Outline

• Review
• Maximum A-Posteriori (MAP) Estimation
• Bayesian Parameter Estimation
• Example: The Gaussian Case
• Recursive Bayesian Incremental Learning
• Problems of Dimensionality
• Linear Algebra review
• Principal Component Analysis
• Fisher Discriminant
Bayesian Decision Theory

- Bayesian decision theory is a fundamental statistical approach to the problem of pattern classification.
  - Decision making when all the probabilistic information is known.
  - For given probabilities the decision is optimal.
  - When new information is added, it is assimilated in optimal fashion for improvement of decisions.
Bayes' formula

\[ P(\omega_j \mid x) = P(x \mid \omega_j) \, P(\omega_j) / P(x), \]

where

\[ P(x) = \sum_{j=1}^{2} p(x \mid \omega_j)P(\omega_j) \]

Posterior = \( \frac{\text{Likelihood} \times \text{Prior}}{\text{Evidence}} \)
Bayes' formula cont.

• \( p(x|\omega_j) \) is called the **likelihood** of \( \omega_j \) with respect to \( x \).
  
  (the \( \omega_j \) category for which \( p(x|\omega_j) \) is large is more "likely" to be the true category)

• \( p(x) \) is the **evidence**
  
  how frequently we will measure a pattern with feature value \( x \).

  Scale factor that guarantees that the posterior probabilities sum to 1.
Bayes' Decision Rule
(Minimizes the probability of error)

\[ \omega_1 : \text{if } P(\omega_1|x) > P(\omega_2|x) \]
\[ \omega_2 : \text{otherwise} \]

or

\[ \omega_1 : \text{if } P(x|\omega_1) P(\omega_1) > P(x|\omega_2) P(\omega_2) \]
\[ \omega_2 : \text{otherwise} \]

and

\[ P(\text{Error}|x) = \min [P(\omega_1|x), P(\omega_2|x)] \]
Normal Density - Univariate Case

- Gaussian density with mean $\mu \in \mathbb{R}$ and standard deviation $\sigma \in \mathbb{R}_+$, $(\sigma^2$ named variance $)$

$$p(x) = \frac{1}{(2\pi)^{1/2} \sigma} \exp \left[ -\frac{1}{2} \left( \frac{x - \mu}{\sigma} \right)^2 \right]$$

$$p(x) \sim N(\mu, \sigma^2)$$

- It can be shown that:

$$\mu = \mathbb{E}[x] = \int_{-\infty}^{\infty} xp(x)dx, \quad \sigma^2 = \mathbb{E}[(x - \mu)^2] = \int_{-\infty}^{\infty} (x - \mu)^2 p(x)dx.$$
Normal Density - Multivariate Case

• The general *multivariate normal density* (MND) in a $d$ dimensions is written as

\[
p(x) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[ -\frac{1}{2} (x - \mu)^t \Sigma^{-1} (x - \mu) \right]
\]

• It can be shown that:

\[
\mu = \mathbb{E}[x] = \int x \ p(x) dx, \quad \Sigma = \mathbb{E}[(x - \mu)(x - \mu)^t]
\]

which means for components

\[
\sigma_{i,j} = \mathbb{E}[(x_i - \mu_i)(x_j - \mu_j)]
\]

• The covariance matrix $\Sigma$ is always symmetric and positive semidefinite.
Normal Density - Multivariate Case

- The general multivariate normal density (MND) in a \(d\) dimensions is written as

\[
p(x) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[ -\frac{1}{2} (x - \mu)^t \Sigma^{-1} (x - \mu) \right]
\]

- It can be shown that:

\[
\mu = E[x] = \int_{\mathbb{R}^d} x \, p(x) \, dx, \quad \Sigma = E[(x - \mu)(x - \mu)^t].
\]

which means for components

\[
\sigma_{i,j} = E[(x_i - \mu_i)(x_j - \mu_j)].
\]
Maximum Likelihood and Bayesian Parameter Estimation

• To design an optimal classifier we need $P(\omega_i)$ and $p(x|\omega_i)$, but usually we do not know them.

• **Solution** – to use training data to estimate the unknown probabilities. Estimation of class-conditional densities is a difficult task.
Maximum Likelihood and Bayesian Parameter Estimation

• **Supervised learning**: we get to see samples from each of the classes “separately” (called tagged or labeled samples).

• Tagged samples are “expensive”. We need to learn the distributions as efficiently as possible.

• **Two methods**: parametric (easier) and non-parametric (harder)
Maximum Likelihood and Bayesian Parameter Estimation

• Program for parametric methods:
  ➢ Assume specific parametric distributions with parameters $\theta \in \Theta \subset \mathbb{R}^p$
  ➢ Estimate parameters $\hat{\theta}(D)$ from training data $D$
  ➢ Replace true value of class-conditional density with approximation and apply the Bayesian framework for decision making.
Suppose we can assume that the relevant (class-conditional) densities are of some parametric form. That is,
\[ p(x|\omega) = p(x|\theta), \text{ where } \theta \in \Theta \subseteq \mathbb{R}^p \]

Examples of parameterized densities:

- Binomial: \( x^{(n)} \) has \( m \) 1’s and \( n-m \) 0’s
  \[
p(x^{(n)} | \theta) = \binom{n}{m} \theta^m (1-\theta)^{n-m}, \quad \Theta = [0,1]
  \]

- Exponential: Each data point \( x \) is distributed according to
  \[
p(x | \theta) = \theta e^{-\theta x}, \quad \Theta = (0, \infty)
  \]
Maximum Likelihood and Bayesian Parameter Estimation cont.

- Two procedures for parameter estimation will be considered:
  - **Maximum likelihood estimation**: choose parameter value $\hat{\theta}$ that makes the data most probable (i.e., maximizes the probability of obtaining the sample that has actually been observed),
    $$p(x \mid D) = p(x \mid \hat{\theta}(D)), \quad \hat{\theta}(D) = \arg \max_{\theta} p(D \mid \theta)$$
  - **Bayesian learning**: define a prior probability on the model space $p(\theta)$ and compute the posterior $p(\theta \mid D)$. Additional samples sharpen the posterior density which peaks near the true values of the parameters.
Sampling Model

- It is assumed that a sample set $S = \{(x_l, \omega_l) : l = 1, \ldots, N\}$ with independently generated samples is available.
- The sample set is partitioned into separate sample sets for each class, $D_j = \{x_l : (x_l, \omega_l) \in D\}$
- A generic sample set will simply be denoted by $D$.
- Each class-conditional $p(x | \omega_j)$ is assumed to have a known parametric form and is uniquely specified by a parameter (vector) $\theta_j$.
- Samples in each set $D_j$ are assumed to be independent and identically distributed (i.i.d.) according to some true probability law $p(x | \omega_j)$. 
Log-Likelihood function and Score Function

• The sample sets are assumed to be functionally independent, i.e., the training set \( S_j \) contains no information about \( \Theta_i \) for \( i \neq j \).
• The i.i.d. assumption implies that

\[
p(D_j | \Theta_j) = \prod_{x \in D_j} p(x | \Theta_j)
\]
• Let \( D \) be a generic sample of size \( n \equiv |D| \).
• Log-likelihood function:

\[
l(\Theta; D) \equiv \ln p(D | \Theta) = \sum_{k=1}^{n} \ln p(x_k | \Theta)
\]
• The log-likelihood function is identical to the logarithm of the probability density function, but is interpreted as a function over the sample space for given parameter \( \Theta \).
Log-Likelihood Illustration

- Assume that all the points in $\mathbf{D}$ are drawn from some (one-dimensional) normal distribution with some (known) variance and unknown mean.
Log-Likelihood function and Score Function cont.

• Maximum likelihood estimator (MLE):

\[ \hat{\theta}(D) = \arg\max_{\theta \in \Theta} l(\theta; D) \]

(tacitly assuming that such a maximum exists!)

• Score function:

\[ U_k(\theta; D) \equiv \frac{\partial l(\theta; D)}{\partial \theta_k} \quad 1 \leq k \leq p \]

and hence

\[ U(\theta; D) \equiv \nabla_{\theta} l(\theta; D) \]

• Necessary condition for MLE (if not on border of domain \( \Theta \)):

\[ U(\theta; D) = 0 \]
Maximum A Posteriory

- **Maximum a posteriori (MAP):**
  Find the value of $\theta$ that maximizes $l(\theta)+\ln(p(\theta))$, where $p(\theta)$ is a prior probability of different parameter values. A MAP estimator finds the peak or mode of a posterior.

  **Drawback of MAP:** after arbitrary nonlinear transformation of the parameter space, the density will change, and the MAP solution will no longer be correct.
Maximum A-Posteriori (MAP) Estimation

- The “most likely value” is given by $\theta$

$$\hat{\theta} = \arg \max_{\theta} p(\theta \mid X^{(n)}) = \arg \max_{\theta} \frac{p_0(\theta)p(X^{(n)} \mid \theta)}{p(X^{(n)})}$$

$$= \arg \max_{\theta} \frac{p_0(\theta) \prod_{i=1}^{n} p(x_i \mid \theta)}{\int p(X^{(n)} \mid \theta') p_0(\theta') d\theta'}$$
Maximum A-Posteriori (MAP) Estimation

\[ p(X^{(n)} | \theta) = \prod_{i=1}^{n} p(x_i | \theta) \]

since the data is i.i.d.

- We can disregard the normalizing factor \( p(X^{(n)}) \) when looking for the maximum
So, the \( \hat{\theta} \) we are looking for is

\[
\hat{\theta} = \arg \max _{\theta} \left[ p_0(\theta) \prod _{i=1}^{n} p(x_i | \theta) \right] \quad \text{(log is monotonically increasing)}
\]

\[
= \arg \max _{\theta} \left( \log \left[ p_0(\theta) \prod _{i=1}^{n} p(x_i | \theta) \right] \right)
\]

\[
= \arg \max _{\theta} \left( \log p_0(\theta) + \log \prod _{i=1}^{n} p(x_i | \theta) \right)
\]

\[
= \arg \max _{\theta} \left( \log p_0(\theta) + \sum _{i=1}^{n} \log p(x_i | \theta) \right)
\]
The Gaussian Case: Unknown Mean

• Suppose that the samples are drawn from a multivariate normal population with mean $\mu$, and covariance matrix $\Sigma$.

• Consider first the case where only the mean is unknown $\theta = \mu$.

• For a sample point $x_k$, we have
\[
\ln P(x_k | \mu) = -\frac{1}{2} \ln [(2\pi)^d |\Sigma|] - \frac{1}{2} (x_k - \mu)^t \Sigma^{-1} (x_k - \mu)
\]

and
\[
\nabla_\mu \ln P(x_k | \mu) = \Sigma^{-1} (x_k - \mu)
\]

• The maximum likelihood estimate for $\mu$ must satisfy

The Gaussian Case: Unknown Mean

\[ \sum_{k=1}^{n} \Sigma^{-1} (x_k - \hat{\mu}) = 0 \]

- Multiplying by \( \Sigma \), and rearranging, we obtain

\[ \hat{\mu} = \frac{1}{n} \sum_{k=1}^{n} x_k \]

- The MLE estimate for the unknown population mean is just the arithmetic average of the training samples (sample mean).

- Geometrically, if we think of the n samples as a cloud of points, the sample mean is the centroid of the cloud.
The Gaussian Case: Unknown Mean and Covariance

- In the general multivariate normal case, neither the mean nor the covariance matrix is known \( \theta = [\mu, \Sigma] \).
- Consider first the univariate case with \( \theta_1 = \mu \) and \( \theta_2 = \sigma^2 \). The log-likelihood of a single point is

\[
\ln p(x_k | \theta) = -\frac{1}{2} \ln 2\pi \theta_2 - \frac{1}{2\theta_2} (x_k - \theta_1)^2
\]

and its derivative is

\[
\nabla_\theta l = \nabla_\theta \ln p(x_k | \theta) = \begin{bmatrix}
\frac{1}{\theta_2} (x_k - \theta_1) \\
-\frac{1}{2\theta_2} + \frac{(x_k - \theta_1)^2}{2\theta_2}
\end{bmatrix}
\]
The Gaussian Case: Unknown Mean and Covariance

• Setting the gradient to zero, and using all the sample points, we get the following necessary conditions:

\[ \sum_{k=1}^{n} \frac{1}{\hat{\theta}_2} (x_k - \hat{\theta}_1) = 0 \quad \text{and} \quad - \sum_{k=1}^{n} \frac{1}{\hat{\theta}_2} + \sum_{k=1}^{n} \frac{(x_k - \hat{\theta}_1)^2}{\hat{\theta}_2^2} = 0 \]

• where \( \hat{\theta}_1 = \hat{\mu} \) and \( \hat{\theta}_2 = \hat{\sigma}^2 \), are the MLE estimates for \( \hat{\mu} \) and \( \hat{\sigma}^2 \) respectively.

• Solving for \( \hat{\mu} \) and \( \hat{\sigma}^2 \), we obtain

\[ \hat{\mu} = \frac{1}{n} \sum_{k=1}^{n} x_k \quad \text{and} \quad \hat{\sigma}^2 = \frac{1}{n} \sum_{k=1}^{n} (x_k - \hat{\mu})^2 \]
The Gaussian multivariate case

• For the multivariate case, it is easy to show that the MLE estimates for are given by
  \[ \hat{\mu} = \frac{1}{n} \sum_{k=1}^{n} x_k \quad \text{and} \quad \hat{\Sigma} = \frac{1}{n} \sum_{k=1}^{n} (x_k - \hat{\mu})(x_k - \hat{\mu})^t \]

• The MLE for the mean vector is the sample mean, and the MLE estimate for the covariance matrix is the arithmetic average of the n matrices \( (x_k - \hat{\mu})(x_k - \hat{\mu})^t \)

• The MLE for \( \sigma^2 \) is biased (i.e., the expected value over all data sets of size n of the sample variance is not equal to the true variance:
  \[ E\left[ \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{\mu})^2 \right] = \frac{n-1}{n} \sigma^2 \neq \sigma^2 \]
The Gaussian multivariate case

- Unbiased estimator for $\mu$ and $\Sigma$ are given by

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

and

$$C = \frac{1}{n-1} \sum_{k=1}^{n} (x_k - \hat{\mu})(x_k - \hat{\mu})^t$$

$C$ is called the sample covariance matrix. $C$ is absolutely unbiased. $\hat{\sigma}^2$ is asymptotically unbiased.
Bayesian Estimation: Class-Conditional Densities

- The aim is to find posteriors $P(\omega_i|x)$ knowing $p(x|\omega_i)$ and $P(\omega_i)$, but they are unknown. How to find them?
- Given the sample $D$, we say that the aim is to find $P(\omega_i|x, D)$
- Bayes formula gives:
  \[
P(\omega_i | x, D) = \frac{p(x | \omega_i, D)P(\omega_i | D)}{\sum_{j=1}^{c} p(x | \omega_i, D)P(\omega_i | D)}.
  \]
- We use the information provided by training samples to determine the class conditional densities and the prior probabilities.
- Generally used assumptions:
  - Priors generally are known or obtainable from a trivial calculations. Thus $P(\omega_i)= P(\omega_i|D)$.
  - The training set can be separated into $c$ subsets: $D_1, ..., D_c$
Bayesian Estimation: Class-Conditional Densities

- The samples $D_j$ have no influence on $p(x|\omega_i, D_i)$ if $i \neq j$

- Thus we can write:

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

- We have $c$ separate problems of the form:

Use a set $D$ of samples drawn independently according to a fixed but unknown probability distribution $p(x)$ to determine $p(x|D)$. 

18 March 2016
Bayesian Estimation: General Theory

- *Bayesian leaning* considers $\theta$ (the parameter vector to be estimated) to be a *random variable*.

Before we observe the data, the parameters are described by a *prior* $p(\theta)$ which is typically very broad. Once we observed the data, we can make use of Bayes’ formula to find *posterior* $p(\theta | D)$. Since some values of the parameters are more consistent with the data than others, the *posterior* is narrower than *prior*. This is *Bayesian learning* (see fig.)
• Density function for \( x \), given the training data set \( D \),
\[
p(x \mid D) = \int p(x, \theta \mid D) d\theta
\]
• From the definition of conditional probability densities
\[
p(x, \theta \mid D) = p(x \mid \theta, D) p(\theta \mid D).
\]
• The first factor is independent of \( D \) since it just our assumed form \( p(x \mid \theta, D) \Rightarrow p(x \mid \theta) \) for parameterized density.
• Therefore
\[
p(x \mid D) = \int p(x \mid \theta) p(\theta \mid D) d\theta
\]
• Instead of choosing a specific value for \( \theta \), the Bayesian approach performs a weighted average over all values of \( \theta \). The weighting factor \( p(\theta \mid D) \), which is a posterior of \( \theta \) is determined by starting from some assumed prior \( p(\theta) \)
• Then update it using Bayes’ formula to take account of data set $D$. Since $D = \{x^1, ..., x^N\}$ are drawn independently which is likelihood function.

$$p(D | \theta) = \prod_{n=1}^{N} p(x^n | \theta) ,$$  

which is likelihood function.

• Posterior for $\theta$ is

$$p(\theta | D) = \frac{p(D | \theta)p(\theta)}{p(D)} = \frac{p(\theta)}{p(D)} \prod_{n=1}^{N} p(x^n | \theta) ,$$  

where normalization factor

$$p(D) = \int p(\theta') \prod_{n=1}^{N} p(x^n | \theta') d\theta'.$$
Bayesian Learning – Univariate Normal Distribution

• Let us use the Bayesian estimation technique to calculate a posteriori density \( p(\theta | D) \) and the desired probability density \( p(x | D) \) for the case \( p(x | \mu) \sim N(\mu, \Sigma) \)

➢ Univariate Case: \( p(\mu | D) \)

Let \( \mu \) be the only unknown parameter

\[
p(x | \mu) \sim N(\mu, \sigma^2)
\]
Bayesian Learning – Univariate Normal Distribution

• Prior probability: normal distribution over \( \mu \),
\[
p(\mu) \sim \mathcal{N}(\mu_0, \sigma_0^2)
\]
\( \mu_0 \) encodes some prior knowledge about the true mean \( \mu \), while \( \sigma_0^2 \) measures our prior uncertainty.

• If \( \mu \) is drawn from \( p(\mu) \) then density for \( x \) is completely determined. Letting \( D = \{x_1, \ldots, x_n\} \) we use
\[
p(\mu \mid D) = \frac{p(D \mid \mu) p(\mu)}{\int p(D \mid \mu) p(\mu) d\mu}
\]
\[
= \alpha \prod_{k=1}^{n} p(x_k \mid \mu) p(\mu)
\]
Bayesian Learning – Univariate Normal Distribution

• Computing the posterior distribution

\[ p(\mu \mid D) \propto p(D \mid \mu) p(\mu) \]

\[ = \alpha' \exp \left[ -\frac{1}{2} \left( \sum_{k=1}^{n} \left( \frac{x_k - \mu}{\sigma} \right)^2 + \left( \frac{\mu - \mu_0}{\sigma_0} \right)^2 \right) \right] \]

\[ = \alpha'' \exp \left[ -\frac{1}{2} \left( \frac{n}{\sigma^2} + \frac{1}{\sigma_0^2} \right) \mu^2 - 2 \left( \frac{1}{\sigma^2} \sum_{k=1}^{n} x_k + \frac{\mu_0}{\sigma_0^2} \right) \mu \right] \]
Bayesian Learning – Univariate Normal Distribution

- Where factors that do not depend on $\mu$ have been absorbed into the constants $\alpha'$ and $\alpha''$
- $p(\mu \mid D)$ is an exponential function of a quadratic function of $\mu$ i.e. it is a normal density.
- $p(\mu \mid D)$ remains normal for any number of training samples.
- If we write

$$p(\mu \mid D) = \frac{1}{\sqrt{2\pi}\sigma_n} \exp \left[ -\frac{1}{2} \left( \frac{\mu - \mu_n}{\sigma_n} \right)^2 \right]$$

then identifying the coefficients, we get
Bayesian Learning – Univariate Normal Distribution

\[
\frac{1}{\sigma_n^2} = \frac{n}{\sigma^2} + \frac{1}{\sigma_0^2} \quad \frac{\mu_n}{\sigma_n^2} = \frac{n}{\sigma^2} \hat{\mu}_n + \frac{\mu_0}{\sigma_0^2}
\]

where \( \hat{\mu}_n = \frac{1}{n} \sum_{k=1}^{n} x_k \) is the sample mean.

- Solving explicitly for \( \mu_n \) and \( \sigma_n^2 \) we obtain

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

and

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

- \( \mu_n \) represents our best guess for \( \mu \) after observing \( n \) samples.
- \( \sigma_n^2 \) measures our uncertainty about this guess.
- \( \sigma_n^2 \) decreases monotonically with \( n \) (approaching \( \sigma^2 / n \) as \( n \) approaches infinity).
Bayesian Learning – Univariate Normal Distribution

• Each additional observation decreases our uncertainty about the true value of $\mu$.

• As $n$ increases, $p(\mu | D)$ becomes more and more sharply peaked, approaching a Dirac delta function as $n$ approaches infinity. This behavior is known as Bayesian Learning.
Bayesian Learning – Univariate Normal Distribution

• In general, $\mu_n$ is a linear combination of $\hat{\mu}_n$ and $\mu_0$, with coefficients that are non-negative and sum to 1.
• Thus $\mu_n$ lies somewhere between $\hat{\mu}_n$ and $\mu_0$.
• If $\sigma_0 \neq 0$, $\mu_n \rightarrow \hat{\mu}_n$ as $n \rightarrow \infty$
• If $\sigma_0 = 0$, our a priori certainty that $\mu = \mu_0$ is so strong that no number of observations can change our opinion.
• If $\sigma_0 > \sigma$, a priori guess is very uncertain, and we take $\mu_n = \hat{\mu}_n$
• The ratio $\sigma^2 / \sigma_0^2$ is called dogmatism.
Bayesian Learning – Univariate Normal Distribution

The Univariate Case: \( p(x \mid D) \)

\[
p(x \mid D) = \int p(x \mid \mu)P(\mu \mid D)d\mu
\]

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

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

where

\[
f(\sigma, \sigma_n) = \int \exp \left[ -\frac{1}{2} \frac{\sigma^2 + \sigma_n^2}{\sigma^2\sigma_n^2} \left( \mu - \frac{\sigma_n^2 x + \sigma^2 \mu_n}{\sigma^2 + \sigma_n^2} \right)^2 \right] d\mu
\]
Bayesian Learning – Univariate Normal Distribution

• Since \( p(x | D) \propto \exp \left[ -\frac{1}{2} \frac{(x - \mu_n)^2}{\sigma^2 + \sigma_n^2} \right] \) we can write

\[
p(x | D) : N(\mu_n, \sigma^2 + \sigma_n^2)
\]

• To obtain the class conditional probability \( p(x | D) \), whose parametric form is known to be \( p(x | \mu) : N(\mu, \sigma) \) we replace \( \mu \) by \( \mu_n \) and \( \sigma^2 \) by \( \sigma^2 + \sigma_n^2 \)

• The conditional mean \( \mu_n \) is treated as if it were the true mean, and the known variance is increased to account for the additional uncertainty in \( x \) resulting from our lack of exact knowledge of the mean \( \mu \).
Example (demo-MAP)

• We have N points which are generated by one dimensional Gaussian,
  \[ p(x \mid \mu) = G_x[\mu, 1]. \] Since we think that the mean should not be very big we use as a prior
  \[ p(\mu) = G_\mu[0, \alpha^2], \] where \( \alpha \) is a hyperparameter. The total objective function is:
  \[
  E \propto - \sum_{n=1}^{N} (x_n - \mu)^2 - \frac{\mu^2}{\alpha^2}
  \]
  which is maximized to give,
  \[
  \mu = \frac{1}{N + \frac{1}{\alpha^2}} \sum_{n=1}^{N} x_n
  \]
  For \( N \ll \frac{1}{\alpha^2} \) influence of prior is negligible and result is ML estimate. But for very strong belief in the prior \( \frac{1}{\alpha^2} \ll N \) the estimate tends to zero. Thus, if few data are available, the prior will bias the estimate towards the prior expected value.
We have seen that $p(D|\theta) = \prod_{k=1}^{n} p(x_k|\theta)$, Let us define $D^n = \{x_1, ..., x_n\}$

Then

$p(D^n|\theta) = p(x_n|\theta) p(D^{n-1}|\theta)$.

Substituting into $p(\theta|D)$, and using Bayes we have:

$$p(\theta|D^n) = \frac{p(D^n|\theta)p(\theta)}{\int p(D^n|\theta)p(\theta)d\theta} = \frac{p(x_n|\theta)p(D^{n-1}|\theta)p(\theta)}{\int p(x_n|\theta)p(D^{n-1}|\theta)p(\theta)d\theta}$$

$$= \frac{p(x_n|\theta)p(\theta|D^{n-1}) \frac{p(D^{n-1})}{p(\theta)}}{\int p(x_n|\theta)p(\theta|D^{n-1}) \frac{p(D^{n-1})}{p(\theta)} p(\theta)d\theta}$$

Finally

$$p(\theta|D^n) = \frac{p(x_n|\theta)p(\theta|D^{n-1})}{\int p(x_n|\theta)p(\theta|D^{n-1})d\theta}$$
Recursive Bayesian Incremental Learning

- While $p(\theta|D^0) = p(\theta)$, repeated use of this eq. produces a sequence
  \[ p(\theta), p(\theta | x_1), p(\theta | x_1, x_1), \ldots \]

- This is called the recursive Bayes approach to the parameter estimation. (Also incremental or on-line learning).

- When this sequence of densities converges to a Dirac delta function centered about the true parameter value, we have Bayesian learning.
Maximal Likelihood vs. Bayesian

• ML and Bayesian estimations are asymptotically equivalent and “consistent”. They yield the same class-conditional densities when the size of the training data grows to infinity.
• ML is typically computationally easier: in ML we need to do (multidimensional) differentiation and in Bayesian (multidimensional) integration.
• ML is often easier to interpret: it returns the single best model (parameter) whereas Bayesian gives a weighted average of models.
• But for a finite training data (and given a reliable prior) Bayesian is more accurate (uses more of the information).
• Bayesian with “flat” prior is essentially ML; with asymmetric and broad priors the methods lead to different solutions.
Consider two-class multivariate normal distributions \( p(x|\omega_i) : N(\mu_i, \Sigma) \) with the same covariance. If priors are equal then Bayesian error rate is given by

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

where \( r^2 \) is the squared Mahalanobis distance:

\[
r^2 = (\mu_1 - \mu_2)^t \Sigma^{-1} (\mu_1 - \mu_2).
\]

Thus the probability of error decreases as \( r \) increases. In the conditionally independent case \( \Sigma = diag(\sigma_i^2, ..., \sigma_d^2) \) and

\[
r^2 = \sum_{i=1}^{d} \left( \frac{\mu_{i1} - \mu_{i2}}{\sigma_i} \right)^2
\]
Problems of Dimensionality

• While classification accuracy can become better with growing of dimensionality (and an amount of training data),

– beyond a certain point, the inclusion of additional features leads to worse than better performance
– computational complexity grows
– the problem of overfitting arises
Occam's Razor

• "Pluralitas non est ponenda sine neccesitate" or "plurality should not be posited without necessity." The words are those of the medieval English philosopher and Franciscan monk William of Occam (ca. 1285-1349).

Decisions based on overly complex models often lead to lower accuracy of the classifier.
What is feature reduction?

- Feature reduction refers to the mapping of the original high-dimensional data onto a lower-dimensional space.
  - Criterion for feature reduction can be different based on different problem settings.
    - Unsupervised setting: minimize the information loss
    - Supervised setting: maximize the class discrimination
- Given a set of data points of \( p \) variables \( \{x_1, x_2, \cdots, x_n\} \)

Compute the linear transformation (projection)

\[
G \in \mathbb{R}^{p \times d} : x \in \mathbb{R}^p \rightarrow y = G^T x \in \mathbb{R}^d \quad (d \ll p)
\]
What is feature reduction?

$\mathbb{R}^d \times p \rightarrow \mathbb{R}^d$

$G \in \mathbb{R}^{p \times d} : X \rightarrow Y = G^T X \in \mathbb{R}^d$

$G^T \in \mathbb{R}^{d \times p}$

Original data

Linear transformation

reduced data