CS559 Machine Learning
Maximum Likelihood Estimation
Bayesian Estimation

Tian Han

Department of Computer Science
Stevens Institute of Technology

Week 3

Outline

- Introduction
  - Univariate Gaussian Example
- Maximum Likelihood Estimation
  - The General Principle
  - Multivariate Gaussian
  - Sequential Estimation
- Bayesian Estimation
  - Example
  - The General Principle
  - Connection to Bayesian Decision

Introduction

Design the Classifier

HAVE prior $P(\omega)$ and class conditional $p(\mathbf{x}|\omega)$.

- Optimal classifier:
  - posterior $p(\omega|\mathbf{x})$
  - conditional risk $R(\alpha_i|\mathbf{x}) = \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j)p(\omega_j|\mathbf{x})$
- In practice, we rarely have this complete information!

Design the Classifier

HAVE prior $P(\omega)$ and class conditional $p(\mathbf{x}|\omega)$.

- Optimal classifier:
  - posterior $p(\omega|\mathbf{x})$
  - conditional risk $R(\alpha_i|\mathbf{x}) = \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j)p(\omega_j|\mathbf{x})$
- In practice, we rarely have this complete information!

ONLY HAVE a number of training samples.

- Prior estimation is easy.
- Class conditional $p(\mathbf{x}|\omega)$ is hard. (sample too small, $\mathbf{x}$ high dimension)

Parametrization of $p(\mathbf{x}|\omega)$

Parametrization: assume the $p(\mathbf{x}|\omega)$ has KNOWN form but UNKNOWN parameters.

- E.g., assume $p(\mathbf{x}|\omega)$ is Gaussian, i.e., $\mathrm{N}(x|\mu, \sigma^2)$, but μ, σ unknown.

Parametrization of $p(\mathbf{x}|\omega)$

Parametrization: assume the $p(\mathbf{x}|\omega)$ has KNOWN form but UNKNOWN parameters.

- E.g., assume $p(\mathbf{x}|\omega)$ is Gaussian, i.e., $\mathrm{N}(x|\mu, \sigma^2)$, but μ, σ unknown.

Estimating the unknown density function $p(\mathbf{x}|\omega)$
$\rightarrow$ **parameter estimation**.

Parametrization of $p(\mathbf{x}|\omega)$

Parametrization: assume the $p(\mathbf{x}|\omega)$ has KNOWN form but UNKNOWN parameters.

- E.g., assume $p(\mathbf{x}|\omega)$ is Gaussian, i.e., $\mathrm{N}(x|\mu, \sigma^2)$, but μ, σ unknown.

Estimating the unknown density function $p(\mathbf{x}|\omega)$
$\rightarrow$ **parameter estimation**.

In this lecture, Maximum Likelihood Estimation (MLE) and Bayesian Estimation (BE).

- Results always identical, but underlying assumptions are different
- Using either estimation, will use $p(\omega|\mathbf{x})$ as our classifier.

Simple Example: Univariate Gaussian

Recall Gaussian Distribution:

$$\mathrm{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

- $\mathbb{E}(x) = \mu$
- $\mathrm{Var}(x) = \sigma^2$

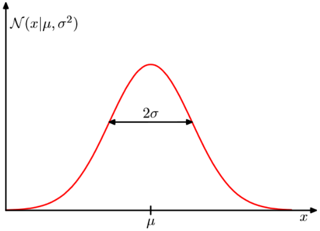

Figure: univariate Gaussian [C.Bishop 2006]

Likelihood Function

Given N training samples $\{x_1, ..., x_N\}$, denote as $\mathcal{D} = (x_1, ..., x_N)^T$, assume:

- drawn independently from Gaussian distribution whose mean μ and variance σ^2 are unknown.
- *independent and identically distributed*, abbreviated as **i.i.d**

The probability of the whole dataset $\mathcal{D}$ is:

$$p(\mathcal{D}|\mu, \sigma^2) = \prod_{n=1}^{N} \mathrm{N}(x_n|\mu, \sigma^2)$$

- Likelihood function for the Gaussian.

Maximize Likelihood

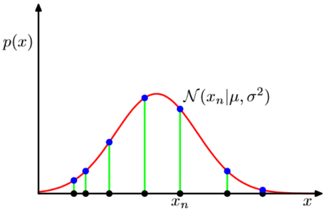

Figure: Likelihood function for Gaussian [C.Bishop 2006]

Use training samples to determine the parameters in a probability distribution:

- Find parameter values that **maximize the likelihood** function.
- i.e., Adjusting the μ and σ^2 of Gaussian so as to **maximize** the product: $\prod_{n=1}^{N} \mathrm{N}(x_n|\mu, \sigma^2)$.

Maximize Likelihood

In practice, consider **maximize the log of the likelihood** function:

$$\arg\max_{\mu,\sigma^2} p(\mathcal{D}|\mu,\sigma^2) \equiv \arg\max_{\mu,\sigma^2} \ln(p(\mathcal{D}|\mu,\sigma^2))$$

Maximize Likelihood

In practice, consider **maximize the log of the likelihood** function:

$$\arg\max_{\mu,\sigma^2} p(\mathcal{D}|\mu,\sigma^2) \equiv \arg\max_{\mu,\sigma^2} \ln(p(\mathcal{D}|\mu,\sigma^2))$$

$$LLD = \ln p(\mathcal{D}|\mu,\sigma^2) = -\frac{1}{2\sigma^2}\sum_{n=1}^{N}(x_n-\mu)^2 - \frac{N}{2}\ln\sigma^2 - \frac{N}{2}\ln(2\pi)$$

Maximize Likelihood

In practice, consider **maximize the log of the likelihood** function:

$$\arg\max_{\mu,\sigma^2} p(\mathcal{D}|\mu,\sigma^2) \equiv \arg\max_{\mu,\sigma^2} \ln(p(\mathcal{D}|\mu,\sigma^2))$$

$$LLD = \ln p(\mathcal{D}|\mu,\sigma^2) = -\frac{1}{2\sigma^2}\sum_{n=1}^{N}(x_n-\mu)^2 - \frac{N}{2}\ln\sigma^2 - \frac{N}{2}\ln(2\pi)$$

Take $\frac{\partial LLD}{\partial \mu} = 0$ and $\frac{\partial LLD}{\partial \sigma^2} = 0$:

Maximize Likelihood

In practice, consider **maximize the log of the likelihood** function:

$$\arg\max_{\mu,\sigma^2} p(\mathcal{D}|\mu,\sigma^2) \equiv \arg\max_{\mu,\sigma^2} \ln(p(\mathcal{D}|\mu,\sigma^2))$$

$$LLD = \ln p(\mathcal{D}|\mu,\sigma^2) = -\frac{1}{2\sigma^2}\sum_{n=1}^{N}(x_n-\mu)^2 - \frac{N}{2}\ln\sigma^2 - \frac{N}{2}\ln(2\pi)$$

Take $\frac{\partial LLD}{\partial \mu} = 0$ and $\frac{\partial LLD}{\partial \sigma^2} = 0$:

- $\hat{\mu} = \frac{1}{N}\sum_{n=1}^{N} x_n$, i.e., sample mean.
- $\hat{\sigma}^2 = \frac{1}{N}\sum_{n=1}^{N}(x_n - \hat{\mu})^2$, i.e., sample variance

Bias

The Maximum Likelihood estimations $\hat{\mu}$, $\hat{\sigma}^2$ depends on training data $\mathcal{D}$ which contains N samples. Consider different possible set of training samples, on average,

$$\begin{aligned} \mathbb{E}(\hat{\mu}) &= \mu \\ \mathbb{E}(\hat{\sigma}^2) &= \frac{N-1}{N}\sigma^2 \end{aligned}$$

Bias

The Maximum Likelihood estimations $\hat{\mu}$, $\hat{\sigma}^2$ depends on training data $\mathcal{D}$ which contains N samples. Consider different possible set of training samples, on average,

$$\begin{aligned} \mathbb{E}(\hat{\mu}) &= \mu \\ \mathbb{E}(\hat{\sigma}^2) &= \frac{N-1}{N}\sigma^2 \end{aligned}$$

- Maximum Likelihood Estimation $\hat{\sigma}^2$ is *biased*. i.e., $\mathbb{E}(\hat{\sigma}^2) \neq \sigma^2$
- Under estimate the true variance σ^2.

Illustration

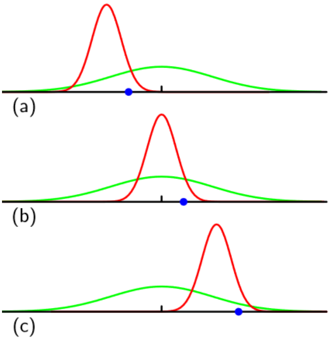

Figure: Averaged across three sets, mean is correct, variance is under-estimated. [C.Bishop 2006]

However, when we have large amount training samples, i.e., $N \to \infty$, the variance estimator tends to become unbiased.

Maximum Likelihood Estimation

General Principle
Multivariate Gaussian
Sequential Estimation

The General Principle

Setting and Assumption

- Training data $\mathcal{D}$ contains the collection of samples from c classes/states, i.e., $\mathcal{D}$ can be partitioned as $\mathcal{D}_1, ..., \mathcal{D}_c$.
- Samples in $\mathcal{D}_j$ are *i.i.d* samples from $p(x|\omega_j)$.
- $p(x|\omega_j)$ has known parametric form (e.g., Gaussian).
- θ_j consists of the unknown parameters that need to be estimated. θ_j for ω_j.
- Goal: use training samples $\mathcal{D}$, estimate unknown parameters $\theta_1, ..., \theta_c$ associated with each category.

Independence Across Classes

We have training data for each class.

When estimating parameters for one class, will only use the data collected for that class.

The samples in $\mathcal{D}_i$ give no information about θ_j if $i \neq j$.

- Handle each class separately.

The General Principle

Use training samples $\mathcal{D} = \{x_1, x_2, ..., x_n\}$ drawn **i.i.d** from probability density $p(x|\theta)$ to estimate the **unknown** parameter vector θ.

The likelihood function for whole dataset $\mathcal{D}$:

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

- Maximum Likelihood Estimation (MLE)of θ, i.e., $\hat{\theta}$, should maximize $p(\mathcal{D}|\theta)$.
- It is the value that best agrees with the observed training data $\mathcal{D}$.

Finding Optimal

- For $\theta = (\theta_1, ..., \theta_p)^T$, define gradient operator:

$$\nabla_\theta \equiv \begin{bmatrix} \frac{\partial}{\partial \theta_1} \\ \vdots \\ \frac{\partial}{\partial \theta_p} \end{bmatrix}$$

- log-likelihood function $l(\theta)$: $l(\theta) \equiv \ln p(\mathcal{D}|\theta)$
- Maximum Likelihood Estimation $\hat{\theta}$:

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

Finding Optimal (Con't)

- Log-likelihood:

$$\begin{aligned} l(\theta) &\equiv \ln p(\mathcal{D}|\theta) \\ &= \sum_{k=1}^{n} \ln p(x_k|\theta) \end{aligned}$$

- Taking gradients w.r.t θ

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

- Necessary condition: $\nabla_\theta l = 0$
- A solution $\hat{\theta}$ might represent local/global minimum/maximum, saddle point etc. Have to check.

Multivariate Gaussian

Multivariate Gaussian

Univariate Gaussian:

$$\mathrm{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Multivariate Gaussian

Univariate Gaussian:

$$\mathrm{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Multivariate Gaussian:

$$\mathrm{N}(\mathbf{x}|\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1} (\mathbf{x}-\mu)\right)$$

where μ is D-dimensional mean vector, Σ is $D \times D$ covariance matrix, and $|\Sigma|$ denotes the determinant of Σ.

MLE for Multivariate Gaussian

Given training samples $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ which assumed to be *i.i.d* samples from multivariate Gaussian $p(\mathbf{x}|\mu, \Sigma)$. μ and Σ are assumed to be unknown and need to be estimated.

- Log-likelihood function for $\mathcal{D}$:

$$\ln p(\mathcal{D}|\mu, \Sigma) = -\frac{ND}{2}\ln(2\pi) - \frac{N}{2}\ln|\Sigma| - \frac{1}{2}\sum_{n=1}^{N}(\mathbf{x}_n - \mu)^T \Sigma^{-1}(\mathbf{x}_n - \mu)$$

- $\frac{\partial}{\partial \mu}\ln p(\mathcal{D}|\mu, \Sigma) = \sum_{n=1}^{N} \Sigma^{-1}(\mathbf{x}_n - \mu) = 0$
- $\frac{\partial}{\partial \Sigma}\ln p(\mathcal{D}|\mu, \Sigma) = 0$, quite involved.

MLE for Multivariate Gaussian

The Maximum Likelihood Estimations $\hat{\mu}$ and $\hat{\Sigma}$ are:

$$\hat{\mu} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n$$

$$\hat{\Sigma} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \hat{\mu})(\mathbf{x}_n - \hat{\mu})^T$$

Similarly, we have:

$$\mathbb{E}(\hat{\mu}) = \mu$$

$$\mathbb{E}(\hat{\Sigma}) = \frac{N-1}{N}\Sigma \neq \Sigma$$

Biased estimator for Σ, may use:
$\tilde{\Sigma} = \frac{1}{N-1}\sum_{n=1}^{N}(\mathbf{x}_n - \hat{\mu})(\mathbf{x}_n - \hat{\mu})^T$ (unbiased)

Sequential Estimation

Motivation

The previous derived Maximum Likelihood Estimation is derived using *whole* dataset. However, in many cases:

- new data available in on-line application.
- the whole training dataset is *too large*.

Sequential estimation: needed in most of the model training, especially the learning of deep models.

Example for Mean Estimation

Consider the MLE of mean, i.e., $\hat{\mu}$, for univariate Gaussian.

$\hat{\mu}^{(N)}$: MLE estimation based on N observations.

Dissect out the contribution from final point x_N, we have:

$$
\begin{aligned}
\hat{\mu}^{(N)} &= \frac{1}{N}\sum_{n=1}^{N} x_n \\
&= \frac{1}{N}x_N + \frac{1}{N}\sum_{n=1}^{N-1} x_n \\
&= \frac{1}{N}x_N + \frac{N-1}{N}\hat{\mu}^{(N-1)} \\
&= \hat{\mu}^{(N-1)} + \frac{1}{N}\left(x_N - \hat{\mu}^{(N-1)}\right)
\end{aligned}
$$

Interpretation

We have:

$$\hat{\mu}^{(N)} = \hat{\mu}^{(N-1)} + \frac{1}{N}(x_N - \hat{\mu}^{(N-1)})$$

- After observing $N-1$ points, we have $\hat{\mu}^{(N-1)}$.
- Now observe x_N, have 'error signal' $(x_N - \hat{\mu}^{(N-1)})$.
- Revise $\hat{\mu}^{(N-1)}$ following direction of 'error signal'.

General Formulation

Consider random variables θ and z which follows joint distribution $p(z, \theta)$.

Define *regression function*:

$$f(\theta) \equiv \mathbb{E}(z|\theta) = \int zp(z|\theta)dz$$

Goal: find root $\theta^\star$, such that $f(\theta^\star) = 0$

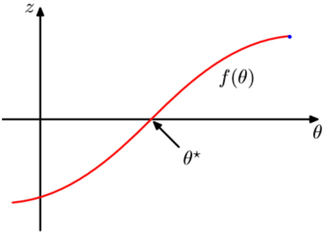

Figure: Regression function $f(\theta)$ and root $\theta^\star$[C.Bishop 2006]

Robbins-Monro Algorithm

Suppose observe one (or batch) z at a time, find the corresponding sequential estimation scheme for $\theta^\star$ (i.e., $f(\theta^\star) = 0$)

Robbins-Monro procedure:

$$\theta^{(N)} = \theta^{(N-1)} - a_{N-1} z(\theta^{(N-1)})$$

where $z(\theta^{(N-1)})$ is an observed value of z when θ takes the value $\theta^{(N)}$.

- Assume conditional variance of z is finite and some conditions on $\{a_N\}$ sequence.
- The procedure converge to root $\theta^\star$ with probability one.

Robbins-Monro for MLE

Suppose we have likelihood function $p(x|\theta)$, then the maximum likelihood estimation $\hat{\theta}$ satisfy:

$$\frac{\partial}{\partial\theta}\left[-\frac{1}{N}\sum_{n=1}^{N}\ln p(x_n|\theta)\right]=0$$

When $N\to\infty$, want:

$$-\lim_{N\to\infty}\frac{1}{N}\sum_{n=1}^{N}\frac{\partial}{\partial\theta}\ln p(x_n|\theta)=\mathbb{E}_x\left[-\frac{\partial}{\partial\theta}\ln p(x|\theta)\right]=0$$

Find the maximum likelihood solution corresponds to finding the root of a regression function.

Robbins-Monro for MLE

Use Robbins-Monro Algorithm for MLE:

$$\theta^{(N)} = \theta^{(N-1)} - a_{N-1} \frac{\partial}{\partial \theta^{(N-1)}} [-\ln p(x_N | \theta^{(N-1)})]$$

Specifically, if likelihood $p(x|\theta)$ is Gaussian (i.e., $\mathrm{N}(x|\mu, \sigma^2)$), then $\theta^{(N)}$ is the MLE estimate $\hat{\mu}^{(N)}$ of the mean of the Gaussian. And random variable z is given by:

$$z = \frac{\partial}{\partial \hat{\mu}} [-\ln p(x|\hat{\mu}, \sigma^2)] = -\frac{1}{\sigma^2}(x - \hat{\mu})$$

Choose $a_N = \frac{\sigma^2}{N}$, we get $\hat{\mu}^{(N)} = \hat{\mu}^{(N-1)} + \frac{1}{N}(x_N - \hat{\mu}^{(N-1)})$

Robbins-Monro for MLE

Recall:

$$z = \frac{\partial}{\partial \hat{\mu}}[-\ln p(x|\hat{\mu}, \sigma^2)] = -\frac{1}{\sigma^2}(x - \hat{\mu})$$

Suppose the training samples $\{x_1, \ldots, x_n\}$ follows from $\mathrm{N}(\mu, \sigma^2)$. The distribution of z is Gaussian with mean $-\frac{1}{\sigma^2}(\mu - \hat{\mu})$ which is also the regression function. The root for such regression function (which is also the maximum likelihood solution) is $\hat{\mu}^\star = \mu$. Thus the sequential MLE using Robbins-Monro could obtain the estimation which is the true mean.

Bayesian Estimation

Example for Gaussian
General Principle
Connecting to Bayesian Decision Problem

Example for Gaussian

Bayesian Inference for Gaussian

Recall: based on training $\mathcal{D} = \{x_1, \ldots, x_n\}$, estimate μ, σ^2 to maximize $p(\mathcal{D}|\mu, \sigma^2)$.

- In Maximum Likelihood framework: μ, σ^2 are unknown but **fixed**.
- In Bayesian Estimation framework: μ, σ^2 are unknown and **random variables**.

Define Prior on μ

Assume σ^2 is known, only estimate/infer μ from N observations, i.e., $\mathcal{D} = \{x_1, \ldots, x_N\}$.

The likelihood function which can be viewed as a function of μ is given by:

$$p(\mathcal{D}|\mu) = \prod_{n=1}^{N} p(x_n|\mu) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left(-\frac{1}{2\sigma^2}\sum_{n=1}^{N}(x_n - \mu)^2\right)$$

Note the only unknown is μ. Prior knowledge about μ can be expressed by *known* prior density $p(\mu)$ which is assumed as:

$$p(\mu) = \mathrm{N}(\mu|\mu_0, \sigma_0^2)$$

Prior on μ

Prior density on μ:

$$p(\mu) = \mathrm{N}(\mu|\mu_0, \sigma_0^2)$$

- μ_0 is our best priori guess for μ, and σ_0 measures the uncertainty about this guess.
- The crucial assumption is we **know** the prior distribution.

Think in this way

- A value is drawn for μ from $p(\mu)$.
- Such value becomes the true value of μ, and will be used to determines the density of training data $\mathcal{D}$.

Think in this way

- A value is drawn for μ from $p(\mu)$.
- Such value becomes the true value of μ, and will be used to determines the density of training data $\mathcal{D}$.

How does the training data $\mathcal{D}$ affects our beliefs about the true value of μ?

Estimating μ: $p(\mu|\mathcal{D})$

Bayes formula to get posterior distribution:

$$\begin{aligned} p(\mu|\mathcal{D}) &= \frac{p(\mathcal{D}|\mu)p(\mu)}{\int p(\mathcal{D}|\mu)p(\mu)d\mu} \\ &= C * \prod_{n=1}^{N} p(x_n|\mu)p(\mu) \end{aligned}$$

C is the normalization constant which depends on $\mathcal{D}$ and independent of μ.

$p(\mu|\mathcal{D})$ is still Gaussian

After some manipulations:

$$p(\mu|\mathcal{D}) = \mathrm{N}(\mu|\mu_N, \sigma_N^2)$$

where:

$$\begin{aligned} \mu_N &= \frac{\sigma^2}{N\sigma_0^2 + \sigma^2}\mu_0 + \frac{N\sigma_0^2}{N\sigma_0^2 + \sigma^2}\hat{\mu} \\ \sigma_N^2 &= \frac{\sigma_0^2\sigma^2}{N\sigma_0^2 + \sigma^2} \\ \hat{\mu} &= \frac{1}{N}\sum_{n=1}^{N} x_n \end{aligned}$$

Recall $\hat{\mu}$ is the maximum likelihood solution.

Interpretation: Vary Number of Samples

$$\begin{aligned}\mu_N &= \frac{\sigma^2}{N\sigma_0^2 + \sigma^2}\mu_0 + \frac{N\sigma_0^2}{N\sigma_0^2 + \sigma^2}\hat{\mu} \\ \sigma_N^2 &= \frac{\sigma_0^2\sigma^2}{N\sigma_0^2 + \sigma^2}\end{aligned}$$

- μ_N represents our best guess for μ after observing N training samples, σ_N^2 measures our uncertainty about this guess.
- μ_N: compromise between the prior mean μ_0 and maximum likelihood solution $\hat{\mu}$.
- $N = 0$: $\mu_N = \mu_0$, $\sigma_N^2 = \sigma_0^2$
- $N \to \infty$: $\mu_N \to \hat{\mu}$, $\sigma_N^2 \to 0$

Bayesian Learning

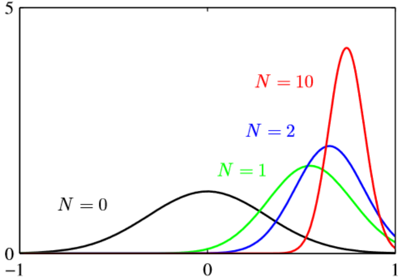

Figure: Bayesian inference for μ. [C.Bishop 2006]

- When number of observations N increase, σ_N^2 decrease monotonically, $p(\mu|\mathcal{D})$ become more and more peaked.
- When infinite number of observations $N \to \infty$, bayesian estimation recovers the maximum likelihood estimation for μ.

Interpretation: σ^2 vs. σ_0^2

$$
\begin{aligned}
\mu_N &= \frac{\sigma^2}{N\sigma_0^2+\sigma^2}\mu_0 + \frac{N\sigma_0^2}{N\sigma_0^2+\sigma^2}\hat{\mu} \\
\sigma_N^2 &= \frac{\sigma_0^2\sigma^2}{N\sigma_0^2+\sigma^2} \\
\hat{\mu} &= \bar{x}_N = \frac{1}{N}\sum_{n=1}^{N} x_n
\end{aligned}
$$

- $\hat{\mu}$: sample mean, reflect the empirical information in the samples.
- If $\sigma_0 = 0$: $\mu_N = \mu_0$, priori certainty is so strong, no observation will change our opinion.
- If $\sigma_0 \gg \sigma$: $\mu_N \to \bar{x}_N$, priori guess is so uncertain, use only samples to estimate.

The General Principle

The General Principle

- The form of density $p(x|\theta)$ is assumed to be *known*, but the value of parameter vector θ is not known exactly.

The General Principle

- The form of density $p(x|\theta)$ is assumed to be *known*, but the value of parameter vector θ is not known exactly.
- Our initial knowledge about θ is assumed to be contained in a *known* priori density $p(\theta)$.

The General Principle

- The form of density $p(x|\theta)$ is assumed to be *known*, but the value of parameter vector θ is not known exactly.
- Our initial knowledge about θ is assumed to be contained in a *known* priori density $p(\theta)$.
- The rest of our knowledge about θ is contained in $\mathcal{D}$ of $\{x_1, \ldots, x_N\}$ drawn independently from unknown density $p(x)$.

The General Principle

- The form of density $p(x|\theta)$ is assumed to be *known*, but the value of parameter vector θ is not known exactly.
- Our initial knowledge about θ is assumed to be contained in a *known* priori density $p(\theta)$.
- The rest of our knowledge about θ is contained in $\mathcal{D}$ of $\{x_1, \ldots, x_N\}$ drawn independently from unknown density $p(x)$.
- Basic problem: find $p(\theta|\mathcal{D})$.

Compute $p(\theta|\mathcal{D})$

Bayes formula:

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{\int p(\mathcal{D}|\theta)p(\theta)d\theta} \tag{1}$$

Where:

$$p(\mathcal{D}|\theta) = \prod_{n=1}^{N} p(x_n|\theta) \tag{2}$$

- MLE: maximum eqn. (2) to get point estimate $\hat{\theta}$.
- Bayesian estimation: use *all* available information (i.e., prior as well as training samples) to get probability estimation for θ, i.e., $p(\theta|\mathcal{D})$.

Sequential Estimation

Recall in MLE, we show that estimation can be done in a sequential manner to utilize the new collected data. The Bayesian paradigm naturally leads to sequential view (write $\mathcal{D}^N = \{x_1, \ldots, x_N\}$):

$$p(\mathcal{D}^N|\theta) = p(x_N|\theta)p(\mathcal{D}^{N-1}|\theta)$$

Then:

$$\begin{aligned} p(\theta|\mathcal{D}^N) &= \frac{p(x_N|\theta)p(\mathcal{D}^{N-1}|\theta)p(\theta)}{\int p(x_N|\theta)p(\mathcal{D}^{N-1}|\theta)p(\theta)d\theta} \\ &= C * \underbrace{\left[p(\theta)\prod_{n=1}^{N-1} p(x_n|\theta)\right]}_{\propto p(\theta|\mathcal{D}^{N-1})} p(x_N|\theta) \end{aligned}$$

Sequential Estimation

$$p(\theta|\mathcal{D}^N) \quad = \quad C * \underbrace{\left[p(\theta)\prod_{n=1}^{N-1} p(x_n|\theta)\right]}_{\propto\, p(\theta|\mathcal{D}^{N-1})} p(x_N|\theta)$$

- Use such sequential procedure, we get $p(\theta)$, $p(\theta|x_1)$, $p(\theta|x_1, x_2)$ and so forth.
- Example of *on-line* learning.

Connecting to Bayesian Decision Problem

Connection to Decision

Suppose we have c state of nature $\omega_1, \ldots, \omega_c$, recall the decision theory discussed in previous chapter is based on posterior $p(\omega_i|x)$.

- $P(\omega_i)$ and $p(x|\omega_i)$ are unknown
- use training samples $\mathcal{D}$ to estimate, denote as $p(\omega_i|x, \mathcal{D})$.

We have:

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

Assume independence across class:

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

Each class is treated independently.

Connection to Decision

Treat each class separately:

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

We have c separate problems of the form: **use a set $\mathcal{D}$ of samples drawn independently according to the fixed but unknown probability density $p(x)$ to determine $p(x|\omega_i,\mathcal{D}_i)$ which is simplified as $p(x|\mathcal{D})$.**

Determine $p(x|\mathcal{D})$

Assume: (1) $\{x_1, \ldots, x_N\} \sim p(x)$, $p(x)$ is unknown but has *known* parametric form, i.e, function $p(x|\theta)$ is completely known, θ is unknown.

Determine $p(x|\mathcal{D})$

Assume: (1) $\{x_1, \ldots, x_N\} \sim p(x)$, $p(x)$ is unknown but has *known* parametric form, i.e, function $p(x|\theta)$ is completely known, θ is unknown.

(2) the prior knowledge about θ is contained in *known* prior density $p(\theta)$.

Determine $p(x|\mathcal{D})$

Assume: (1) $\{x_1, \ldots, x_N\} \sim p(x)$, $p(x)$ is unknown but has *known* parametric form, i.e, function $p(x|\theta)$ is completely known, θ is unknown.

(2) the prior knowledge about θ is contained in *known* prior density $p(\theta)$.

Goal: compute $p(x|\mathcal{D})$ which is as close as we can get to obtaining unknown $p(x)$.

Determine $p(x|\mathcal{D})$

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

- The distribution of x is known completely when we know value of the parameter vector θ.
- Links $p(x|\mathcal{D})$ to the posterior density $p(\theta|\mathcal{D})$ for the unknown parameter vector.
- The integration may need Monte-Carlo simulation which is computation intensive.

Example on Univariate Gaussian

Recall the previous example that estimate μ: we assume $p(\mu) \sim \mathrm{N}(\mu|\mu_0, \sigma_0^2)$, $p(x_i|\mu) \sim \mathrm{N}(x|\mu, \sigma^2)$ where σ^2 is known. Then for training set $\mathcal{D}$, we have:

$$p(\mu|\mathcal{D}) = \mathrm{N}(\mu|\mu_N, \sigma_N^2)$$

Example on Univariate Gaussian

Recall the previous example that estimate μ: we assume $p(\mu) \sim \mathrm{N}(\mu|\mu_0, \sigma_0^2)$, $p(x_i|\mu) \sim \mathrm{N}(x|\mu, \sigma^2)$ where σ^2 is known.
Then for training set $\mathcal{D}$, we have:

$$p(\mu|\mathcal{D}) = \mathrm{N}(\mu|\mu_N, \sigma_N^2)$$

Take step further:

$$\begin{aligned} p(x|\mathcal{D}) &= \int p(x|\mu)p(\mu|\mathcal{D})d\mu \\ &= \mathrm{N}(\mu_N, \sigma^2 + \sigma_N^2) \end{aligned}$$

Example on Univariate Gaussian

Recall the previous example that estimate μ: we assume $p(\mu) \sim \mathrm{N}(\mu|\mu_0, \sigma_0^2)$, $p(x_i|\mu) \sim \mathrm{N}(x|\mu, \sigma^2)$ where σ^2 is known. Then for training set $\mathcal{D}$, we have:

$$p(\mu|\mathcal{D}) = \mathrm{N}(\mu|\mu_N, \sigma_N^2)$$

Take step further:

$$\begin{aligned} p(x|\mathcal{D}) &= \int p(x|\mu)p(\mu|\mathcal{D})d\mu \\ &= \mathrm{N}(\mu_N, \sigma^2 + \sigma_N^2) \end{aligned}$$

Note: increased variance to account for additional uncertainty in x due to inexact knowledge of μ.

Example on Univariate Gaussian

Recall the previous example that estimate μ: we assume $p(\mu) \sim \mathrm{N}(\mu|\mu_0, \sigma_0^2)$, $p(x_i|\mu) \sim \mathrm{N}(x|\mu, \sigma^2)$ where σ^2 is known. Then for training set $\mathcal{D}$, we have:

$$p(\mu|\mathcal{D}) = \mathrm{N}(\mu|\mu_N, \sigma_N^2)$$

Take step further:

$$\begin{aligned} p(x|\mathcal{D}) &= \int p(x|\mu)p(\mu|\mathcal{D})d\mu \\ &= \mathrm{N}(\mu_N, \sigma^2 + \sigma_N^2) \end{aligned}$$

Note: increased variance to account for additional uncertainty in x due to inexact knowledge of μ.

$p(x|\mathcal{D})$ is the desired class conditional density $p(x|\omega_i, \mathcal{D}_i)$, together with prior $P(\omega_i)$, we define the posterior $p(\omega_i|x, \mathcal{D})$ based on which classifier is built.

Maximum Likelihood and Bayesian Estimation

- When $N \to \infty$ (infinite training data), maximum likelihood and Bayesian solutions are equivalent. i.e., $p(x|\mathcal{D}) \approx p(x|\hat{\theta})$.

Maximum Likelihood and Bayesian Estimation

- When $N \to \infty$ (infinite training data), maximum likelihood and Bayesian solutions are equivalent. i.e., $p(x|\mathcal{D}) \approx p(x|\hat{\theta})$.
- When use "flat" or uniform prior $p(\theta)$, Bayesian estimation is equivalent to maximum likelihood estimation.

Maximum Likelihood and Bayesian Estimation

- When $N \to \infty$ (infinite training data), maximum likelihood and Bayesian solutions are equivalent. i.e., $p(x|\mathcal{D}) \approx p(x|\hat{\theta})$.
- When use "flat" or uniform prior $p(\theta)$, Bayesian estimation is equivalent to maximum likelihood estimation.
- Maximum Likelihood method is computational efficient, i.e., only need gradient. Bayesian methods needs integration which are hard to approximate. (recall $p(x|\mathcal{D}) = \int p(x|\theta)p(\theta|\mathcal{D})d\theta$)

Maximum Likelihood and Bayesian Estimation

- When $N \to \infty$ (infinite training data), maximum likelihood and Bayesian solutions are equivalent. i.e., $p(x|\mathcal{D}) \approx p(x|\hat{\theta})$.
- When use "flat" or uniform prior $p(\theta)$, Bayesian estimation is equivalent to maximum likelihood estimation.
- Maximum Likelihood method is computational efficient, i.e., only need gradient. Bayesian methods needs integration which are hard to approximate. (recall $p(x|\mathcal{D}) = \int p(x|\theta)p(\theta|\mathcal{D})d\theta$)
- Maximum likelihood give single best model, while bayesian method give a weighted average of models.

Maximum Likelihood and Bayesian Estimation

- When $N \to \infty$ (infinite training data), maximum likelihood and Bayesian solutions are equivalent. i.e., $p(x|\mathcal{D}) \approx p(x|\hat{\theta})$.
- When use "flat" or uniform prior $p(\theta)$, Bayesian estimation is equivalent to maximum likelihood estimation.
- Maximum Likelihood method is computational efficient, i.e., only need gradient. Bayesian methods needs integration which are hard to approximate. (recall $p(x|\mathcal{D}) = \int p(x|\theta)p(\theta|\mathcal{D})d\theta$)
- Maximum likelihood give single best model, while bayesian method give a weighted average of models.
- Bayesian methods use more of the information through $p(\theta|\mathcal{D})$, if such information is reliable, then its better than maximum likelihood.