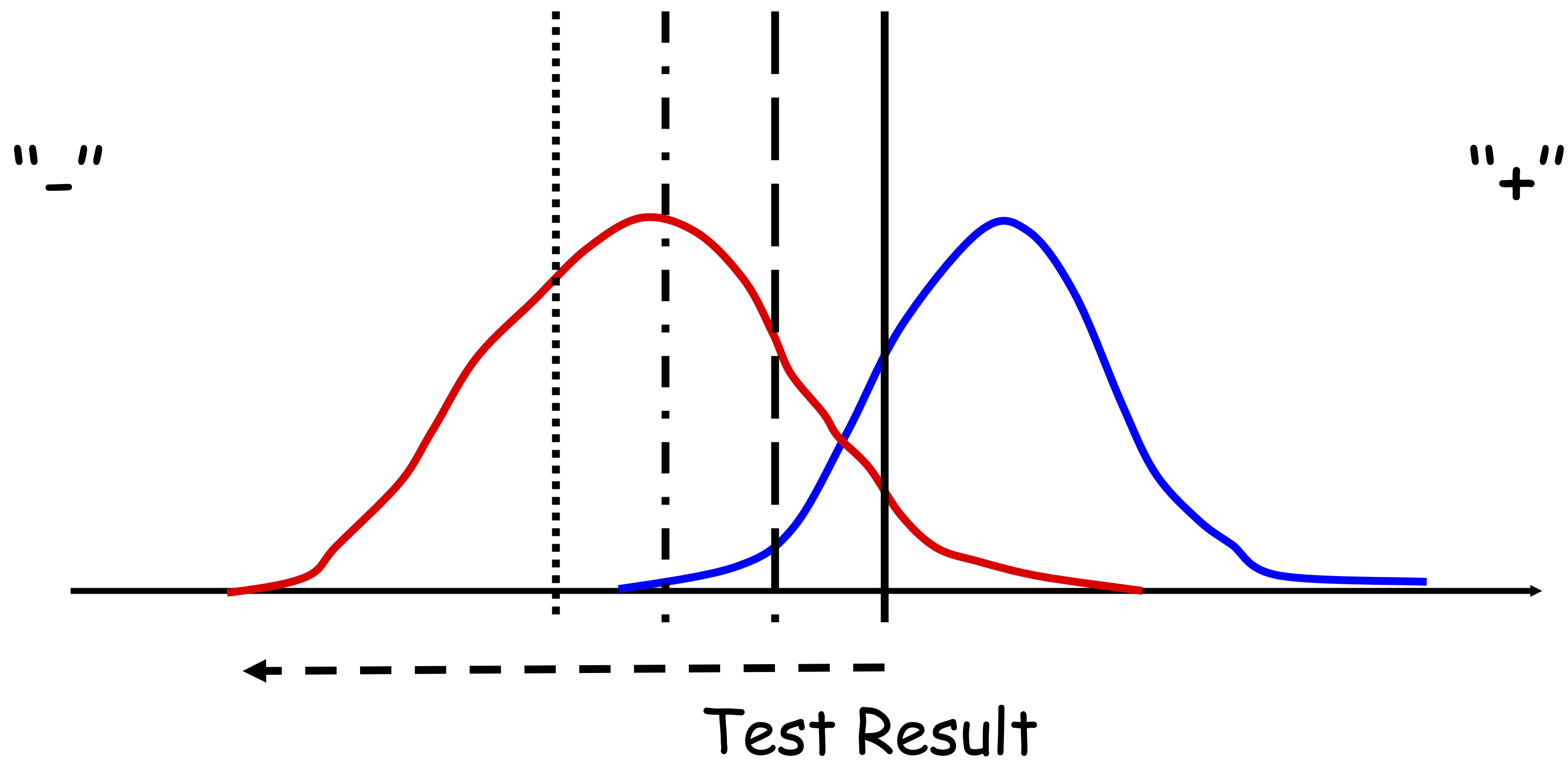
False Negative



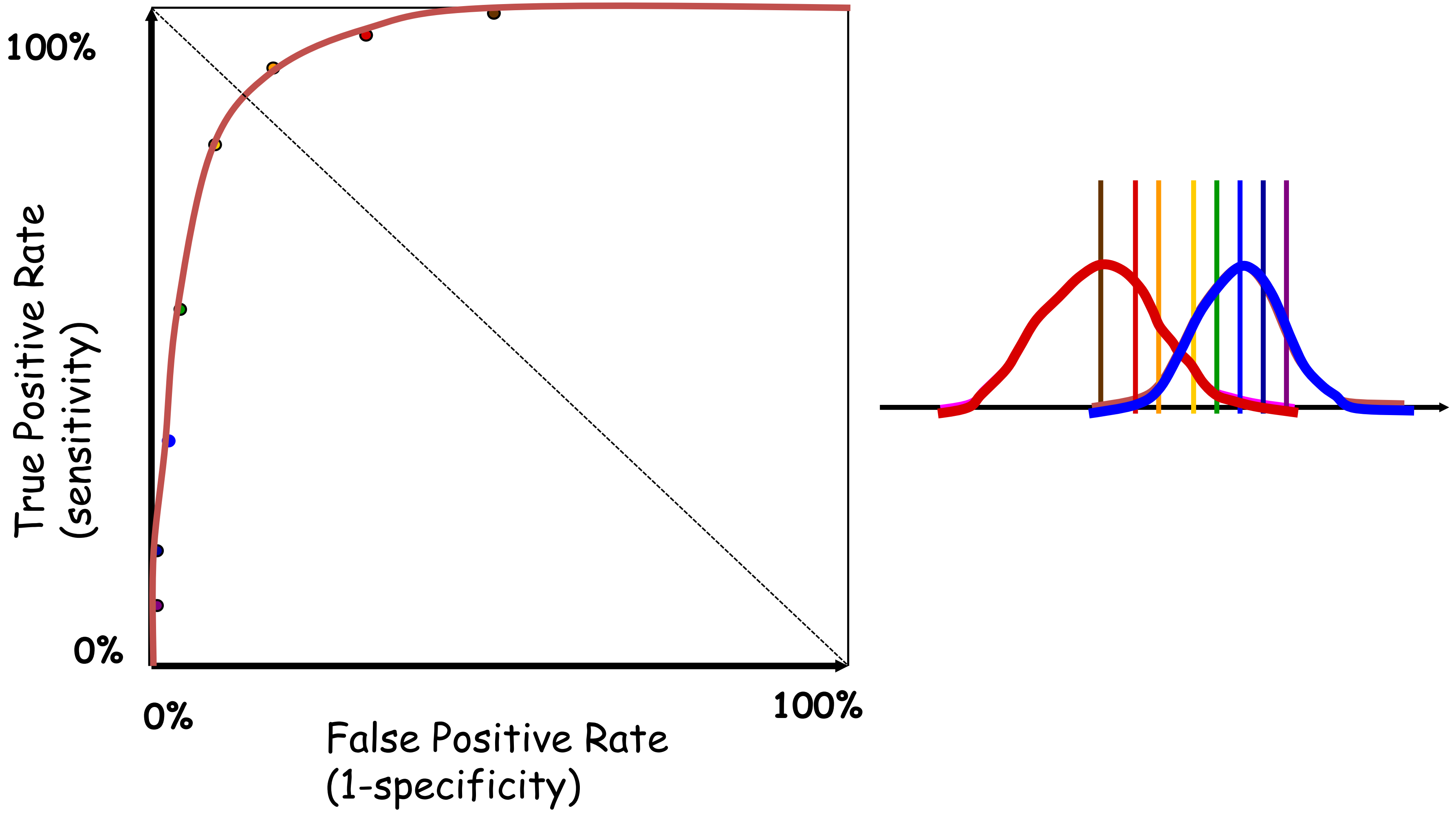
Moving the Threshold: Right



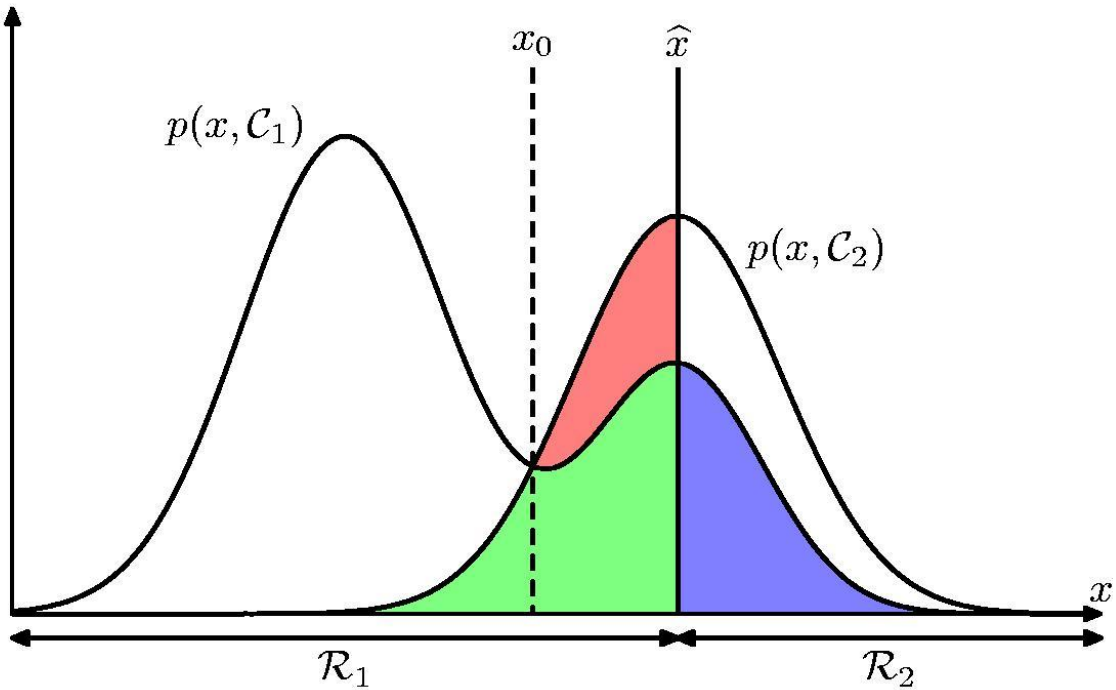
Moving the Threshold: Left



ROC Curve


100%
True Positive Rate
(sensitivity)
0%
0%
100%
False Positive Rate
(1-specificity)

Minimum Misclassification Rate



$$\begin{aligned} p(\text{mistake}) &= p(\mathbf{x} \in \mathcal{R}_1, \mathcal{C}_2) + p(\mathbf{x} \in \mathcal{R}_2, \mathcal{C}_1) \\ &= \int_{\mathcal{R}_1} p(\mathbf{x}, \mathcal{C}_2)\,\mathrm{d}\mathbf{x} + \int_{\mathcal{R}_2} p(\mathbf{x}, \mathcal{C}_1)\,\mathrm{d}\mathbf{x}. \end{aligned}$$

Minimum Expected Loss

Example: classify medical images as ‘cancer’ or ‘normal’

Decision

$$\text{Truth}\quad \begin{array}{cc} & \begin{array}{cc}\text{cancer} & \text{normal}\end{array} \\ \begin{array}{c}\text{cancer}\\ \text{normal}\end{array} & \begin{pmatrix} 0 & 1000 \\ 1 & 0 \end{pmatrix}\end{array}$$

False Positive

False Negatives

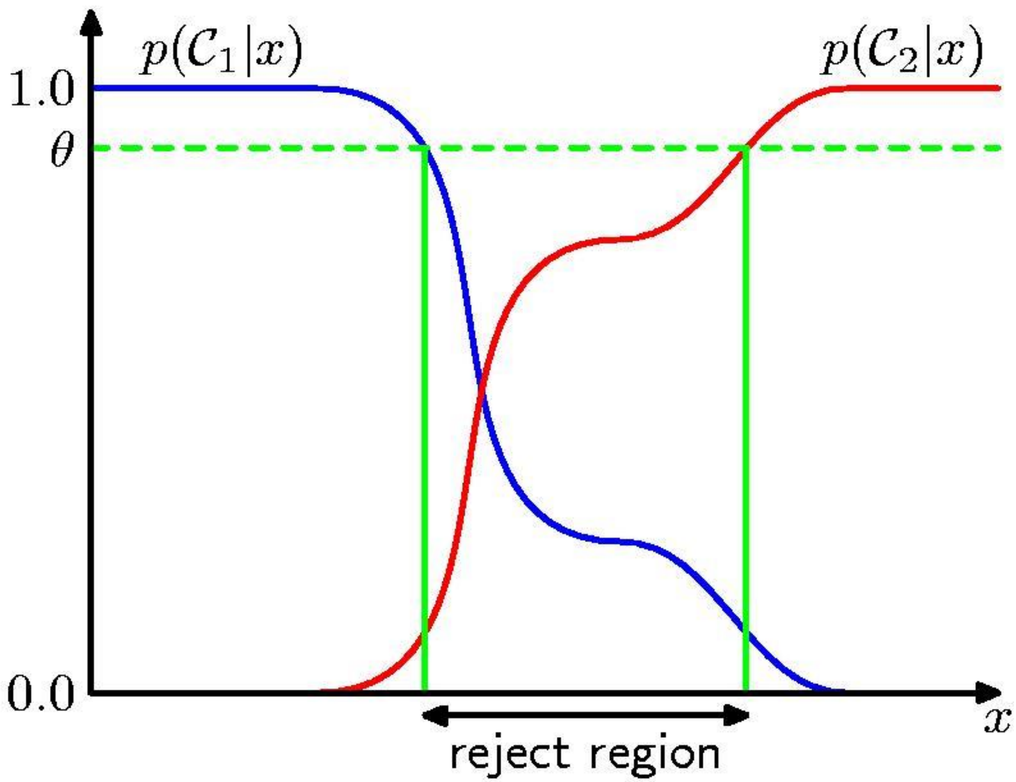
Minimum Expected Loss

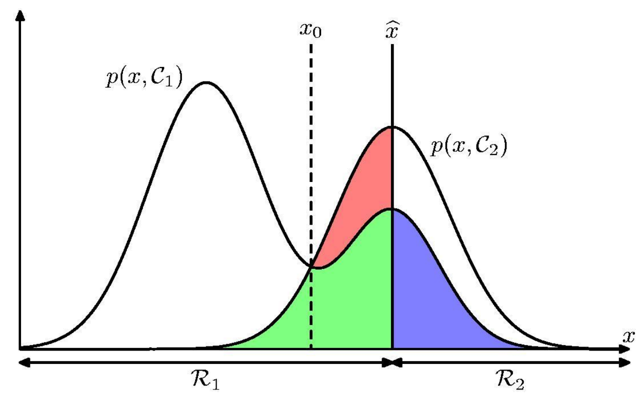
$$\mathbb{E}[L] = \sum_k \sum_j \int_{\mathcal{R}_j} L_{kj} p(\mathbf{x}, \mathcal{C}_k) \, \mathrm{d}\mathbf{x}$$

Regions $\mathcal{R}_j$ are chosen to minimize

$$\mathbb{E}[L] = \sum_k L_{kj} p(\mathcal{C}_k | \mathbf{x})$$

Reject Option





Posterior Probability

The posterior probability of class 1 is given by:

$$p(C_1 \mid \mathbf{x}) = \frac{p(C_1)\,p(\mathbf{x} \mid C_1)}{p(C_1)\,p(\mathbf{x} \mid C_1) + p(C_0)\,p(\mathbf{x} \mid C_0)} = \frac{1}{1+e^{-z}} = \sigma(z)$$

$$where \quad z = \ln \frac{p(C_1)\,p(\mathbf{x} \mid C_1)}{p(C_0)\,p(\mathbf{x} \mid C_0)} = \boxed{\ln \frac{p(C_1 \mid \mathbf{x})}{1 - p(C_1 \mid \mathbf{x})}}$$

z is called the logit and is given by the log odds

Why Separate Inference and Decision?

- Minimizing risk (loss matrix may change over time)
- Reject option
- Unbalanced class priors
- Combining models

Decision Theory for Regression

Inference step

Determine $p(\mathbf{x}, t)$.

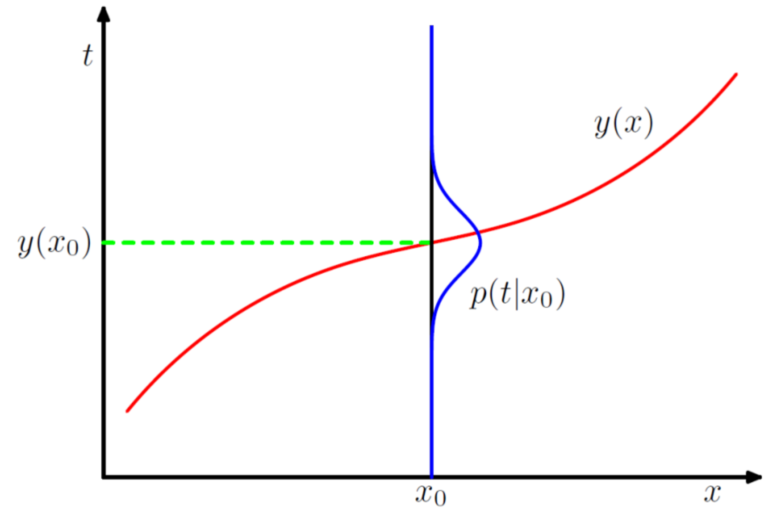
Decision step

For given x, make optimal prediction, $y(\mathrm{x})$, for t.

Loss function: $$\mathbb{E}[L] = \iint L(t, y(\mathbf{x}))p(\mathbf{x}, t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}t$$

The Expected Squared Loss Function

$$\mathbb{E}[L] = \iint \{y(\mathbf{x}) - t\}^2 p(\mathbf{x}, t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}t$$



$$\begin{aligned}\{y(\mathbf{x}) - t\}^2 &= \{y(\mathbf{x}) - \mathbb{E}[t|\mathbf{x}] + \mathbb{E}[t|\mathbf{x}] - t\}^2 \\ &= \{y(\mathbf{x}) - \mathbb{E}[t|\mathbf{x}]\}^2 + 2\{y(\mathbf{x}) - \mathbb{E}[t|\mathbf{x}]\}\{\mathbb{E}[t|\mathbf{x}] - t\} + \{\mathbb{E}[t|\mathbf{x}] - t\}^2\end{aligned}$$

$$\mathbb{E}[L] = \int \{y(\mathbf{x}) - \mathbb{E}[t|\mathbf{x}]\}^2 p(\mathbf{x})\,\mathrm{d}\mathbf{x} + \int \mathrm{var}[t|\mathbf{x}]\, p(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

$\Rightarrow$ $y(\mathbf{x}) = \mathbb{E}[t|\mathbf{x}]$

predictor

noise

$y(x)$: an estimator of the mean of t for given $\mathbf{x}$

https://stats.stackexchange.com/questions/228561/loss-functions-for-regression-proof

Generative vs Discriminative

Generative approach:

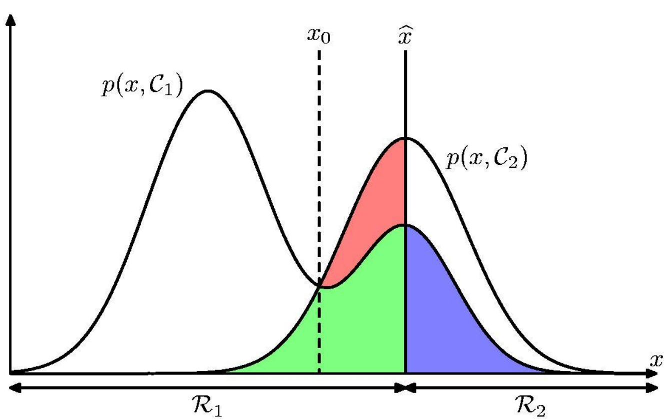
Model $p(t, \mathbf{x}) = p(\mathbf{x}|t)p(t)$

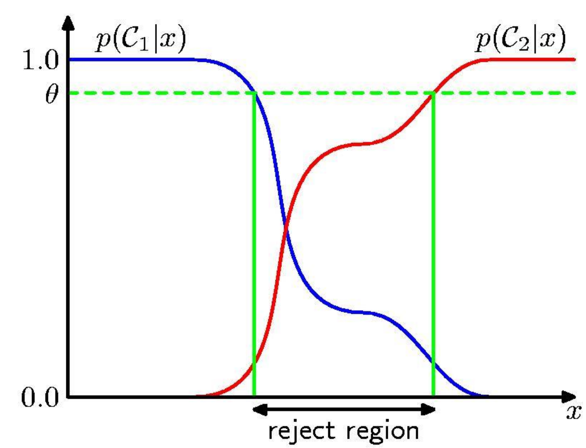
Use Bayes' theorem $p(t|\mathbf{x}) = \dfrac{p(\mathbf{x}|t)p(t)}{p(\mathbf{x})}$

Discriminative approach:

Model $p(t|\mathbf{x})$ directly

t : category





Outlines

- Pattern Recognition and Machine Learning
- Curve Fitting and Regularization
- Probabilities and Gaussian Distributions
- Bayesian Inferences (ML and MAP)
- Curse of Dimensionality
- Decision Theories
- Entropy and Information

Entropy

$$\mathrm{H}[x] = -\sum_{x} p(x) \log_2 p(x)$$

Important quantity in

- coding theory
- statistical physics
- machine learning

Entropy

Coding theory: x discrete with 8 possible states; how many bits to transmit the state of x?

All states equally likely

$$\mathrm{H}[x] = -8 \times \frac{1}{8} \log_2 \frac{1}{8} = 3 \text{ bits}.$$

Entropy

x	a	b	c	d	e	f	g	h
$p(x)$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{64}$	$\frac{1}{64}$	$\frac{1}{64}$	$\frac{1}{64}$
code	0	10	110	1110	111100	111101	111110	111111

$$\begin{aligned} \mathrm{H}[x] &= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{8}\log_2\frac{1}{8} - \frac{1}{16}\log_2\frac{1}{16} - \frac{4}{64}\log_2\frac{1}{64} \\ &= 2 \text{ bits} \end{aligned}$$

$$\begin{aligned} \text{average code length} &= \frac{1}{2}\times 1 + \frac{1}{4}\times 2 + \frac{1}{8}\times 3 + \frac{1}{16}\times 4 + 4\times\frac{1}{64}\times 6 \\ &= 2 \text{ bits} \end{aligned}$$

Entropy

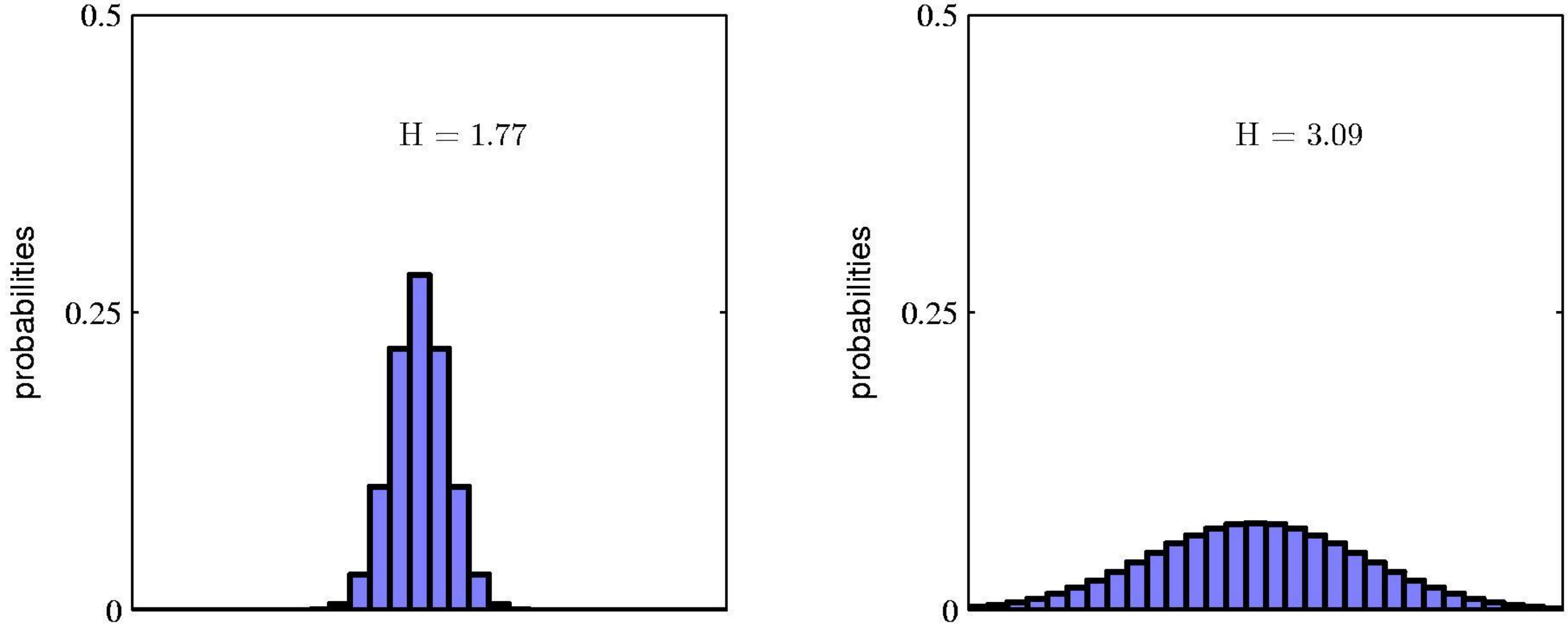
In how many ways can N identical objects be allocated M bins?

$$W = \frac{N!}{\prod_i n_i!}$$

$$\mathrm{H} = \frac{1}{N}\ln W \simeq -\lim_{N\to\infty}\sum_i \left(\frac{n_i}{N}\right)\ln\left(\frac{n_i}{N}\right) = -\sum_i p_i \ln p_i$$

Entropy maximized when $\forall i : p_i = \frac{1}{M}$

Entropy



Differential Entropy

Put bins of width Δ along the real line

$$\lim_{\Delta \to 0} \left\{ -\sum_i p(x_i) \Delta \ln p(x_i) \right\} = -\int p(x) \ln p(x) \, \mathrm{d}x$$

Differential entropy maximized (for fixed σ^2) when

$$p(x) = \mathcal{N}(x|\mu, \sigma^2)$$

in which case

$$\mathrm{H}[x] = \frac{1}{2} \left\{ 1 + \ln(2\pi\sigma^2) \right\}.$$

Conditional Entropy

$$\mathrm{H}[\mathbf{y}|\mathbf{x}] = -\iint p(\mathbf{y}, \mathbf{x}) \ln p(\mathbf{y}|\mathbf{x}) \,\mathrm{d}\mathbf{y}\,\mathrm{d}\mathbf{x}$$

$$\mathrm{H}[\mathbf{x}, \mathbf{y}] = \mathrm{H}[\mathbf{y}|\mathbf{x}] + \mathrm{H}[\mathbf{x}]$$

The Kullback-Leibler Divergence

$$\mathrm{KL}(p\|q) = -\int p(\mathbf{x}) \underbrace{\ln q(\mathbf{x})}_{\text{Cross Entropy } \mathrm{C}(p||q)} \,\mathrm{d}\mathbf{x} - \left(-\int \underbrace{p(\mathbf{x}) \ln p(\mathbf{x})}_{\text{Entropy } \mathrm{H}(p)} \,\mathrm{d}\mathbf{x}\right)$$

$$= -\int p(\mathbf{x}) \ln \left\{ \frac{q(\mathbf{x})}{p(\mathbf{x})} \right\} \mathrm{d}\mathbf{x}$$

$$\mathrm{KL}(p\|q) \simeq \frac{1}{N} \sum_{n=1}^{N} \{ \underbrace{-\ln q(\mathbf{x}_n|\boldsymbol{\theta})}_{\text{Cross Entropy}} + \underbrace{\ln p(\mathbf{x}_n)}_{\text{Negative Entropy}} \}$$

$$\mathrm{KL}(p\|q) \geqslant 0 \qquad \mathrm{KL}(p\|q) \not\equiv \mathrm{KL}(q\|p)$$

KL divergence describes a distance between model p and model q

Cross Entropy for Machine Learning

Goal of Machine Learning： $p(\textit{ real data }) \approx p(\textit{ model } \mid \theta)$

we assume： $p(\textit{ training data }) \approx p(\textit{ real data })$

Operation of Machine Learning： $p(\textit{training data}) \approx p(\textit{model} \mid \theta)$

$$\boxed{\min_{\theta} \mathrm{KL}(p(\textit{ training data }) \,||\, p(\textit{model} \mid \theta))}$$

$$\Leftrightarrow \quad \boxed{\min_{\theta} \mathrm{C}(p(\textit{ training data }) \,||\, p(\textit{model} \mid \theta))} \quad \text{as } \mathrm{H}(p(\textit{ training data })) \text{ is fixed}$$

Cross Entropy for Machine Learning

$$\boxed{C(p(\,training\ data\,)\,||\,p(model\,|\,\theta)\,)}$$

Bernoulli model: $p(\,model\,|\,\theta\,) = \rho^{t}(1-\rho)^{1-t}$

Cross entropy : $C = -\frac{1}{N}\sum_n t_n \ln\rho + (1-t_n)\ln(1-\rho)$

t_n: *training data*

ρ: *model parameter*

Gaussian model: $p(\,model\,|\,\theta\,) \propto e^{-0.5(t-\mu)^2}$

Cross entropy : $C \propto \frac{1}{N}\sum_n (t_n - \mu)^2$

t_n: *training data*

μ: *model parameter*

Mutual Information

$$
\begin{aligned}
\mathrm{I}[\mathbf{x}, \mathbf{y}] &\equiv \mathrm{KL}(p(\mathbf{x}, \mathbf{y}) \| p(\mathbf{x}) p(\mathbf{y})) \\
&= -\iint p(\mathbf{x}, \mathbf{y}) \ln \left(\frac{p(\mathbf{x}) p(\mathbf{y})}{p(\mathbf{x}, \mathbf{y})} \right) \mathrm{d}\mathbf{x}\, \mathrm{d}\mathbf{y}
\end{aligned}
$$

$$
\mathrm{I}[\mathbf{x}, \mathbf{y}] = \mathrm{H}[\mathbf{x}] - \mathrm{H}[\mathbf{x}|\mathbf{y}] = \mathrm{H}[\mathbf{y}] - \mathrm{H}[\mathbf{y}|\mathbf{x}]
$$

Mutual information describes the degree of dependence between **x** and **y**

Information Gain

$\mathrm{H}[\mathbf{x}]$: uncertain of balls

$\mathrm{H}[\mathbf{x}|\mathbf{y}]$:
uncertain of balls after weighing once

$\mathbf{x}$: one ball lighter

$\mathbf{y}$: weighing once

$\mathbf{x}|\mathbf{y}$: one ball lighter after weighing once

$$\mathrm{I}[\mathbf{x},\mathbf{y}] = \mathrm{H}[\mathbf{x}] - \mathrm{H}[\mathbf{x}|\mathbf{y}] = \log_2 3$$

$$\mathrm{H}[\mathbf{x}] = \log_2 N$$

After weighing $\frac{\mathrm{N}}{3}$ times, all the uncertainties can be removed

Independent Signal Separation

Independent Component Analysis

Mixture Coefficients

Data

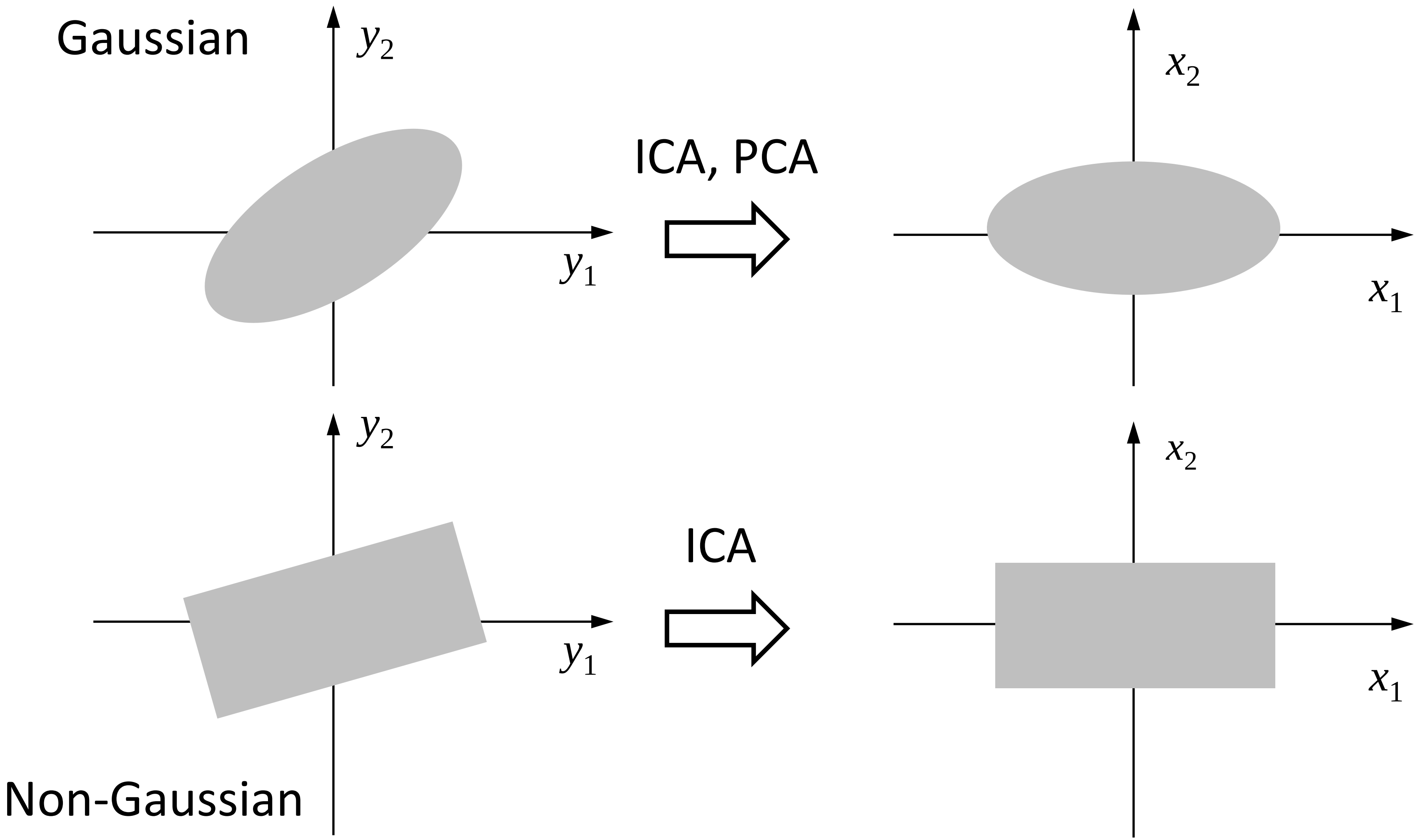
Components

$$Y = AX$$

$$\boxed{\min_{\mathrm{A}} I([X_1, X_2, \ldots, X_M] | A, Y)}$$

After optimization, the components of ***X*** become as much independent as possible

Illustration of ICA Operation



Summary

- Pattern Recognition
- Model Training and Regularization
- Probabilities and Gaussian Distributions
- Bayesian Inferences (ML and MAP)
- Curse of Dimensionality
- Decision Theory
- Entropy and Information