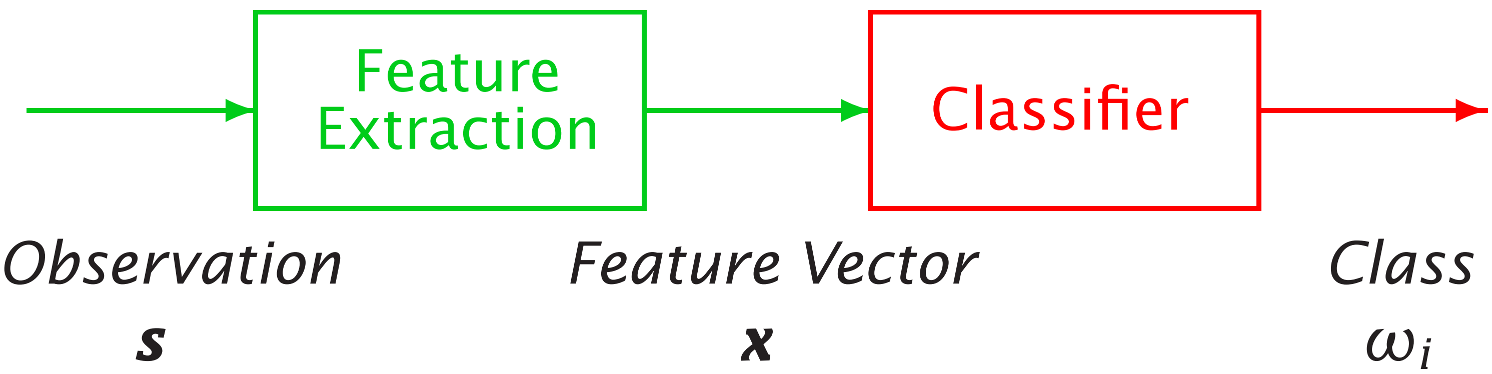
Pattern Classification

- Introduction
- Parametric classifiers
- Semi-parametric classifiers
- Dimensionality reduction
- Significance testing

Pattern Classification

Goal: To classify objects (or patterns) into categories (or classes)



Types of Problems:

1. *Supervised:* Classes are known beforehand, and data samples of each class are available
2. *Unsupervised:* Classes (and/or number of classes) are not known beforehand, and must be inferred from data

Probability Basics

- Discrete probability mass function (PMF): $P(\omega_i)$

$$\sum_i P(\omega_i) = 1$$

- Continuous probability density function (PDF): $p(x)$

$$\int p(x)dx = 1$$

- Expected value: $E(x)$

$$E(x) = \int xp(x)dx$$

Kullback-Liebler Distance

- Can be used to compute a distance between two probability mass distributions, $P(z_i)$, and $Q(z_i)$

$$D(P \parallel Q) = \sum_i P(z_i) \log \frac{P(z_i)}{Q(z_i)} \geq 0$$

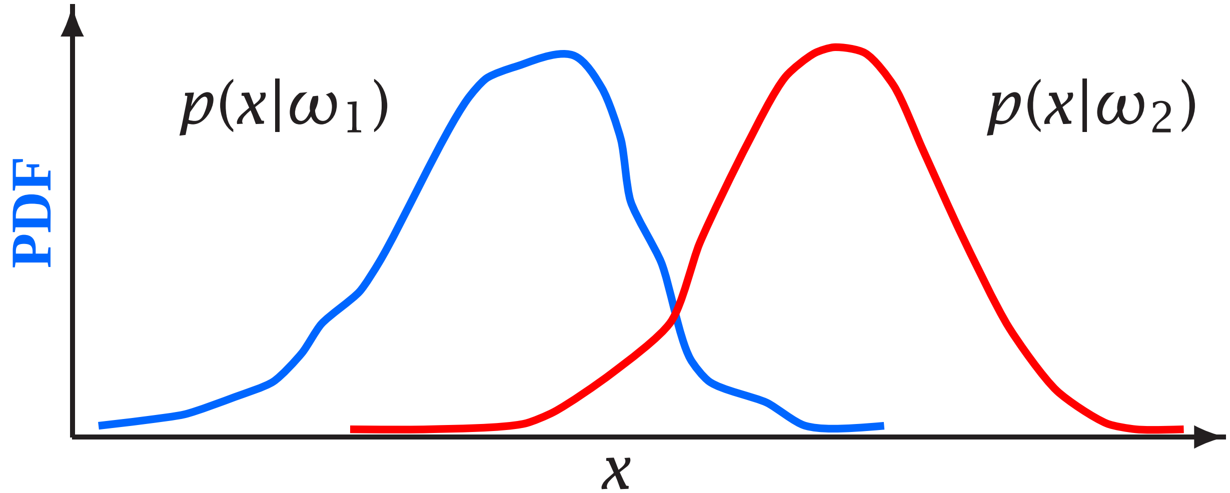
- Makes use of inequality $\log x \leq x - 1$

$$\sum_i P(z_i) \log \frac{Q(z_i)}{P(z_i)} \leq \sum_i P(z_i)\left(\frac{Q(z_i)}{P(z_i)} - 1\right) = \sum_i Q(z_i) - P(z_i) = 0$$

- Known as *relative entropy* in information theory
- The *divergence* of $P(z_i)$ and $Q(z_i)$ is the symmetric sum

$$D(P \parallel Q) + D(Q \parallel P)$$

Bayes Theorem



Define:	$\{\omega_i\}$	a set of M mutually exclusive classes
	$P(\omega_i)$	a priori probability for class ω_i
	$p(\mathbf{x}|\omega_i)$	PDF for feature vector $\mathbf{x}$ in class ω_i
	$P(\omega_i|\mathbf{x})$	a posteriori probability of ω_i given $\mathbf{x}$

From Bayes Rule: $$P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{p(\mathbf{x})}$$

where $$p(\mathbf{x}) = \sum_{i=1}^{M} p(\mathbf{x}|\omega_i)P(\omega_i)$$

Bayes Decision Theory

- The probability of making an error given $\boldsymbol{x}$ is:

$$P(error|\boldsymbol{x}) = 1 - P(\omega_i|\boldsymbol{x}) \qquad \text{if decide class } \omega_i$$

- To minimize $P(error|\boldsymbol{x})$ (and $P(error)$):

$$\text{Choose } \omega_i \text{ if } mathP(\omega_i|\boldsymbol{x}) > P(\omega_j|\boldsymbol{x}) \qquad \forall j \neq i$$

- For a two class problem this decision rule means:

$$\text{Choose } \omega_1 \text{ if } \frac{p(\boldsymbol{x}|\omega_1)P(\omega_1)}{p(\boldsymbol{x})} > \frac{p(\boldsymbol{x}|\omega_2)P(\omega_2)}{p(\boldsymbol{x})}; \text{ else } \omega_2$$

- This rule can be expressed as a likelihood ratio:

$$\text{Choose } \omega_1 \text{ if } \frac{p(\boldsymbol{x}|\omega_1)}{p(\boldsymbol{x}|\omega_2)} > \frac{P(\omega_2)}{P(\omega_1)}; \text{ else choose } \omega_2$$

Bayes Risk

- Define cost function λ_{ij} and conditional risk $R(\omega_i|\boldsymbol{x})$:
  - λ_{ij} is cost of classifying $\boldsymbol{x}$ as ω_i when it is really ω_j
  - $R(\omega_i|\boldsymbol{x})$ is the risk for classifying $\boldsymbol{x}$ as class ω_i

$$R(\omega_i|\boldsymbol{x}) = \sum_{j=1}^{M} \lambda_{ij} P(\omega_j|\boldsymbol{x})$$

- Bayes risk is the minimum risk which can be achieved:

  Choose ω_i if $R(\omega_i|\boldsymbol{x}) < R(\omega_j|\boldsymbol{x}) \qquad \forall j \neq i$

- Bayes risk corresponds to minimum $P(error|\boldsymbol{x})$ when
  - All errors have equal cost ($\lambda_{ij} = 1, \quad i \neq j$)
  - There is no cost for being correct ($\lambda_{ii} = 0$)

$$R(\omega_i|\boldsymbol{x}) = \sum_{j \neq i} P(\omega_j|\boldsymbol{x}) = 1 - P(\omega_i|\boldsymbol{x})$$

Discriminant Functions

- Alternative formulation of Bayes decision rule
- Define a discriminant function, $g_i(\boldsymbol{x})$, for each class ω_i

$$\text{Choose } \omega_i \text{ if } g_i(\boldsymbol{x}) > g_j(\boldsymbol{x}) \qquad \forall j \neq i$$

- Functions yielding identical classification results:

$$\begin{aligned} g_i(\boldsymbol{x}) &= P(\omega_i|\boldsymbol{x}) \\ &= p(\boldsymbol{x}|\omega_i)P(\omega_i) \\ &= \log p(\boldsymbol{x}|\omega_i) + \log P(\omega_i) \end{aligned}$$

- Choice of function impacts computation costs
- Discriminant functions partition feature space into decision regions, separated by decision boundaries

Density Estimation

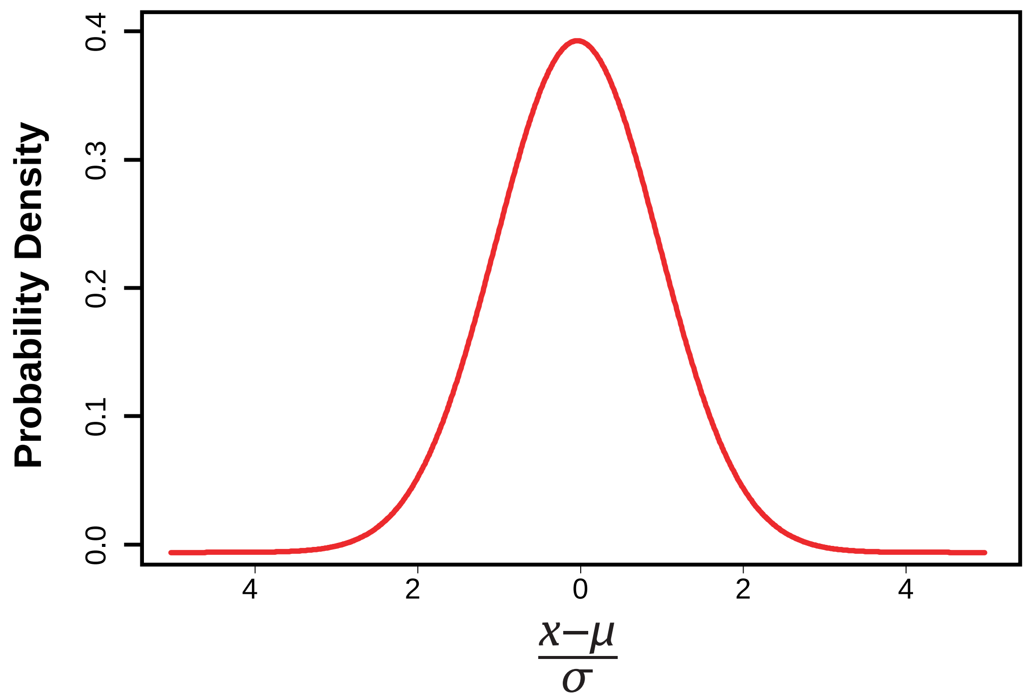
- Used to estimate the underlying PDF $p(\boldsymbol{x}|\omega_i)$
- Parametric methods:
  - Assume a specific functional form for the PDF
  - Optimize PDF parameters to fit data
- Non-parametric methods:
  - Determine the form of the PDF from the data
  - Grow parameter set size with the amount of data
- Semi-parametric methods:
  - Use a general class of functional forms for the PDF
  - Can vary parameter set independently from data
  - Use unsupervised methods to estimate parameters

Parametric Classifiers

- Gaussian distributions
- Maximum likelihood (ML) parameter estimation
- Multivariate Gaussians
- Gaussian classifiers

Gaussian Distributions

- Gaussian PDF's are reasonable when a feature vector can be viewed as perturbation around a reference



- Simple estimation procedures for model parameters
- Classification often reduced to simple distance metrics
- Gaussian distributions also called *Normal*

Gaussian Distributions: One Dimension

- One-dimensional Gaussian PDF's can be expressed as:

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \sim N(\mu, \sigma^2)$$

- The PDF is centered around the mean

$$\mu = E(x) = \int xp(x)dx$$

- The *spread* of the PDF is determined by the variance

$$\sigma^2 = E((x-\mu)^2) = \int (x-\mu)^2 p(x)dx$$

MIT

Maximum Likelihood Parameter Estimation

- Maximum likelihood parameter estimation determines an estimate $\hat{\theta}$ for parameter θ by maximizing the likelihood $L(\theta)$ of observing data $\mathcal{X} = \{x_1, \ldots, x_n\}$

$$\hat{\theta} = \arg\max_{\theta} \quad L(\theta)$$

- Assuming independent, identically distributed data

$$L(\theta) = p(\mathcal{X}|\theta) \; = p(x_1, \ldots, x_n|\theta) \; = \prod_{i=1}^{n} p(x_i|\theta)$$

- ML solutions can often be obtained via the derivative

$$\frac{\partial}{\partial \theta} L(\theta) \; = 0$$

- For Gaussian distributions $\log L(\theta)$ is easier to solve

Gaussian ML Estimation: One Dimension

- The maximum likelihood estimate for μ is given by:

$$
\begin{aligned}
L(\mu) &= \prod_{i=1}^{n} p(x_i|\mu) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \\
\log L(\mu) &= -\frac{1}{2\sigma^2} \sum_i (x_i-\mu)^2 - n \log \sqrt{2\pi}\sigma \\
\frac{\partial \log L(\mu)}{\partial \mu} &= \frac{1}{\sigma^2} \sum_i (x_i-\mu) = 0
\end{aligned}
$$

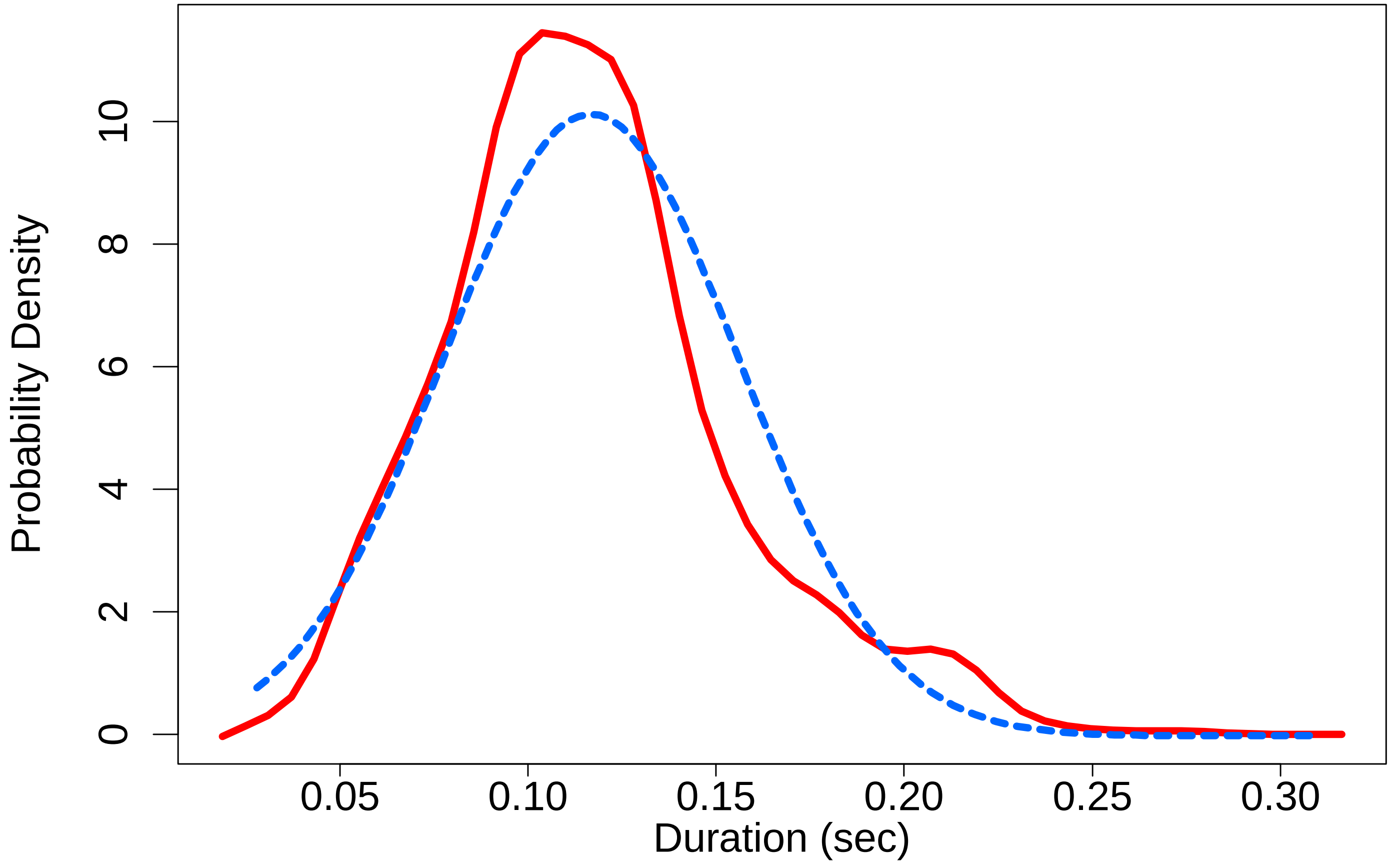
$$\hat{\mu} = \frac{1}{n} \sum_i x_i$$

- The maximum likelihood estimate for σ is given by:

$$\hat{\sigma}^2 = \frac{1}{n} \sum_i (x_i - \hat{\mu})^2$$

Gaussian ML Estimation: One Dimension

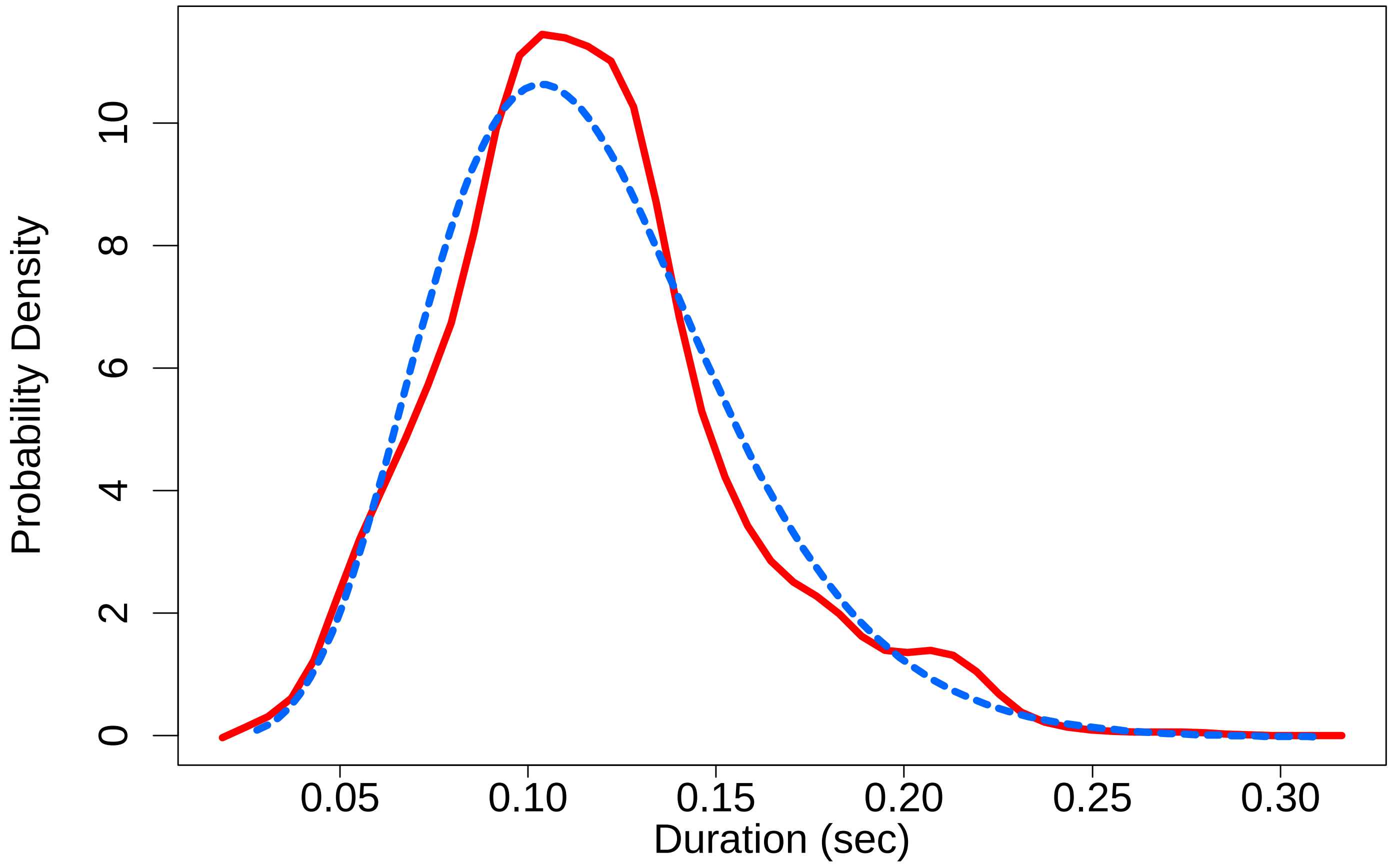
[s] Duration (1000 utterances, 100 speakers)



($\hat{\mu} \approx 120$ ms, $\hat{\sigma} \approx 40$ ms)

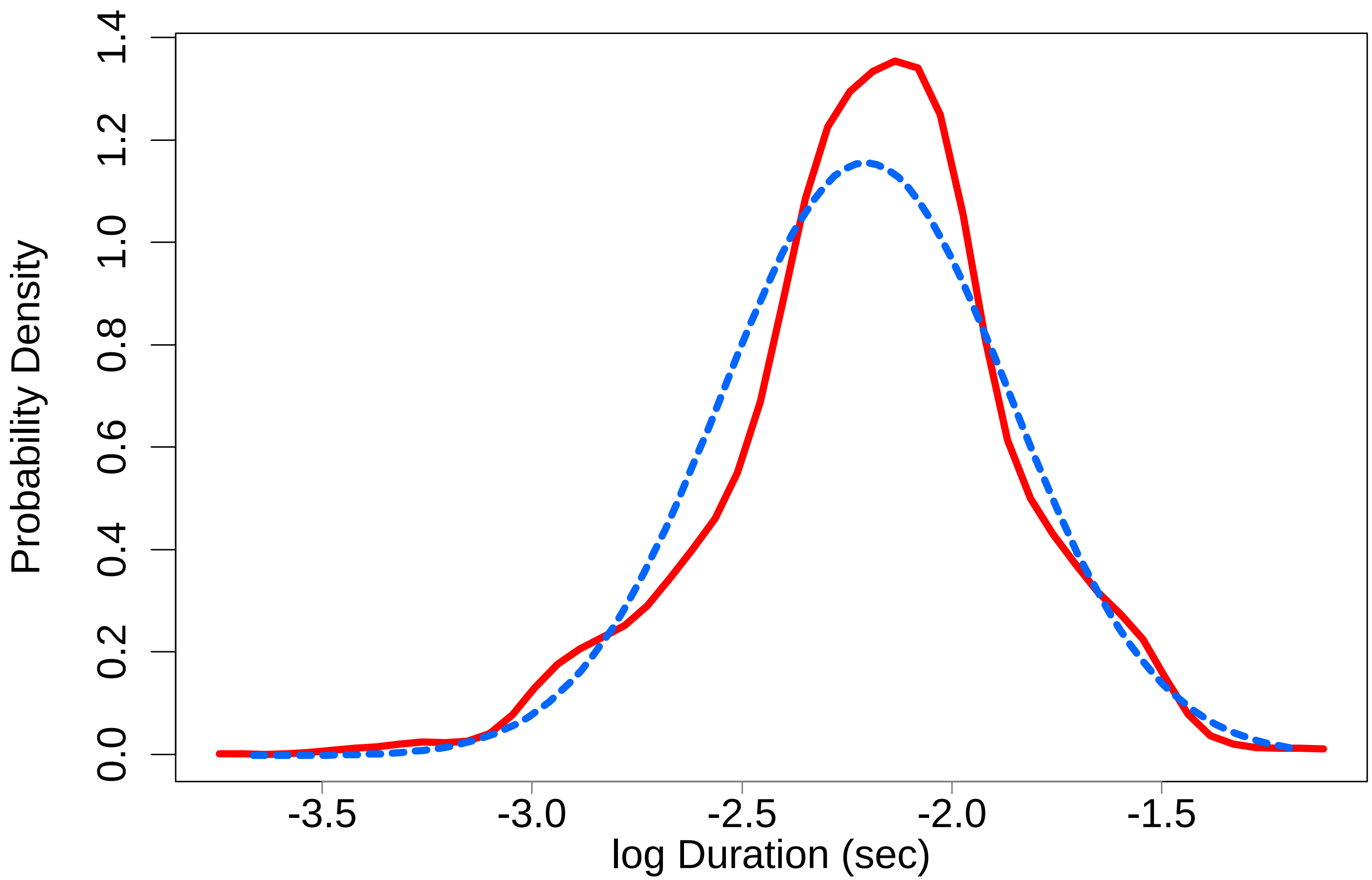
ML Estimation: Alternative Distributions

[s] Duration: Gamma Distribution



ML Estimation: Alternative Distributions

[s] Log Duration: Normal Distribution



Gaussian Distributions: Multiple Dimensions

- A multi-dimensional Gaussian PDF can be expressed as:

$$p(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^t \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

- d is the number of dimensions
- $\boldsymbol{x} = \{x_1, \ldots, x_d\}$ is the input vector
- $\boldsymbol{\mu} = E(\boldsymbol{x}) = \{\mu_1, \ldots, \mu_d\}$ is the mean vector
- $\boldsymbol{\Sigma} = E((\boldsymbol{x}-\boldsymbol{\mu})(\boldsymbol{x}-\boldsymbol{\mu})^t)$ is the covariance matrix with elements σ_{ij}, inverse $\boldsymbol{\Sigma}^{-1}$, and determinant $|\boldsymbol{\Sigma}|$
- $\sigma_{ij} = \sigma_{ji} = E((x_i - \mu_i)(x_j - \mu_j)) = E(x_i x_j) - \mu_i \mu_j$

Gaussian Distributions: Multi-Dimensional Properties

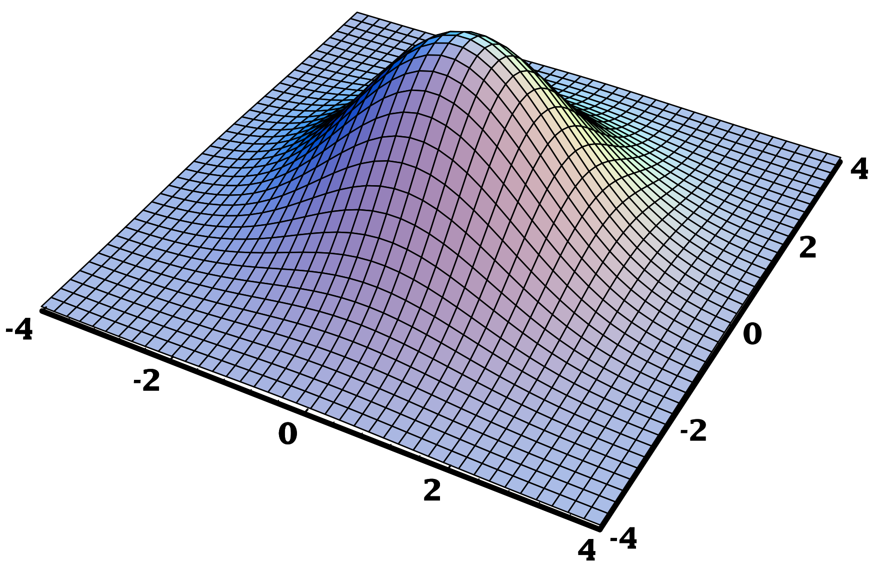
- If the i^{th} and j^{th} dimensions are statistically or linearly independent then $E(x_i x_j) = E(x_i)E(x_j)$ and $\sigma_{ij} = 0$
- If all dimensions are statistically or linearly independent, then $\sigma_{ij} = 0 \quad \forall i \neq j$ and $\boldsymbol{\Sigma}$ has non-zero elements only on the diagonal
- If the underlying density is Gaussian and $\boldsymbol{\Sigma}$ is a diagonal matrix, then the dimensions are statistically independent and

$$p(\boldsymbol{x}) = \prod_{i=1}^{d} p(x_i) \qquad p(x_i) \sim N(\mu_i, \sigma_{ii}) \qquad \sigma_{ii} = \sigma_i^2$$

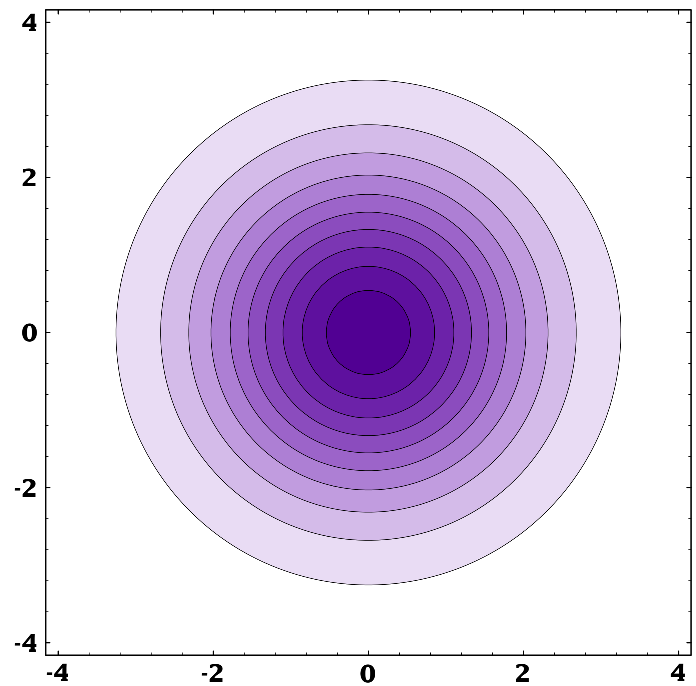
Diagonal Covariance Matrix: $\Sigma = \sigma^2 \boldsymbol{I}$

$$\Sigma = \begin{vmatrix} 2 & 0 \\ 0 & 2 \end{vmatrix}$$

3-Dimensional PDF



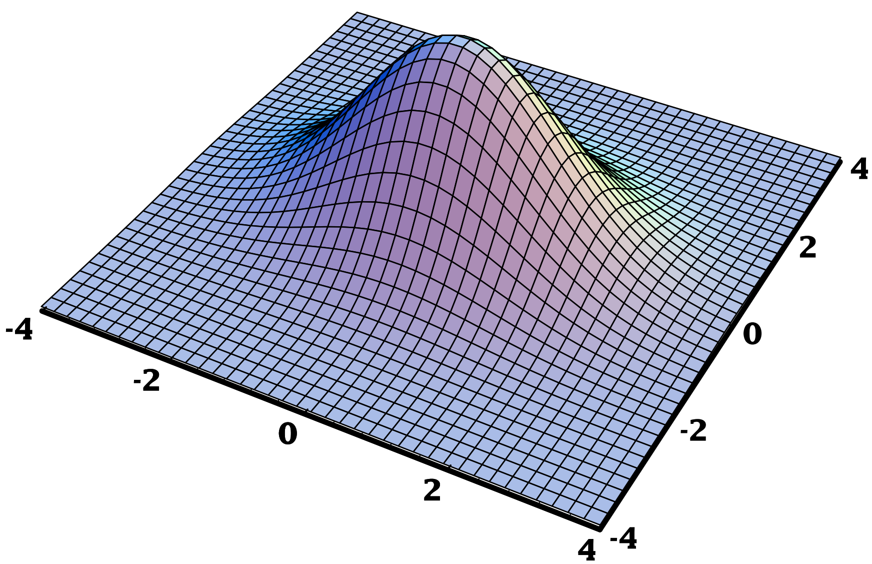
PDF Contour



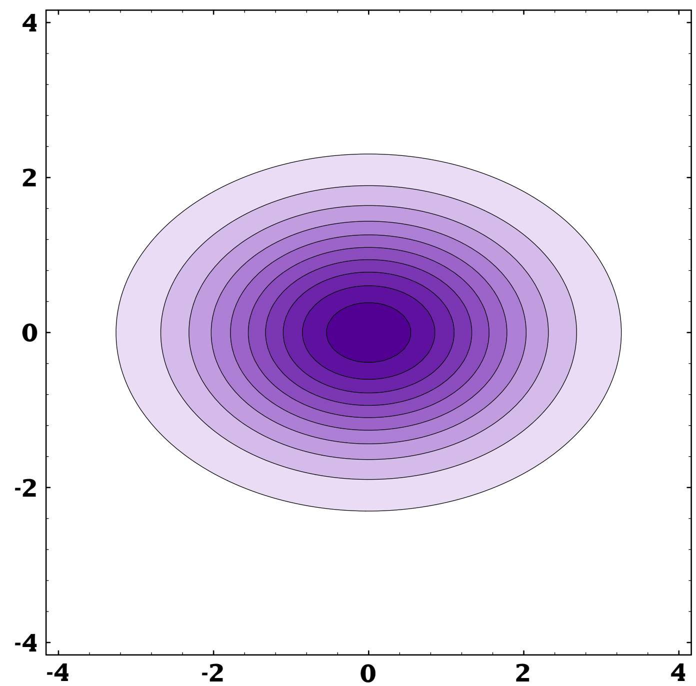
Diagonal Covariance Matrix: $\sigma_{ij} = 0 \quad \forall i \neq j$

$$\Sigma = \left| \begin{matrix} 2 & 0 \\ 0 & 1 \end{matrix} \right|$$

3-Dimensional PDF



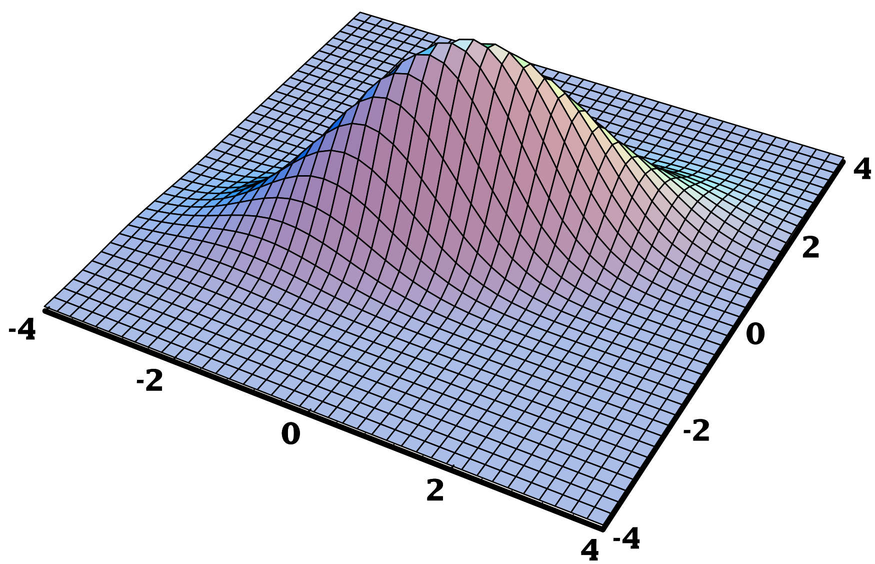
PDF Contour



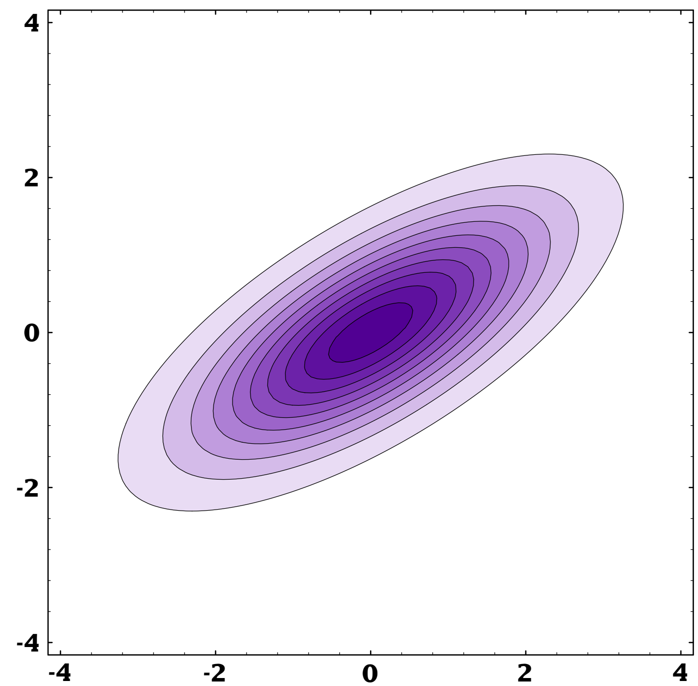
General Covariance Matrix: $\sigma_{ij} \neq 0$

$$\Sigma = \begin{vmatrix} 2 & 1 \\ 1 & 1 \end{vmatrix}$$

3-Dimensional PDF



PDF Contour



Multivariate ML Estimation

- The ML estimates for parameters $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_l\}$ are determined by maximizing the joint likelihood $L(\boldsymbol{\theta})$ of a set of i.i.d. data $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$

$$L(\boldsymbol{\theta}) = p(\mathcal{X}|\boldsymbol{\theta}) = p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n|\boldsymbol{\theta}) = \prod_{i=1}^{n} p(\boldsymbol{x}_i|\boldsymbol{\theta})$$

- To find $\hat{\boldsymbol{\theta}}$ we solve $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \boldsymbol{0}$, or $\nabla_{\boldsymbol{\theta}} \log L(\boldsymbol{\theta}) = \boldsymbol{0}$

$$\nabla_{\boldsymbol{\theta}} = \{\frac{\partial}{\partial \theta_1}, \cdots, \frac{\partial}{\partial \theta_l}\}$$

- The ML estimates of $\boldsymbol{\mu}$ and Σ are:

$$\hat{\boldsymbol{\mu}} = \frac{1}{n} \sum_i \boldsymbol{x}_i \qquad \hat{\Sigma} = \frac{1}{n} \sum_i (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})^t$$

Multivariate Gaussian Classifier

$$p(\boldsymbol{x}) \sim N(\boldsymbol{\mu}, \Sigma)$$

- Requires a mean vector $\boldsymbol{\mu}_i$, and a covariance matrix $\boldsymbol{\Sigma}_i$ for each of M classes $\{\omega_1, \cdots, \omega_M\}$
- The minimum error discriminant functions are of form:

$$g_i(\boldsymbol{x}) = \log P(\omega_i|\boldsymbol{x}) = \log p(\boldsymbol{x}|\omega_i) + \log P(\omega_i)$$

$$g_i(\boldsymbol{x}) = -\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_i)^t \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_i) - \frac{d}{2}\log 2\pi - \frac{1}{2}\log|\boldsymbol{\Sigma}_i| + \log P(\omega_i)$$

- Classification can be reduced to simple distance metrics for many situations

Gaussian Classifier: $\Sigma_i = \sigma^2 \boldsymbol{I}$

- Each class has the same covariance structure: statistically independent dimensions with variance σ^2
- The equivalent discriminant functions are:

$$g_i(\boldsymbol{x}) = -\frac{\|\boldsymbol{x} - \boldsymbol{\mu}_i\|^2}{2\sigma^2} + \log P(\omega_i)$$

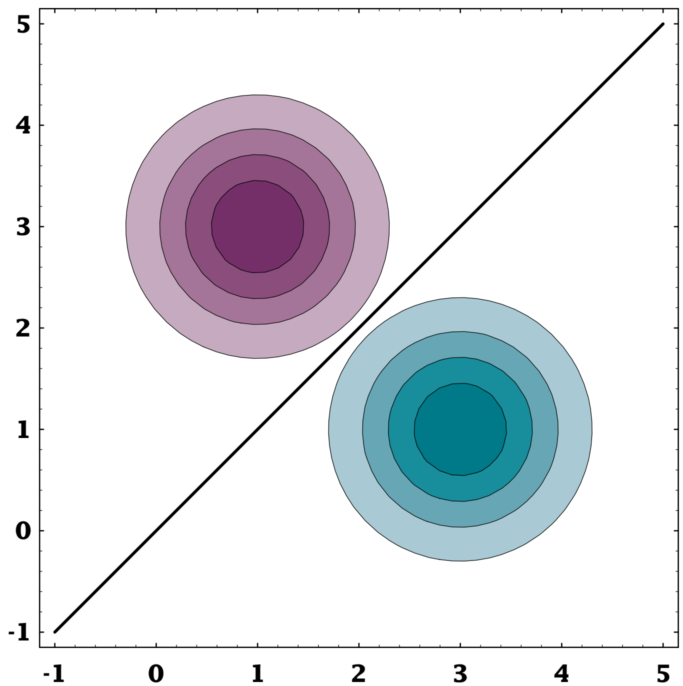
- If each class is equally likely, this is a minimum distance classifier, a form of template matching
- The discriminant functions can be replaced by the following linear expression:

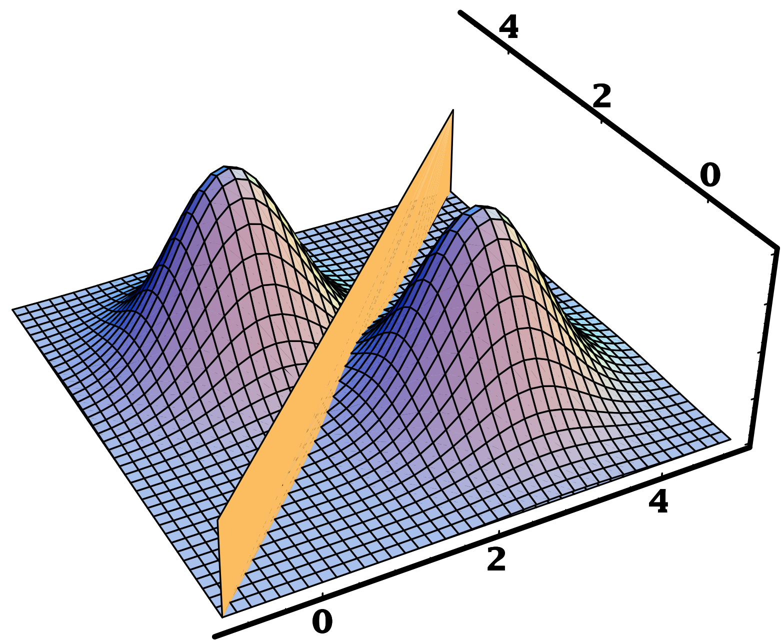
$$g_i(\boldsymbol{x}) = \boldsymbol{w}_i^t \boldsymbol{x} + \omega_{i0}$$

  where $\boldsymbol{w}_i = \frac{1}{\sigma^2}\boldsymbol{\mu}_i$ and $\omega_{i0} = -\frac{1}{2\sigma^2}\boldsymbol{\mu}_i^t\boldsymbol{\mu}_i + \log P(\omega_i)$

Gaussian Classifier: $\Sigma_i = \sigma^2 \boldsymbol{I}$

For distributions with a common covariance structure the decision regions are hyper-planes.





Gaussian Classifier: $\Sigma_i = \Sigma$

- Each class has the same covariance structure $\mathbf{\Sigma}$
- The equivalent discriminant functions are:

$$g_i(\boldsymbol{x}) = -\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_i)^t \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_i) + \log P(\omega_i)$$

- If each class is equally likely, the minimum error decision rule is the squared Mahalanobis distance
- The discriminant functions remain linear expressions:

$$g_i(\boldsymbol{x}) = \boldsymbol{w}_i^t \boldsymbol{x} + \omega_{i0}$$

where

$$\boldsymbol{w}_i = \Sigma^{-1} \boldsymbol{\mu}_i$$

$$\omega_{i0} = -\frac{1}{2}\boldsymbol{\mu}_i^t \Sigma^{-1} \boldsymbol{\mu}_i + \log P(\omega_i)$$

Gaussian Classifier: Σ_i Arbitrary

- Each class has a different covariance structure $\boldsymbol{\Sigma}_i$
- The equivalent discriminant functions are:

$$g_i(\boldsymbol{x}) = -\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_i)^t \Sigma_i^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_i) - \frac{1}{2} \log |\Sigma_i| + \log P(\omega_i)$$

- The discriminant functions are inherently quadratic:

$$g_i(\boldsymbol{x}) = \boldsymbol{x}^t \boldsymbol{W}_i \boldsymbol{x} + \boldsymbol{w}_i^t \boldsymbol{x} + \omega_{i0}$$

where

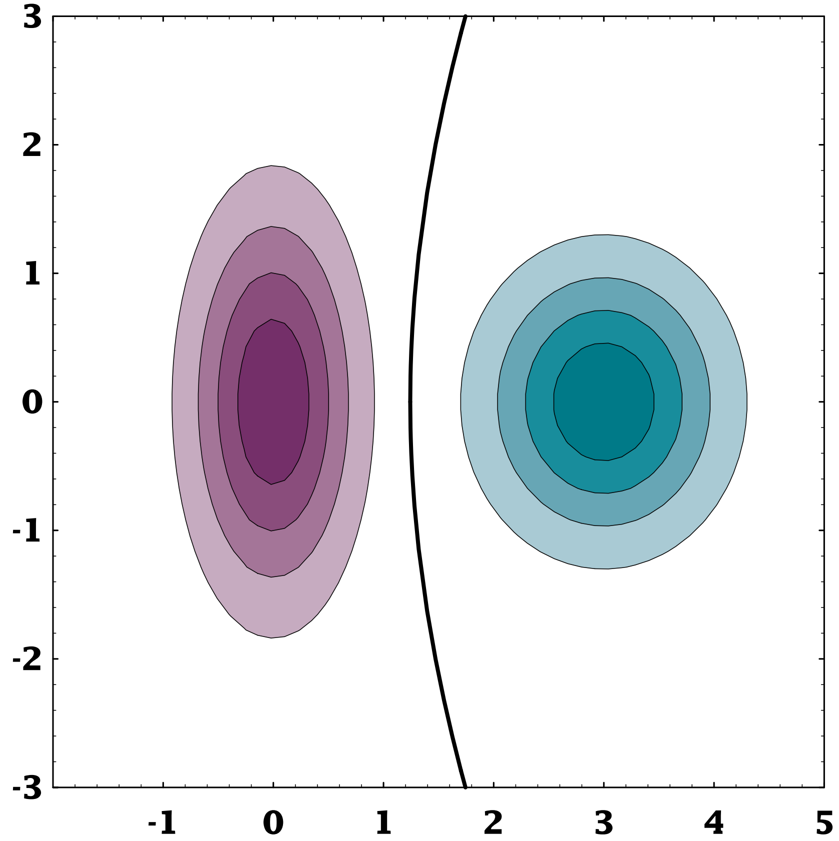
$$\boldsymbol{W}_i = -\frac{1}{2}\Sigma_i^{-1}$$

$$\boldsymbol{w}_i = \Sigma_i^{-1}\boldsymbol{\mu}_i$$

$$\omega_{i0} = -\frac{1}{2}\boldsymbol{\mu}_i^t \Sigma_i^{-1} \boldsymbol{\mu}_i - \frac{1}{2} \log |\Sigma_i| + \log P(\omega_i)$$

Gaussian Classifier: Σ_i Arbitrary

For distributions with arbitrary covariance structures the decision regions are defined by hyper-spheres.



MIT

3 Class Classification (Atal & Rabiner, 1976)

- Distinguish between silence, unvoiced, and voiced sounds
- Use 5 features:
  - Zero crossing count
  - Log energy
  - Normalized first autocorrelation coefficient
  - First predictor coefficient, and
  - Normalized prediction error
- Multivariate Gaussian classifier, ML estimation
- Decision by squared Mahalanobis distance
- Trained on four speakers (2 sentences/speaker), tested on 2 speakers (1 sentence/speaker)

Maximum A Posteriori Parameter Estimation

- Bayesian estimation approaches assume the form of the PDF $p(x|\theta)$ is known, but the value of θ is not
- Knowledge of θ is contained in:
  - An initial *a priori* PDF $p(\theta)$
  - A set of i.i.d. data $\mathcal{X} = \{x_1, \ldots, x_n\}$
- The desired PDF for x is of the form

$$p(x|\mathcal{X}) = \int p(x, \theta|\mathcal{X}) d\theta = \int p(x|\theta) p(\theta|\mathcal{X}) d\theta$$

- The value $\hat{\theta}$ that maximizes $p(\theta|\mathcal{X})$ is called the maximum a posteriori (MAP) estimate of θ

$$p(\theta|\mathcal{X}) = \frac{p(\mathcal{X}|\theta)p(\theta)}{p(\mathcal{X})} = \alpha \prod_{i=1}^{n} p(x_i|\theta)p(\theta)$$

Gaussian MAP Estimation: One Dimension

- For a Gaussian distribution with unknown mean μ:

$$p(x|\mu) \sim N(\mu, \sigma^2) \qquad p(\mu) \sim N(\mu_0, \sigma_0^2)$$

- MAP estimates of μ and x are given by:

$$p(\mu|\mathcal{X}) = \alpha \prod_{i=1}^{n} p(x_i|\mu) p(\mu) \ \sim N(\mu_n, \sigma_n^2)$$

$$p(x|\mathcal{X}) = \int p(x|\mu) p(\mu|\mathcal{X}) d\mu \ \sim N(\mu_n, \sigma^2 + \sigma_n^2)$$

where $$\mu_n = \frac{n\sigma_0^2}{n\sigma_0^2 + \sigma^2}\hat{\mu} + \frac{\sigma^2}{n\sigma_0^2 + \sigma^2}\mu_0 \qquad \sigma_n^2 = \frac{\sigma_0^2 \sigma^2}{n\sigma_0^2 + \sigma^2}$$

- As n increases, $p(\mu|\mathcal{X})$ converges to $\hat{\mu}$, and $p(x|\mathcal{X})$ converges to the ML estimate $\sim N(\hat{\mu}, \sigma^2)$

References

- Huang, Acero, and Hon, *Spoken Language Processing*, Prentice-Hall, 2001.
- Duda, Hart and Stork, *Pattern Classification*, John Wiley & Sons, 2001.
- Atal and Rabiner, A Pattern Recognition Approach to Voiced-Unvoiced-Silence Classification with Applications to Speech Recognition, *IEEE Trans ASSP*, 24(3), 1976.