Chapter 3: Maximum-Likelihood and Bayesian Parameter Estimation (part 2)

- Bayesian Estimation (MAP)
- Bayesian Parameter Estimation: Gaussian Case
- LM vs MAP
- Problems of Dimensionality
- Problem of Overfitting
- Computational Complexity

BAYESIAN CLASSIFIER

$$P(w_k|\,x,D)=\frac{P(x\,|\,w_k,D)P(w_k\,|\,D)}{\sum_{j-1}^{n}P(x\,|\,w_j,D)P(w_j\,|\,D)}=\frac{P(x\,|\,w_k,D)P(w_k\,|\,D)}{P(x\,|\,D)}$$

- $P(w_k|\,x,\mathrm{D})$ — **Info needed:** the probability of class w_k given pattern x and training set D.
- $P(x\,|\,w_k,D)$ — **Interesting data to be found through training:** how likely is class w_k given the training set D, to produce feature vector x.
- $P(w_k|D)$ — How likely is class w_k within the training set D (easy to determine).
- $P(x\,|\,D)$ — How likely is pattern x within the training set D (easy to determine).

Bayesian Parameter Estimation: General Theory

- **P(x | $\mathcal{D}$) computation can be applied to any situation in which the unknown density can be parametrized: the basic assumptions are:**
  - **The form of P(x | θ) is assumed known, but the value of θ is not known exactly**
  - **Our knowledge about θ is assumed to be contained in a known prior density P(θ)**
  - **The rest of our knowledge θ is contained in a set $\mathcal{D}$ of n random variables $x_1, x_2, \ldots, x_n$ that follows P(x)**

PROBABILISTIC PARAMETERS

- **Assumptions:**
  - Although prior probability $P(x)$ is unknown, it is assumed to have a normal distribution whose parameters $\theta=[\mu,\sigma^2]$ are random variables with their own (Gaussian) distribution.
  - D is a set of N (i.i.d) samples $X_1, X_2, \ldots, X_N$.

- **Conclusion:**
  - Any information about θ is contained in a (known) prior density $P(\theta)$. Observation of the samples converts this to a posterior probability $P(\theta|D)$ that peaks at the actual value of θ.

$$P(x \mid D) = \sum P(x \mid \theta_j) P(\theta_j \mid D) \quad \Rightarrow \quad P(x \mid D) = \int P(x \mid \theta) P(\theta \mid D) d\theta$$

$$P(D \mid \theta) = \prod_{k=1}^{N} P(X_k \mid \theta) \quad \Rightarrow \quad P(\theta \mid D) = \frac{P(D \mid \theta) P(\theta)}{P(D)}$$

PARAMETER ESTIMATION

- **ML** estimation is based on the premise that the class-conditional probabilities $P(x|C_k, D)$ have normal distributions whose parameters $\theta=[\mu,\sigma^2]$ are constants and their actual values are to be found by maximizing the distribution probability of the training data conditioned on the parameters, i.e. $P(D|\theta)$.

- **MAP** (Bayesian) estimation follows an arguably more natural approach by attempting to find the most probable values of the parameters, conditioned on the available data. Thus, parameters are regarded as random variables with their own distribution, and that is the posterior distribution:

$$P(\theta \mid D) = \frac{P(D \mid \theta)P(\theta)}{P(D)} \rightarrow P(D \mid \theta)P(\theta)$$

where $P(\theta)$ is the prior distribution of the parameters, and $P(D)$ is independent of any class distribution and it can be ignored.

MAP ESTIMATE

- The value of parameters θ maximizing the posterior distribution *P(θ|D)* also maximizes its logarithm. Maximizing the log of the posterior estimate with respect to parameters θ yields the **maximum a posteriori** (or **MAP**) estimate:

$$\theta_{MAP} = \arg\max_{\theta} \left[\ln P(\theta \mid D)\right] = \arg\max_{\theta} \left[\ln P(D \mid \theta) + \ln P(\theta)\right]$$

- If the prior distribution *P(θ)* is flat (that is, independent of θ), then the MAP estimate matches the ML estimate.

$$\theta_{MAP} = \arg\max_{\theta} \left[\ln P(D \mid \theta)\right] = \theta_{ML}$$

UNIVARIATE CASE

- **Goal:** apply MAP estimation to calculate $P(\theta|D)$ and $P(x|D)$ under the assumption that $P(x|\theta)$ follows a normal distribution: $P(x|\theta) \sim N(\mu, \sigma^2)$.
- **Data**: a set D of N (i.i.d) training samples $X_1, X_2, \ldots, X_N$.

- **Univariate case:** normal distribution with unknown mean μ (and known variance σ^2):
  - $P(x|\theta) \Rightarrow P(x|\mu) \sim N(\mu, \sigma^2)$
  - $P(\mu) \sim N(\mu_0, \sigma_0^2)$; μ_0 = best guess for μ (with uncertainty measured by σ_0^2).

- It can be shown that prior information can be combined with empirical information in the samples to obtain the "a posteriori" distribution $P(\mu|D)$.

$$P(\mu \mid D) = \frac{P(D \mid \mu)P(\mu)}{P(D)} = \prod_{k=1}^{N} P(X_k \mid \mu)P(\mu) = \boldsymbol{\alpha} \quad \frac{1}{\sqrt{2\pi\sigma_N^2}} e^{\left[-\frac{1}{2}\frac{(\mu-\mu_N)^2}{\sigma_N^2}\right]}$$

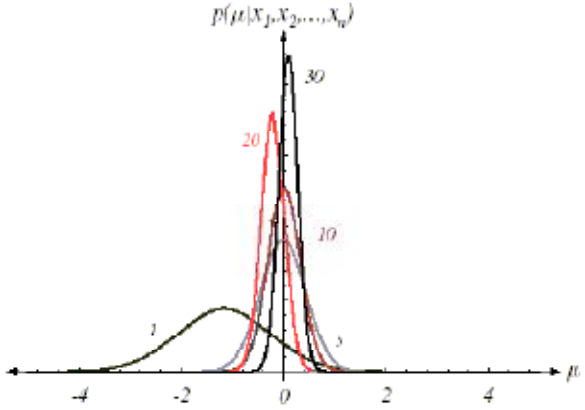
$$\overline{\mu} = \frac{1}{N}\sum_{k=1}^{N} X_k \qquad \mu_N = \frac{N\sigma_0^2}{N\sigma_0^2+\sigma^2}\overline{\mu} + \frac{\sigma^2}{N\sigma_0^2+\sigma^2}\mu_0 \qquad \sigma_N^2 = \frac{\sigma_0^2\sigma^2}{N\sigma_0^2+\sigma^2}$$

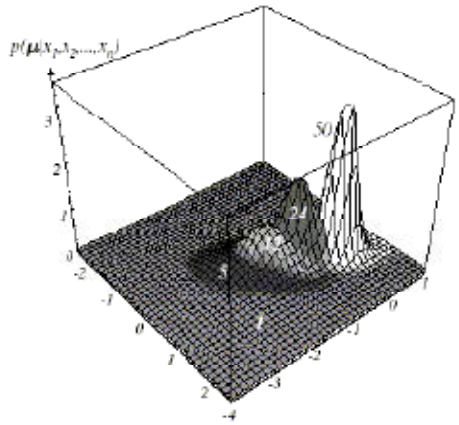
BAYESIAN LEARNING

- **MAP analysis:** as the number of samples increases, $P(\mu|D)$ becomes more and more sharply peaked, approaching the **Dirac delta function** as $N \to \infty$.

$$\lim_{N\to\infty} \mu_N = \overline{\mu} \qquad \sigma_N^2 \approx \frac{\sigma^2}{N} \to 0$$

- **Bayesian learning:** as the size of the training set D increases, the peak in $P(D|\theta)$, and subsequently in the probabilistic estimate $P(\theta|D)$, tends to be more and more pronounced. At the limit, as $N \to \infty$, it becomes the Dirac delta function (i.e. the limit of a normal distribution with a variance of zero).





ML vs. MAP

	ML	MAP
Parameters	Constants	Random variables
Assumptions	It avoids assumptions on prior information and it is analytically easier to solve, although some estimates can be biased.	It permits including a priori information about the unknown, but the analytical derivation are cumbersome.
Accuracy		More accurate statistically (because it considers the uncertainty in estimating the parameters).
Size of training set	Estimates do not change significantly as the training set size increases.	Estimates continuously improves as the training set size increases.
	For ordinary cases, both approaches give similar results with sufficient sample data.	
Computation Complexity	Low	High

CONCLUSIONS

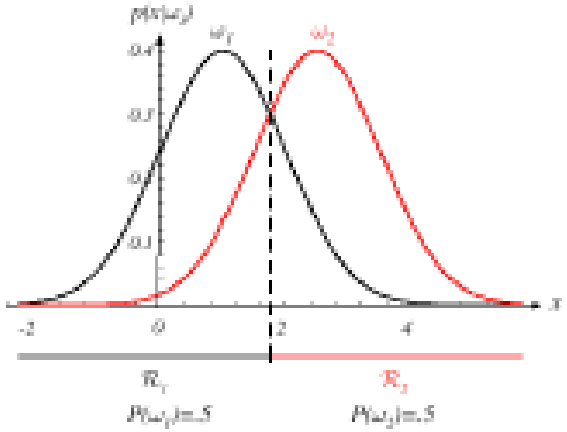
- If the class-conditional probabilities have a parametric form (derived from a Gaussian distribution), then the learning task is reduced to finding the parameter $\theta_k = [\mu_k, \sigma^2_k]$ associated with each class C_k.
- **Maximum-likelihood** method assumes that the parameters are constants and seeks to find their values that best match the training data.
- **Bayesian** method considers the parameters as being random variables with known prior density, and employs the training data to convert this to an "a posteriori" density.
- While Bayesian estimation is more accurate and it should be preferred (in principle), the maximum-likelihood method is easier to implement and provides a nearly as accurate result in the context of large training sets.
- **Problems with parametric techniques:**
  - Usually the density form is not known.
  - Most distributions in practice are multi-modal, whereas most models are unimodal.
  - Approximating a multivariate distribution as a product of univariate distributions does not work well in practice.

PROBLEMS OF DIMENSIONALITY

- Are more features better than fewer features? Would adding more features improve classification performance?
- Which features are more informative?
- How about complexity? Would adding more features make the design of the classifier more complicated?
- Shall we aim getting the best possible classification performance on the training data?
- How to avoid overfitting?

Problems of Dimensionality

- Problems involving 50 or 100 features (binary valued)
  - Classification accuracy depends upon the dimensionality and the amount of training data
  - Case of two classes multivariate normal with the same covariance



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

where : $r^2 = (m_1 - m_2)^t \Sigma^{-1} (m_1 - m_2)$

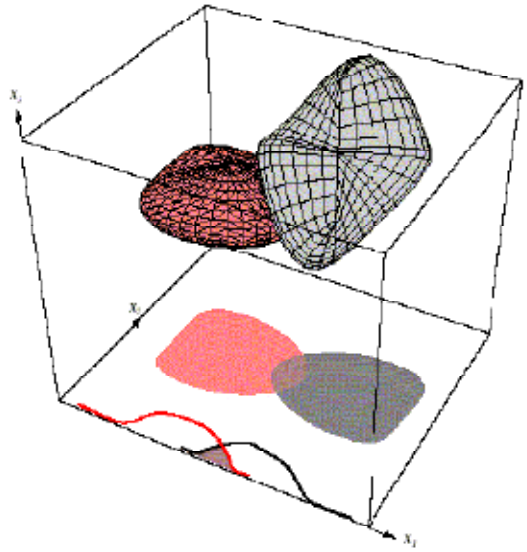
$$\lim_{r \to \infty} P(\text{error}) = 0$$

ACCURACY vs. DIMENSION

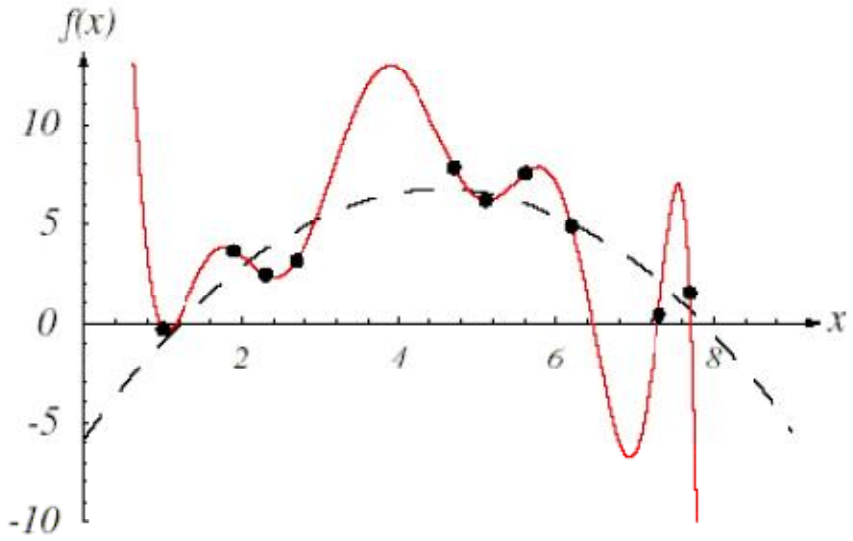
- Although increasing the number of features increases the cost and complexity of both the feature extractor and the classifier, it is often reasonable to believe that the performance will improve. At worst, the Bayes classifier will ignore the additional features, but if new features provide any additional information, the performance must improve.

- It has been observed in practice that, beyond a certain point, the inclusion of additional features leads to worse rather than better performance. Large number of features may cause overfitting, if they are not relevant features, and if the underlying distributions are not estimated accurately.

Two three-dimensional distributions have non-overlapping densities, and thus in three dimensions the Bayes error vanishes. When projected to a subspace (here, the two-dimensional x_1 - x_2 subspace or a one-dimensional x_1 subspace) there can be greater overlap of the projected distributions, and hence greater Bayes error.



PROBLEM OF OVERFITTING



The "training data" (black dots) were selected from a quadratic function plus Gaussian noise, i.e.

$f(x) = ax^2 + bx + c + \varepsilon$ where $p(\varepsilon) \sim N(0, \sigma^2)$.

The 10th-degree polynomial shown fits the data perfectly, but we desire instead the second-order function f(x), because it would lead to better predictions for new samples.

- **If features are independent then:**

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

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

- **Most useful features are the ones for which the difference between the means is large relative to the standard deviation**

- **It has frequently been observed <u>in practice</u> that, beyond a certain point, the inclusion of additional features leads to worse rather than better performance: <u>we have the wrong model !</u>**

Computational Complexity

Our design methodology is affected by the computational difficulty

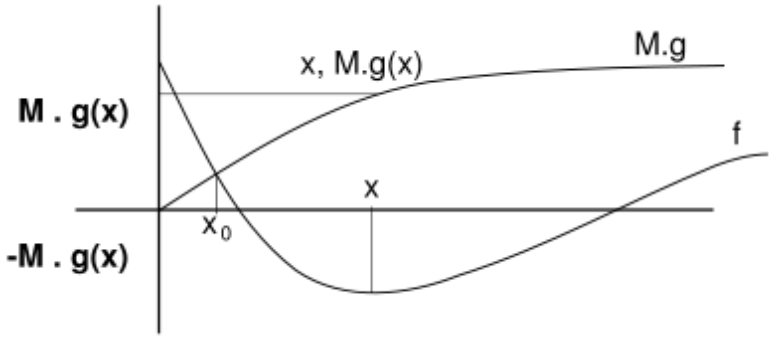
O notation

f(x) = O(h(x)) "big oh of h(x)" or we can say f has order g

If:

$$M \quad positive \in \Re \; ; \; |f(x)| \leq M|h(x)|, \quad x_0 \leq x$$

(An upper bound on f(x) grows no worse than h(x) for sufficiently large x!)



Example

Take the polynomials:

$$f(x) = 6x^4 - 2x^3 + 5$$

$$g(x) = x^4.$$

We say $f(x)$ has order $O(g(x))$ or $O(x^4)$ (as $x \to \infty$)

From the definition of order

$$|f(x)| \leq M|g(x)| \text{ for } x > x_0.$$

Proof:

for all $x > 1$ (we take $x_0 = 1$):

$$|6x^4 - 2x^3 + 5| \leq 6x^4 + 2x^3 + 5$$

$$|6x^4 - 2x^3 + 5| \leq 6x^4 + 2x^4 + 5x^4$$

$$|6x^4 - 2x^3 + 5| \leq 13x^4$$

$$|6x^4 - 2x^3 + 5| \leq 13\,|x^4|.$$

where M = 13 in this example

- O is not unique!

  $f(x) = O(x^2)$; $f(x) = O(x^3)$; $f(x) = O(x^4)$

- Θ "big theta" notation

  $f(x) = \Theta(h(x))$

  If:

  $$\exists(x_0, c_1, c_2) \in \Re^3; \forall x > x_0$$

  $$0 \leq c_1 h(x) \leq f(x) \leq c_2 h(x)$$

$f(x) = \Theta(x^2)$ but $f(x) \neq \Theta(x^3)$

Complexity of the ML Estimation

- Gaussian priors in d dimensions classifier with n training samples for each of c classes

- For each category, we have to compute the discriminant function

$$g(x) = -\frac{1}{2}(x - \hat{\mu})^t \Sigma^{-1} (x - \hat{\mu}) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\hat{\Sigma}| + \ln P(\omega)$$

$O(d.n)$ $O(n.d^2)$ $O(1)$ $O(d^2.n)$ $O(n)$

Total = $O(d^2 \cdot n)$

Total for c classes = $O(cd^2.n) \approx O(d^2.n)$

- Cost increases when d and n are large!