Chapter 3
MAXIMUM-LIKELIHOOD AND BAYESIAN PARAMETER ESTIMATION

Wei-Yang Lin

Department of Computer Science

& Information Engineering

mailto:wylin@cs.ccu.edu.tw

3.1 **Introduction**

3.2 Maximum-Likelihood Estimation

3.3 Bayesian Estimation

3.4 Bayesian Parameter Estimation: Gaussian Case

3.5 Bayesian Parameter Estimation: General Theory

3.7 Problems of Dimensionality

3.8 Component Analysis and Discriminants

3.9 Expectation-Maximization (EM)

3.10 Hidden Markov Models

3.1 Introduction

- In the previous chapter, we learn how to design an optimal classifier if we knew the priors and conditional densities.
- Unfortunately, we rarely have this kind of complete knowledge about a classification problem.
- In a typical case, we merely have some knowledge about the situation, together with a number of **training samples**.
- So, how could we design a classifier now?

- One approach is to use training samples to estimate the unknown prior probabilities and conditional densities.
- For example, we can assume that a conditional density is normal with $\mathbf{m}_i$ and covariance $\mathbf{\Sigma}_i$, and then use training samples to estimate the parameters $\mathbf{m}_i$ and $\mathbf{\Sigma}_i$.
- The parameter estimation is a classical problem in statistics.
- We shall now introduce two popular procedures, namely, **maximum-likelihood** estimation and **Bayesian** estimation.

- Maximum-likelihood treats the parameters as unknown quantities whose values are fixed.
  - The best estimate is defined to be the one that maximizes a likelihood function.
- Bayesian method views the parameters as random variables having some known distributions.
  - In the Bayesian case, a typical effect of observing a new data is to sharpen a density, causing it to peak near the true value of a parameter.
  - This phenomenon is known as **Bayesian learning**.

3.1 Introduction

3.2 **Maximum-Likelihood Estimation**

3.3 Bayesian Estimation

3.4 Bayesian Parameter Estimation: Gaussian Case

3.5 Bayesian Parameter Estimation: General Theory

3.7 Problems of Dimensionality

3.8 Component Analysis and Discriminants

3.9 Expectation-Maximization (EM)

3.10 Hidden Markov Models

3.2.1 The General Principle

- Suppose we have c data sets, $\mathcal{D}_1, \ldots, \mathcal{D}_c$, with the samples in $\mathcal{D}_j$ having been drawn independently from the probability distribution $p(\mathbf{x}|\omega_j)$.
- We say such samples are **i.i.d.** – independent and identically distributed.
- We assume that $p(\mathbf{x}|\omega_j)$ has a known parametric form and therefore is determined by the value of its parameter $\boldsymbol{\theta}_j$.

- For example, we might have $p(\mathbf{x}|\omega_j) \sim N(\boldsymbol{\theta}_j)$, where $\boldsymbol{\theta}_j$ consists of $\mathbf{m}_j$ and $\boldsymbol{\Sigma}_j$
- To show the dependency on $\boldsymbol{\theta}_j$ explicitly, we write $p(\mathbf{x}|\omega_j)$ as $p(\mathbf{x}|\omega_j, \boldsymbol{\theta}_j)$.
- Our problem is to use training samples to obtain good estimates for the unknown parameters $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_c$ associated with each category.

- For simplicity, we assume that samples in $\mathcal{D}_i$ give no information about $\boldsymbol{\theta}_j$ if $i \neq j$.
- With this assumption, we have c separate problems of the following form: Use a set $\mathcal{D}$ of training samples drawn independently from the density $p(\mathbf{x}|\boldsymbol{\theta})$ to estimate the unknown parameter $\boldsymbol{\theta}$.

- Suppose that $\mathcal{D}$ contains n sample, $\mathbf{x}_1, \ldots, \mathbf{x}_n$. Because the samples are drawn independently, we have

$$p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{k=1}^{n} p(\mathbf{x}_k|\boldsymbol{\theta}) \tag{1}$$

- The $p(\mathcal{D}|\boldsymbol{\theta})$ is a function of $\boldsymbol{\theta}$ and is called the **likelihood** of $\boldsymbol{\theta}$ with respect to the set of samples.
- The **maximum likelihood estimate** of $\boldsymbol{\theta}$ is the value $\hat{\boldsymbol{\theta}}$ that maximizes $p(\mathcal{D}|\boldsymbol{\theta})$.

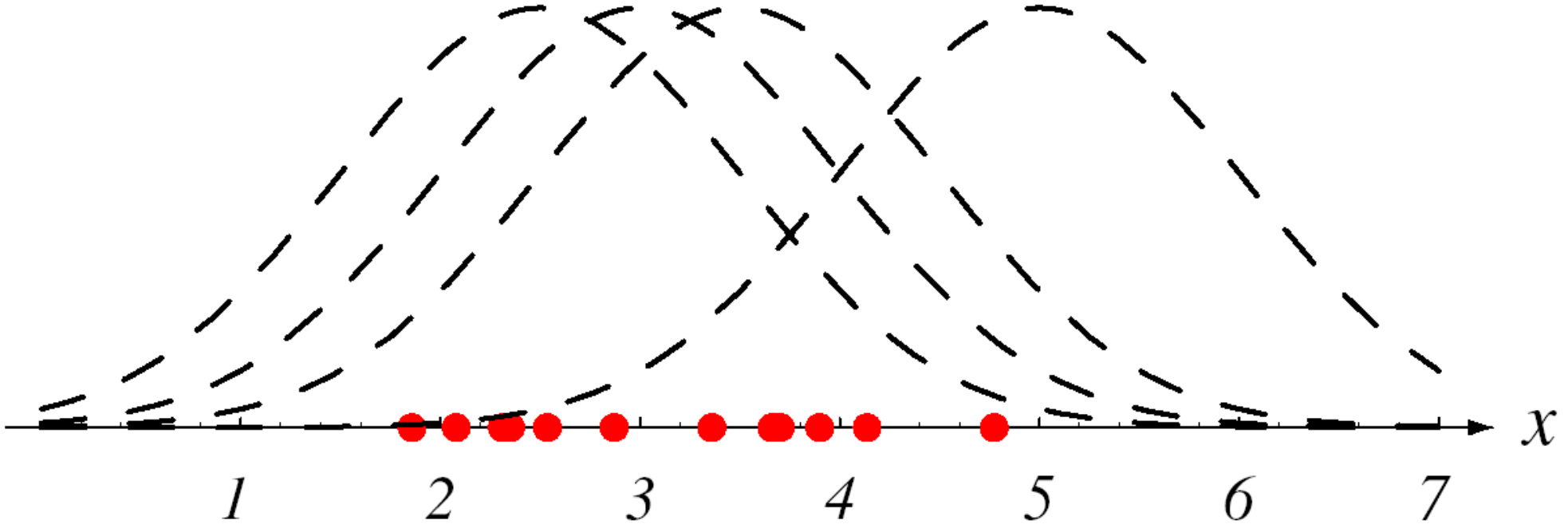


Figure 1: The data points (red dots) are assumed or known to be drawn from a Gaussian distribution of a particular variance, but unknown mean. Four candidate distributions are shown in dashed line.

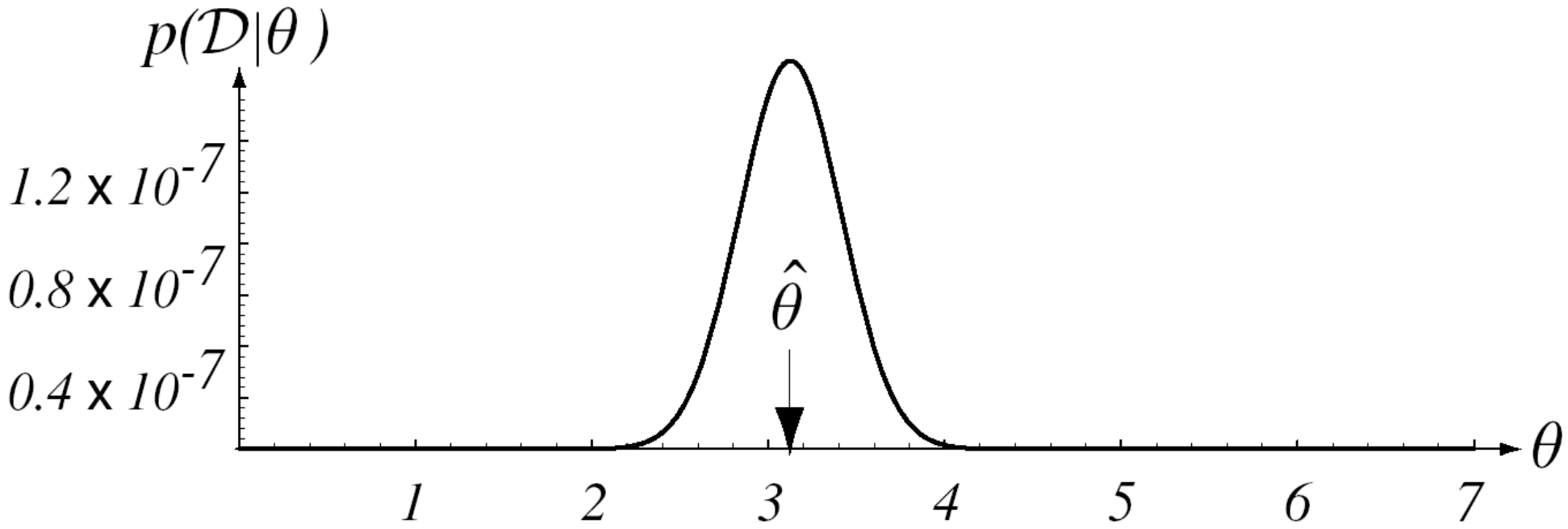


Figure 2: The likelihood $p(\mathcal{D}|\theta)$ is shown as a function of θ. The value that maximizes the likelihood is marked $\hat{\theta}$.

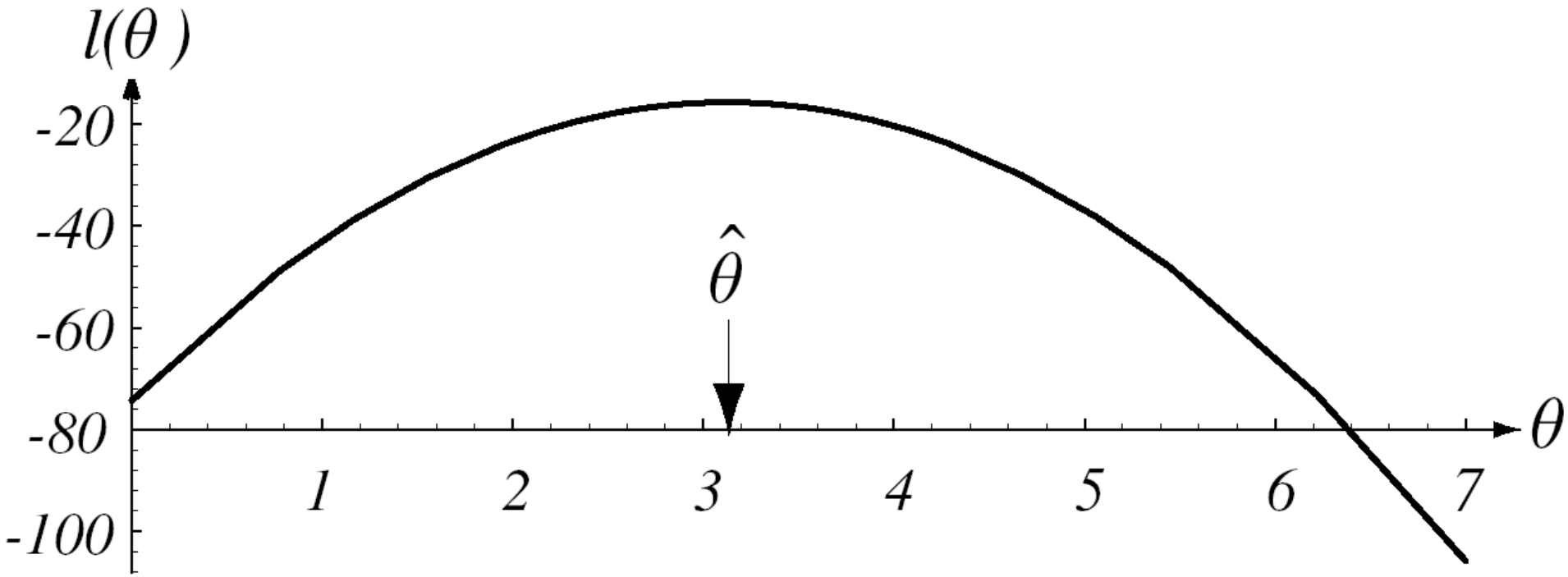


Figure 3: The $\hat{\theta}$ also maximizes the log-likelihood $l(\theta)$.

- It is usually easier to work with the logarithm of a likelihood than with the likelihood itself.
- Because the logarithm is monotonically increasing, the $\hat{\boldsymbol{\theta}}$ that maximizes the log-likelihood also maximizes the likelihood itself.
- Suppose $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)^t$, and we let $\boldsymbol{\nabla}_{\boldsymbol{\theta}}$ denote the gradient operator.

$$\boldsymbol{\nabla}_{\boldsymbol{\theta}} = \left[\begin{array}{ccc} \frac{\partial}{\partial \theta_1} & \cdots & \frac{\partial}{\partial \theta_p} \end{array} \right]^t \tag{2}$$

- Let $l(\boldsymbol{\theta})$ denote the **log-likelihood** function

$$l(\boldsymbol{\theta}) = \ln p(\mathcal{D}|\boldsymbol{\theta}) \tag{3}$$

We can then formulate the problem as finding the argument $\boldsymbol{\theta}$ that maximizes the log-likelihood, that is,

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} l(\boldsymbol{\theta}) \tag{4}$$

- By taking the logarithm of both sides of (1), we have

$$l(\boldsymbol{\theta}) = \sum_{k=1}^{n} \ln p(\mathbf{x}_k|\boldsymbol{\theta}) \tag{5}$$

and

$$\boldsymbol{\nabla}_{\boldsymbol{\theta}} l(\boldsymbol{\theta}) = \sum_{k=1}^{n} \boldsymbol{\nabla}_{\boldsymbol{\theta}} \ln p(\mathbf{x}_k|\boldsymbol{\theta}) \tag{6}$$

- The maximum-likelihood estimate for $\boldsymbol{\theta}$ can be found by solving the set of p equations.

$$\boldsymbol{\nabla}_{\boldsymbol{\theta}} l(\boldsymbol{\theta}) = \mathbf{0} \tag{7}$$

- A solution to Eq. (7) could be a global maximum, a local maximum or minimum, or an inflection point of $l(\boldsymbol{\theta})$.

- One must also check if the extremum occurs at the boundary of parameter space.

3.2.2 The Gaussian Case: Unknown μ

- Suppose that the samples are drawn from a multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$.
- For simplicity, we further assume that only the mean is unknown.
- Under this condition, we have

$$\ln p(\mathbf{x}_k|\boldsymbol{\mu}) = -\frac{1}{2}\ln\left[(2\pi)^d|\boldsymbol{\Sigma}|\right] - \frac{1}{2}(\mathbf{x}_k - \boldsymbol{\mu})^t\boldsymbol{\Sigma}^{-1}(\mathbf{x}_k - \boldsymbol{\mu}) \tag{8}$$

and

$$\boldsymbol{\nabla}_{\boldsymbol{\mu}} \ln p(\mathbf{x}_k|\boldsymbol{\mu}) = \boldsymbol{\Sigma}^{-1}(\mathbf{x}_k - \boldsymbol{\mu}) \tag{9}$$

- We see from Eqs. (6) and (7) that the maximum-likelihood estimate for $\boldsymbol{\mu}$ must satisfy

$$\sum_{k=1}^{n} \boldsymbol{\Sigma}^{-1}(\mathbf{x}_k - \hat{\boldsymbol{\mu}}) = \mathbf{0} \tag{10}$$

- After some algebraic manipulations, we obtain

$$\hat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{k=1}^{n} \mathbf{x}_k \tag{11}$$

- This is a satisfying result.
- It says that the maximum-likelihood estimate is the average of training samples – the **sample mean**.
- One would be inclined to use this intuitive estimate without knowing that it is the maximum-likelihood solution.

3.2.3 The Gaussian Case: Unknown μ and Σ

- In a more general case, neither the mean $\boldsymbol{\mu}$ nor the covariance $\boldsymbol{\Sigma}$ is known.
- We first consider the univariate case with $\theta_1 = \mu$ and $\theta_2 = \sigma^2$.
- Hence, the log-likelihood is

$$\ln p(x_k|\boldsymbol{\theta}) = -\frac{1}{2}\ln\left[2\pi\theta_2\right] - \frac{1}{2\theta_2}(x_k - \theta_1)^2 \tag{12}$$

and its gradient is

$$\boldsymbol{\nabla}_{\boldsymbol{\theta}} \ln p(x_k|\boldsymbol{\theta}) = \begin{bmatrix} \frac{1}{\theta_2}(x_k - \theta_1) \\ -\frac{1}{2\theta_2} + \frac{(x_k-\theta_1)^2}{2\theta_2^2} \end{bmatrix} \tag{13}$$

- Once again, we see from Eqs. (6) and (7) that the maximum-likelihood estimate for $\boldsymbol{\theta}$ must satisfy

$$\sum_{k=1}^{n} \frac{1}{\hat{\theta}_2}(x_k - \hat{\theta}_1) = 0 \tag{14}$$

and

$$-\sum_{k=1}^{n} \frac{1}{\hat{\theta}_2} + \sum_{k=1}^{n} \frac{(x_k - \hat{\theta}_1)^2}{\hat{\theta}_2^2} = 0 \tag{15}$$

- By substituting $\hat{\mu} = \hat{\theta}_1$ and $\hat{\sigma}^2 = \hat{\theta}_2$, we obtain the following maximum-likelihood estimates:

$$\hat{\mu} = \frac{1}{n} \sum_{k=1}^{n} x_k \tag{16}$$

and

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{k=1}^{n} (x_k - \hat{\mu})^2 \tag{17}$$

- Again, the maximum-likelihood estimate for the mean is sample mean.
- The maximum-likelihood estimate for the variance is the average of $(x_k - \hat{\mu})^2$.
- Because the true variance is the expected value of $(x - \mu)^2$, this is also a satisfying result.

- The analysis of the multivariate case is very similar (Problem 6).
- The maximum-likelihood estimates for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are given by

$$\hat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_k \tag{18}$$

and

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{n}\sum_{k=1}^{n}(\mathbf{x}_k - \hat{\boldsymbol{\mu}})(\mathbf{x}_k - \hat{\boldsymbol{\mu}})^t \tag{19}$$

3.2.4 Bias

- The maximum-likelihood estimate for the variance σ^2 is **biased**; that is, the expected value of the estimate is not equal to the true value:

$$\mathcal{E}\left[\frac{1}{n}\sum_{k=1}^{n}(x_k - \bar{x})^2\right] = \frac{n-1}{n}\sigma^2 \neq \sigma^2 \tag{20}$$

- If an estimator tends to become unbiased as the number of samples becomes very large, as for instance Eq. (20), then the estimator is **asymptotically unbiased**.

- In the applications with large training data, asymptotically unbiased estimators are acceptable.

- The maximum-likelihood estimate for the covariance matrix is similarly biased.
- An intuitive **unbiased** estimator for $\boldsymbol{\Sigma}$ is given by

$$\mathbf{C} = \frac{1}{n-1} \sum_{k=1}^{n} (\mathbf{x}_k - \hat{\boldsymbol{\mu}})(\mathbf{x}_k - \hat{\boldsymbol{\mu}})^t \tag{21}$$

  where $\mathbf{C}$ is called **sample covariance matrix**.
- Clearly, $\widehat{\boldsymbol{\Sigma}} = [(n-1)/n]\mathbf{C}$; these two estimates are essentially identical when n is large.

$$
\begin{aligned}
& \mathcal{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left\{x_i-\bar{x})\right\}^2\right] \\
= \quad & \mathcal{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left\{x_i-\frac{1}{n}\sum_{j=1}^{n}x_j\right\}^2\right] \\
= \quad & \mathcal{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left\{x_i-\mu+\mu-\frac{1}{n}\sum_{j=1}^{n}x_j\right\}^2\right] \\
= \quad & \frac{1}{n}\sum_{i=1}^{n}\mathcal{E}\left[\left\{x_i-\mu-\frac{1}{n}\sum_{j=1}^{n}(x_j-\mu)\right\}^2\right] \\
= \quad & \frac{1}{n}\sum_{i=1}^{n}\mathcal{E}\left[(x_i-\mu)^2-2(x_i-\mu)\frac{1}{n}\sum_{j=1}^{n}(x_j-\mu)+\frac{1}{n^2}\Big\{\sum_{j=1}^{n}(x_j-\mu)\Big\}^2\right]
\end{aligned}
$$

Continued ...

$$\frac{1}{n}\sum_{i=1}^{n}\mathcal{E}\left[(x_i-\mu)^2-2(x_i-\mu)\frac{1}{n}\sum_{j=1}^{n}(x_j-\mu)+\frac{1}{n^2}\{\sum_{j=1}^{n}(x_j-\mu)\}^2\right]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\sigma^2-\frac{2\sigma^2}{n}+\frac{1}{n^2}\sum_{j=1}^{n}\mathcal{E}\left[(x_j-\mu)^2\right]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\sigma^2-\frac{2\sigma^2}{n}+\frac{\sigma^2}{n}$$

$$= \frac{\sigma^2}{n}\sum_{i=1}^{n}1-\frac{2}{n}+\frac{1}{n}$$

$$= \frac{\sigma^2}{n}\sum_{i=1}^{n}\frac{n-1}{n}=\frac{n-1}{n}\sigma^2 \qquad (22)$$

3.1 Introduction

3.2 Maximum-Likelihood Estimation

3.3 **Bayesian Estimation**

3.4 Bayesian Parameter Estimation: Gaussian Case

3.5 Bayesian Parameter Estimation: General Theory

3.7 Problems of Dimensionality

3.8 Component Analysis and Discriminants

3.9 Expectation-Maximization (EM)

3.10 Hidden Markov Models

3.3 Bayesian Estimation

- We now introduce the Bayesian estimation, also known as Bayesian learning.
- There is a conceptual difference between the maximum-likelihood and Bayesian approaches.
  - In maximum-likelihood estimation, the value of an unknown parameter is fixed.
  - In Bayesian learning, we consider $\boldsymbol{\theta}$ to be a random vector.

3.3.1 The Class-Conditional Densities

- Let $\mathcal{D}$ denote the set of samples, and we would like to emphasize the role of samples by writing the posterior probabilities as $P(\omega_i|\mathbf{x}, \mathcal{D})$.
- Given the samples $\mathcal{D}$, Bayes formula is

$$P(\omega_i|\mathbf{x}, \mathcal{D}) = \frac{p(\mathbf{x}|\omega_i, \mathcal{D})P(\omega_i|\mathcal{D})}{\sum_{j=1}^{c} p(\mathbf{x}|\omega_j, \mathcal{D})P(\omega_j|\mathcal{D})} \tag{23}$$

- It says that we could use the information provided by the training samples to help determine both the conditional densities and priors.

- For simplicity, we assume that the true values of prior probabilities are known; thus we substitute $P(\omega_i) = P(\omega_i|\mathcal{D})$.
- This allows us to write Eq. (23) as

$$P(\omega_i|\mathbf{x}, \mathcal{D}) = \frac{p(\mathbf{x}|\omega_i, \mathcal{D})P(\omega_i)}{\sum_{j=1}^{c} p(\mathbf{x}|\omega_j, \mathcal{D})P(\omega_j)} \tag{24}$$

- Furthermore, each category can be treated independently.
- Thus, we have c separate problems of the following form: Obtain an estimate $p(\mathbf{x}|\mathcal{D})$ by using samples drawn independently from a fixed but unknown distribution $p(\mathbf{x})$.

3.3.2 The Parameter Distribution

- Although the desired density $p(\mathbf{x})$ is unknown, we assume that it has a known parametric form.
- The only thing assumed unknown is the value of parameter $\boldsymbol{\theta}$.
- We could express the assumption that $p(\mathbf{x})$ is unknown but has a known parametric form by saying that the function $p(\mathbf{x}|\boldsymbol{\theta})$ is completely known.

- Now, we have convert the problem of estimating a density function to one of estimating a parameter.
- Any information we might have about $\boldsymbol{\theta}$ is assumed to contained in a prior density $p(\boldsymbol{\theta})$.
- Then, our goal is to convert $p(\boldsymbol{\theta})$ to a posterior density $p(\boldsymbol{\theta}|\mathcal{D})$, which is expected to sharply peak about the true value of $\boldsymbol{\theta}$.

- We do this by integrating the joint density $p(\mathbf{x}, \boldsymbol{\theta}|\mathcal{D})$ over $\boldsymbol{\theta}$. That is,

$$p(\mathbf{x}|\mathcal{D}) = \int p(\mathbf{x}, \boldsymbol{\theta}|\mathcal{D}) d\boldsymbol{\theta} \tag{25}$$

  where the integration extends over the entire parameter space.

- Note that we can always write $p(\mathbf{x}, \boldsymbol{\theta}|\mathcal{D})$ as $p(\mathbf{x}|\boldsymbol{\theta}, \mathcal{D})p(\boldsymbol{\theta}|\mathcal{D})$.

- Also, the distribution of $\mathbf{x}$ is known completely once we know the value of parameter $\boldsymbol{\theta}$, i.e. $p(\mathbf{x}|\boldsymbol{\theta}, \mathcal{D}) = p(\mathbf{x}|\boldsymbol{\theta})$

- Thus, Eq. (25) can be rewritten as

$$\begin{aligned} p(\mathbf{x}|\mathcal{D}) &= \int p(\mathbf{x}|\boldsymbol{\theta},\mathcal{D})p(\boldsymbol{\theta}|\mathcal{D})d\boldsymbol{\theta} \\ &= \int p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{D})d\boldsymbol{\theta} \end{aligned} \tag{26}$$

- This equation links the desired density function $p(\mathbf{x}|\mathcal{D})$ to the posterior density $p(\boldsymbol{\theta}|\mathcal{D})$.
- It says that the samples exert their influence on $p(\mathbf{x}|\mathcal{D})$ through the posterior density $p(\boldsymbol{\theta}|\mathcal{D})$.

- In general, Bayesian estimation leads us to average $p(\mathbf{x}|\boldsymbol{\theta})$ over the possible values of $\boldsymbol{\theta}$.
- If $p(\boldsymbol{\theta}|\mathcal{D})$ peaks very sharply about some value $\hat{\boldsymbol{\theta}}$, we obtain $p(\mathbf{x}|\mathcal{D}) \simeq p(\mathbf{x}|\hat{\boldsymbol{\theta}})$, i.e., the result we would obtain by substituting the estimate $\hat{\boldsymbol{\theta}}$.

3.1 Introduction

3.2 Maximum-Likelihood Estimation

3.3 Bayesian Estimation

3.4 **Bayesian Parameter Estimation: Gaussian Case**

3.5 Bayesian Parameter Estimation: General Theory

3.7 Problems of Dimensionality

3.8 Component Analysis and Discriminants

3.9 Expectation-Maximization (EM)

3.10 Hidden Markov Models

3.4 Bayesian Parameter Estimation: Gaussian Case

- In this section, we use Bayesian estimation techniques to calculate the **a posteriori** density $p(\boldsymbol{\theta}|\mathcal{D})$ for the cases where $p(\mathbf{x}|\omega_i) \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$.

3.4.1 The Univariate Case: $p(\mu|\mathcal{D})$

- For simplicity, we first consider the the univariate case, that is,

$$p(x|\mu) \sim N(\mu, \sigma^2), \tag{27}$$

  where the only unknown is the mean μ.

- We shall make a further assumption that

$$p(\mu) \sim N(\mu_0, \sigma_0^2), \tag{28}$$

  where both μ_0 and σ_0^2 are known.

- Roughly speaking, μ_0 represents our initial guess for μ and σ_0^2 represents our uncertainty about this guess.

- Suppose that n samples $x_1, \ldots, x_n$ are independently drawn from the desired density function $p(x)$.
- Letting $\mathcal{D} = \{x_1, \ldots, x_n\}$, we have

$$\begin{aligned} p(\mu|\mathcal{D}) &= \frac{p(\mathcal{D}|\mu)p(\mu)}{\int p(\mathcal{D}|\mu)p(\mu)d\mu} \\ &= \alpha \prod_{k=1}^{n} p(x_k|\mu)p(\mu) \end{aligned} \tag{29}$$

  where α is a normalization factor that only depends on $\mathcal{D}$.
- This equation shows how the observations affect the distribution of μ; it relates the prior density $p(\mu)$ to the a posteriori density $p(\mu|\mathcal{D})$.

- Because $p(x|\mu) \sim N(\mu, \sigma^2)$ and $p(\mu) \sim N(\mu_0, \sigma_0^2)$, we have

$$\begin{aligned} & p(\mu|\mathcal{D}) \\ = \; & \alpha \prod_{k=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left[\frac{-1}{2}\frac{(x_k-\mu)^2}{\sigma^2}\right] \frac{1}{\sqrt{2\pi}\sigma_0} \exp\left[\frac{-1}{2}\frac{(\mu-\mu_0)^2}{\sigma_0^2}\right] \\ = \; & \alpha' \exp\left[\frac{-1}{2}\left(\sum_{k=1}^{n}\frac{(x_k-\mu)^2}{\sigma^2} + \frac{(\mu-\mu_0)^2}{\sigma_0^2}\right)\right] \\ = \; & \alpha'' \exp\left[\frac{-1}{2}\left[\left(\frac{n}{\sigma^2}+\frac{1}{\sigma_0^2}\right)\mu^2 - 2\left(\frac{1}{\sigma^2}\sum_{k=1}^{n} x_k + \frac{\mu_0}{\sigma_0^2}\right)\mu\right]\right] \end{aligned} \tag{30}$$

  where factors that do not depend on μ have been absorbed in to the constants α, α', and α''.

- $p(\mu|\mathcal{D})$ is an exponential function of a quadratic function of μ, i.e., is again a normal density.
- And, this is true for any number of training samples.
- We could write $p(\mu|\mathcal{D}) \sim N(\mu_n, \sigma_n^2)$, where μ_n and σ_n^2 is found by coefficients in Eq. (30).

$$\frac{1}{\sigma_n^2} = \frac{n}{\sigma^2} + \frac{1}{\sigma_0^2} \tag{31}$$

$$\frac{\mu_n}{\sigma_n^2} = \frac{1}{\sigma^2}\sum_{k=1}^{n} x_k + \frac{\mu_0}{\sigma_0^2} \tag{32}$$

- Then, we solve explicitly for μ_n and σ_n^2

$$\mu_n = \left(\frac{n\sigma_0^2}{n\sigma_0^2 + \sigma^2} \right) \hat{\mu}_n + \frac{\sigma^2}{n\sigma_0^2 + \sigma^2} \mu_0 \tag{33}$$

and

$$\sigma_n^2 = \frac{\sigma_0^2 \sigma^2}{n\sigma_0^2 + \sigma^2} \tag{34}$$

where

$$\hat{\mu}_n = \frac{1}{n} \sum_{k=1}^{n} x_k \tag{35}$$

- These equations show how the prior information is combined with the information contained in training samples.

Some observations

- Roughly speaking, μ_n represents our best guess after observing n samples, and σ_n^2 measures our uncertainty about this guess.
- σ_n^2 decreases monotonically with n – each additional sample decreases the uncertainty of our estimate.
- As n increases, $p(\mu|\mathcal{D})$ becomes more and more sharply peaked.
- This behavior is commonly known as **Bayesian learning** (Figs. 4 and 5).

More observations

- Note that μ_n is a linear combination of $\hat{\mu}_n$ and μ_0. Thus, μ_n always lies somewhere between $\hat{\mu}_n$ and μ_0.
  - If $\sigma_0 \neq 0$, μ_n approaches the sample mean as n approaches infinity.
  - If $\sigma_0 = 0$, we have a case in which our prior certainty is so strong that observations can not change our opinion.
  - At the other extreme, if $\sigma_0 \gg \sigma$, we are so uncertain about our initial guess that we only take sample mean as an estimate.

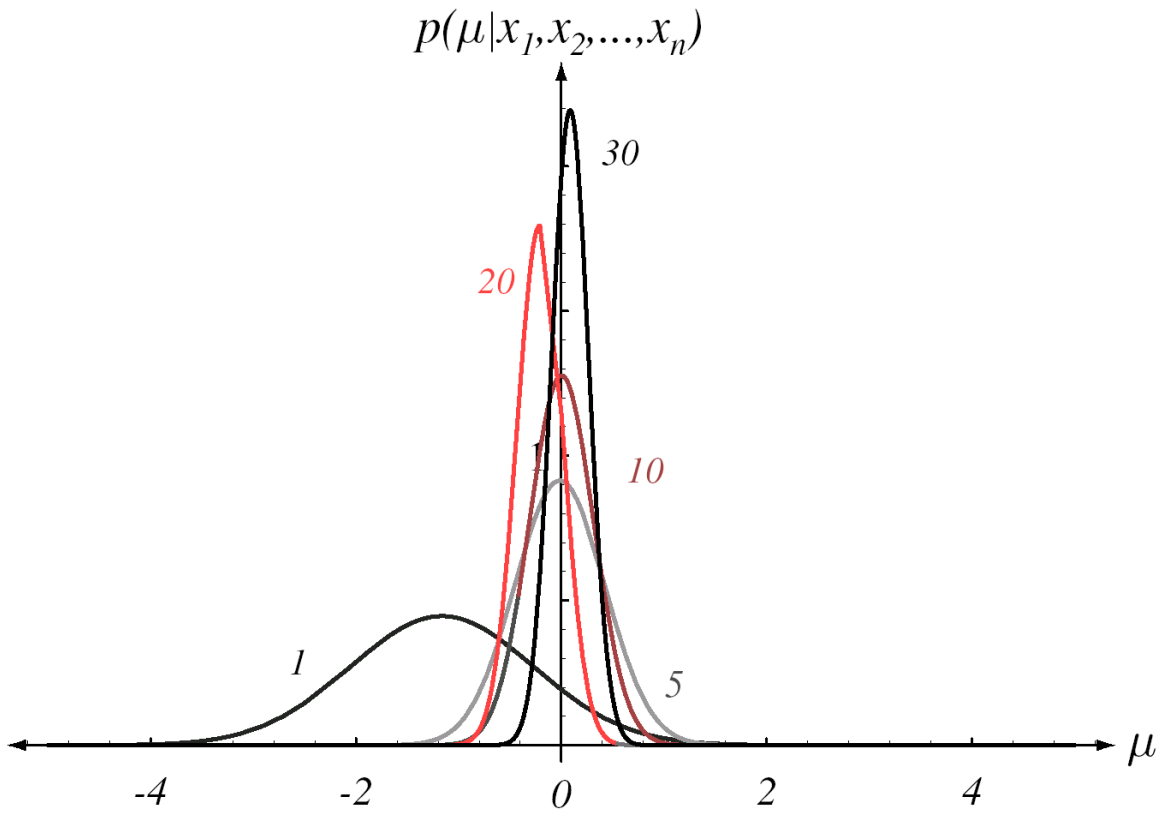


Figure 4: Bayesian learning in one dimension. The posterior distributions are labeled by the number of training samples.

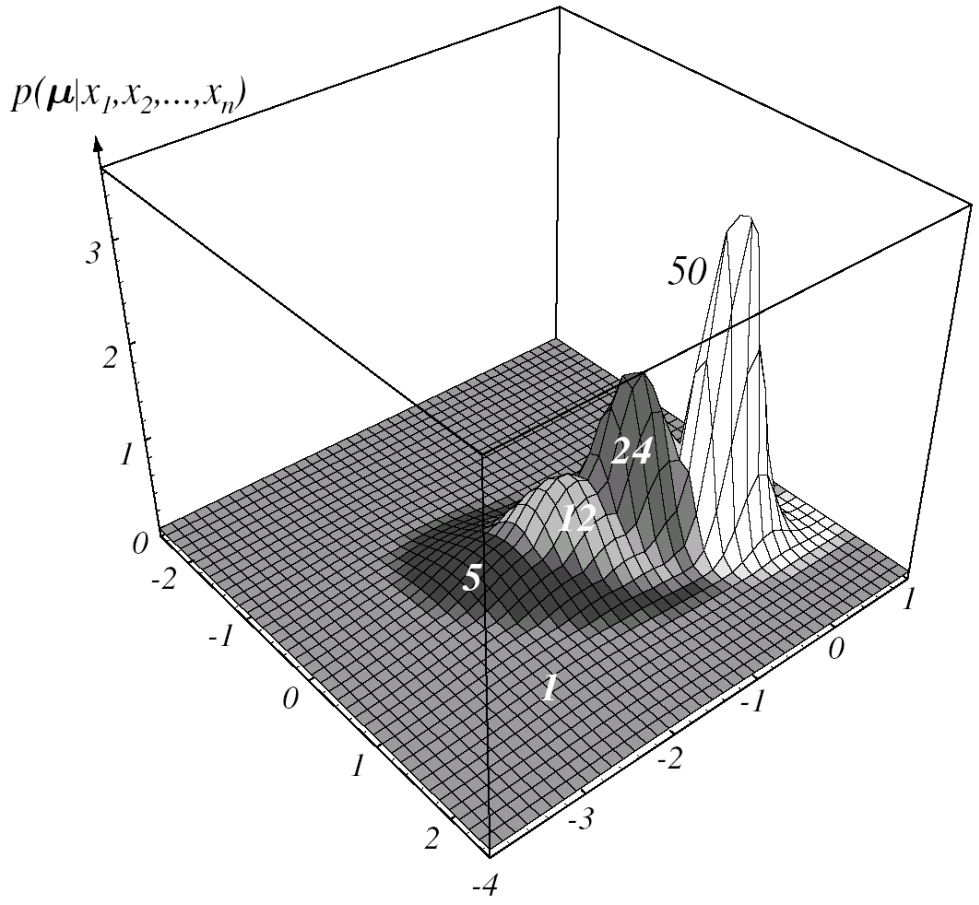


Figure 5: Bayesian learning in two dimension. The posterior distributions are labeled by the number of training samples.

3.4.2 The Univariate Case: $p(\mathbf{x}|\mathcal{D})$

3.4.3 The Multivariate Case

- We have just seen a special case – the univariate Gaussian.
- This approach can be generally applied to any situation in which an unknown density can be parameterized.
- The basic assumptions are summarized as follows:
  - The form of a density $p(\mathbf{x}|\boldsymbol{\theta})$ is assumed to be known, but the value of parameter $\boldsymbol{\theta}$ is not known exactly.
  - Our knowledge about $\boldsymbol{\theta}$ is translated to a prior density $p(\boldsymbol{\theta})$.
  - A set of n samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ drawn independently from the probability distribution $p(\mathbf{x}|\boldsymbol{\theta})$ provides additional information about $\boldsymbol{\theta}$.

- The fundamental issue is to compute the posterior density $p(\boldsymbol{\theta}|\mathcal{D})$. Then, the desired density $p(\mathbf{x}|\mathcal{D})$ is given by

$$p(\mathbf{x}|\mathcal{D}) = \int p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{D})d\boldsymbol{\theta} \tag{36}$$

- By Bayes formula, we have

$$p(\boldsymbol{\theta}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}} \tag{37}$$

- And, by independence assumption

$$p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{k=1}^{n} p(\mathbf{x}_k|\boldsymbol{\theta}) \tag{38}$$

- These constitute the formal solution to the problem.

- While we have obtain the formal Bayesian solution, a number of questions remain.
- One concerns the difficulty of carrying out these computation.
- We shall discuss the issue of computation briefly.
- Let $\mathcal{D}^n = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$. From the independence assumption, we have

$$p(\mathcal{D}^n|\boldsymbol{\theta}) = p(\mathbf{x}_n|\boldsymbol{\theta})p(\mathcal{D}^{n-1}|\boldsymbol{\theta}) \tag{39}$$

- By substituting this in Eq. (37), we obtain the following recursion relation.

$$p(\boldsymbol{\theta}|\mathcal{D}^n) = \frac{p(\mathbf{x}_n|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{D}^{n-1})}{\int p(\mathbf{x}_n|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{D}^{n-1})d\boldsymbol{\theta}} \tag{40}$$

- With the understanding that $p(\boldsymbol{\theta}|\mathcal{D}^0) = p(\boldsymbol{\theta})$, this equation produces a sequence of densities $p(\boldsymbol{\theta}|\mathbf{x}_1), p(\boldsymbol{\theta}|\mathbf{x}_1, \mathbf{x}_2)$ and so forth.
- This is called **recursive Bayes** approach to parameter estimation.
- This is an example of **incremental** or **on-line** learning method, where learning goes on as the data are collected.

Example 1

Recursive Bayes Learning

Suppose we believe our samples come from a uniform distribution

$$p(x|\theta) \sim U(0,\theta) = \begin{cases} 1/\theta & 0 \le x \le \theta \\ 0 & \text{otherwise,} \end{cases} \tag{41}$$

but initially we know only that the parameter is bounded. Given the data $\mathcal{D} = \{4, 7, 2, 8\}$, we could estimate the value of θ by using the recursive Bayes method.

- Before any data arrive, we have the prior $p(\theta) \sim U(0, 10)$.
- When the first data $x_1 = 4$ arrives, we use Eq. (40) to get an improved estimate:

$$p(\theta|\mathcal{D}^1) \propto p(x_1 = 4|\theta)p(\theta|\mathcal{D}^0) = \begin{cases} 1/\theta & 4 \leq \theta \leq 10 \\ 0 & \text{otherwise,} \end{cases} \tag{42}$$

- When the next point arrives, we have

$$p(\theta|\mathcal{D}^2) \propto p(x_2 = 7|\theta)p(\theta|\mathcal{D}^1) = \begin{cases} 1/\theta^2 & 7 \leq \theta \leq 10 \\ 0 & \text{otherwise,} \end{cases} \tag{43}$$

  and similarly for the remaining data points.

- It should be clear that the general form of our solution is $p(\theta|\mathcal{D}^n) \propto 1/\theta^n$ for $\max[\mathcal{D}^n] \leq \theta \leq 10$, as shown in Fig. 6.
- The maximum-likelihood solution here is $\hat{\theta} = 8$, and this implies $p(x|\mathcal{D}) \sim U(0, 8)$.
- The Bayesian solution, which requires the integration in Eq. (36), has a tail at higher values – indicating the influence of the prior $p(\theta)$ (Fig. 7).

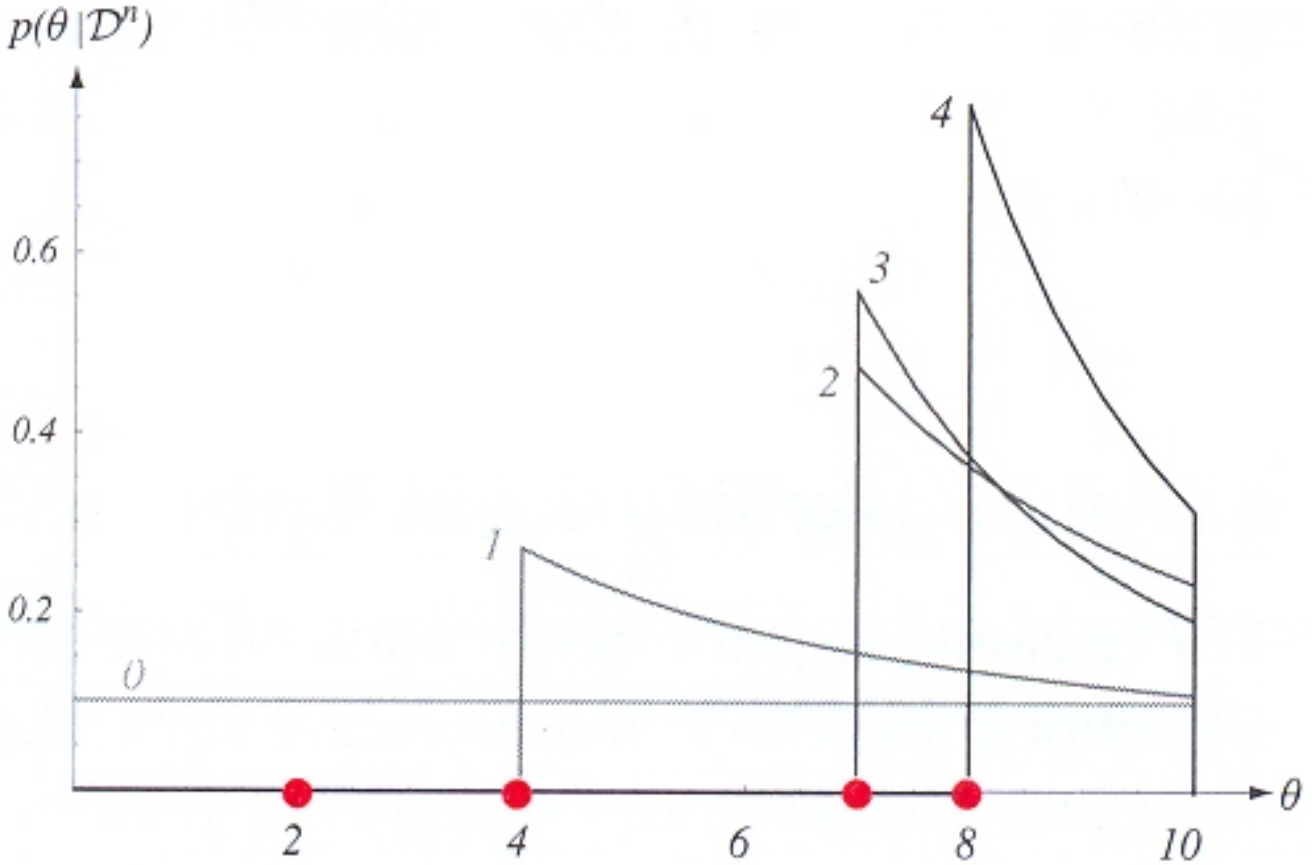


Figure 6: The posterior $p(\theta|\mathcal{D}^n)$

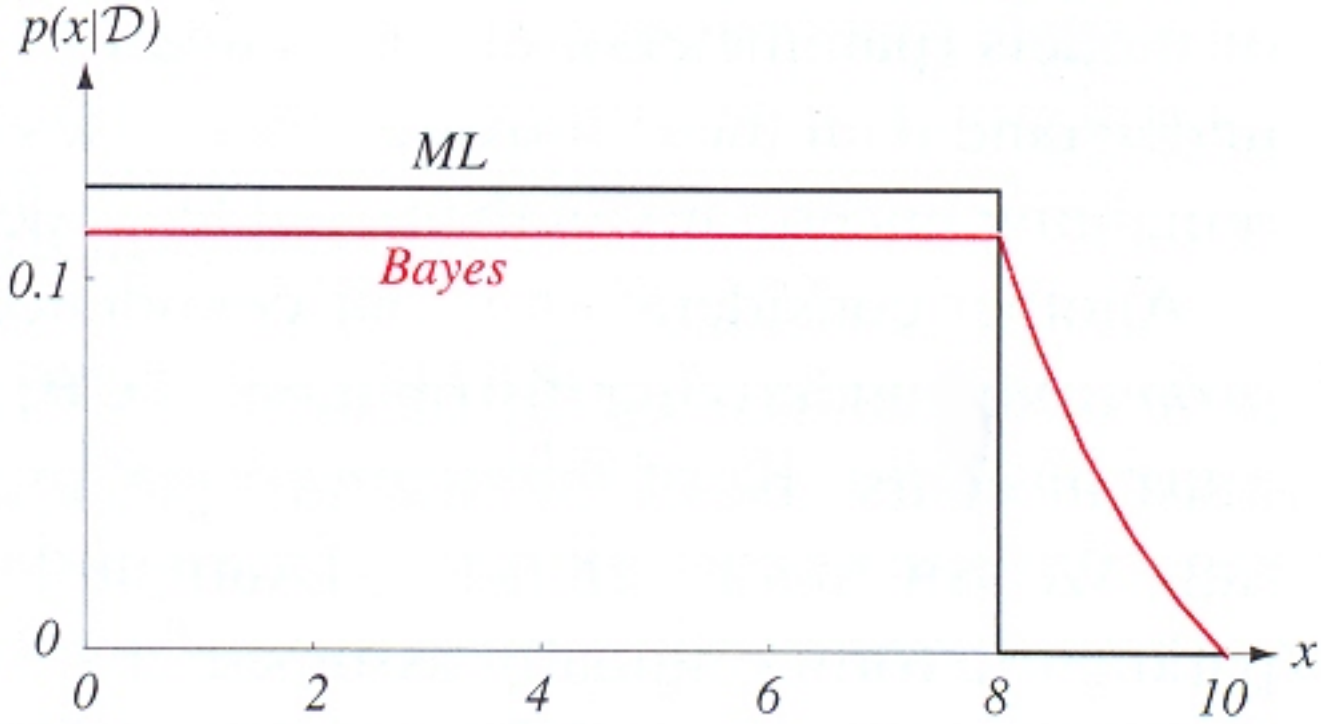


Figure 7: The density estimated by Bayesian method has a tail above $x = 8$.

3.1 Introduction

3.2 Maximum-Likelihood Estimation

3.3 Bayesian Estimation

3.4 Bayesian Parameter Estimation: Gaussian Case

3.5 Bayesian Parameter Estimation: General Theory

3.7 **Problems of Dimensionality**

3.8 Component Analysis and Discriminants

3.9 Expectation-Maximization (EM)

3.10 Hidden Markov Models

3.7 Problems of Dimensionality

- In practical applications, we typically believe that each feature is useful for at least some of the discrimination.
- In the following subsections, we will discuss how classification accuracy depends upon dimensionality.

3.7.1 Accuracy, Dimension, and Training Sample Size

- If the features are statistically independent, there are some theoretical results that suggest excellent performance.
- For example, consider the two-category problem with $p(\mathbf{x}|\omega_j) \sim N(\boldsymbol{\mu}_j, \boldsymbol{\Sigma})$
- If prior probabilities are equal, the Bayes error rate is given by (Problem 30 in chapter 2)

$$P(error) = \frac{1}{\sqrt{2\pi}} \int_{r/2}^{\infty} e^{-u^2/2} du \tag{44}$$

where r^2 is the squared Mahalalanobis distance:

$$r^2 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^t \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \tag{45}$$

- Thus, the probability of error decreases as r increases.
- In the conditionally independent case,

$$\boldsymbol{\Sigma} = diag(\sigma_1^2, \sigma_2^2, \ldots, \sigma_d^2) \tag{46}$$

$$r^2 = \sum_{i=1}^{d} \left(\frac{\mu_{i1} - \mu_{i2}}{\sigma_i} \right)^2 \tag{47}$$

- This shows how each feature contributes to reducing the probability of error.

Some observations

- The most useful features are the ones for which the difference between the means is large relative to the standard deviations.
- A feature is useless if its means for two classes are the same.
- The probability of error can be made arbitrary small by introducing new, independent features.

- In general, if the performance is not satisfactory, it is natural to consider adding new features.
- Although increasing the number of features increases the complexity, it is reasonable to believe that the performance will improve.
- If the new features provide any additional information, the performance must improve (FIG. 8).

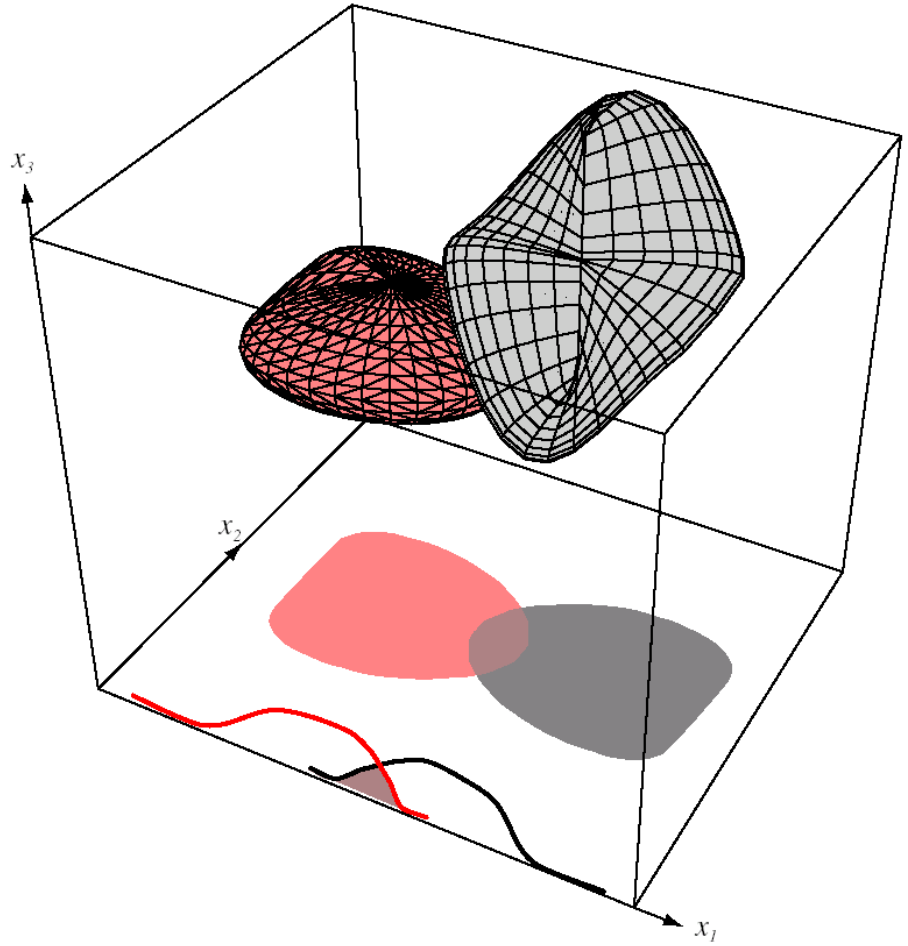


Figure 8: Bayes error vanishes in three-dimensional feature space.

Curse of dimensionality

- Unfortunately, it has been observed in practice that, beyond a certain point, the inclusion of additional features leads to worse performance.
- This paradox presents a serious problem for classifier design.
- The sources of the difficulty include:
  - We have a wrong model, e.g., the Gaussian assumption is inappropriate.
  - The number of samples is insufficient for reliable density estimation.

3.7.2 Computational Complexity

- The computational complexity is an important issue in the design of a classifier.
- Recall the technical notation for computational complexity : we say that $f(x)$ is "of the order of h(x)" – written $\mathcal{O}(h(x))$ and generally read "**big oh** of $h(x)$" – if there exist constants c and x_0 such that $|f(x)| \leq c|h(x)|$ for all $x > x_0$.
- For instance, suppose $f(x) = a_0 + a_1 x + a_2 x^2$; in this case we have a complexity of $\mathcal{O}(x^2)$. Because for sufficiently large x, the constant, linear, and quadratic terms can be overcome by proper choice of c and x_0.

- In describing the computational complexity of a classifier, we are interested in the number of mathematical operations, such as additions, multiplications, and divisions it requires.
- Consider the complexity of the maximum-likelihood parameter estimators in d-dimensional normal densities, with n training samples for each of c categories.
- For each category, it is necessary to calculate

$$g(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \overbrace{\hat{\boldsymbol{\mu}}}^{\mathcal{O}(dn)})^t \overbrace{\widehat{\boldsymbol{\Sigma}}^{-1}}^{\mathcal{O}(nd^2)} (\mathbf{x} - \hat{\boldsymbol{\mu}}) - \overbrace{\frac{d}{2}\ln 2\pi}^{\mathcal{O}(1)} - \overbrace{\frac{1}{2}\ln|\widehat{\boldsymbol{\Sigma}}|}^{\mathcal{O}(d^2n)} + \overbrace{\ln P(\omega)}^{\mathcal{O}(n)}$$

- The complexity of finding $\hat{\boldsymbol{\mu}}$ is $\mathcal{O}(dn)$, because for each of the d dimensions we must add n component values.
- The division by n in sample mean is independent of the number of samples, and hence does not affect the complexity.
- For each of the $d(d+1)/2$ components of the sample covariance matrix $\widehat{\boldsymbol{\Sigma}}$, there are n multiplications and additions.
- The inverse can be calculated in $\mathcal{O}(d^3)$, e.g., Gaussian elimination.
- The complexity of estimating $P(\omega)$ is of course $\mathcal{O}(n)$.

- Usually, we assume that $n > d$ and thus the overall complexity of calculating a discriminant function is dominated by the $\mathcal{O}(d^2 n)$ term.
- The parameter estimation is done for each category. Hence, the overall complexity for training a classifier is $\mathcal{O}(cd^2 n)$.
- Since c is typically a constant much smaller than d^2 or n, we can call the complexity $\mathcal{O}(d^2 n)$.
- We saw in Section 3.4 that it is generally desirable to have more training data from a larger-dimensional space; the complexity analysis shows the steep cost in doing so.

- We now consider the matter of estimating a covariance matrix.
- This requires the estimation of d diagonal elements and $d(d-1)/2$ off-diagonal elements.
- The maximum-likelihood estimate is an $\mathcal{O}(nd^2)$ calculation.

$$\widehat{\mathbf{\Sigma}} = \frac{1}{n}\sum_{k=1}^{n}(\mathbf{x}_k - \mathbf{m}_n)(\mathbf{x}_k - \mathbf{m}_n)^t \tag{48}$$

- It is the sum of $n-1$ independent d-by-d matrices of rank one, and thus is guaranteed to be singular if $n \leq d$.
- Because we must compute the inverse of $\widehat{\mathbf{\Sigma}}$, we have a requirement for at least $d+1$ samples.
- To obtain a good estimate, it is not surprising that much more samples were needed.

- The computational complexity for performing classification is less.
- Given a test data $\mathbf{x}$, we first compute $\mathbf{x} - \hat{\boldsymbol{\mu}}$, an $\mathcal{O}(d)$ calculation.
- Moreover, we multiply the inverse of sample covariance matrix, an $\mathcal{O}(d^2)$ calculation.
- The decision $\max_i g_i(\mathbf{x})$ is an $\mathcal{O}(c)$ operation.
- For small c, classification is an $\mathcal{O}(d^2)$ operation.

- A common distinction is made between **polynomially** complex and **exponentially** complex algorithms – $\mathcal{O}(a^k)$ for some constant a and variable k of a problem.
- Exponential algorithms are generally too complex to be performed in practice.

3.1 Introduction

3.2 Maximum-Likelihood Estimation

3.3 Bayesian Estimation

3.4 Bayesian Parameter Estimation: Gaussian Case

3.5 Bayesian Parameter Estimation: General Theory

3.7 Problems of Dimensionality

3.8 **Component Analysis and Discriminants**

3.9 Expectation-Maximization (EM)

3.10 Hidden Markov Models

3.8 Component Analysis and Discriminants

- One approach to coping with the problem of excessive dimensionality is to reduce the dimensionality.
- Linear methods are particularly attractive.
- Geometrically, linear methods project a high-dimensional data onto a lower dimensional space.
- There are two classical approaches:
  - Principal Component Analysis (PCA)
  - Multiple Discriminant Analysis (MDA)

3.8.1 Principal Component Analysis (PCA)

- Suppose that we want to find a point $\mathbf{x}_0$ such that the sum of the squared distances between $\mathbf{x}_0$ and a set of d-dimensional samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ is as small as possible.
- We define the criterion function $J(\mathbf{x}_0)$ by

$$J(\mathbf{x}_0) = \sum_{k=1}^{n} \|\mathbf{x}_0 - \mathbf{x}_k\|^2 \tag{49}$$

  and seek the value of $\mathbf{x}_0$ that minimizes J.
- It is easy to show that the solution is given by,

$$\mathbf{x}_0 = \mathbf{m} = \frac{1}{n} \sum_{k=1}^{n} \mathbf{x}_k \tag{50}$$

This can be verified by

$$
\begin{aligned}
J(\mathbf{x}_0) &= \sum_{k=1}^{n} \|(\mathbf{x}_0 - \mathbf{m}) - (\mathbf{x}_k - \mathbf{m})\|^2 \\
&= \sum_{k=1}^{n} \|\mathbf{x}_0 - \mathbf{m}\|^2 - 2\sum_{k=1}^{n} (\mathbf{x}_0 - \mathbf{m})^t(\mathbf{x}_k - \mathbf{m}) + \sum_{k=1}^{n} \|\mathbf{x}_k - \mathbf{m}\|^2 \\
&= \sum_{k=1}^{n} \|\mathbf{x}_0 - \mathbf{m}\|^2 - 2(\mathbf{x}_0 - \mathbf{m})^t \sum_{k=1}^{n} (\mathbf{x}_k - \mathbf{m}) + \sum_{k=1}^{n} \|\mathbf{x}_k - \mathbf{m}\|^2 \\
&= \sum_{k=1}^{n} \|\mathbf{x}_0 - \mathbf{m}\|^2 + \underbrace{\sum_{k=1}^{n} \|\mathbf{x}_k - \mathbf{m}\|^2}_{independent\ of\ \mathbf{x}_0} \qquad (51)
\end{aligned}
$$

- The sample mean $\mathbf{m}$ is a zero-dimensional representation of the data set in least-squares sense.
- Similarly, we can obtain a one-dimensional representation for a set of d-dimensional samples.
- Let $\mathbf{e}$ be a unit vector. The line $\mathbf{x} = \mathbf{m} + a\mathbf{e}$, where $a \in \mathbb{R}$ represents a one-dimensional vector space with the origin $\mathbf{m}$.

- Let $\mathbf{m} + a_k\mathbf{e}$ denote the projection of $\mathbf{x}_k$ onto the one-dimensional space.
- We can find the set of coefficients a_k that minimizes the following criterion function:

$$\begin{aligned} J(\mathbf{e}) &= \sum_{k=1}^{n} \|(\mathbf{m} + a_k\mathbf{e}) - \mathbf{x}_k\|^2 \\ &= \sum_{k=1}^{n} \|a_k\mathbf{e} - (\mathbf{x}_k - \mathbf{m})\|^2 \\ &= \sum_{k=1}^{n} a_k^2\|\mathbf{e}\|^2 - 2\sum_{k=1}^{n} a_k\mathbf{e}^t(\mathbf{x}_k - \mathbf{m}) + \sum_{k=1}^{n} \|\mathbf{x}_k - \mathbf{m}\|^2 \end{aligned} \tag{52}$$

- Note that $\|\mathbf{e}\|^2 = 1$.
- By taking the partial derivatives with respect to a_k and setting the derivatives to zero, we obtain

$$a_k = \mathbf{e}^t(\mathbf{x}_k - \mathbf{m}) \tag{53}$$

- Geometrically, this says that we obtain a_k by projecting the $\mathbf{x}_k$ onto the line in the direction of $\mathbf{e}$ that passes through the sample mean.

- The problem of finding the direction $\mathbf{e}$ remains unsolved.
- The solution to this problem involves the so-called **scatter matrix** $\mathbf{S}$ defined by

$$\mathbf{S} = \sum_{k=1}^{n} (\mathbf{x}_k - \mathbf{m})(\mathbf{x}_k - \mathbf{m})^t \tag{54}$$

- The scatter matrix should look familiar – it is merely $n - 1$ times the sample covariance matrix.

We substitute Eq. (53) into Eq. (52) and obtain

$$\begin{aligned} J(\mathbf{e}) &= \sum_{k=1}^{n} a_k^2 - 2\sum_{k=1}^{n} a_k^2 + \sum_{k=1}^{n} \|\mathbf{x}_k - \mathbf{m}\|^2 \\ &= -\sum_{k=1}^{n} \left[\mathbf{e}^t(\mathbf{x}_k - \mathbf{m})\right]^2 + \sum_{k=1}^{n} \|\mathbf{x}_k - \mathbf{m}\|^2 \\ &= -\sum_{k=1}^{n} \mathbf{e}^t(\mathbf{x}_k - \mathbf{m})(\mathbf{x}_k - \mathbf{m})^t\mathbf{e} + \sum_{k=1}^{n} \|\mathbf{x}_k - \mathbf{m}\|^2 \\ &= -\mathbf{e}^t\mathbf{S}\mathbf{e} + \sum_{k=1}^{n} \|\mathbf{x}_k - \mathbf{m}\|^2 \end{aligned} \tag{55}$$

- Clearly, the vector $\mathbf{e}$ that minimizes $J(\mathbf{e})$ also maximizes $\mathbf{e}^t\mathbf{S}\mathbf{e}$.
- We use the method of Lagrange multipliers to find the vector $\mathbf{e}$ that maximizes $\mathbf{e}^t\mathbf{S}\mathbf{e}$ subject to the constraint $\|\mathbf{e}\|^2 = 1$.
- Letting λ be the multiplier, we differentiate

$$u = \mathbf{e}^t\mathbf{S}\mathbf{e} - \lambda(\mathbf{e}^t\mathbf{e} - 1) \tag{56}$$

with respect to $\mathbf{e}$ and obtain

$$\frac{\partial u}{\partial \mathbf{e}} = 2\mathbf{S}\mathbf{e} - 2\lambda\mathbf{e} \tag{57}$$

- By setting the gradient to zero vector, we see that $\mathbf{e}$ must be an eigenvector of the scatter matrix:

$$\mathbf{S}\mathbf{e} = \lambda\mathbf{e}. \tag{58}$$

- Because $\mathbf{e}^t\mathbf{S}\mathbf{e} = \lambda\mathbf{e}^t\mathbf{e} = \lambda$, it follows that we want to select the eigenvector corresponding to the largest eigenvalue.
- In other words, we project the data onto the line through the sample mean in the direction of the eigenvector having the largest eigenvalue.

- The result can be readily extended to d'-dimensional case. We write

$$\mathbf{x}_k = \mathbf{m} + \sum_{i=1}^{d'} a_{ki}\mathbf{e}_i \tag{59}$$

where $d' \leq d$.

- It is not difficult to show that the criterion function

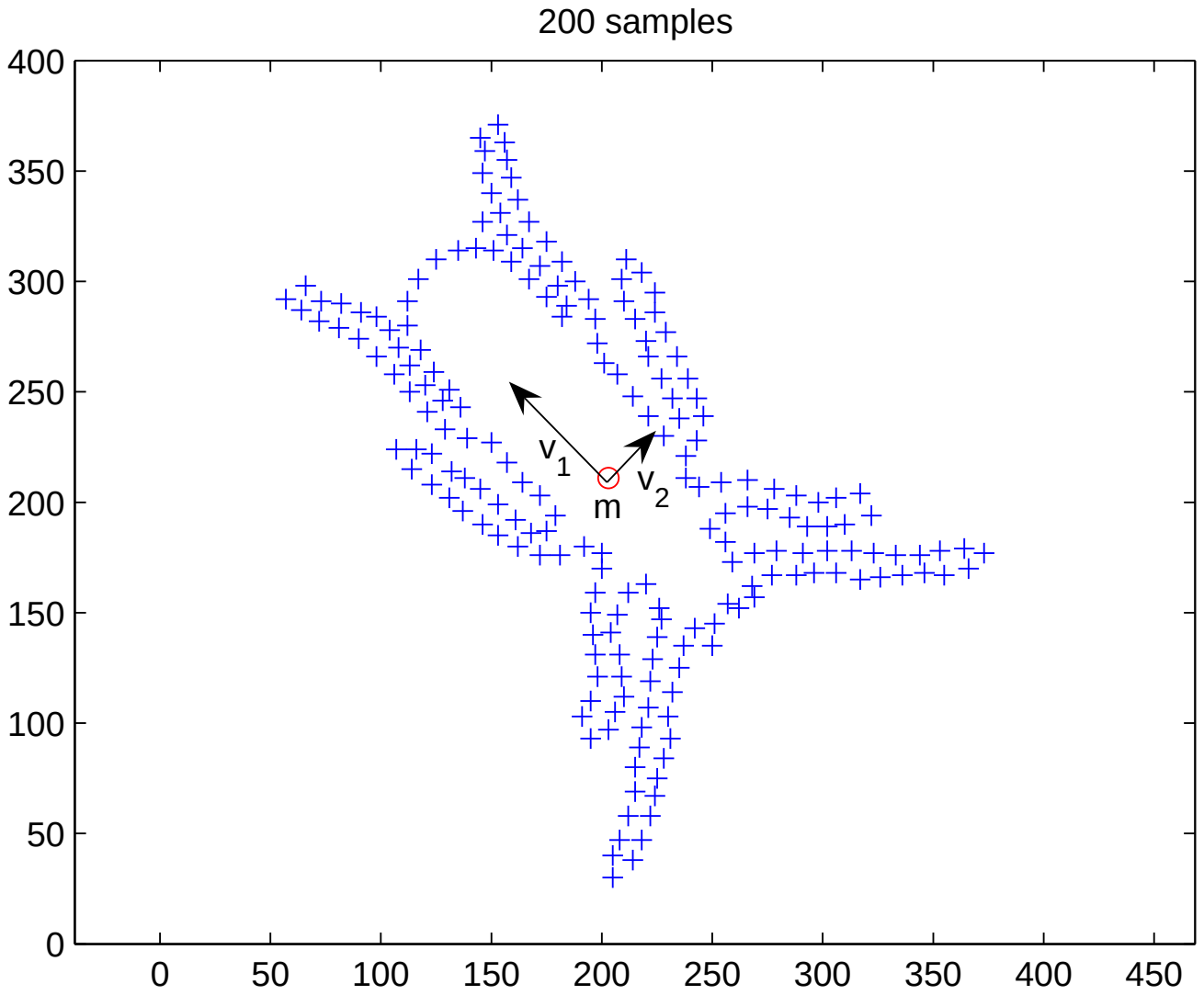
$$J_{d'} = \sum_{k=1}^{n} \|(\mathbf{m} + \sum_{i=1}^{d'} a_{ki}\mathbf{e}_i) - \mathbf{x}_k\|^2 \tag{60}$$

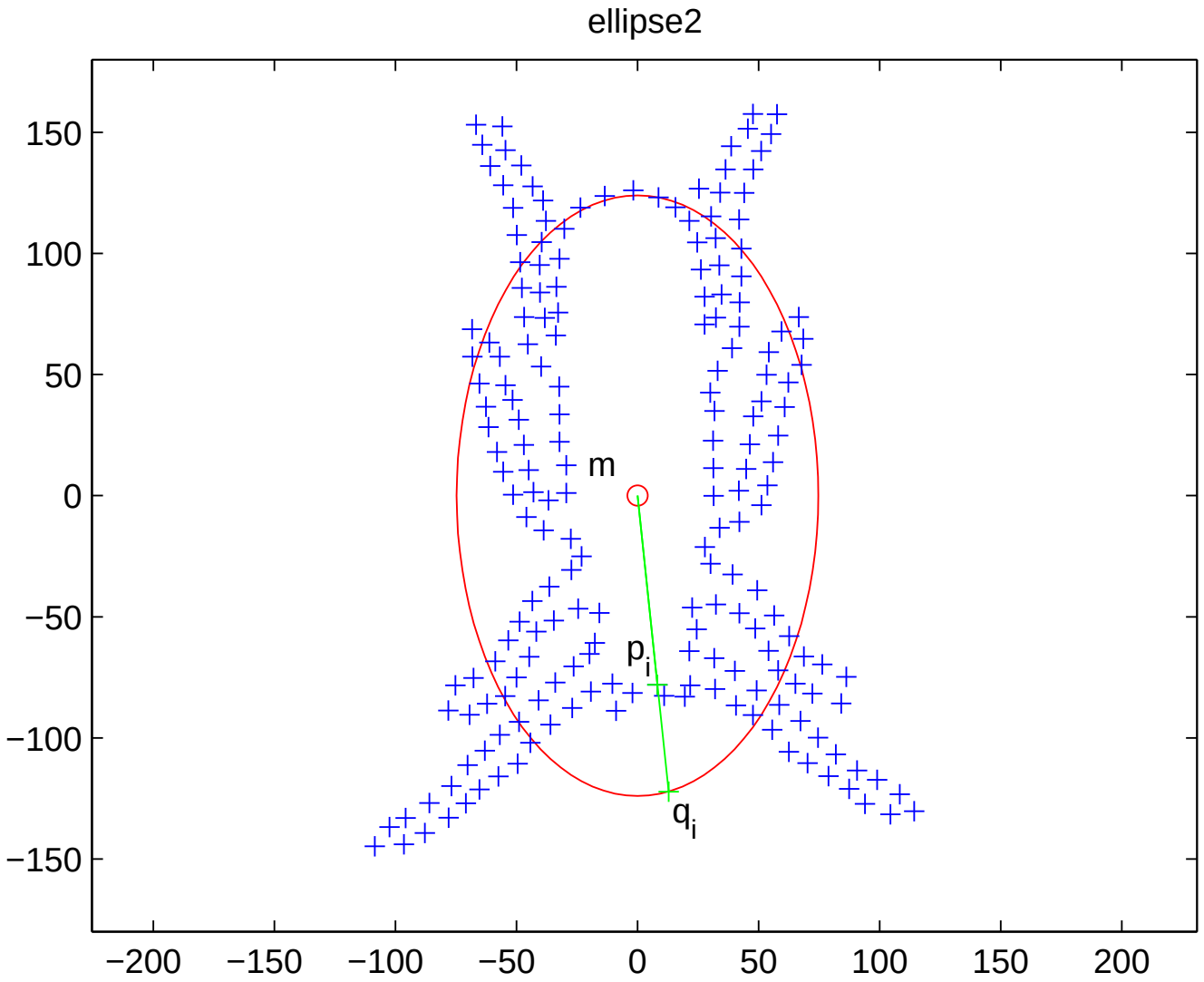
is minimized when $\mathbf{e}_1, \ldots, \mathbf{e}_{d'}$ are the eigenvectors of the scatter matrix having the d' largest eigenvalues.

- The coefficients a_{ki} are called the **principal components**.

Some observations

- Geometrically, if we picture the points $\mathbf{x}_1, \ldots, \mathbf{x}_n$ as forming a hyperellipsoidally shaped cloud, then the eigenvectors are the principal axes of that hyperellipsoid.
- PCA reduces the dimensionality by restricting attention to those directions along which the scatter of the projected points is the greatest.


200 samples
400
350
300
250
200
150
100
50
0
0
50
100
150
200
250
300
350
400
450
v1
v2
m


ellipse2
150
100
50
0
−50
−100
−150
−200
−150
−100
−50
0
50
100
150
200
m
p_i
q_i

3.8.2 Fisher Linear Discriminant

Discriminant Analysis

- The discriminant analysis seeks a direction that is efficient for discrimination.
- Consider the problem of projecting data from d dimensions onto a line. The goal of discriminant analysis is to find a projection direction for which the projected samples are well separated.

Suppose we have a set of labeled training data $\{\mathbf{x}_i, y_i; i = 1, \ldots, N\}$, where $\mathbf{x}_i \in \mathbb{R}^D$ are the samples and $y_i \in \{1, -1\}$ are the labels.

If we perform a linear combination of the components of $\mathbf{x}$, we obtain the scalar dot product

$$\pi = \mathbf{w}^T \mathbf{x}$$

and a corresponding set of samples $\{\pi_1, \ldots, \pi_N\}$.

Geometrically, if $\|\mathbf{w}\| = 1$, each π_i is the projection of the corresponding $\mathbf{x}_i$ onto the direction of $\mathbf{w}$.

- The magnitude of $\mathbf{w}$ is of no real significance.
- The direction of $\mathbf{w}$ is very important.
- Figure 9 illustrates the effect of choosing different values for $\mathbf{w}$.

Note that if the original distributions are highly overlapping, even the "best" $\mathbf{w}$ is unlikely to provide an adequate separation.

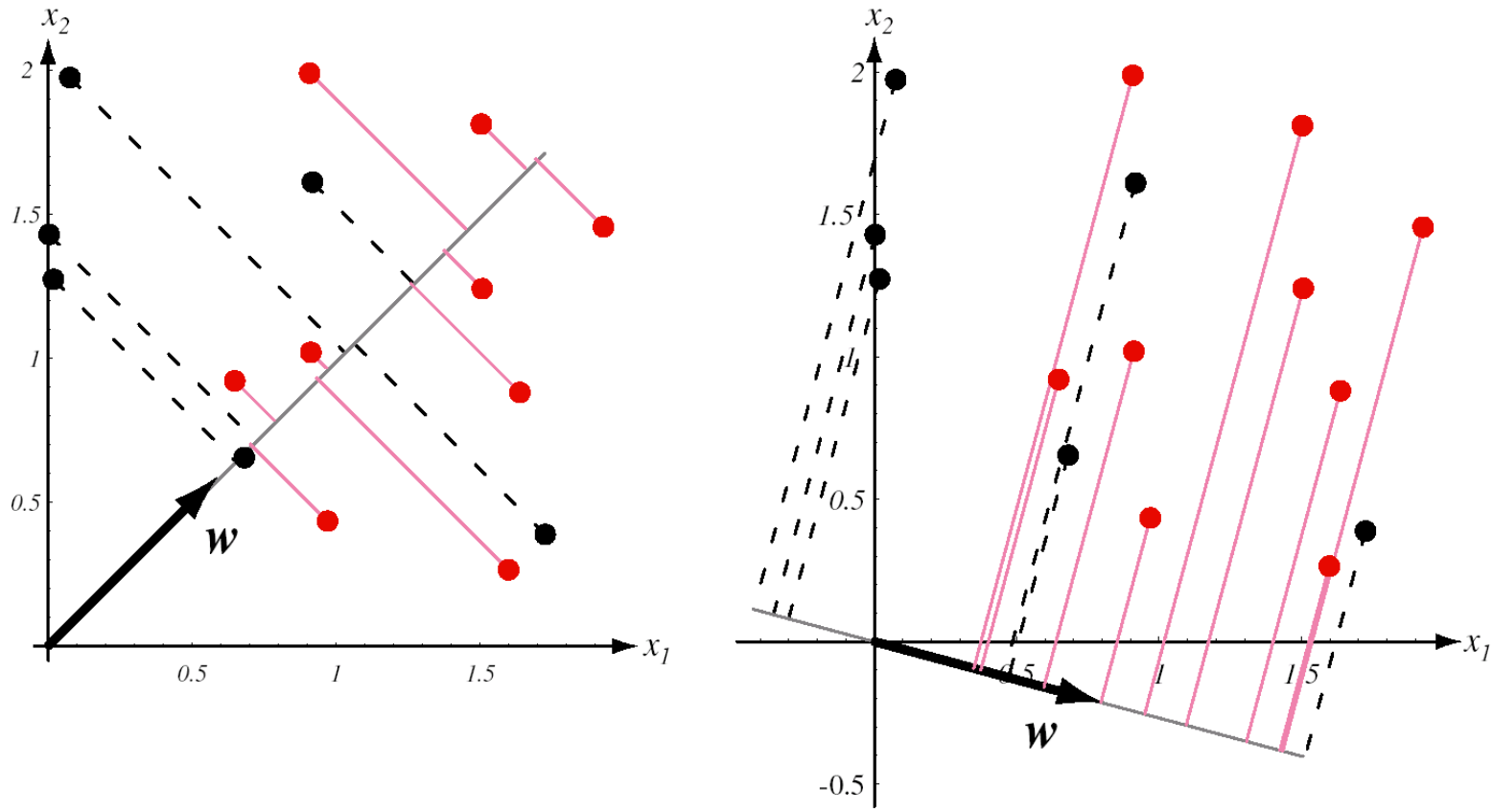


Figure 9: The figure on the right shows greater separation between the red and black projected points.

Find the Best Projection Direction

- We now turn to the matter of finding the best $\mathbf{w}$.
- Fisher's idea is to seek a projection direction $\mathbf{w}$ that separates projected class means well while achieving a small variance around each of them.

If the sample means of the two classes are given by

$$\mathbf{c}_1 = \frac{1}{N_1} \sum_{i:y_i=+1} \mathbf{x}_i \quad \mathbf{c}_2 = \frac{1}{N_2} \sum_{i:y_i=-1} \mathbf{x}_i$$

then the sample means for the projected samples are given by

$$\begin{aligned} \mu_1 &= \frac{1}{N_1} \sum_{i:y_i=+1} \pi_i = \frac{1}{N_1} \sum_{i:y_i=+1} \mathbf{w}^T \mathbf{x}_i = \mathbf{w}^T \mathbf{c}_1 \\ \mu_2 &= \frac{1}{N_2} \sum_{i:y_i=-1} \pi_i = \frac{1}{N_2} \sum_{i:y_i=-1} \mathbf{w}^T \mathbf{x}_i = \mathbf{w}^T \mathbf{c}_2 \end{aligned}$$

The distance between the projected means is

$$|\mu_1 - \mu_2| \ = |\mathbf{w}^T(\mathbf{c}_1 - \mathbf{c}_2)|$$

To obtain a good separation, we want the difference between the projected means to be large relative to some measure of the standard deviations for each class.

Rather than forming sample variances, we define the *scatter* for the projected samples by

$$\begin{aligned} s_1^2 &= \sum_{i:y_i=+1} (\mathbf{w}^T \mathbf{x}_i - \mu_1)^2 \\ s_2^2 &= \sum_{i:y_i=-1} (\mathbf{w}^T \mathbf{x}_i - \mu_2)^2 \end{aligned}$$

$s_1^2 + s_2^2$ is called the total *within-class scatter* of the projected samples.

The goal of Fisher's linear discriminant is to find the optimal $\mathbf{w}$ that maximizes the criterion function

$$J(\mathbf{w}) = \frac{|\mu_1 - \mu_2|^2}{s_1^2 + s_2^2}$$

In other words, maximizing $J(\cdot)$ leads to the best separation between the projected samples.

To obtain $J(\cdot)$ as an explicit function of $\mathbf{w}$, we define the *scatter matrices* by

$$\mathbf{S}_1 = \sum_{i:y_i=+1} (\mathbf{x}_i - \mathbf{c}_1)(\mathbf{x}_i - \mathbf{c}_1)^T$$

and

$$\mathbf{S}_2 = \sum_{i:y_i=-1} (\mathbf{x}_i - \mathbf{c}_2)(\mathbf{x}_i - \mathbf{c}_2)^T$$

Then, we can write

$$\begin{aligned} s_1^2 &= \sum_{i:y_i=+1} (\mathbf{w}^T\mathbf{x}_i - \mu_1)^2 \\ &= \sum_{i:y_i=+1} (\mathbf{w}^T\mathbf{x}_i - \mathbf{w}^T\mathbf{c}_1)^2 \\ &= \sum_{i:y_i=+1} \mathbf{w}^T(\mathbf{x}_i - \mathbf{c}_1)(\mathbf{x}_i - \mathbf{c}_1)^T\mathbf{w} \\ &= \mathbf{w}^T\mathbf{S}_1\mathbf{w} \end{aligned}$$

Similarly,

$$s_2^2 = \mathbf{w}^T \mathbf{S}_2 \mathbf{w}$$

Therefore the denominator of $J(\mathbf{w})$ can be written as follows

$$s_1^2 + s_2^2 = \mathbf{w}^T \mathbf{S}_W \mathbf{w}$$

where $\mathbf{S}_W = \mathbf{S}_1 + \mathbf{S}_2$

The separation between the projected means is given by

$$\begin{aligned}(\mu_1 - \mu_2)^2 &= (\mathbf{w}^T\mathbf{c}_1 - \mathbf{w}^T\mathbf{c}_2)^2 \\ &= \mathbf{w}^T(\mathbf{c}_1 - \mathbf{c}_2)(\mathbf{c}_1 - \mathbf{c}_2)^T\mathbf{w} \\ &= \mathbf{w}^T\mathbf{S}_B\mathbf{w}\end{aligned}$$

where

$$\mathbf{S}_B = (\mathbf{c}_1 - \mathbf{c}_2)(\mathbf{c}_1 - \mathbf{c}_2)^T$$

- $\mathbf{S}_W$ is called the *within-class scatter matrix.*
- $\mathbf{S}_B$ is called the *between-class scatter matrix.*
- For any $\mathbf{w}$, $\mathbf{S}_B\mathbf{w}$ is in the direction of $\mathbf{c}_1 - \mathbf{c}_2$.
- In terms of $\mathbf{S}_B$ and $\mathbf{S}_W$, the criterion function can be written as

$$J(\mathbf{w}) = \frac{\mathbf{w}^T\mathbf{S}_B\mathbf{w}}{\mathbf{w}^T\mathbf{S}_W\mathbf{w}}$$

- It is easy to show that a vector that maximizes $J(\mathbf{w})$ must satisfy

$$\mathbf{S}_B\mathbf{w} = \lambda\mathbf{S}_W\mathbf{w}$$

  for some constant λ, which is a generalized eigenvalue problem.

- If $\mathbf{S}_W$ is nonsingular, we can obtain a conventional eigenvalue problem by writing

$$\mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{w} = \lambda\mathbf{w}$$

- In this case, it is unnecessary to solve for eigenvalues and eigenvectors of $\mathbf{S}_W^{-1}\mathbf{S}_B$ due to the fact that $\mathbf{S}_B\mathbf{w}$ is in the direction of $\mathbf{c}_1 - \mathbf{c}_2$.
- Because the magnitude of $\mathbf{w}$ is immaterial, we can write the $\mathbf{w}^*$ that maximizes $J(\mathbf{w})$ as follows

$$\mathbf{w}^* = \mathbf{S}_W^{-1}(\mathbf{c}_1 - \mathbf{c}_2)$$

A Numerical Example

Given the samples belonging to class 1

$$\mathbf{x}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \mathbf{x}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \mathbf{x}_3 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

$$\mathbf{c}_1 = \frac{1}{3}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 2 \\ 2 \end{bmatrix}\right) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

and the samples belonging to class 2

$$\mathbf{x}_4 = \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \mathbf{x}_5 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \mathbf{x}_6 = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$$

$$\mathbf{c}_2 = \frac{1}{3}\left(\begin{bmatrix} -1 \\ 0 \end{bmatrix} + \begin{bmatrix} -1 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 \\ 2 \end{bmatrix}\right) = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

$$\mathbf{S}_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \begin{bmatrix} -1 & -1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$$

$$\mathbf{S}_2 = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \begin{bmatrix} 0 & -1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix}$$

$$\mathbf{S}_W = \mathbf{S}_1 + \mathbf{S}_2 = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 2 & 4 \end{bmatrix}$$

$$\mathbf{S}_B = (\mathbf{c}_1 - \mathbf{c}_2)(\mathbf{c}_1 - \mathbf{c}_2)^T = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \begin{bmatrix} 2 & 0 \end{bmatrix} = \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\mathbf{S}_W^{-1} = \frac{1}{4}\begin{bmatrix} 4 & -2 \\ -2 & 2 \end{bmatrix}$$

$$\mathbf{w}^* = \mathbf{S}_W^{-1}(\mathbf{c}_1 - \mathbf{c}_2) = \frac{1}{4}\begin{bmatrix} 4 & -2 \\ -2 & 2 \end{bmatrix}\begin{bmatrix} 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \end{bmatrix}$$

3.8.3 Multiple Discriminant Analysis

- Multiple discriminant methods seek the subspace with the greatest separation of the projected distributions (Figure 10).
- In particular, what we seek is a d-by-d' projection matrix that in some sense maximizes the ratio of the between-class scatter to the within-class scatter.
- A simple scalar measure of scatter is the determinant of the scatter matrix.
- The determinant is the product of the eigenvalues, and hence is the product of the variances in the principal directions.

- Using this measure, we obtain the criterion function

$$J(\mathbf{W}) = \frac{|\mathbf{W}^T \mathbf{S}_B \mathbf{W}|}{|\mathbf{W}^T \mathbf{S}_W \mathbf{W}|}$$

- The columns of the optimal $\mathbf{W}$ are the generalized eigenvectors that correspond to the largest eigenvalues in

$$\mathbf{S}_B \mathbf{w}_i = \lambda_i \mathbf{S}_W \mathbf{w}_i.$$

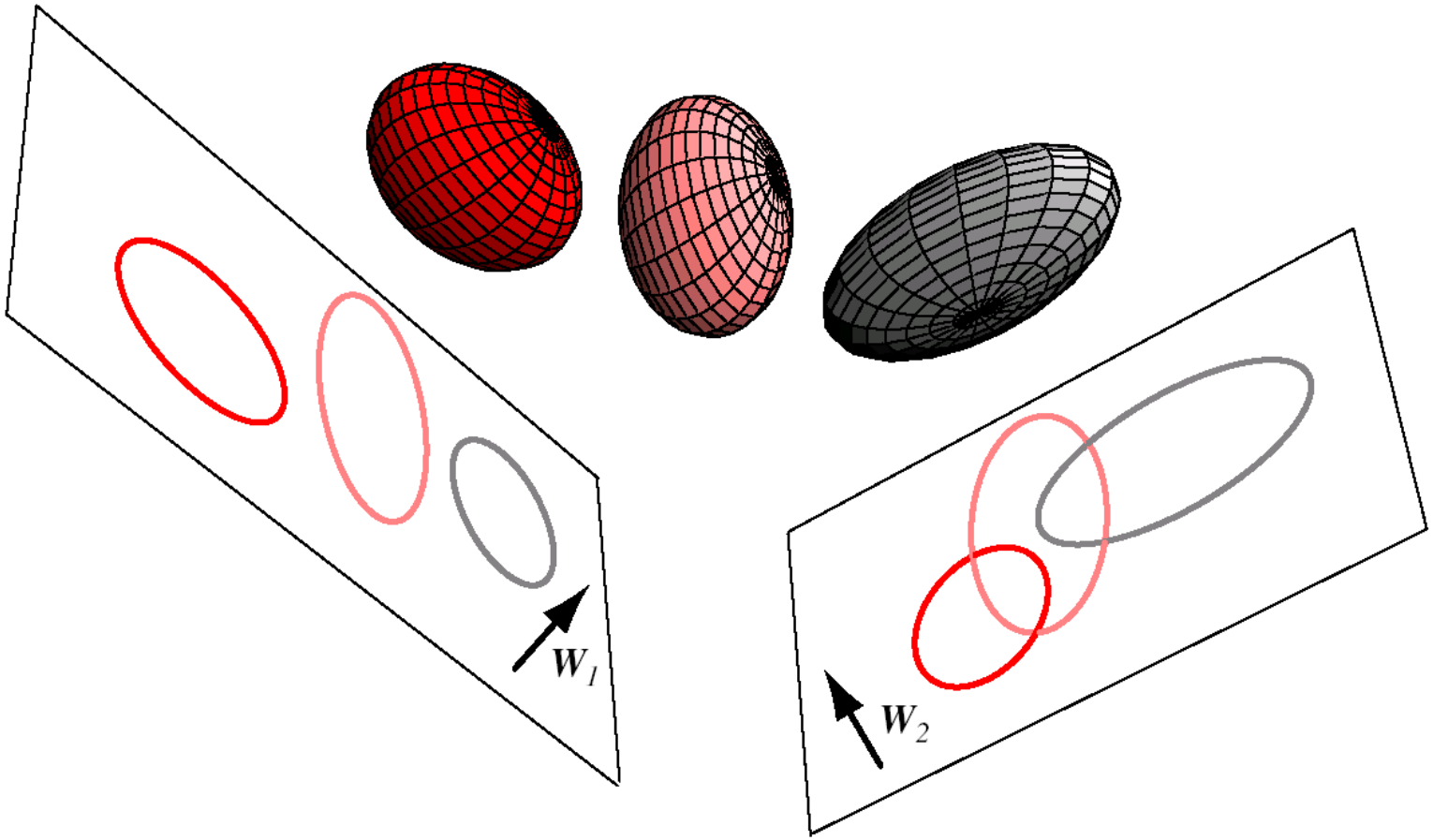


Figure 10: Three three-dimensional distributions are projected onto two-dimensional subspaces, described by normal vectors $\mathbf{W}_1$ and $\mathbf{W}_2$.

Historical Remarks

- Principal component analysis is a classical statistical technique [1]; it is very useful in a wide range of engineering applications.
- Fisher's early work on linear discriminants [2] is well described in [3] and a number of textbooks [4].

3.9 Expectation-Maximization (EM)

Historical Remarks

- The expectation-maximation algorithm is due to Dempster et al. [5], and a thorough overview appears in [6].
- On-line or increamental version of EM are described in [7, 8].

3.1 Introduction

3.2 Maximum-Likelihood Estimation

3.3 Bayesian Estimation

3.4 Bayesian Parameter Estimation: Gaussian Case

3.5 Bayesian Parameter Estimation: General Theory

3.7 Problems of Dimensionality

3.8 Component Analysis and Discriminants

3.9 Expectation-Maximization (EM)

3.10 **Hidden Markov Models**

3.10 Hidden Markov Models

- We now turn to the problem of making a sequence of decisions.
- In problems that have an inherent temporality we may have state at time t that is influenced by the state at time $t-1$.
- Hidden Markov models have found greatest use in such problems – for instance, speech recognition.
- While notations are inevitably more complicated than the models considered up to this point, we emphasize that the same underlying ideas are exploited.

3.10.1 First-Order Markov Models

- We consider a sequence of states at successive times; the state at time t is denoted by $\omega(t)$.
- A sequence of length T is denoted by $\boldsymbol{\omega}^T = \{\omega(1), \omega(2), \ldots, \omega(T)\}$, as for instance we might have $\boldsymbol{\omega}^6 = \{\omega_1, \omega_4, \omega_2, \omega_2, \omega_1, \omega_4\}$
- Note that the system can revisit a state at different steps, and not every state need to be visited.

- Our model for the production of a sequence is described by **transition probabilities** $P(\omega(t+1) = \omega_j \mid \omega(t) = \omega_i) = a_{ij}$ – the probability of having ω_j at time $t+1$ given that the state at time t was ω_i.
- In general, the transition probabilities are nonsymmetric, i.e., $a_{ij} \neq a_{ji}$, and a state may be visited in succession, i.e., $a_{ii} \neq 0$.

- Suppose we are given a model $\boldsymbol{\theta}$, i.e., the values of a_{ij}, as well as a sequence $\boldsymbol{\omega}^T$
- In order to calculate the probability that the model generated the sequence, we simply multiply the successive probabilities.
- For instance, to find the probability that a particular model generated the sequence $\boldsymbol{\omega}^6 = \{\omega_1, \omega_4, \omega_2, \omega_2, \omega_1, \omega_4\}$, we have $P(\boldsymbol{\omega}^6|\boldsymbol{\theta}) = a_{14} \cdot a_{42} \cdot a_{22} \cdot a_{21} \cdot a_{14}$.
- If there is a prior probability on the first state $P(\omega(1) = \omega_i)$, we could include such a factor as well.

- Up to here we have been discussing a Markov model, or technically speaking, a first-order Morkov model, because the state at $t+1$ depends only on the state at t.
- However, we do not always have access to the state $\omega(t)$.
- Thus, we will have to augment Markov model to allow for **visible states** – which are directly accessible – as separated from the **hidden states**, which are not.

3.10.2 First-Order Hidden Markov Models

- Let $\mathbf{V}^T = \{v(1), v(2), \ldots, v(T)\}$ denote a sequence of visible states and thus we might have $\mathbf{V}^6 = \{v_5, v_1, v_1, v_5, v_2, v_3\}$.
- At time t, a hidden state $\omega(t) = \omega_j$ has the probability of emitting a visible state $v(t) = v_k$. We denote this probability by $P(v(t) = v_k \mid \omega(t) = \omega_j) = b_{jk}$.
- Such a model is called **Hidden Markov model** (Fig. 11).

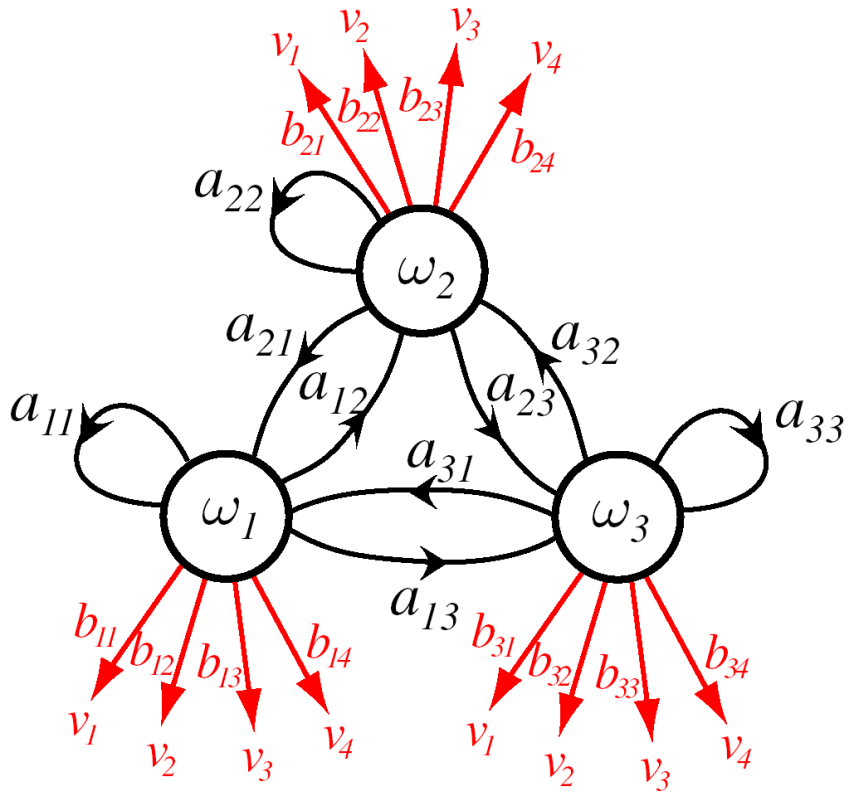


Figure 11: The transitions between hidden states are shown in black while the emission probabilities of visible states are shown in red.

3.10.3 Hidden Markov Model Computation

- We continue to define some new terms and clarify our notation.
- A **final** or **absorbing state** ω_0 is one which, if entered, is never left. (i.e., $a_{00} = 1$).
- As mentioned. we denote the transition probabilities among hidden states a_{ij} and b_{jk} for the probability of the emission of a visible state:

$$a_{ij} = P(\omega_j(t+1) \mid \omega_i(t)) \tag{61}$$

$$b_{jk} = P(v_k(t) \mid \omega_j(t)) \tag{62}$$

- We assume that a transition occurs from t to $t+1$ and that a visible state is emitted at each time step.
- Thus, we have

$$\sum_j a_{ij} = 1 \qquad \text{for all } i \tag{63}$$

$$\sum_k b_{jk} = 1 \qquad \text{for all } j \tag{64}$$

- With these preliminaries, we now discuss the three central issues in HMMs, namely, the evaluation, decoding, and learning.

3.10.4 Evaluation

- The probability that the model produces a sequence of visible states is

$$P(\mathbf{V}^T) = \sum_{r=1}^{r_{max}} P(\mathbf{V}^T \mid \boldsymbol{\omega}_r^T) \cdot P(\boldsymbol{\omega}_r^T) \tag{65}$$

  where r indexes a particular sequence
  $\boldsymbol{\omega}_r^T = \{\omega(1), \omega(2), \ldots, \omega(T)\}$ of T hidden states.

- In the case of c hidden states, there will be $r_{max} = c^T$ possible terms in the sum of Eq. (65).

- The second term in Eq. (65) can be rewritten as:

$$P(\boldsymbol{\omega}_r^T) = \prod_{t=1}^{T} P(\omega(t) \mid \omega(t-1)) \tag{66}$$

that is, a product of the a_{ij} according to the sequence of hidden states.

- We can also write the first term in Eq. (65) as:

$$P(\mathbf{V}^T \mid \boldsymbol{\omega}_r^T) = \prod_{t=1}^{T} P(v(t) \mid \omega(t)) \tag{67}$$

that is, a product of b_{jk} according the sequence of visible states and the corresponding hidden states.

- We can now express Eq. (65) as

$$P(\mathbf{V}^T) = \sum_{r=1}^{r_{max}} \prod_{t=1}^{T} P(v(t) \mid \omega(t)) \cdot P(\omega(t) \mid \omega(t-1)) \qquad (68)$$

- The probability that we observe a particular visible sequence is equal to sum over all possible sequences of hidden states.

- **This is an $\mathcal{O}(c^T T)$ calculation, which is quite prohibitive in practice.**
- For instance, if $c = 10$ and $T = 20$, we must perform on the order of 10^{21} calculations.
- In order to reduce the computational complexity, we calculate $P(\mathbf{V}^T)$ recursively.
- Note that each term $P(v(t) \mid \omega(t)) \cdot P(\omega(t) \mid \omega(t-1))$ involves only $v(t), \omega(t)$ and $\omega(t-1)$.

- We define

$$\alpha_j(t) = \begin{cases} 0 & t = 0 \text{ and } j \neq \text{initial state} \\ 1 & t = 0 \text{ and } j = \text{initial state} \\ \left[\sum_i \alpha_i(t-1)a_{ij}\right] b_{jk}v(t) & \text{otherwise,} \end{cases} \tag{69}$$

  where $b_{jk}v(t)$ means the transition probability b_{jk} selected by the visible state $v(t)$.

- Consequently, $\alpha_i(t)$ represents the probability that the HMM is in hidden state ω_i at step t having generated the first t elements of $\mathbf{V}^T$.

Algorithm 2. (HMM Forward)

1 **initialize** $t \leftarrow 0, a_{ij}, b_{jk}, \mathbf{V}^T, \alpha_j(0)$

2 **for** $t \leftarrow t + 1$

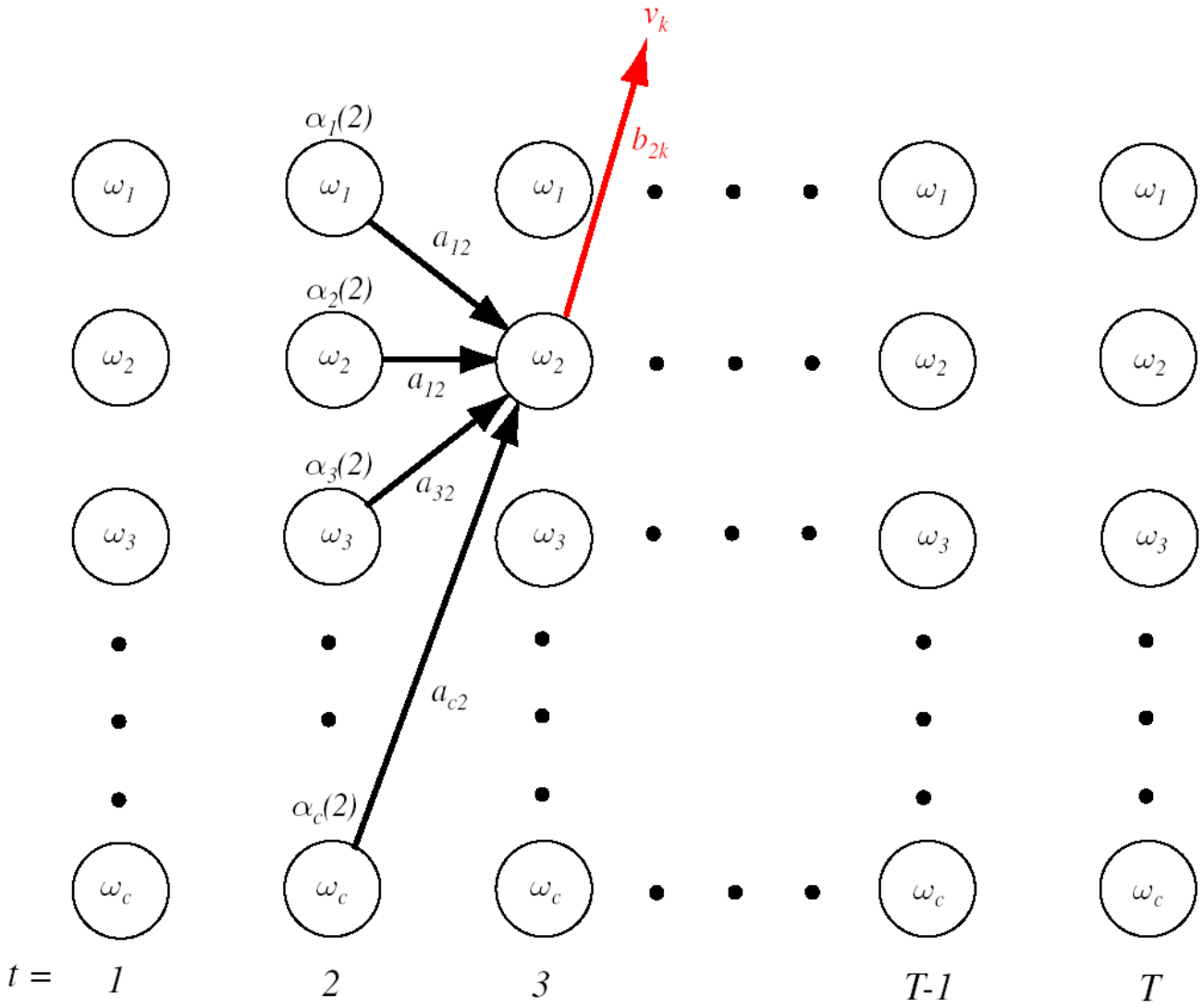
3 $\alpha_j(t) \leftarrow b_{jk} v(t) \sum_{i=1}^{c} \alpha_i(t-1) a_{ij}$

4 **until** $t = T$

5 **return** $P(\mathbf{V}^T) \leftarrow \alpha_0(T)$ for the final state

6 **end**

- In line 5, α_0 denotes the probability of the associated sequence ending to the known final state.
- The *Forward Algorithm* has a computational complexity of $\mathcal{O}(c^2\ T)$.
- For $c = 10$ and $T = 20$, we would need only on the order of 2000 calculations.

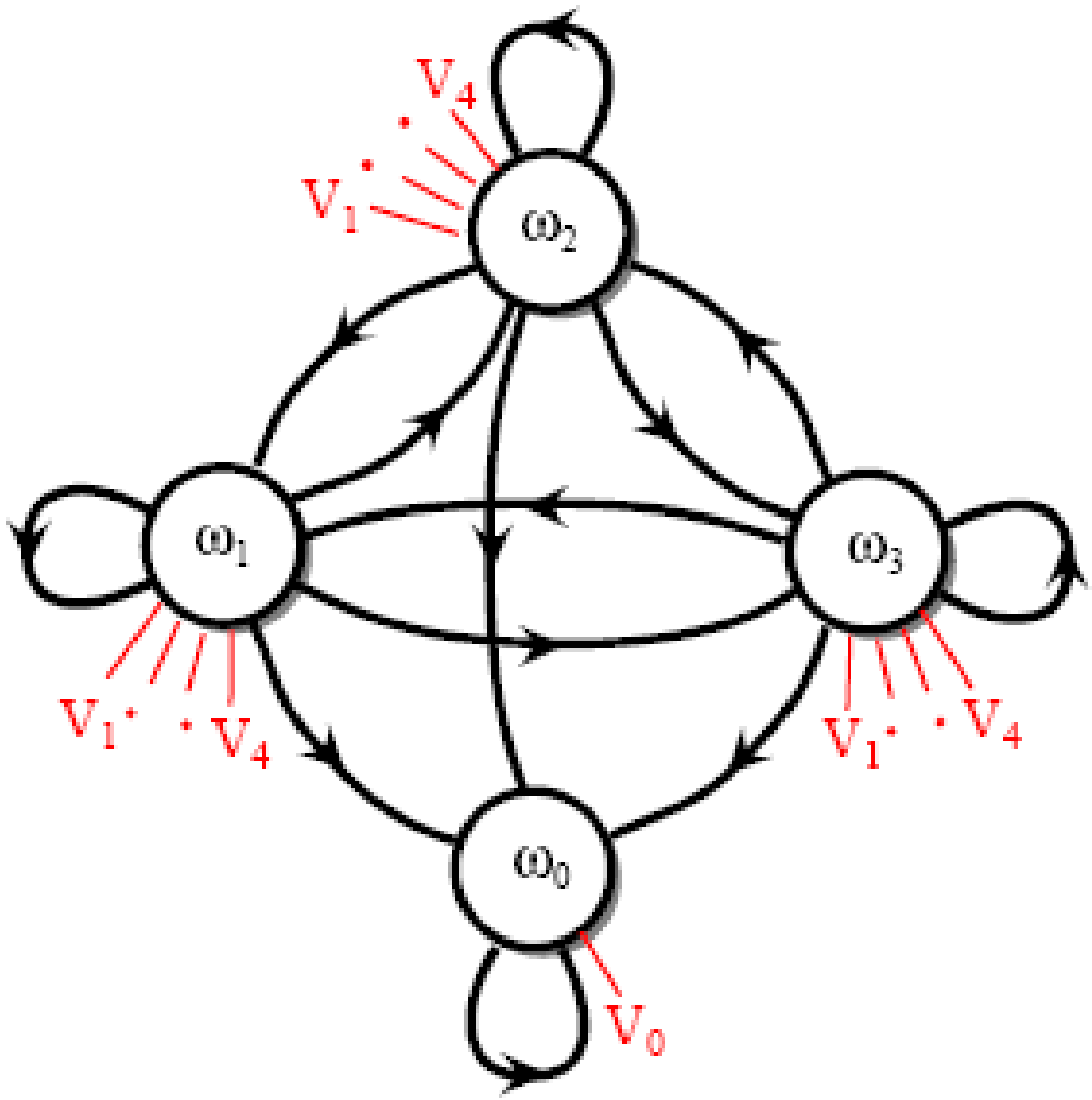

v_k
b_{2k}
$\alpha_1(2)$
$\alpha_2(2)$
$\alpha_3(2)$
$\alpha_c(2)$
a_{12}
a_{12}
a_{32}
a_{c2}
ω_1
ω_2
ω_3
ω_c
t = 1
2
3
T-1
T

- The forward algorithm can be visualized by means of a trellis – unfolding of the HMM through time.
- Suppose we seek the probability that the HMM was in state ω_2 at $t = 3$ and generate the observed sequence up through that step.
- At time step $t = 2$, the probability that the HMM was in state ω_i and generated the observed sequence is denoted by $\alpha_i(2)$.
- For the previous figure, we have $\alpha_2(3) = b_{2k} \sum_{i=1}^{c} \alpha_i(2) \cdot a_{i2}$.

Example 2

Hidden Markov Model

Consider the HMM with an absorber state ω_0 and a null visible state v_0.


V_4
V_1
ω_2
ω_1
ω_3
V_1
V_4
V_1
V_4
ω_0
V_0

The transition probabilities are given by

$$a_{ij} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0.2 & 0.3 & 0.1 & 0.4 \\ 0.2 & 0.5 & 0.2 & 0.1 \\ 0.8 & 0.1 & 0.0 & 0.1 \end{bmatrix} \tag{70}$$

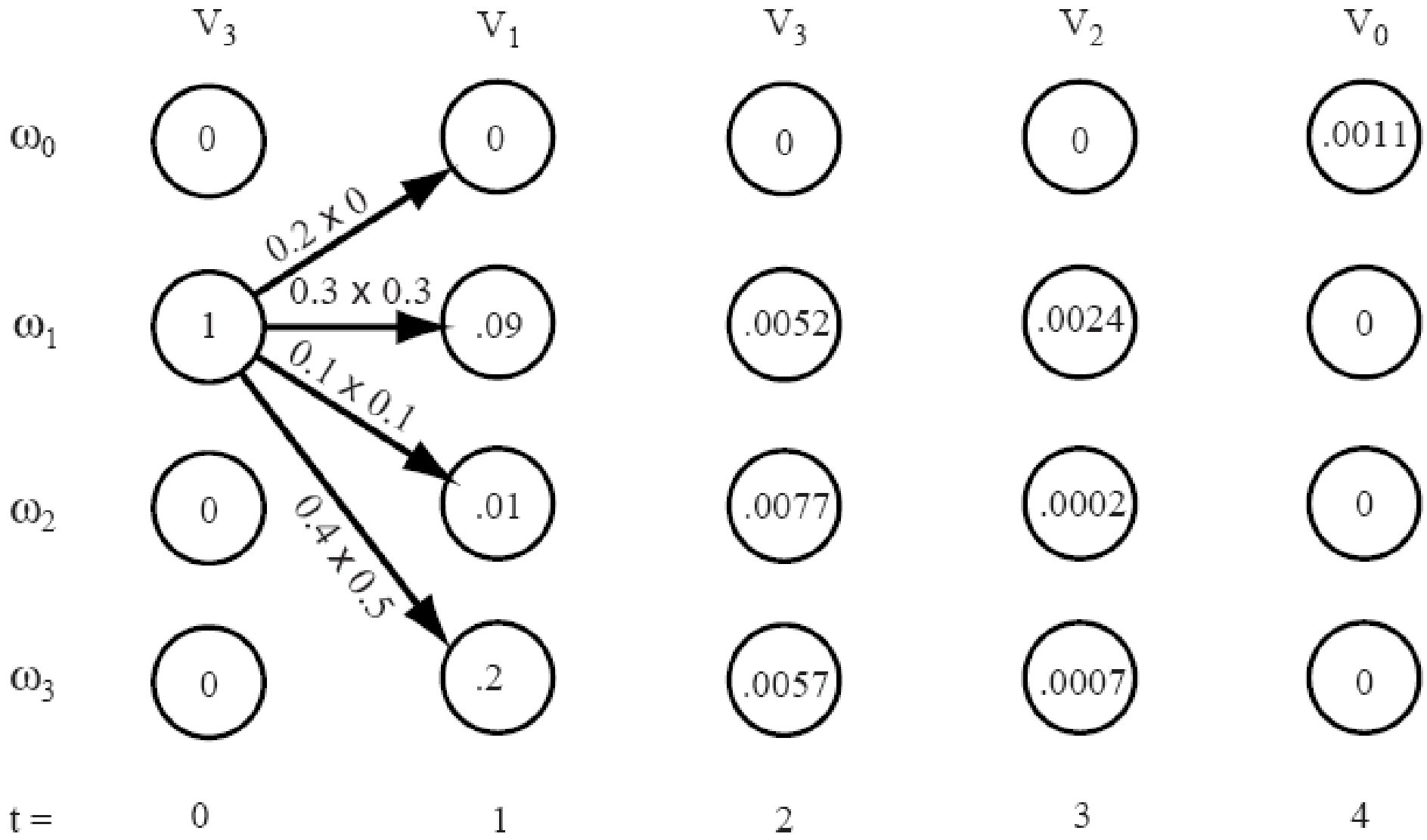
and

$$b_{ij} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.3 & 0.4 & 0.1 & 0.2 \\ 0 & 0.1 & 0.1 & 0.7 & 0.1 \\ 0 & 0.5 & 0.2 & 0.1 & 0.2 \end{bmatrix} \tag{71}$$

- What is the probability it generates the sequence $\mathbf{V}^4 = \{v_1, v_3, v_2, v_0\}$?

- The HMM consists of four hidden states (each emits one of five visible states).
- Suppose we know the initial hidden state to be ω_1, and thus $\alpha_1(0) = 1$ and $\alpha_j(0) = 0$ for $j \neq 1$..
- The visible state at each time step is shown at the top of the figure.
- $\alpha_j(t)$, the probability that the model generated the observed sequence up to time t, is shown in each circle.
- The product $a_{ij} \cdot b_{jk}$ is shown along each transition link for the step $t = 0$ to $t = 1$.

- Because visible state v_1 was emitted at $t = 1$, we have $\alpha_0(1) = \alpha_1(0) \cdot a_{10} \cdot b_{01}$. Likewise, $\alpha_1(1) = \alpha_1(0) \cdot a_{11} \cdot b_{11}$
- In this example, the calculation of $\alpha_j(1)$ is particularly simple because of the known initial hidden state.
- For subsequent times, the calculation requires a sum over all hidden states at previous time.
- The final probability, $P(\mathbf{V}^T \mid \boldsymbol{\theta})$, is hence 0.0011.


V3
V1
V3
V2
V0
ω0
ω1
ω2
ω3
0
0
0
0
.0011
1
.09
.0052
.0024
0
0
.01
.0077
.0002
0
0
.2
.0057
.0007
0
0.2 x 0
0.3 x 0.3
0.1 x 0.1
0.4 x 0.5
t =
0
1
2
3
4

- If we denote our model – a_{ij} and b_{jk} – by $\boldsymbol{\theta}$, the probability of the model given the observations $\mathbf{V}^T$ is

$$P(\boldsymbol{\theta} \mid \mathbf{V}^T) = \frac{P(\mathbf{V}^T \mid \boldsymbol{\theta}) \cdot P(\boldsymbol{\theta})}{P(\mathbf{V}^T)} \tag{72}$$

- We would have a number of HMMs, one for each category, and associate an observed sequence to the model with the highest probability.

- For instance, we would have a model for "cat" and the other one for "dog" in speech recognition. For a test utterance, we determine which model has the highest probability.

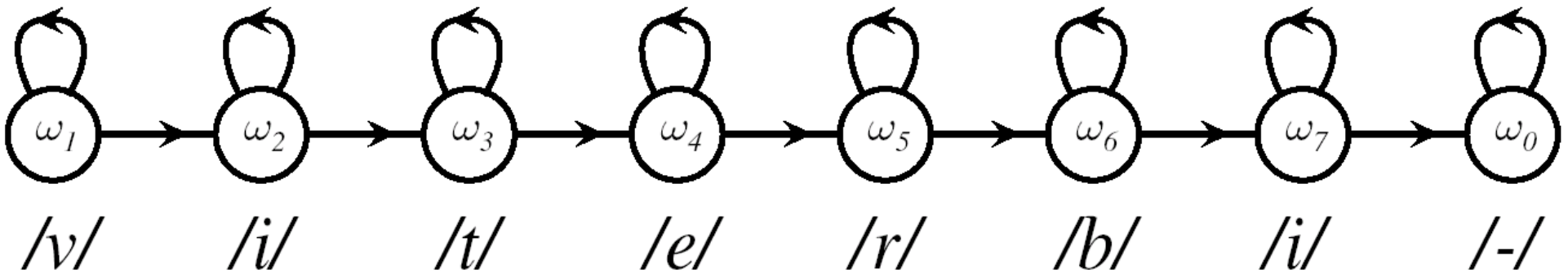


Figure 12: A left-to-right HMM is commonly used in speech recognition.

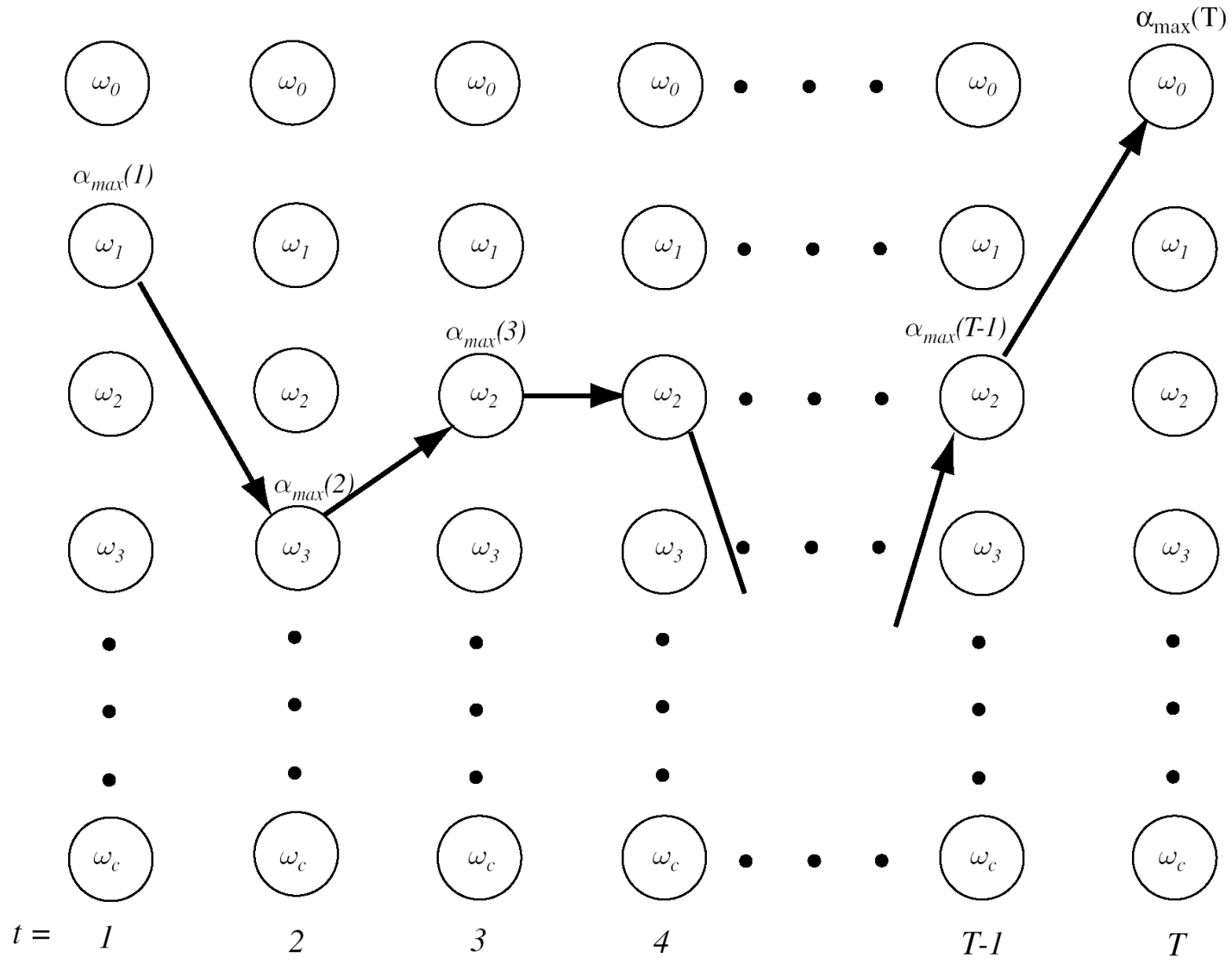
3.10.5 Decoding

- Given an observed sequence $\mathbf{V}^T$, the decoding problem is to find the most probable sequence of hidden states.
- While we might enumerate every possible sequence of hidden states, this is an $\mathcal{O}(c^T T)$ calculation and prohibitive.
- Instead, we use perhaps the simplest decoding algorithm, shown in the next slide.

Algorithm 4. (HMM Decoding)

01 **begin initialize** $Path \leftarrow \{\}, t \leftarrow 0$
02 **for** $t \leftarrow t + 1$
03 $j \leftarrow 0$
04 **for** $j \leftarrow j + 1$
05 $\alpha_j(t) \leftarrow b_{jk} v(t) \sum_{i=1}^{c} \alpha_i(t-1) a_{ij}$
06 **until** $j = c$
07 $j' \leftarrow \arg\min_j \alpha_j(t)$
08 Append $\omega_{j'}$ to Path
09 **until** $t = T$
10 **return** Path
11 **end**

- The black line in the following figure corresponds to Path.
- The Path connects the hidden states with the highest value of α_j at each step t.
- Note that these is no guarantee that the path is a valid one.
- For instance, it is possible that a path is forbidden by the model, as illustrated in Example 3.


$\alpha_{max}(T)$
ω_0
$\alpha_{max}(1)$
ω_1
$\alpha_{max}(3)$
$\alpha_{max}(T\text{-}1)$
ω_2
$\alpha_{max}(2)$
ω_3
ω_c
t = 1 2 3 4 T-1 T

Example 3

HMM Decoding

We find the path $\{\omega_1, \omega_3, \omega_2, \omega_1, \omega_0\}$ for the data of Example 2. However, the transition from ω_3 to ω_2 is not allowed according to the a_{ij} in Example 2.

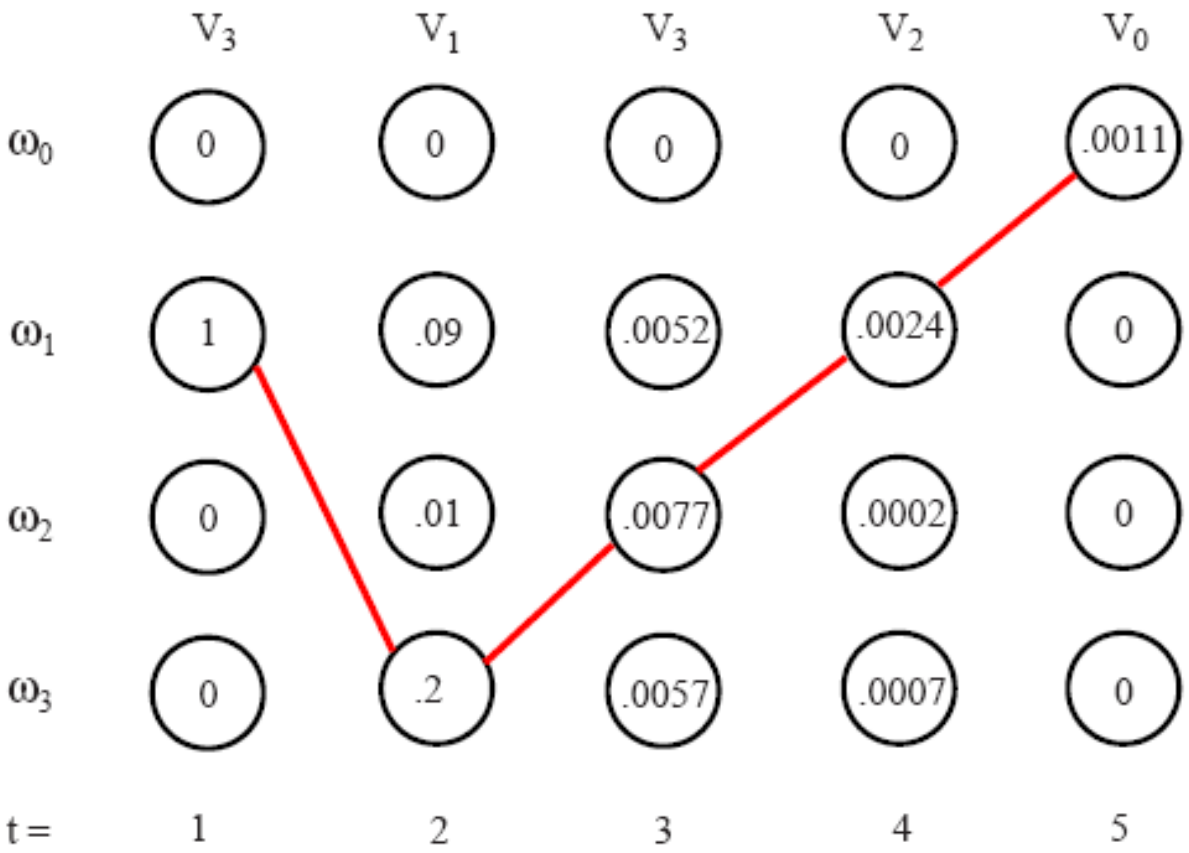


Figure 13: The locally optimal path through the HMM trellis of Example 2.

3.10.6 Learning

- Here, the goal is to determine model parameters from a set of training samples.
- There is no known method for obtaining the optimal set of parameters.
- But, we can obtain a good solution by using the following algorithm.

The Forward-Backward Algorithm

- The parameters are iteratively updated so that the model could better explain the observed sequence.
- Recall that $\alpha_i(t)$ is defined as the probability that the model is in state $\omega_i(t)$ and has generated the observed sequence up to time t.
- We can analogously define $\beta_i(t)$ to be the probability that the model is in state $\omega_i(t)$ and *will generate* the reminder of the observed sequence, that is, from $t+1 \rightarrow T$.

- We express $\beta_i(t)$ as

$$\beta_i(t) = \begin{cases} 0 & \omega_i(T) \neq \omega_0 \\ 1 & \omega_i(T) = \omega_0 \\ \sum_j \beta_j(t+1) a_{ij} b_{jk} v(t+1) & \text{otherwise.} \end{cases} \tag{73}$$

- To understand Eq. (73), imaging we know $\alpha_i(t)$ up to time $T-1$, and we want to calculate the probability that the model would generate the remaining visible state.
- This probability, $\beta_i(T)$, is just the probability that we make a transition to $\omega_i(T)$ multiplied by the probability that this hidden state emits the correct final visible state.
- This will be either 0 (if $\omega_i(T)$ is not the final state) or 1 (if it is).
- Thus, it is clear that $\beta_i(T-1) = \sum_j a_{ij} b_{jk} v(T) \beta_j(T)$.
- We can repeat the process, to determine $\beta_i(T-2)$, and so on.

- But, we don't know the true value of a_{ij} and b_{jk}.
- To deal with this issue, we define $\gamma_{ij}(t)$ as follows:

$$\gamma_{ij}(t) = \frac{\alpha_i(t-1)a_{ij}b_{jk}\beta_j(t)}{P(\mathbf{V}^T|\boldsymbol{\theta})} \tag{74}$$

  where $P(\mathbf{V}^T|\boldsymbol{\theta})$ is the probability that the model generates $\mathbf{V}^T$.
- Thus, $\gamma_{ij}(t)$ is the probability of a transition from $\omega_i(t-1)$ to $\omega_j(t)$ given that the model generates the observed sequence $\mathbf{V}^T$.

- The estimate $\hat{a}_{ij}$ is found by taking the ratio between the expected number of transition from ω_i to ω_j and the total expected number of any transitions from ω_i.

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T} \gamma_{ij}(t)}{\sum_{t=1}^{T} \sum_{k} \gamma_{ik}(t)} \tag{75}$$

- The estimate $\hat{b}_{jk}$ is calculated by the ratio between the frequency that v_k is emitted and that for any symbol.

$$\hat{b}_{jk} = \frac{\sum_{t=1, v(t)=v_k}^{T} \sum_{l} \gamma_{jl}(t)}{\sum_{t=1}^{T} \sum_{l} \gamma_{jl}(t)} \tag{76}$$

- In short, we start with rough estimates of a_{ij} and b_{jk}, calculate improved estimates by Eqs (75) and (76), and repeat until some convergence criterion is met.
- This is also known as **Baum-Welch algorithm**.
- The forward-backward algorithm is an instance of generalized expectation-maximization algorithm.

Algorithm 5. (Forward-Backward)

1 **begin initialize** $a_{ij}, b_{jk}, \mathbf{V}^T, \theta, z \leftarrow 0$

2 **do** $z \leftarrow z + 1$

3 compute $\hat{a}(z)$ from $a(z-1)$ and $b(z-1)$ by Eq. (75)

4 compute $\hat{b}(z)$ from $a(z-1)$ and $b(z-1)$ by Eq. (76)

5 $a_{ij}(z) \leftarrow \hat{a}_{ij}(z)$

6 $b_{jk}(z) \leftarrow \hat{b}_{jk}(z)$

7 **until** $\max_{i,j,k} [a_{ij}(z) - a_{ij}(z-1), b_{jk}(z) - b_{jk}(z-1)] < \theta$

8 **return** $a_{ij} \leftarrow a_{ij}(z), b_{jk} \leftarrow b_{jk}(z)$

9 **end**

Historical Remarks

- Hidden Markov models were introduced by Baum and collaborators [9, 10] and had their greatest applications in speech recognition [11, 12].
- Hidden Markov methods have been extended to two-dimensions and applied to recognizing characters [13].
- The decoding algorithm is related to pioneering work of Viterbi and followers [14, 15].

Summary

Parameter Estimation

- The maximum-likelihood method seeks to find the parameter values that is best supported by the data.
- In Bayesian estimation, the parameters are considered random variables having a know prior prior density; the samples convert this to an **a posteriori** density.
- The recursive Bayes method update a posterior incrementally.
- Maximum-likelihood methods are generally easier to implement while the Bayesian estimation is, in principle, to be preferred.

Fisher Linear Discriminant

- The fisher linear discriminant finds a subspace where categories are best separated; classification can then be performed in the subspace.
- Fisher's method can be extended to cases with multiple categories.

Expectation Maximization

- EM is an iterative scheme to find model parameter when some data are missing.
- The **E step** requires marginalizing over the missing variables given the current model.
- The model parameters are updated in the **M step**.

Hidden Markov Models

- HMMs consist of nodes representing hidden states, interconnected by links describing the transition probabilities.
- Each hidden state has an associated set of visible states
- The transition probabilities can be learned from sample sequences by means of **forward-backword** or **Baum-welch** algorithm.
- Classification proceeds by finding the model among candidates that is most likely to have produced a given observed sequence.

References

[1] I. Jolliffe, *Principal Component Analysis.* Springer, 2002.

[2] R. Fisher, "The use of multiple measurement in taxonomy problems," *Annals of Eugenics*, vol. 7, pp. 179–188, 1936.

[3] G. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition.* Wiley, 1992.

[4] K. Fukunaga, *Introduction to Statistical Pattern Recognition.* Academic Press, 1990.

[5] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the*

Royal Statistical Society. Series B (Methodological), vol. 39, no. 1, pp. 1–38, 1977.

[6] G. McLachlin and T. Krishnan, *The EM Algorithm and Extensions*. Wiley Series in Probability and Statistics, 1997.

[7] R. Jacobs and M. Jordan, "Hierarchical mixtures of experts and the EM algorithm," *Neural Computation*, vol. 6, pp. 181–214, 1992.

[8] D. Titterington, "Recursive parameter estimation using incomplete data," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 46, no. 2, pp. 257–267, 1984.

[9] L. Baum and T. Petrie, "Statistical inference for probabilistic

functions of finite state Markov chains," *The Annals of Mathematical Statistics*, vol. 37, no. 6, pp. 1554–1563, 1966.

[10] L. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *The Annals of Mathematical Statistics*, vol. 41, no. 1, pp. 164–171, 1970.

[11] L. Rabiner, "A tutorial on hidden Markov models and selected applications inspeech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[12] L. Rabiner and B. Juang, *Fundamentals of speech recognition*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1993.

[13] G. Kopec and P. Chou, "Document image decoding using Markov source models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 6, pp. 602–617, 1994.

[14] G. Forney Jr, "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.

[15] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.