Università di Verona

A.Y. 2021-22

Machine Learning & Artificial Intelligence

Parameter Estimation: Maximum Likelihood approach and Bayesian approach

Vittorio Murino

Introduction

- To create an optimal classifier that uses the Bayesian decision rule you need to know:
  - **Prior probabilities** $P(\omega_i)$
  - **Class-conditional densities** $p(\mathbf{x} \mid \omega_i)$

- The performance of a classifier <u>strongly</u> depends on the **goodness** of these components

- ***BUT PRACTICALLY ALL THIS INFORMATION IS NEVER AVAILABLE!!***

- More often we only have:
  - *A vague knowledge of the problem*, from which to extract vague a-priori probabilities.
  - *Some particularly representative patterns*, ***training data***, used to train the classifier (often too few!)

- Estimating a-priori probabilities is usually not particularly difficult.

- Estimating conditional densities is more complex.

- Given that the knowledge, although approximate, of a-priori densities does not present problems, regarding conditional densities the problems can be divided into:
  1. ***estimate** the **unknown function*** $p(\mathbf{x} \mid \omega_j)$
  2. ***estimate unknown parameters of known function*** $p(\mathbf{x} \mid \omega_j)$

e.g., estimate the vector $\boldsymbol{\theta}_j = (\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$

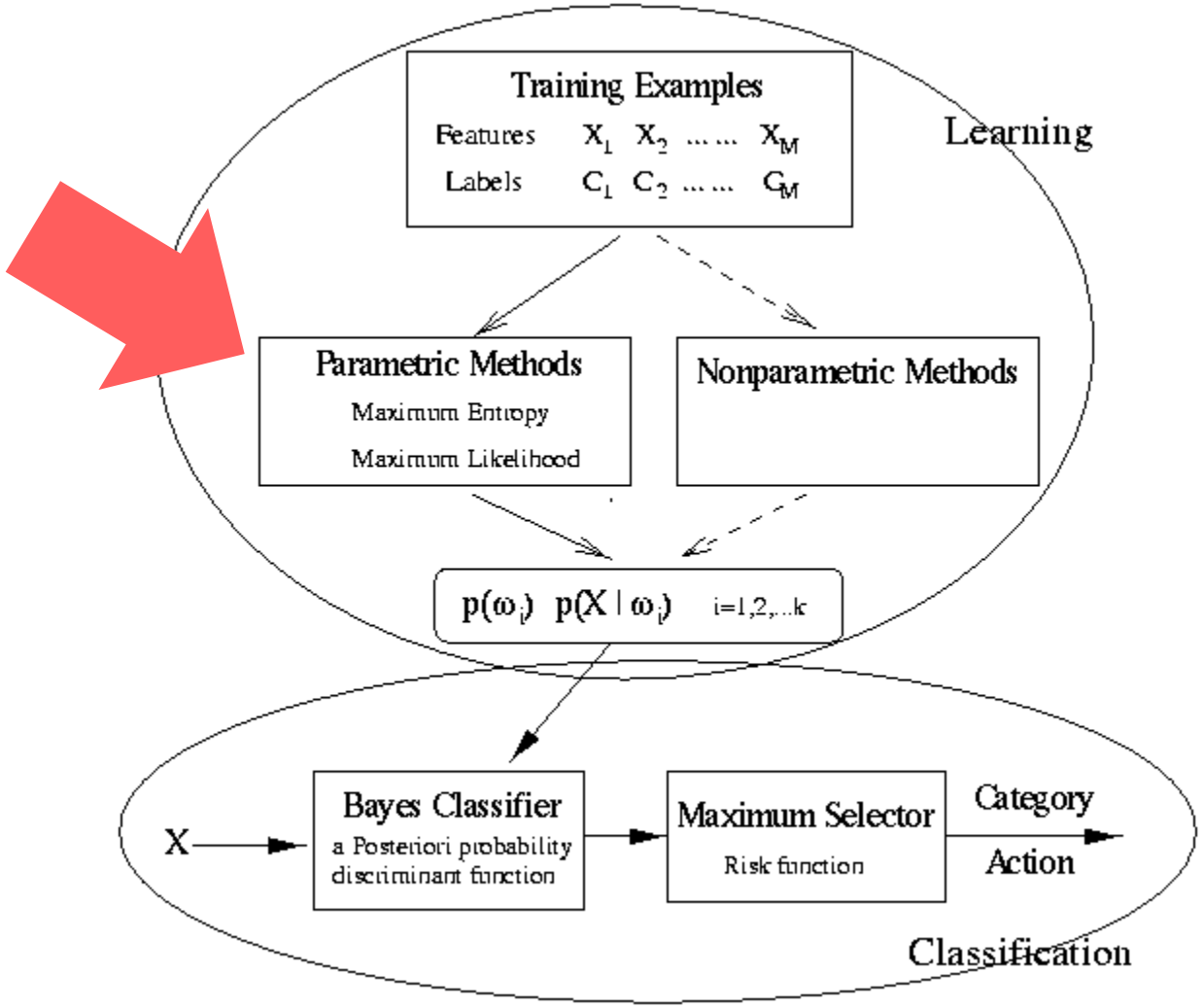
when $p(\mathbf{x} \mid \omega_j) \approx N(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$

Parameter estimation

- The second point is much simpler (though complex!) and represents a classic problem in statistics.

- Transferred to *pattern recognition*, a possible approach is to
  1) estimate the parameters from the training data
  2) use the resulting estimates as true values
  3) use the Bayesian decision theory to build a classifier

Overview


Training Examples
Features x_1 x_2 …… x_M
Labels c_1 c_2 …… c_M
Learning
Parametric Methods
Maximum Entropy
Maximum Likelihood
Nonparametric Methods
$p(\omega_i)$ $p(X \mid \omega_i)$ i=1,2,..k
X
Bayes Classifier
a Posteriori probability
discriminant function
Maximum Selector
Risk function
Category
Action
Classification

Parameter estimation - A priori probability

- Suppose we have a set of n training data in which each pattern is assigned an identity label (i.e., I know for sure which state ω_i the k-*th* pattern belongs to)

  ➔ ***Supervised* parameter learning problem**

- Then $$P(\omega_i) = \frac{n_i}{n}$$

  where n_i is the number of samples with label ω_i (operation that can be formally proved)

- This easy operation is not so useful, because the a-priori probabilities, in practice, are not so useful if compared to the conditional densities.

Parameter estimation - Problem instance

- Suppose we have *c* sets of data $D_1, D_2, \ldots, D_c$ sampled independently according to the density $p(x|\omega_j)$, assuming that $p(x|\omega_j)$ has a known parametric form

- The parameter estimation problem consists in estimating the parameters that define $p(x|\omega_j)$

- To simplify the problem, we also assume that:
  - the samples belonging to the set D_i do not give information about the parameters of $p(x|\ \omega_j)$ if $i \neq j$

Parameter estimation – Two approaches

- Specifically, the problem can be formulated as:
  - Given a training set D={x_1, x_2, ..., x_n}
  - $p(\mathbf{x}|\omega)$ is determined by $\boldsymbol{\theta}$, which is a vector representing the necessary parameters
    (e.g., $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$ if $p(\mathbf{x} \mid \omega) \approx N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$)
  - We want to find the best parameter $\boldsymbol{\theta}$ using the training set.

- There are two approaches:
  - Maximum likelihood estimation (ML)
  - Bayesian estimation

Parameter estimation – Two approaches (2)

- *Maximum Likelihood approach*
  - Parameters are seen as quantities whose values are *fixed* but *unknown*
  - The best estimation of their value is defined to be the one *that maximizes the probability of obtaining the samples actually observed (training data).*

- Bayesian approach
  - Parameters are seen as *random variables* having some known a-priori distribution
  - The observation of the samples converts these probabilities to a posterior density, revising our opinion about the true values of the parameters.
  - Adding training samples, the result is to refine the shape of the posterior density function, leading to a peak near the true values of the parameters (*Bayesian Learning* phenomenon).

- The results of the two approaches, although procedurally different, are qualitatively similar.

Maximum Likelihood approach

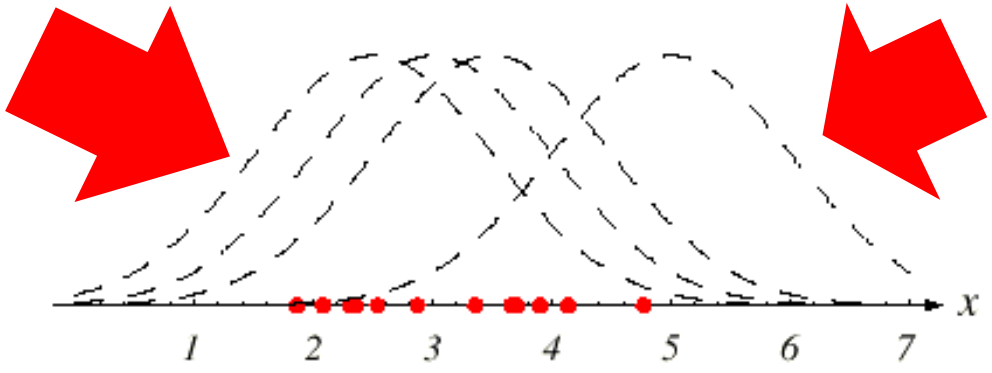
- Given the starting hypothesis of the problem, since the patterns of the set $\mathbf{D}$ are i.i.d. – independent and identically distributed, we have that:

$$p(\mathbf{D} \mid \boldsymbol{\theta}) = \prod_{k=1}^{n} p(x_k \mid \boldsymbol{\theta})$$

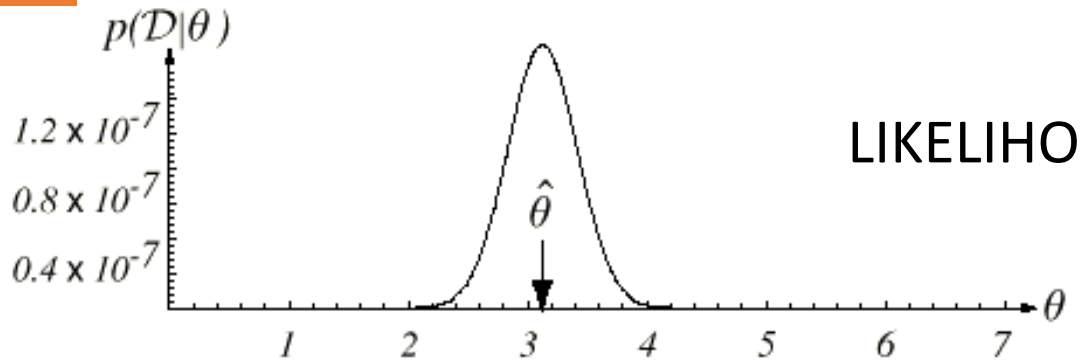
- Viewed as a function of $\boldsymbol{\theta}$, $p(\mathbf{D} \mid \boldsymbol{\theta})$ is called the ***likelihood*** of $\boldsymbol{\theta}$ with respect to the set of samples $\mathbf{D}$.
- The maximum likelihood estimate of $\boldsymbol{\theta}$ is, by definition, the value $\hat{\boldsymbol{\theta}}$ that maximizes $p(\mathbf{D} \mid \boldsymbol{\theta})$;
- Remember the assumption that $\boldsymbol{\theta}$ is fixed but unknown

Maximum Likelihood approach (2)

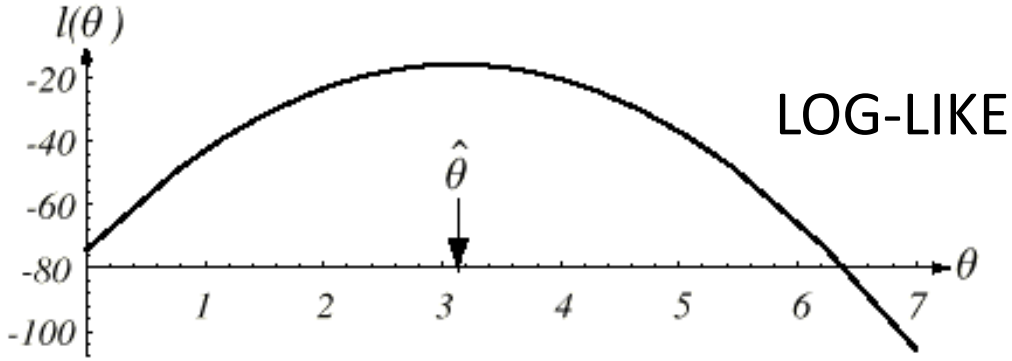
Training points in one dimension, known or assumed to be drawn from a Gaussian of a fixed variance, but unknown mean



4 among the infinite possible Gaussians



LIKELIHOOD



LOG-LIKELIHOOD

NB: the likelihood $p(\mathrm{D}|\theta)$ is a function of the mean θ, while the conditional density $p(\mathrm{x}|\theta)$ is a function of x

Maximum Likelihood approach (3)

- If the number of parameters to be set is *p*, let's $\boldsymbol{\theta}$ denote the *p*-component vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)^t$ and let

$$\nabla\boldsymbol{\theta} \equiv \begin{bmatrix} \dfrac{\partial}{\partial\theta_1} \\ \vdots \\ \dfrac{\partial}{\partial\theta_p} \end{bmatrix}$$

be the gradient operator.

- For analytical purposes it is easier to work with the logarithm of the likelihood.
- We define $l(\boldsymbol{\theta})$ as the log-likelihood function

$$l(\theta) \equiv \ln p(D \mid \theta) = \sum_{k=1}^{n} \ln p(x_k \mid \theta)$$

Maximum Likelihood approach (4)

- The goal is to determine the vector

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

where the dependence on the data set $\mathbf{D}$ is implicit.

- Thus, to obtain the maximum:

$$l(\boldsymbol{\theta}) \equiv \ln p(\mathbf{D} \mid \boldsymbol{\theta}) = \sum_{k=1}^{n} \ln p(x_k \mid \boldsymbol{\theta})$$

$$\nabla_{\theta} l(\boldsymbol{\theta}) = \sum_{k=1}^{n} \nabla_{\theta} \ln p(x_k \mid \boldsymbol{\theta})$$

from which we want to obtain $\nabla_{\theta} l(\boldsymbol{\theta}) = 0$

Maximum Likelihood approach (5)

- Formally, once the set of parameters has been estimated, it is necessary to check that the solution found is actually a global maximum, rather than a local maximum or an inflection point or, even worse, a minimum point.

- You also need to control what happens at the borders (boundaries) of the parameters space.

- We now apply the ML approach to some specific cases.

Maximum Likelihood: the Gaussian Case

- Suppose that the samples are drawn from a multivariate normal population with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.
- For simplicity, let's first consider the case where only the mean $\boldsymbol{\mu}$ is unknown.
- Under this condition, we consider a sample point $\mathbf{x}_k$ and find

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

$$\Downarrow$$

$$\nabla_{\boldsymbol{\mu}} \ln p(\mathbf{x}_k \mid \boldsymbol{\mu}) = \boldsymbol{\Sigma}^{-1}(\mathbf{x}_k - \boldsymbol{\mu})$$

Maximum Likelihood: the Gaussian Case (2)

- Identifying $\boldsymbol{\theta}$ with $\boldsymbol{\mu}$, we see that the Maximum-Likelihood estimate for $\boldsymbol{\mu}$ must satisfy:

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

- Multiplying for $\boldsymbol{\Sigma}$ and rearranging the sum, we obtain

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

that is the arithmetic ***average*** of the training samples, sometimes written as $\hat{\boldsymbol{\mu}}_n$ to clarify its dependence on the number of samples.

Maximum Likelihood: the Gaussian Case (3)

- In the more general (and more typical) multivariate normal case, **neither the mean μ nor the covariance matrix Σ is known**.
- Consider first the univariate case with $\boldsymbol{\theta} = (\theta_1, \theta_2) = (\mu, \sigma^2)$
- The log-likelihood of a single point is

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

And its derivative is

$$\nabla_{\boldsymbol{\theta}} l = \nabla_{\boldsymbol{\theta}} \ln\, p(x_k \mid \boldsymbol{\theta}) = \begin{bmatrix} \frac{1}{\theta_2}(x_k - \theta_1) \\ -\frac{1}{2\theta_2} + \frac{(x_k - \theta_1)^2}{2\theta_2^2} \end{bmatrix}$$

Maximum Likelihood: the Gaussian Case (4)

- Equalizing to 0 and considering all the points we get:

$$\sum_{k=1}^{n} \frac{1}{\theta_2}(x_k - \hat{\theta}_1) = 0 \qquad -\sum_{k=1}^{n} \frac{1}{\hat{\theta}_2} + \sum_{k=1}^{n} \frac{(x_k - \hat{\theta}_1)^2}{\hat{\theta}_2^{\,2}} = 0$$

where $\hat{\theta}_1$ and $\hat{\theta}_2$ are the ML estimates for θ_1 and θ_2.

- By substituting $\hat{\mu} = \hat{\theta}_1$ and $\sigma^2 = \hat{\theta}_2$ we obtain the ML estimates for mean and variance

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

Maximum Likelihood: the Gaussian Case (5)

- The analysis of the multivariate case is basically very similar, just involving more manipulations. The result is:

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

- The maximum likelihood estimate for the variance is biased, that is, the expected value over all data sets of size *n* of the sample variance is not equal to the true variance

$$E\left\{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2\right\} = \frac{n-1}{n}\sigma^2 \neq \sigma^2$$

Maximum-Likelihood: other cases

- Besides the Gaussian density, there are also other density families that constitute as many families of parameters:
  - ***Exponential distribution*** $$p(x \mid \theta) = \begin{cases} \theta\, e^{-\theta\, x} & x \geq 0 \\ 0 & \text{altrimenti} \end{cases}$$
  - ***Uniform distribution*** $$p(x \mid \theta) = \begin{cases} 1/\theta & 0 \leq x \leq \theta \\ 0 & \text{altrimenti} \end{cases}$$
  - ***Distribution of multivariate Bernoulli***

Maximum-Likelihood – Error model

- In general, if the parametric models are valid, the maximum-likelihood classifier provides excellent results.

- Instead, if incorrect parametric families are used, the classifier produces strong errors
  - This happens even if it is known the parametric family to use, for example, if it is estimated as a parameter within a Gaussian distribution a too wide variance.

Maximum-Likelihood – Error model (2)

- *In fact, there is no error model that gives a confidence or reliability value to the parameterization obtained.*

- In addition, all training data must be available to apply the Maximum-Likelihood estimation
  - If we want to use <u>new</u> training data, we need to repeat again the Maximum-Likelihood estimation procedure.

Bayesian parameter estimation

- Unlike the ML approach, in which we assume $\boldsymbol{\theta}$ as fixed but unknown, the Bayesian parameter estimation approach considers $\boldsymbol{\theta}$ as a random variable.

- In this case the training dataset **D** allows us to convert a prior distribution $p(\boldsymbol{\theta})$ into a posterior probability density $p(\boldsymbol{\theta}|\mathbf{D})$

$$p(\boldsymbol{\theta}) \Rightarrow p(\boldsymbol{\theta}|\mathbf{D})$$

- Given the difficulty of the argument, it is necessary a step back to the concept of Bayesian classification

Bayesian estimation approach – Core idea

- The computation of the posterior probabilities $P(\omega_i|\boldsymbol{x})$ lies at the heart of Bayesian classification

- To create an optimal classifier that uses the Bayesian decision rule you need to know:

  → **Prior probabilities** $P(\omega_i)$

  → **Conditional densities** $p(\boldsymbol{x}|\omega_i)$

- When these quantities are unknown, the best we can do is to compute $p(\boldsymbol{x}|\omega_i)$ using ***all of the information at our disposal***.

Bayesian estimation approach – Core idea (2)

- Part of this *information* may come from:
  1. **Prior knowledge**
     - *Functional forms for unknown densities*
     - *Ranges for the values of unknown parameters*
  2. **Training samples**
     - If we let ***D*** denote the *set of samples*: our goal becomes to compute the posterior probabilities $P(\omega_i | \boldsymbol{x}, \boldsymbol{D})$

- From these probabilities, we can obtain the Bayes classifier.

Bayesian estimation approach – Core idea (3)

- Given the training set D, the Bayes' formula then becomes:

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

- Assumptions:
  - Reasonably, $P(\omega_i \mid D) \Rightarrow P(\omega_i)$
  - Since we are treating the supervised learning case, the training set D can be partitioned into c subsets $D_1, D_2, \ldots, D_c$, with the samples in D_i belonging to ω_i
  - The samples belonging to D_i have no influence on the parameters of $p(\mathbf{x} \mid \omega_j, D)$ if $i \neq j$.

Bayesian estimation approach – Core idea (4)

- These assumptions lead us to two consequences:
  1. It allows us to work with each class separately, i.e.

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

$$P(\omega_i \mid \mathbf{x}, D) = \frac{p(\mathbf{x} \mid \omega_i, D_i) P(\omega_i)}{\sum_{j=1}^{c} p(\mathbf{x} \mid \omega_j, D_j) P(\omega_j)}$$

Bayesian estimation approach – Core idea (5)

2. Because each class can be treated independently, we do not need distinctions among classes, so we can simplify our notation **reducing to** c **different istances of the same problem**, i.e.:

$$P(\omega_i \mid \mathbf{x}, D) = \frac{p(\mathbf{x} \mid \omega_i, D_i) P(\omega_i)}{\sum_{j=1}^{c} p(\mathbf{x} \mid \omega_j, D_j) P(\omega_j)}$$

$$p(\mathbf{x} \mid D)$$

$$p(\mathbf{x})$$

Use a set of samples D, drawn independently according to the fixed but unknown probability distribution $p(\mathbf{x})$, *to determine* $p(\mathbf{x}|D)$

Bayesian estimation approach – Core idea (6)

- In practice, the Bayesian learning process *estimates a model implicitly*, that is, it does not return a vector of parameters $\boldsymbol{\theta}$ visible, but a *distribution* on it, given by the training set available.
- The fact that $p(\mathbf{x})$ is unknown but with known parametric form is expressed by saying that $p(\mathbf{x}|\,\boldsymbol{\theta})$ is completely known.
- It is preferred to write $p(\mathbf{x}|\mathrm{D})$ instead of $p(\mathbf{x}|\,\boldsymbol{\theta})$ because it is more meaningful, although an underlying model exists (in fact the term $p(\mathbf{x}|\,\boldsymbol{\theta})$ will appear later).
- Any information you have before observing the samples is assumed to be contained in the <u>known</u> a-priori density $p(\boldsymbol{\theta})$.
- Observations convert the prior $p(\boldsymbol{\theta})$ into a posterior distribution $p(\boldsymbol{\theta}|\mathrm{D})$ that hopefully assumes a maximum in the true value of $\boldsymbol{\theta}$.

The parameter distribution

- Ingredients:
  - $p(\mathbf{x})$: unknown, but with known parametric form;
  - $\boldsymbol{\theta}$: *parameter vector, unknown*;
  - $p(\mathbf{x}|\boldsymbol{\theta})$: completely known (being the parametric form $p(\mathbf{x})$);
  - $p(\boldsymbol{\theta})$: *any <u>a-priori</u> information we might have to observe the samples*
    The observation of the samples converts this to a …
  - $p(\boldsymbol{\theta}|\mathrm{D})$: … *<u>posterior</u> density*, sharply peaked about the true value of $\boldsymbol{\theta}$.

The parameter distribution (2)

- What we are doing is actually observing how $p(\mathbf{x}|\mathrm{D})$ is obtained via an implicit parameter model $\boldsymbol{\theta}$.

- We are therefore realizing the calculation of $p(\mathbf{x}|\mathrm{D})$ to estimate $p(\mathbf{x})$, converting the problem of estimating a probability density to a problem of estimating a parameter vector.
- Reasonably, we have

$$p(\mathbf{x} \mid D) = \int p(\mathbf{x}, \boldsymbol{\theta} \mid D) d\boldsymbol{\theta}$$

where the integration extends over the entire parameter space.

The parameter distribution (3)

- Then $$p(\mathbf{x} \mid D) = \int p(\mathbf{x}, \boldsymbol{\theta} \mid D) d\boldsymbol{\theta}$$
$$= \int p(\mathbf{x} \mid \boldsymbol{\theta}, D) p(\boldsymbol{\theta} \mid D) d\boldsymbol{\theta}$$

- Since, by hypothesis, the selection of $\mathbf{x}$ is done independently from the training samples in D, given $\boldsymbol{\theta}$,
$$p(\mathbf{x} \mid D) = \int p(\mathbf{x} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta} \mid D) d\boldsymbol{\theta}$$

- That is, the distribution of $p(\mathbf{x})$ is known completely once we know the value of the parameter vector $\boldsymbol{\theta}$

The parameter distribution (4)

- The previous equation links explicitly the desired class-conditional density $p(\mathbf{x}|\mathrm{D})$ to the posterior density $p(\boldsymbol{\theta}|\mathrm{D})$ through the unknown parameter vector $\boldsymbol{\theta}$.
- If $p(\boldsymbol{\theta}|\mathrm{D})$ peaks very sharply about some value $\hat{\boldsymbol{\theta}}$, we obtain an estimation of the most likely vector, so

$$p(\mathbf{x}|\mathrm{D}) \approx p(\mathbf{x} \mid \hat{\boldsymbol{\theta}})$$

- But this approach allows to **take into account the effects of all other models**, described by the value of the **integral function**, for ***all possible models***.

$$p(\mathbf{x} \mid D) = \int p(\mathbf{x} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta} \mid D)\, d\boldsymbol{\theta}$$

Example: Gaussian Case

- We use the Bayesian estimation techniques to calculate the posterior density $p(\boldsymbol{\theta}|\mathrm{D})$ and the desired density $p(\mathbf{x}|\mathrm{D})$ for the case where

$$p(\mathbf{x}\,|\,\boldsymbol{\theta}) \equiv p(\mathbf{x}\,|\,\boldsymbol{\mu}) \approx N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

- **<u>THE UNIVARIATE CASE</u>:**

$$p(\mathbf{x}\,|\,\boldsymbol{\mu}) \equiv p(x\,|\,\mu) \approx N(\mu, \sigma^2)$$

The mean μ is the only unknown parameter

$$p(\mu) \approx N(\mu_0, \sigma_0^2)$$

Conjugate prior

The prior knowledge about μ, can be expressed by a prior density, where the mean and the variance are known

In practice, μ_0 represents our best a-priori guess for the parameter μ, and σ_0^2 measures our uncertainty about this guess

Example: Gaussian Case (2)

- We can draw μ from $N(\mu_0, \sigma_0^2)$
- It becomes the true value for μ and completely determines the density for x.
- Suppose to have n training samples $D = \{x_1, x_2, \ldots, x_n\}$ and calculate:

Reproduced density

$$p(\mu \mid D) = \frac{p(D \mid \mu)\, p(\mu)}{\int p(D \mid \mu)\, p(\mu)\, d\mu}$$

$$= \alpha \prod_{k=1}^{n} p(x_k \mid \mu)\, p(\mu)$$

where α is a normalization factor that depends on D and independent from μ

Example: Gaussian Case (3)

- This equation shows how the observation of a set of training samples affects our idea about the true value of μ: ***it relates the prior density p(μ) to a posterior density p(μ|D).***

- Developing the calculations, we realize that, thanks to the normal prior, $p(\mu|D)$ is also normal, and changes depending on the number of samples that form the training set, evolving in a Dirac impulse for $n \to \infty$ (<u>*Bayesian Learning*</u> phenomenon).

- Formally the following formulas are reached:

Example: Gaussian Case (4)

$$p(\mu \mid D) = \frac{p(D \mid \mu)\, p(\mu)}{\int p(D \mid \mu)\, p(\mu)\, d\mu} = \frac{1}{\sqrt{2\pi}\sigma_n} \exp\{-\frac{(\mu - \mu_n)^2}{2\sigma_n^{\ 2}}\}$$

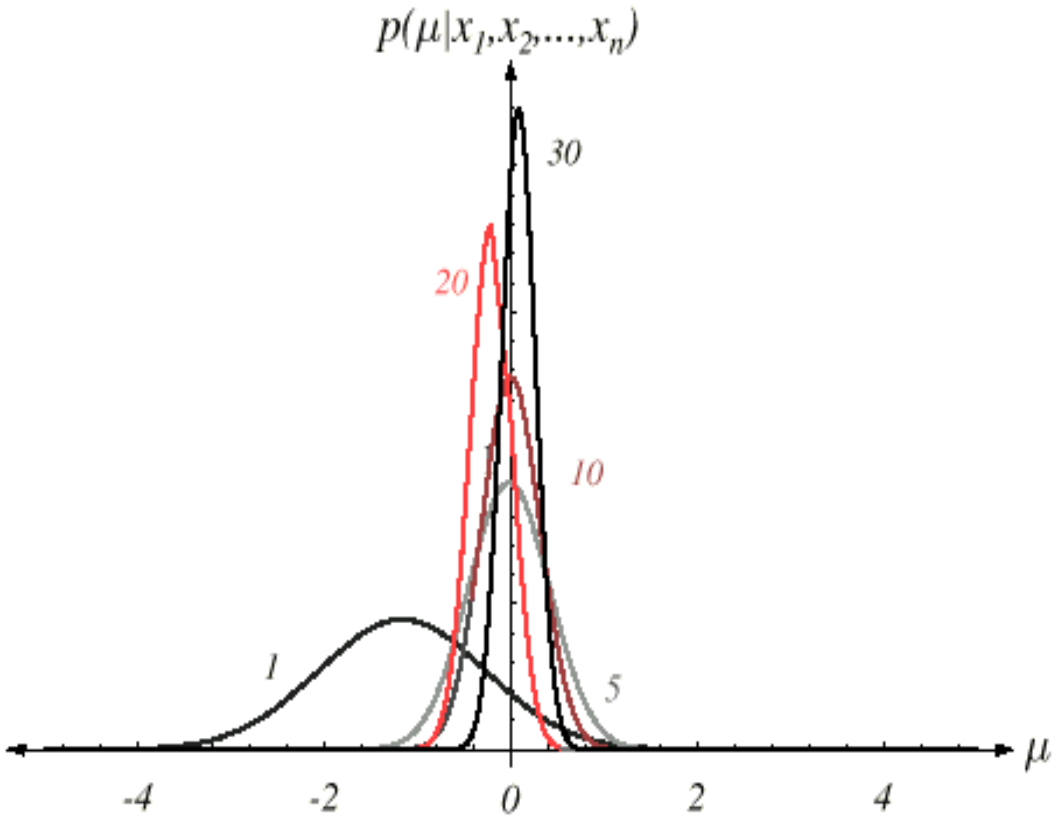
$$\text{dove} \quad \mu_n = \frac{n\sigma_0^{\ 2}}{n\sigma_0^{\ 2} + \sigma^2}\left(\frac{1}{n}\sum_{k=1}^{n} x_k\right) + \frac{\sigma^2}{n\sigma_0^{\ 2} + \sigma^2}\mu_0$$

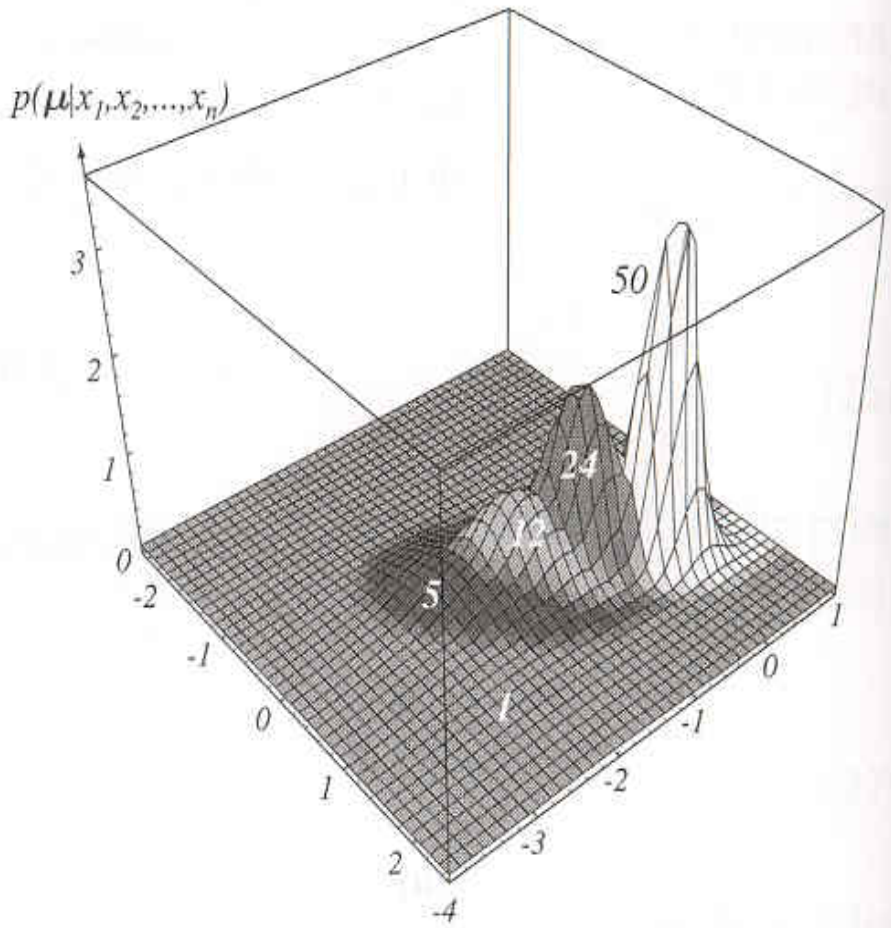
$$\sigma_n^2 = \frac{\sigma_0^{\ 2}\sigma^2}{n\sigma_0^{\ 2} + \sigma^2}$$

μ_n represents our best choice for μ after observing n samples.

$\sigma_n^{\ 2}$ measures the uncertainty about this guess

Example: Gaussian Case (5)





Example: Gaussian Case (6)

- Having obtained the a-posteriori density for the mean, $p(\mu|D)$, all that remains is to *obtain the "class-conditional" density for* $p(x|\mathrm{D})$, which, in exact notation, is $p(x|\omega_i, D_i)$. So, we have:

$$p(x|\mathcal{D}) = \int p(x|\mu)p(\mu|\mathcal{D})\,d\mu$$

$$= \int \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right] \frac{1}{\sqrt{2\pi}\sigma_n} \exp\left[-\frac{1}{2}\left(\frac{\mu-\mu_n}{\sigma_n}\right)^2\right] d\mu$$

$$= \frac{1}{2\pi\sigma\sigma_n} \exp\left[-\frac{1}{2}\frac{(x-\mu_n)^2}{\sigma^2+\sigma_n^2}\right] f(\sigma,\sigma_n), \qquad (36)$$

Example: Gaussian Case (7)

$$f(\sigma, \sigma_n) = \int \exp\left[-\frac{1}{2}\frac{\sigma^2+\sigma_n^2}{\sigma^2\sigma_n^2}\left(\mu - \frac{\sigma_n^2 x + \sigma^2\mu_n}{\sigma^2+\sigma_n^2}\right)^2\right] d\mu.$$

- Observing the equation (36), we notice that

$$p(x \mid D) \approx N(\mu_n, \sigma^2 + \sigma_n^2)$$

- If we compare the class conditional-density $p(x|\mathrm{D})$, with its parametric form $p(x \mid \mu) \approx N(\mu, \sigma^2)$, we observe that the conditional mean is treated as if it were the true mean, and the known variance is proportional to the current degree of uncertainty.

Example: Gaussian Case (8)

- To sum up, the obtained density $p(x|D)$ is the desired class-conditional density

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

which, together with the prior probabilities $P(\omega_i)$, gives us the probabilistic information needed to design the classifier, in contrast with ML methods that only makes point estimates for $\hat{\mu}$ e $\hat{\sigma}^2$

Bayesian parameter estimation: general theory

- Summarizing and extending them to the general case, the main formulas seen are:

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

$$p(\mu\,|\,D) = \frac{p(D\,|\,\mu)\,p(\mu)}{\int p(D\,|\,\mu)\,p(\mu)\,d\mu} = \frac{p(D\,|\,\boldsymbol{\theta})\,p(\boldsymbol{\theta})}{\int p(D\,|\,\boldsymbol{\theta})\,p(\boldsymbol{\theta})\,d\boldsymbol{\theta}} = p(\boldsymbol{\theta}\,|\,D)$$

$$p(D\,|\,\boldsymbol{\theta}) = \prod_{k=1}^{n} p(x_k\,|\,\boldsymbol{\theta})$$

- Note the similarity with the ML approach, with the difference that here we do not look for the max point $\hat{\boldsymbol{\theta}}$

Bayesian parameter estimation: general theory (2)

- There are still questions to be clarified:
  - *Convergence of* $p(\mathbf{x}|\mathrm{D})$ *to* $p(\mathbf{x})$

- <u>Convergence</u>: let's suppose $\mathrm{D}^n=\{x_1, \ldots, x_n\}$, $n > 1$:

$$p(D^n \mid \boldsymbol{\theta}) = p(\mathbf{x}_n \mid \boldsymbol{\theta})\, p(\mathrm{D}^{n-1} \mid \boldsymbol{\theta})$$

$$p(\boldsymbol{\theta} \mid D) = \frac{p(D \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{\int p(D \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})\, d\boldsymbol{\theta}}$$

On line method
of *Bayesian learning*

$$p(\boldsymbol{\theta} \mid D^n) = \frac{p(\mathbf{x}_n \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta} \mid D^{n-1})}{\int p(\mathbf{x}_n \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta} \mid D^{n-1})\, d\boldsymbol{\theta}}$$

Assuming that

$$p(\boldsymbol{\theta} \mid D^0) = p(\boldsymbol{\theta})$$

Bayesian approach – Conclusions

- To conclude, extending the notation to the various classes ω_i and corresponding training set D_i, the design of a Bayesian classifier through estimation of parameters with Bayesian approach is subject to the following formulas:

$$p(\theta \mid D_i, \omega_i) = \frac{p(D_i \mid \theta, \omega_i)\, p(\theta \mid \omega_i)}{\int p(D_i \mid \theta, \omega_i)\, p(\theta \mid \omega_i)\, d\theta}$$

$$= \frac{\prod_{k=1}^{n_i} p(x_{i,k} \mid \theta) \;\; p(\theta \mid \omega_i)}{\int \prod_{k=1}^{n_i} p(x_{i,k} \mid \theta) \;\; p(\theta \mid \omega_i)\, d\theta}$$

Bayesian approach – Conclusions (2)

- Let $D_i^n = \{x_{i,1},\ldots, x_{i,n}\}$

$$p(\theta \mid D_i^{\,n}, \omega_i) = \frac{\prod_{k=1}^{n_i} p(x_{i,k} \mid \theta, \omega_i)\ \ p(\theta \mid \omega_i)}{\int \prod_{k=1}^{n_i} p(x_{i,k} \mid \theta, \omega_i)\ \ p(\theta \mid \omega_i) d\theta}$$

$$= \frac{p(x_{i,n_i} \mid \theta)\, p(\theta \mid D_i^{\,n-1}, \omega_i)}{\int p(x_{i,n_i} \mid \theta)\, p(\theta \mid D_i^{\,n-1}, \omega_i) d\theta}$$

Bayesian approach – Conclusions (3)

- The classifier ***minimum error rate*** results
  - Decide ω_i if $P(\omega_i|x) \geq P(\omega_j|x)$, for $j=1,\ldots,c$

$$P(\omega_i \mid x, D_i) = \frac{p(x \mid \omega_i, D_i) P(\omega_i)}{p(x \mid D_i)}$$

$$p(x \mid \omega_i, D_i) = \int p(x, \theta \mid \omega_i, D_i) d\theta$$

$$= \int p(x \mid \theta) p(\theta \mid \omega_i, D_i) d\theta$$

Comparison ML – Bayes estimation

- ML gives us a point estimate $\hat{\boldsymbol{\theta}}$, instead, Bayes approach gives us a distribution on $\boldsymbol{\theta}$.

- ML and Bayes solutions are equivalent in the asymptotic limit of infinite training data.
  - To the limit, $p(\boldsymbol{\theta}|D)$ converges to a Dirac delta function

- Practically, the approaches are different for various reasons:
  - *Computational complexity*
  - *Interpretability*
  - *Confidence in the prior information*
  - *Compromise between estimation accuracy and variance*