Brain-like Computing
&
Machine Intelligence

Model Inference and Averaging

Dept. Computer Science & Engineering
Shanghai Jiao Tong University

Contents

Course CS &30... SJTU Statistical Learning theory & Application

OBE of The Chapter

- To understand the relations between bootstrap and MLE
- To understand the basic concept of MLE, and grasp the technical implementation of MLE
- To master theoretical issues on MLE
  - Convergence, Convergent rate, Advantages
- To understand the formulation & implementation of Bayesians
- To grasp the formulation EM algorithm
  - Convergence and Implementation

Course CS & 04... SJTU Statistical Learning theory & Application

Bootstrap by Basis Expansions

- Consider a linear expansion $$\mu(x) = \sum_{j=1}^{N} \beta_j h_j(x)$$
- The least square error solution $$\hat{\beta} = (H^T H)^{-1} H^T y$$
- The Covariance of β

$$cor(\hat{\beta}) = (H^T H)^{-1} \hat{\sigma}^2; \qquad \hat{\sigma}^2 = \sum (y_i - \hat{\mu}(x_i))^2 / N$$

Course CS &304H SJTU Statistical Learning theory & Application

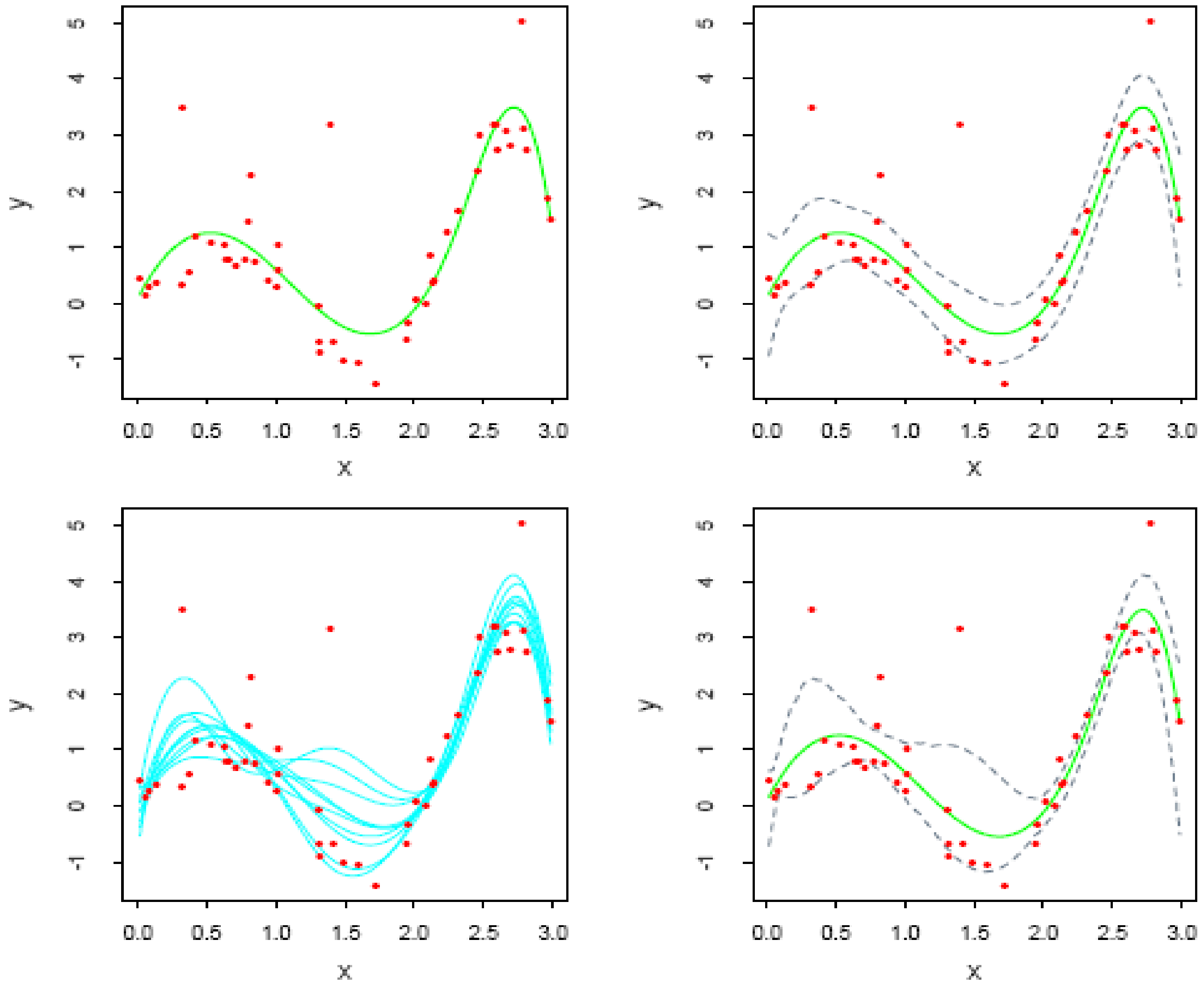


FIGURE 8.2. *(Top left:) B-spline smooth of data. (Top right:) B-spline smooth plus and minus 1.96× standard error bands. (Bottom left:) Ten bootstrap replicates of the B-spline smooth. (Bottom right:) B-spline smooth with 95% standard error bands computed from the bootstrap distribution.*

nd Averaging

Parametric Model

- Assume a parameterized probability density (parametric model) for observations

$$z_i \sim g_\theta(z)$$

E.g. normal distribution $\theta = (\mu, \sigma^2)$; $g_\theta(z) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(z-\mu)^2/\sigma^2}$

Course CS &304H SJTU Statistical Learning theory & Application

Maximum Likelihood Inference

- Suppose we are trying to measure the true value of some quantity (x_T).
  - We make repeated measurements of this quantity $\{x_1, x_2, \ldots, x_n\}$.
  - The standard way to estimate x_T from our measurements is to calculate the mean value:

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i$$

and set $x_T = \mu_x$.

Course CS &304H SJTU Statistical Learning Theory & Application

Maximum Likelihood Inference

- Suppose we are trying to measure the true value of some quantity (x_T).
  - We make repeated measurements of this quantity $\{x_1, x_2, \ldots, x_n\}$.
  - The standard way to estimate x_T from our measurements is to calculate the mean value:

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i$$

and set $x_T = \mu_x$.

DOES THIS PROCEDURE MAKE SENSE?

The maximum likelihood method (MLM) answers this question and provides a general method for estimating parameters of interest from data.

MH SJTU Statistical Learning Theory & Application

The Maximum Likelihood Method

Course CS 3304H SJTU Statistical Learning Theory & Applications

- Statement of the Maximum Likelihood Method
  - Assume we have made N measurements of x: $\{x_1, x_2, \ldots, x_n\}$
  - Assume we know the probability distribution function that describes x: $f(x, a)$.
  - Assume we want to determine the parameter α.
- MLM: pick α to maximize the probability of getting the measurements !

$$L = f(x_1,\alpha)dx \cdot f(x_2,\alpha)dx \cdots f(x_n,\alpha)dx$$

$$= f(x_1,\alpha) \cdot f(x_2,\alpha) \cdots f(x_n,\alpha)[dx^n]$$

$L(\alpha)$ is called $\boxed{\text{the Likelihood Function:}}$ $\quad L(\alpha) = \prod_{i=1}^{N} f(x_i,\alpha)$

Log Maximum Likelihood Method

- Maximizes $L(\alpha)$ by solving the following equation

$$\left.\frac{\partial L(\alpha)}{\partial \alpha}\right|_{\alpha=\alpha^*} = 0$$

  - Often easier to maximize $\ln L(\alpha)$
  - $L(\alpha)$ and $\ln L(\alpha)$ achieve maximum at the same location.

- $\ln L(\alpha)$ converts the product into a summation.

$$\ln L = \sum_{i=1}^{N} \ln f(x_i, \alpha)$$

Course CS 3304, SJTU Statistical Learning theory & Application

Log Maximum Likelihood Method

- The new maximization condition is:

$$\left.\frac{\partial \ln L}{\partial \alpha}\right|_{\alpha=\alpha^*} = \sum_{i=1}^{N} \frac{\partial}{\partial \alpha} \ln f(x_i,\alpha)\Big|_{\alpha=\alpha^*} = 0$$

- Resultant equations: simple linear equations or coupled non-linear equations.

- **Example:** Let $f(x,\alpha)$ be given by a Gaussian distribution function.
- Let $\alpha = \mu$ be the mean of the Gaussian.
- To find the best estimate of α from our set of *n* measurements $\{x_1, x_2, \ldots, x_n\}$.

SJTU Statistical Learning theory & Application

An Example: Gaussian

- Gaussian PDF

$$f(x_i,\alpha)=\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\sum_{i=1}^{n}\frac{(x_i-\alpha)^2}{2\sigma^2}\right)$$

- The likelihood function for this problem is:

$$L=\prod_{i=1}^{n}f(x_i,\alpha)=\prod_{i=1}^{n}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\sum_{i=1}^{n}\frac{(x_i-\alpha)^2}{2\sigma^2}\right)$$

$$=\left[\frac{1}{\sigma\sqrt{2\pi}}\right]^n e^{-\frac{(x_1-\alpha)^2}{2\sigma^2}}e^{-\frac{(x_2-\alpha)^2}{2\sigma^2}}\cdots e^{-\frac{(x_n-\alpha)^2}{2\sigma^2}}=\left[\frac{1}{\sigma\sqrt{2\pi}}\right]^n e^{-\sum_{i=1}^{n}\frac{(x_i-\alpha)^2}{2\sigma^2}}$$

Course CS &204H SJTU Statistical Learning theory & Application

An Example: Gaussian

$$\ln L = \ln \prod_{i=1}^{n} f(x_i, \alpha) = \ln\left([\frac{1}{\sigma\sqrt{2\pi}}]^n \exp\left(-\sum_{i=1}^{n} \frac{(x_i - \alpha)^2}{2\sigma^2} \right)\right)$$

$$= n\ln(\frac{1}{\sigma\sqrt{2\pi}}) - \sum_{i=1}^{n} \frac{(x_i - \alpha)^2}{2\sigma^2}$$

- We want to find the *a* that maximizes the log likelihood function:

$$\frac{\partial \ln L}{\partial \alpha} = \frac{\partial}{\partial \alpha}\left[n\ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right) - \sum_{i=1}^{n} \frac{(x_i - \alpha)^2}{2\sigma^2} \right] = 0$$

$$\frac{\partial}{\partial \alpha}\sum_{i=1}^{n}(x_i - \alpha)^2 = 0; \qquad \sum_{i=1}^{n} 2(x_i - \alpha)(-1) = 0 \quad \alpha = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Course CS & 041 SJTU Statistical Learning theory & Application

An Example: Poisson

- Let $f(x,\alpha)$ be given by a Poisson distribution.
- Let α be the mean of the Poisson.
- We want the best estimate of *a* from our set of *n* measurements $\{x_1, x_2, \ldots, x_n\}$
- Poisson PDF:

$$f(x,\alpha) = \frac{e^{-\alpha}\alpha^x}{x!}$$

Course CS &304H SJTU Statistical Learning theory & Application

An Example: Poisson

- The likelihood function for this problem is:

$$L(\alpha) = \prod_{i=1}^{n} f(x_i, \alpha) = \prod_{i=1}^{n} \frac{e^{-\alpha} \alpha^{x_i}}{x_i !} = \frac{e^{-n\alpha} \alpha^{\sum_{i=1}^{n} x_i}}{x_1 ! x_2 ! .. x_n !}$$

$$\frac{d \ln L}{d\alpha} = \frac{d}{d\alpha}\left(-n\alpha + \ln \alpha \cdot \sum_{i=1}^{n} x_i - \ln(x_1 ! x_2 ! .. x_n !) \right) = -n + \frac{1}{\alpha} \sum_{i=1}^{n} x_i = 0$$

$$\alpha = \frac{1}{n} \sum_{i=1}^{n} x_i$$

Average

Course CS &304H SJTU Statistical Learning theory & Application

General properties of MLM

- For large data samples (large *n*) the likelihood function, *L*, approaches a Gaussian distribution.
- Maximum likelihood estimates are usually *consistent*.
  - For large *n* the estimates converge to the true value of the parameters we wish to determine.
- Maximum likelihood estimates are usually *unbiased*.
  - For all sample sizes the parameter of interest is calculated correctly.

Course CS &304H SJTU Statistical Learning Theory & Application

General properties of MLM

- Maximum likelihood estimate is *efficient*: the estimate has the smallest variance.
- Maximum likelihood estimate is *sufficient*: it uses all the information in the observations (the x_i's).
- The solution from MLM is unique.
- Bad news: we must know the correct probability distribution for the problem at hand!

Course CS &304H SJTU Statistical Learning theory & Application

Contents

Course CS3304H SJTU Statistical Learning theory & Application

Maximum Likelihood

- We maximize the likelihood function $$L(\theta; \mathbf{Z}) = \prod_{i=1}^{N} g_\theta(z_i)$$
- Log-likelihood function $$\ell(\theta; \mathbf{Z}) = \sum_{i=1}^{N} \log g_\theta(z_i) = \sum_{i=1}^{N} \ell(\theta; z_i)$$
- Estimate θ using the **score function**

$$\dot{\ell}(\theta; \mathbf{Z}) = \sum_{i=1}^{N} \dot{\ell}(\theta; z_i), \qquad \text{where } \dot{\ell}(\theta; z_i) = \frac{\partial \ell(\theta; z_i)}{\partial \theta}$$

- Assume that L takes its maximum in the interior parameter space. Then

$$\dot{\ell}(\hat{\theta}; \mathbf{Z}) = 0$$

Course CS &304H SJTU Statistical Learning theory and application

Fisher Information

- Negative sum of second derivatives is the information matrix

$$\mathbf{I}(\theta) = -\sum_{i=1}^{N} \frac{\partial^2 \ell(\theta; z_i)}{\partial\theta\partial\theta^T}$$

is called the observed information, should be greater 0.

- Fisher information (expected information) is

$$\mathbf{i}(\theta) = E_\theta\left[\mathbf{I}(\theta)\right]$$

Assume that θ_0 is the true value of θ

Course CS &304H SJTU Statistical Learning theory & Application

Sampling Theory

- Basic result of sampling theory
- The sampling distribution of the max-likelihood estimator approaches the following normal distribution, as $N \to \infty$

$$\hat{\theta} \to N(\theta_0, \mathbf{i}(\theta_0)^{-1})$$

when we sample independently from $g_{\theta_0}(z)$

- This suggests to approximate the distribution with

$$N(\hat{\theta}, \mathbf{i}(\hat{\theta})^{-1})$$

Course CS &504H SJTU Statistical Learning theory & Application

Error Bound

- The corresponding error estimates are obtained from

$$\sqrt{\mathbf{i}(\hat{\theta})^{-1}_{jj}} \quad \text{and} \quad \sqrt{\mathbf{I}(\hat{\theta})^{-1}_{jj}}$$

- The confidence points have the form

$$\hat{\theta}_j - z^{(1-\alpha)} \cdot \sqrt{\mathbf{i}(\hat{\theta})^{-1}_{jj}}$$

$$\text{and} \quad \hat{\theta}_j - z^{(1-\alpha)} \cdot \sqrt{\mathbf{I}(\hat{\theta})^{-1}_{jj}}$$

$z^{(1-\alpha)}$ is the 1-α percentile of the normal distribution

Course CS &304H SJTU Statistical Learning theory & Application

Approximate form of the Fisher information

- The Fisher information equals

$$I(\theta) = -\mathrm{E}\left[\frac{\partial^2 \log f(X;\theta)}{\partial\theta^2}\right]$$

- The Cramér–Rao bound can then be written as

$$\operatorname{var}(\hat{\theta}) \geq \frac{1}{I(\theta)} = \frac{1}{-\mathrm{E}\left[\frac{\partial^2 \log f(X;\theta)}{\partial\theta^2}\right]}$$

Course CS &304H SJTU Statistical Learning theory & Application

Contents

- The Bootstrap and Maximum Likelihood Methods
  - Basic theory on MLE (Convergence, Lower bound)
  - Typical Example: regression model
- Bayesian Methods
- The EM Algorithm
- MCMC for Sampling from the Posterior
- Model Averaging and Stacking

Course CS &3041: SJTU Statistical Learning theory & Application

An Example

- Consider a linear expansion

$$y = \sum_{j=1}^{N} \beta_j h_j(x) + \varepsilon \qquad\qquad f(\varepsilon, \sigma^2) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{\varepsilon^2}{2\sigma^2}}$$

- The log-likelihood

$$l(\theta) = -\frac{N}{2} \log \sigma^2 2\pi - \boxed{\frac{1}{2\sigma^2} \sum_{i=1}^{N} \left(y_i - \beta^T h(x_i) \right)^2}$$

- Estimating equations:

$$\frac{\partial l(\theta)}{\partial \beta} = 0; \qquad \frac{\partial l(\theta)}{\partial \sigma^2} = 0.$$

Course CS &304H SJTU Statistical Learning Theory & Applications

An Example

- Consider a linear expansion

$$\mu(x)=\sum_{j=1}^{N}\beta_j h_j(x)$$

- The least square error solution

$$\hat{\beta}=(H^T H)^{-1}H^T y$$

- The Covariance of $\hat{\beta}$

$$cor(\hat{\beta})=(H^T H)^{-1}\hat{\sigma}^2; \qquad \hat{\sigma}^2=\sum(y_i-\hat{\mu}(x_i))^2/N$$

Course CS3204H SJTU Statistical Learning theory & Application

An Example

Consider prediction model $\hat{\mu}(x) = \sum_{j=1}^{N} \hat{\beta}_j h_j(x),$

The standard deviation

$$se[\hat{\mu}(x)] = [h(x)^T (H^T H)^{-1} h(x)]^{1/2} \hat{\sigma}$$

- The confidence region

$$\hat{\mu}(x) \pm 1.96 se[\hat{\mu}(x)]$$

Course CS &304H SJTU Statistical Learning theory & Application

Contents

- The Bootstrap and Maximum Likelihood Methods
- **Bayesian Methods**
  - How to update pdf after seeing the data and use the prior
- The EM Algorithm
- MCMC for Sampling from the Posterior
- Model Averaging and Stacking

Course CS &304H SJTU Statistical Learning theory & Application

Bayesian Methods

- Given a sampling model $\Pr(\mathbf{Z}|\theta)$ and a prior $\Pr(\theta)$ for the parameters, estimate the posterior probability

$$\Pr(\theta|\mathbf{Z}) = \frac{\Pr(\mathbf{Z}|\theta)\cdot\Pr(\theta)}{\int \Pr(\mathbf{Z}|\theta)\cdot\Pr(\theta)d\theta}$$

- By drawing samples or estimating its mean or parameters
- Differences to mere counting (frequentist approach)
  - Prior: allow for uncertainties present before seeing the data
  - Posterior: allow for uncertainties present after seeing the data

Course CS 2004H SJTU Statistical_Learning Theory & Application

Bayesian Methods

- The posterior distribution affords also a predictive distribution of seeing future values z^{new}

$$\Pr\left(z^{\text{new}} \middle| \mathbf{Z}\right) = \int \Pr\left(z^{\text{new}} \middle| \theta\right) \cdot \Pr\left(\theta \middle| \mathbf{Z}\right) d\theta$$

- In contrast, the max-likelihood approach would predict future data on the basis of not accounting for the uncertainty in the parameters $\Pr\left(z^{\text{new}} \middle| \hat{\theta}\right)$

Course CS &304H SJTU Statistical Learning theory & Application

An Example

- Consider a linear expansion

$$y = \mu_\beta(x) + \varepsilon = \sum_{j=1}^{N} \beta_j h_j(x) + \varepsilon; \qquad p(\varepsilon) = \frac{1}{(2\pi)^{1/2}\sigma_\varepsilon} \exp(-\tfrac{\varepsilon^2}{2\sigma^2})$$

- The prior probability $\beta \;\propto\; N(0, \tau\,\Sigma)$

- The maximum a posterior

$$p(\beta | y, x) \approx p(\beta)\, p(y \,|\, x, \beta) = N(0, \tau\,\Sigma) N_y(\mu_\beta(x) \,|\, \sigma_\varepsilon^2)$$

Course CS3204H SJTU Statistical Learning theory & Application

MAP (Maximum A Posterior)

- The posterior distribution for β is also Gaussian, with mean and covariance

$$\mathrm{E}(\beta \mid \mathbf{Z}) = \left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\Sigma^{-1}\right)^{-1}\mathbf{H}^T\mathrm{y}, \qquad \mathrm{cov}(\beta \mid \mathbf{Z}) = \left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\Sigma^{-1}\right)^{-1}\sigma^2.$$

- The corresponding posterior values for $\mu(x) = \sum_{j=1}^{N}\beta_j h_j(x)$

$$\mathrm{E}(\mu(x) \mid \mathbf{Z}) = h(x)^T\left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\Sigma^{-1}\right)^{-1}\mathbf{H}^T\mathrm{Y},$$

$$\mathrm{cov}\left[\mu(x), \mu(x') \mid \mathbf{Z}\right] = h(x)^T\left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\Sigma^{-1}\right)^{-1}h(x')\sigma^2.$$

SJTU Statistical Learning theory & Application

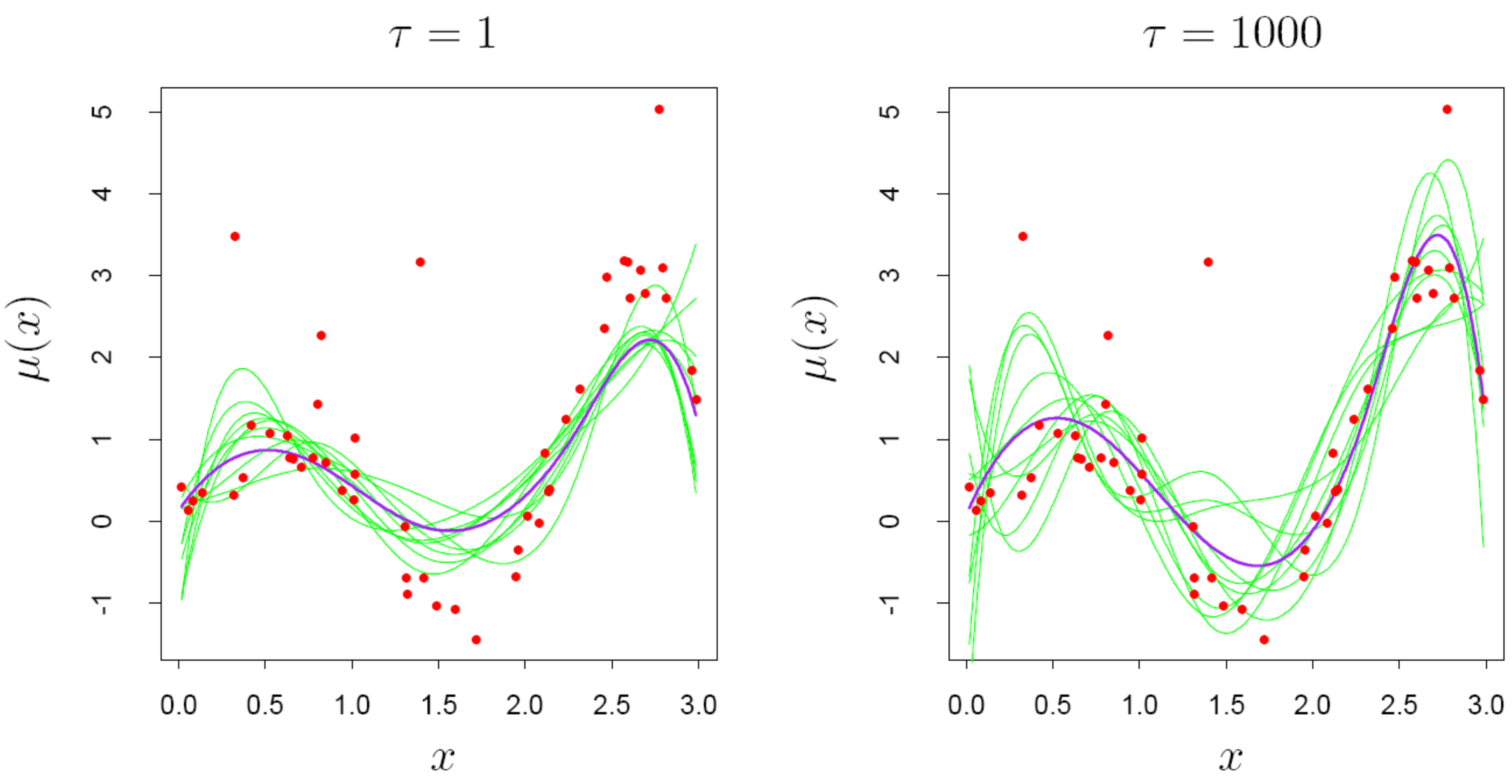


FIGURE 8.4. *Smoothing example: Ten draws from the posterior distribution for the function $\mu(x)$, for two different values of the prior variance τ. The purple curves are the posterior means.*

Bootstrap vs Bayesian

Course CS &304H SJTU Statistical Learning theory & Application

- The bootstrap mean is an approximate posterior average
- Simple example:
  - Single observation z drawn from a normal distribution $z \propto N(\theta, 1)$
  - Assume a normal prior for θ : $\theta \propto N(0, \tau)$
  - Resulting posterior distribution

$$\theta | z \propto N\left(\frac{z}{1+1/\tau}, \frac{1}{1+1/\tau}\right)$$

Bootstrap vs Bayesian

- Three ingredients make this work
  - The choice of a noninformative prior for θ
  - The dependence of $\ell(\theta; \mathbf{Z})$ on Z only through the max-likelihood estimate $\hat{\theta}$
  - The symmetry of

$$\ell(\theta; \mathbf{Z}) = \ell(\theta; \hat{\theta})$$

$$\ell(\theta; \hat{\theta}) = \ell(\hat{\theta}; \theta) + \text{constant}.$$

Course CS &3044: SJTU Statistical Learning theory & Application

Bootstrap vs Bayesian

- The bootstrap distribution represents an (approximate) nonparametric, noninformative posterior distribution for our parameter.
- But this bootstrap distribution is obtained painlessly without having to formally specify a prior and without having to sample from the posterior distribution.
- Hence we might think of the bootstrap distribution as a “poor man's” Bayes posterior. By perturbing the data, the bootstrap approximates the Bayesian effect of perturbing the parameters, and is typically much simpler to carry out.

Course CS 3304H SJTU Statistical Learning theory & Application

Contents

Course CS &304H SJTU Statistical Learning theory & Application

The EM Algorithm

- Gaussian Mixture Model

$$f(x) = \sum_{m=1}^{M} \alpha_m \phi(x; \mu_m, \Sigma_m)$$

- EM algorithm for 2 Gaussian mixtures
  - Given $x_1, x_2, \ldots, x_n,$ log-likelihood:

$$l(y, \theta) = \sum_{i=1}^{N} \log\left[\alpha \phi_{\theta_1}(x_i) + (1-\alpha)\phi_{\theta_2}(x_i)\right]$$

Bad formulation

  - Suppose we observe Latent Binary

$$L(x, z, \theta) = \sum_{\substack{i=1 \\ z_i=1}}^{N} \log\left[\alpha \varphi_{\theta_1}(x_i)\right] + \sum_{\substack{i=1 \\ z_i=0}}^{N} \log\left[(1-\alpha)\varphi_{\theta_2}(x_i)\right]$$

Good formulation

z such that $z = 1 \Rightarrow x \sim \varphi_{\theta_1}, \quad z = 0 \Rightarrow x \sim \varphi_{\theta_2}$

Course CS3304H SJTU Statistical Learning theory & Application

The EM Algorithm

- The EM algorithm for two-component Gaussian mixtures
  - Take initial guesses $\hat{\pi}, \hat{\mu}_1, \hat{\theta}_1, \hat{\mu}_2, \hat{\theta}_2$ for the parameters
  - Expectation Step: Compute the responsibilities

$$\hat{\gamma}_i = \frac{\hat{\pi}\phi_{\hat{\theta}_2}(y_i)}{(1-\hat{\pi})\phi_{\hat{\theta}_1}(y_i) + \hat{\pi}\phi_{\hat{\theta}_2}(y_i)}, \quad i = 1, \ldots, N$$

Course CS &304H SJTU Statistical Learning theory & Application

The EM Algorithm

– Maximization Step: Compute the weighted means and variances

$$\hat{\mu}_1 = \frac{\sum_{i=1}^{N}(1-\hat{\gamma}_i)y_i}{\sum_{i=1}^{N}(1-\hat{\gamma}_i)}, \qquad \hat{\sigma}_1^2 = \frac{\sum_{i=1}^{N}(1-\hat{\gamma}_i)(y_i-\hat{\mu}_1)^2}{\sum_{i=1}^{N}(1-\hat{\gamma}_i)},$$

$$\hat{\mu}_2 = \frac{\sum_{i=1}^{N}\hat{\gamma}_i y_i}{\sum_{i=1}^{N}\hat{\gamma}_i}, \qquad \hat{\sigma}_2^2 = \frac{\sum_{i=1}^{N}\hat{\gamma}_i(y_i-\hat{\mu}_1)^2}{\sum_{i=1}^{N}\hat{\gamma}_i},$$

$$\hat{\pi} = \sum_{i=1}^{N}\hat{\gamma}_i \,/\, N$$

– Iterate 2 and 3 until convergence

Course CS &304H SJTU Statistical Learning theory & Application

The EM Algorithm in General

- $\mathbf{Z}$ input data, with log-likelihood $\ell(\theta, \mathbf{Z})$
- $\mathbf{Z}^m$ latent data (in our example Δ_i)
- $\mathbf{T} = (\mathbf{Z}, \mathbf{Z}^m)$ complete data with log-likelihood $\ell_0(\theta, \mathbf{T})$

$$\Pr\left(\mathbf{Z}^m \middle| \mathbf{Z}, \theta'\right) = \frac{\Pr\left(\mathbf{Z}^m, \mathbf{Z} \middle| \theta'\right)}{\Pr\left(\mathbf{Z} \middle| \theta'\right)}; \qquad \Pr\left(\mathbf{Z} \middle| \theta'\right) = \frac{\Pr\left(\mathbf{T} \middle| \theta'\right)}{\Pr\left(\mathbf{Z}^m \middle| \mathbf{Z}, \theta'\right)}$$

we have $\ell(\theta'; \mathbf{Z}) = \ell_0(\theta'; \mathbf{T}) - \ell_1(\theta'; \mathbf{Z}^m | \mathbf{Z})$

Course CS &304H SJTU Statistical Learning theory & Application

The EM Algorithm in General

$$\text{we have } \ell(\theta'; \mathbf{Z}) = \ell_0(\theta'; \mathbf{T}) - \ell_1(\theta'; \mathbf{Z}^m \,|\mathbf{Z})$$

- Taking conditional expectations with respect to the distribution of T|Z governed by parameter θ gives

$$\begin{aligned}
\ell(\theta'; \mathbf{Z}) &= \boxed{\int \ell_0(\theta'; \mathbf{T}) \Pr(\mathbf{Z}^m \,|\mathbf{Z}, \theta) d\mathbf{Z}^m} \\
&\quad - \boxed{\int \ell_1(\theta'; \mathbf{Z}^m) \Pr(\mathbf{Z}^m \,|\mathbf{Z}, \theta) d\mathbf{Z}^m} \\
&= \boxed{\mathrm{E}\left(\ell_0(\theta'; \mathbf{T}) \,\middle|\mathbf{Z}, \theta\right)} - \boxed{E\left(\ell_1(\theta'; \mathbf{Z}^m) \,\middle|\mathbf{Z}, \theta\right)} \\
&= Q(\theta'; \theta) - R(\theta'; \theta)
\end{aligned}$$

Course CS-MH SJTU Statistical Learning Theory & Application

The EM Algorithm in General

- Taking conditional expectations with respect to the distribution of T|Z governed by parameter θ gives

$$
\begin{aligned}
R(\theta'; \theta) &= E\left(\ell_1(\theta'; \mathbf{Z}^m \,|\mathbf{Z}) \,\middle|\mathbf{Z}, \theta\right) \\
&= \int \log\left(p(\mathbf{Z}^m \,|\mathbf{Z}, \theta')\right) p(\mathbf{Z}^m \,|\mathbf{Z}, \theta)\, d\mathbf{Z}^m \\
&\leq \int \log\left(p(\mathbf{Z}^m \,|\mathbf{Z}, \theta)\right) p(\mathbf{Z}^m \,|\mathbf{Z}, \theta)\, d\mathbf{Z}^m \\
&= R(\theta; \theta)
\end{aligned}
$$

Course CS 8304H SJTU Statistical Learning theory & Application

Proof of $R(\theta'; \theta) \le R(\theta; \theta)$

If $\phi(x)$ is convex, then $\mathrm{E}[\phi(x)] \ge \phi(E[x])$

$$R(\theta'; \theta) - R(\theta; \theta) =$$

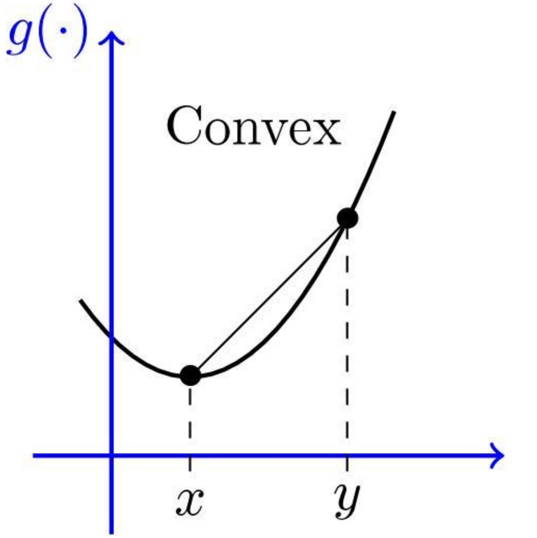
$$= \int [\log(p(\mathbf{Z}^m | \mathbf{Z}, \theta')) - \log(p(\mathbf{Z}^m | \mathbf{Z}, \theta))] p(\mathbf{Z}^m | \mathbf{Z}, \theta)\, d\mathbf{Z}^m$$

$$\le \int \log\left(\frac{p(\mathbf{Z}^m | \mathbf{Z}, \theta')}{p(\mathbf{Z}^m | \mathbf{Z}, \theta)}\right) p(\mathbf{Z}^m | \mathbf{Z}, \theta)\, d\mathbf{Z}^m$$

$$= \mathrm{E}_{p^m}\left[\log\left(\frac{p(\mathbf{Z}^m | \mathbf{Z}, \theta')}{p(\mathbf{Z}^m | \mathbf{Z}, \theta)}\right)\right] \le \log\left[\mathrm{E}_{p^m}\left(\frac{p(\mathbf{Z}^m | \mathbf{Z}, \theta')}{p(\mathbf{Z}^m | \mathbf{Z}, \theta)}\right)\right]$$

$$= \log \int \left(\frac{p(\mathbf{Z}^m | \mathbf{Z}, \theta')}{p(\mathbf{Z}^m | \mathbf{Z}, \theta)}\right) p(\mathbf{Z}^m | \mathbf{Z}, \theta)\, d\mathbf{Z}^m$$

$$= \log \int p(\mathbf{Z}^m | \mathbf{Z}, \theta')\, d\mathbf{Z}^m = \log(1) = 0$$



Course CS3304H SJTU Statistical Learning Theory & Application

The EM Algorithm in General

In the M step, the EM algorithm maximizes $Q(\theta', \theta)$ over θ′, rather than the actual objective function .

$$\ell(\theta'; \mathbf{Z}) = \mathrm{E}\left(\ell_0(\theta'; \mathbf{T}) \middle| Z, \theta\right) - E\left(\ell_1(\theta'; \mathbf{Z}^m | \mathbf{Z}) \middle| Z, \theta\right)$$

$$= Q(\theta'; \theta) - R(\theta'; \theta)$$

$$\ell(\theta'; \mathbf{Z}) - \ell(\theta; \mathbf{Z}) = Q(\theta'; \theta) - Q(\theta; \theta) - \left(R(\theta'; \theta) - R(\theta; \theta)\right) \geq 0$$

Note that $R(\theta*, \theta)$ is the expectation of a log-likelihood of a density (indexed by $\theta*$), with respect to the same density indexed by θ, and hence (by Jensen's inequality) is maximized as a function of $\theta*$, when $\theta* = \theta$

Course CS &304H SJTU Statistical Learning theory & Application

The EM Algorithm in General

1. Start with initial params $\hat{\theta}$
2. **Expectation Step**: at the j-th step compute

$$Q(\theta', \hat{\theta}^{(j)}) = \mathrm{E}(\ell_0(\theta', \mathbf{T}) \mid \mathbf{Z}, \hat{\theta}^{(j)})$$

   as a function of the dummy argument θ'

3. **Maximization Step**: Determine the new params $\hat{\theta}^{(j+1)}$ by maximizing

$$Q(\theta', \hat{\theta}^{(j)})$$

- Iterate 2 and 3 until convergence

Course CS&304H SJTU Statistical Learning theory & Application

Contents

- The Bootstrap and Maximum Likelihood Methods
- Bayesian Methods
- Relationship Between the Bootstrap and Bayesian Inference
- **The EM Algorithm**
  - **E-step is expensive in general**
  - **Variational approach; approximation by sampling**
- MCMC for Sampling from the Posterior
- Model Averaging and Stacking

Course CS &304H SJTU Statistical Learning theory & Application

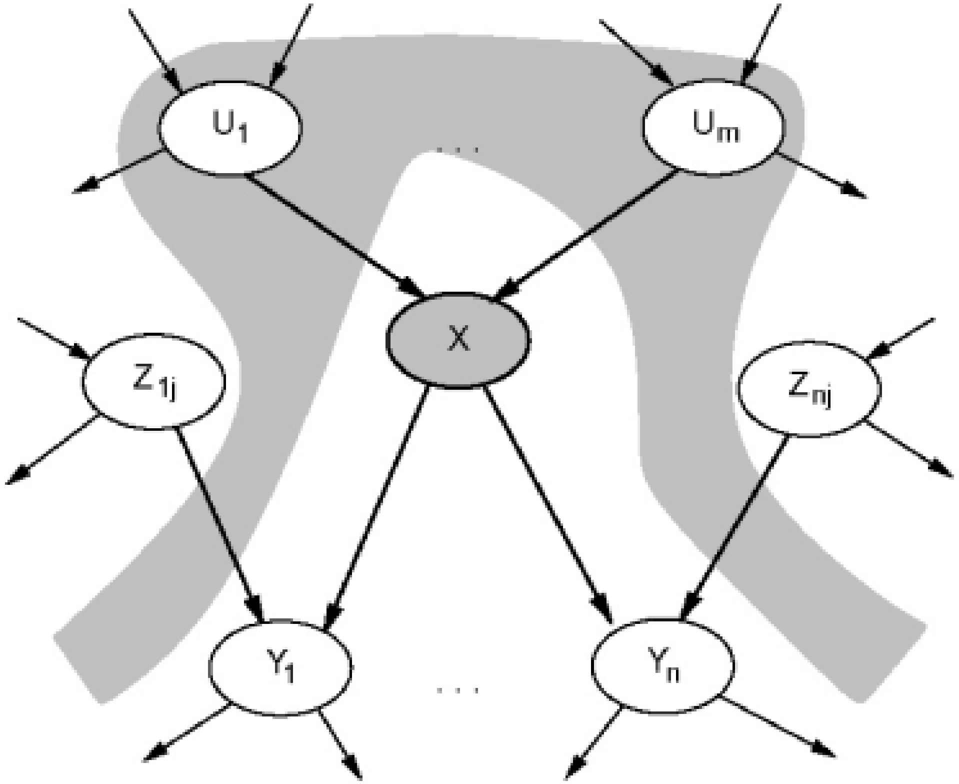
Algorithm 8.3 *Gibbs Sampler.*

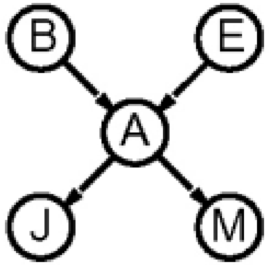
1. Take some initial values $U_k^{(0)}, k = 1, 2, \ldots, K$.
2. Repeat for $t = 1, 2, \ldots, .$:

   For $k = 1, 2, \ldots, K$ generate $U_k^{(t)}$ from
   $\Pr(U_k^{(t)} | U_1^{(t)}, \ldots, U_{k-1}^{(t)}, U_{k+1}^{(t-1)}, \ldots, U_K^{(t-1)})$.
3. Continue step 2 until the joint distribution of $(U_1^{(t)}, U_2^{(t)}, \ldots, U_K^{(t)})$ *does not change.*

Local semantics of Bayesian network

Local semantics: each node is conditionally independent of its nondescendants given its parents

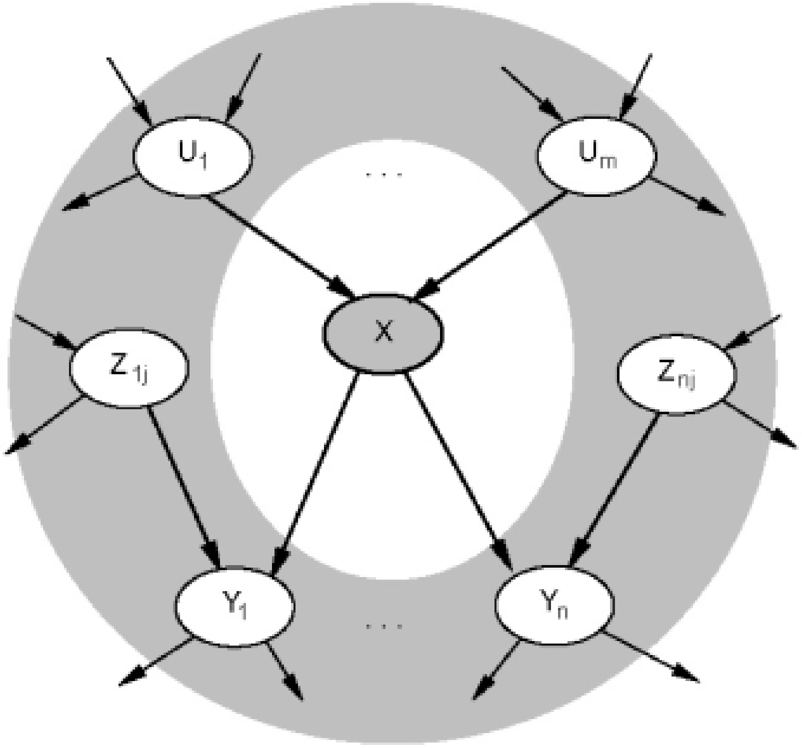


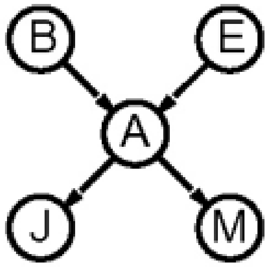


e.g., *JohnCalls* is independent of *Burglary* and *Earthquake*, given the value of *Alarm*.

Markov Blanket

Each node is conditionally independent of all others given its Markov blanket: parents + children + children's parents





e.g., *Burglary* is independent of *JohnCalls* and *MaryCalls* , given the value of *Alarm* and *Earthquake*.

Approximate inference using MCMC*

"State" of network = current assignment to all variables.

Generate next state by sampling one variable given Markov blanket
Sample each variable in turn, keeping evidence fixed

function MCMC-ASK(X, $\mathbf{e}$, bn, N) **returns** an estimate of $P(X|\mathbf{e})$
  local variables: $\mathbf{N}[X]$, a vector of counts over X, initially zero
    $\mathbf{Z}$, the nonevidence variables in bn
    $\mathbf{x}$, the current state of the network, initially copied from $\mathbf{e}$

  initialize $\mathbf{x}$ with random values for the variables in $\mathbf{Y}$
  for $j = 1$ to N **do**
    $\mathbf{N}[x] \leftarrow \mathbf{N}[x] + 1$ where x is the value of X in $\mathbf{x}$
    for each Z_i in $\mathbf{Z}$ **do**
      sample the value of Z_i in $\mathbf{x}$ from $\mathbf{P}(Z_i|MB(Z_i))$ given the values of $MB(Z_i)$ in $\mathbf{x}$
  return NORMALIZE($\mathbf{N}[X]$)

Can also choose a variable to sample at random each time

Contents

- The Bootstrap and Maximum Likelihood Methods
- Bayesian Methods
- Relationship Between the Bootstrap and Bayesian Inference
- The EM Algorithm
- MCMC for Sampling from the Posterior
- Model Averaging and Stacking

Course CS &304H SJTU Statistical Learning theory & Application

Model Averaging and Stacking

- Given predictions $\hat{f}_1(x),\ \hat{f}_2(x), \dots,\ \hat{f}_M(x)$
- Under square-error loss, seek weights

$$\hat{w} = (\hat{w}_1, \hat{w}_2, \dots, \hat{w}_M)$$

- Such that

$$\hat{w} = \arg\min_{w} \mathrm{E}_{\mathcal{P}} \left[Y - \sum_{m=1}^{M} w_m \hat{f}_m(x) \right]^2$$

- Here the input x is fixed and the N observations in Z are distributed according to P

Course CS &304H SJTU Statistical Learning theory & Application

Model Averaging and Stacking

- The solution is the population linear regression of Y on $\hat{F}(x)^T$, namely

$$\hat{F}(x)^T = \left[\hat{f}_1(x),\ \hat{f}_2(x), \ldots,\ \hat{f}_M(x) \right]$$

- Now the full regression has smaller error, namely

$$\hat{w} = \mathrm{E}_{\mathcal{P}} \left[\hat{F}(x)\hat{F}(x)^T \right]^{-1} \mathrm{E}_{\mathcal{P}} \left[\hat{F}(x)Y \right]$$

- Population linear regression is not available, thus replace it by the linear regression over the training set

$$\mathrm{E}_{\mathcal{P}} \left[Y - \sum_{m=1}^{M} \hat{w}_m \hat{f}_m(x) \right]^2 \leq \mathrm{E}_{\mathcal{P}} \left[Y - \hat{f}_m(x) \right]^2 \quad \forall m$$

Course CS3304H SJTU Statistical Learning theory & Application

Contents

- The Bootstrap and Maximum Likelihood Methods
- Bayesian Methods
- Relationship Between the Bootstrap and Bayesian Inference
- The EM Algorithm
- MCMC for Sampling from the Posterior
- Model Averaging and Stacking

Course CS 3304H SJTU Statistical Learning Theory & Application

THE END OF TALK