Chapter 8: Model Inference and Averaging

The Elements of Statistical Learning

Biaobin Jiang

Department of Biological Sciences
Purdue University

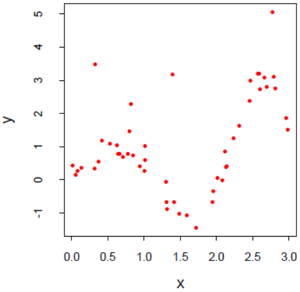
bjiang@purdue.edu

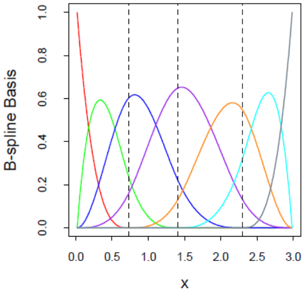
July 28, 2014

Overview

A Smoothing Example

Input: the training data $(x_i, y_i), i = 1, 2, \ldots, N$.
Output: the estimated coefficients β_j.
Method: Fitting B-Spline basis functions $h_j(x)$ using least squares.





Step 1: *expand* $\mathbf{x} \in \mathbb{R}^{N\times 1}$ into $\mathbf{H} \in \mathbb{R}^{N\times 7}$ using $\mathcal{R}$ package spline.

$$H \leftarrow bs(x, knots = quantile(x, p = c(1/4, 2/4, 3/4)))$$

Fitting by Least Squares

Step 2: Fitting a 7-dimensional linear combination model (Eq. (8.1)).

$$\hat{\mathbf{y}} = \mathrm{E}(Y|X = x) = \mu(\mathbf{x}) = \sum_{i=1}^{7} \beta_j h_j(x) = \mathbf{Hb}$$

Minimize the squared error:

$$\begin{aligned}\min_{\mathbf{b}} \frac{1}{2}\|\mathbf{y} - \hat{\mathbf{y}}\|^2 &= \frac{1}{2}(\mathbf{y} - \mathbf{Hb})^T(\mathbf{y} - \mathbf{Hb}) \\ &= \frac{1}{2}(\mathbf{b}^T\mathbf{H}^T\mathbf{Hb} - 2\mathbf{y}^T\mathbf{Hb} + \mathbf{y}^T\mathbf{y})\end{aligned}$$

Take the derivative and set to zero:

$$\mathbf{H}^T\mathbf{Hb} - \mathbf{H}^T\mathbf{y} = 0$$

And we finally obtain Equation (8.2):

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

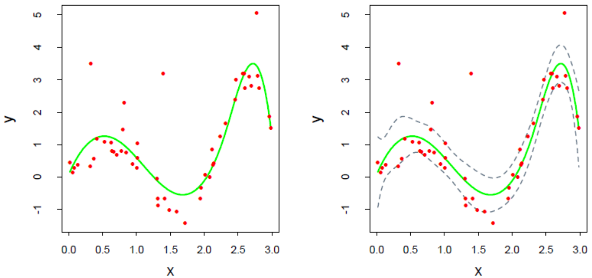
Uncertainty of Least Squares Fitting

Step 3: Consider β is a random variable, and we can investigate its uncertainty. The estimated covariance matrix of $\hat{\beta}$ is given by Eq. (8.3)

$$\widehat{\mathrm{Var}}(\hat{\beta}) = (\mathbf{H}^T\mathbf{H})^{-1}\hat{\sigma}^2$$

where we have estimated $\hat{\sigma}^2 = \sum_{i=1}^{N}(y_i - \hat{y}_i)^2/N$.
And the standard error of a prediction $\hat{y} = h(x^{\text{new}})^T\hat{\beta}$ is Eq. (8.4)

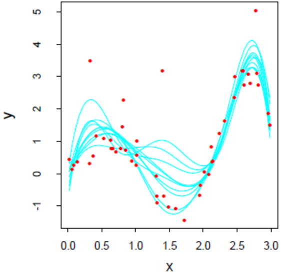
$$\widehat{\mathrm{se}}[\hat{y}] = \sqrt{h(x^{\text{new}})^T(\mathbf{H}^T\mathbf{H})^{-1}h(x^{\text{new}})} \cdot \hat{\sigma}$$

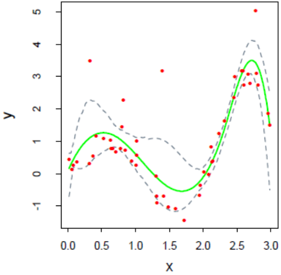


What is Bootstrap?

The bootstrap is a useful tool for constructing **confidence intervals** and calculating **standard errors** for difficult statistics (e.g., median).

In practice, the bootstrap principle is always carried out using simulation.





How to Bootstrap?

For example, to estimate a median from a data set of n observations:

1. Sample n observations **with replacement** from the observed data resulting in one simulated complete data set;
2. Take the median of the simulated data set;
3. Repeat these two steps B times, resulting in B simulated medians;
4. Then we can:
    - Draw a histogram of them;
    - Calculate standard deviation;
    - Estimate confidence intervals.

This bootstrap method is called the *nonparametric bootstrap*.

Likelihood Function

- A *probability* function is a function of random variables $\mathbf{y}$.
- A *likelihood* function is a function of parameters β.

We begin by specifying a probability density function for our observation

$$z_i \sim g_\theta(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(z-\mu)^2/\sigma^2}$$

where $\theta = (\mu, \sigma^2)$.
Then the likelihood function is given by

$$L(\theta; \mathbf{Z}) = \prod_{i=1}^{N} g_\theta(z_i)$$

And the logarithm of the likelihood function is

$$\ell(\theta; \mathbf{Z}) = \sum_{i=1}^{N} \log g_\theta(z_i)$$

MLE for the Smoothing Example

The log-likelihood function is Eq. (8.20)

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

The maximum likelihood estimate is obtained by setting $\partial\ell/\partial\beta = 0$ and $\partial\ell/\partial\sigma^2 = 0$, giving Eq. (8.21)

$$\hat{\beta} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y}$$
$$\hat{\sigma}^2 = \frac{1}{N}\sum(y_i - \hat{\mu}(x_i))^2$$

where $\hat{\mu}(x_i) = \hat{y}_i = h(x_i)^T\hat{\beta}$, which are the same as the LS solutions in Eq. (8.2) and (8.3).

Conclusion: MLE is identical to the Least Squares solution.

Bootstrap vs. Maximum Likelihood

In essence the bootstrap is a computer implementation of maximum likelihood. The advantage of the bootstrap over the maximum likelihood formula is that it allows us to compute maximum likelihood estimates of standard errors and other quantities in settings where no formulas are available.

Bayesian Inference

Specifying a Bayesian Prior

To control the model complexity, instead of regularization, we now define a (Gaussian) *prior* distribution Eq. (8.25)

$$\beta \sim \mathcal{N}(0, \tau\Sigma)$$

where τ is variance and Σ is correlation matrix.

Then we can compute the *posterior* distribution over β via Bayes' rule:

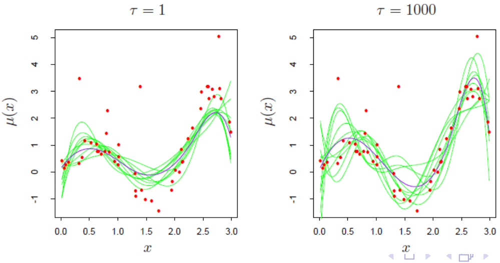
$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{normalizing factor}}$$

Posterior Inference

As a consequence of combining a Gaussian prior and a linear model within a Gaussian likelihood, the posterior is also conveniently Gaussian, with mean and covariance Eq. (8.27)

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

$$\mathrm{cov}(\beta|\mathbf{Z}) = \left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\Sigma^{-1}\right)^{-1}\sigma^2$$



MAP Estimation

Here, we call the single most probable value under the above posterior distribution *Maximum A Posteriori* estimate

$$\beta_{\text{MAP}} = \left(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\Sigma^{-1}\right)^{-1}\mathbf{H}^T\mathbf{y}$$

which is identical to the Least Squares solution with $L2$-norm regularization (Ridge Regression).

The General Bayesian Predictive Framework

The *true Bayesian* way is to integrate out, or marginalize over, the uncertain variables θ (all parameters) in order to obtain the *predictive distribution* Eq. (8.24)

$$\mathbb{P}(z^{\text{new}}|\mathbf{Z}) = \int \mathbb{P}(z^{\text{new}}|\theta) \cdot \mathbb{P}(\theta|\mathbf{Z}) d\theta$$

And this is nearly always analytically intractable to compute!

Summary: Before picking a method, MLE or Bayesian, think about what you need: a value, or a distribution.

EM Algorithm

What is EM?

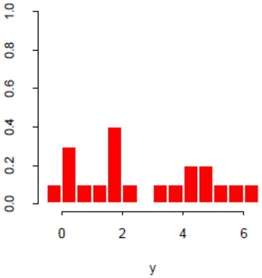
- Expectation Maximization (EM) algorithm is a popular tool for simplifying difficult maximum likelihood problems.
- It is commonly well known as the Baum-Welch algorithm in the context of Hidden Markov Model (HMM).
- In terms of machine learning, its variant called K-mean clustering is widely used as an unsupervised learning.
- It only returns a local maxima.

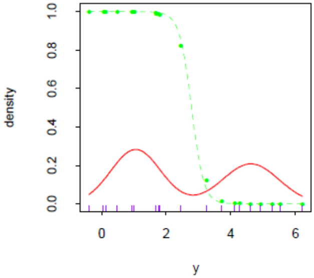
Two-Component Mixture Model: An Example

We model Y as a mixture of two Gaussian distributions (Eq. (8.36)):

$$\begin{aligned} Y_1 &= \mathcal{N}(\mu_1, \sigma_1^2) \\ Y_2 &= \mathcal{N}(\mu_2, \sigma_2^2) \\ Y &= (1 - \triangle) \cdot Y_1 + \triangle \cdot Y_2 \end{aligned}$$

where $\triangle \in \{0, 1\}$ with $\mathbb{P}(\triangle = 1) = \pi$.





How EM works?

The log-likelihood based on the N training cases is Eq. (8.39)

$$\ell(\theta; \mathbf{Z}) = \sum_{i=1}^{N} \log\left[(1-\pi)\phi_{\theta_1}(y_i) + \pi\phi_{\theta_2}(y_i)\right]$$

Since directly maximizing $\ell(\theta; \mathbf{Z})$ is quite difficult numerically, we suppose the values of the unobserved latent variables $\triangle_i$ and transform the log-likelihood function equivalently as Eq. (8.40)

$$\begin{aligned}\ell_0(\theta; \mathbf{Z}, \triangle) = & \sum_{i=1}^{N}\left[(1-\triangle_i)\log\phi_{\theta_1}(y_i) + \triangle_i \log\phi_{\theta_2}(y_i)\right] \\ & + \sum_{i=1}^{N}\left[(1-\triangle_i)\log(1-\pi) + \triangle_i \log\pi\right]\end{aligned}$$

Introducing a Responsibility term

Recall Eq. (8.40)

$$\ell_0(\theta; \mathbf{Z}, \triangle) = \sum_{i=1}^{N} [(1 - \triangle_i) \log \phi_{\theta_1}(y_i) + \triangle_i \log \phi_{\theta_2}(y_i)] + \sum_{i=1}^{N} [(1 - \triangle_i) \log(1 - \pi) + \triangle_i \log \pi]$$

Since the values of the $\triangle_i$ are actually *unknown*, we proceed in an iterative fashion, substituting for each $\triangle_i$ in Eq. (8.40) its *expected value* in Eq. (8.41)

$$\gamma_i(\theta) = \mathrm{E}(\triangle_i | \theta, \mathbf{Z}) = \mathbb{P}(\triangle_i = 1 | \theta, \mathbf{Z})$$

which is also called the *responsibility* of model 2 for observation i.

Algorithm Details

Algorithm 8.1 *EM Algorithm for Two-component Gaussian Mixture.*

1. Take initial guesses for the parameters $\hat{\mu}_1, \hat{\sigma}_1^2, \hat{\mu}_2, \hat{\sigma}_2^2, \hat{\pi}$ (see text).

2. *Expectation Step*: compute the responsibilities

$$\hat{\gamma}_i = \frac{\hat{\pi}\phi_{\hat{\theta}_2}(y_i)}{(1-\hat{\pi})\phi_{\hat{\theta}_1}(y_i) + \hat{\pi}\phi_{\hat{\theta}_2}(y_i)}, \quad i = 1, 2, \ldots, N. \tag{8.42}$$

3. *Maximization Step*: compute the weighted means and variances:

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

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

   and the mixing probability $\hat{\pi} = \sum_{i=1}^N \hat{\gamma}_i / N$.

4. Iterate steps 2 and 3 until convergence.

EM in General

Algorithm 8.2 *The EM Algorithm.*

1. Start with initial guesses for the parameters $\hat{\theta}^{(0)}$.
2. *Expectation Step*: at the jth step, compute
$$Q(\theta', \hat{\theta}^{(j)}) = \mathrm{E}(\ell_0(\theta'; \mathbf{T})|\mathbf{Z}, \hat{\theta}^{(j)}) \tag{8.43}$$
as a function of the dummy argument θ'.
3. *Maximization Step*: determine the new estimate $\hat{\theta}^{(j+1)}$ as the maximizer of $Q(\theta', \hat{\theta}^{(j)})$ over θ'.
4. Iterate steps 2 and 3 until convergence.

MCMC

What is MCMC?

The *Markov Chain Monte Carlo* (MCMC) approach is primarily used for posterior sampling given a Bayesian model.

We will see that *Gibbs Sampling*, an MCMC procedure, is closely related to the EM algorithm: the main difference is that it samples from the conditional distributions rather than maximizing over them.

Gibbs Sampling

- Gibbs Sampling uses conditional sampling of each parameter given the rest.
- After the procedure reaches stationarity, the marginal density of any subset of the variables can be approximated by a density estimate applied to the sample values.
- More formally, Gibbs sampling produces a Markov Chain whose stationary distribution is the true joint distribution of all variables.

Algorithm Details

Algorithm 8.4 *Gibbs sampling for mixtures.*

1. Take some initial values $\theta^{(0)} = (\mu_1^{(0)}, \mu_2^{(0)})$.
2. Repeat for $t = 1, 2, \ldots,$.
   (a) For $i = 1, 2, \ldots, N$ generate $\Delta_i^{(t)} \in \{0, 1\}$ with $\Pr(\Delta_i^{(t)} = 1) = \hat{\gamma}_i(\theta^{(t)})$, from equation (8.42).
   (b) Set

$$\begin{aligned} \hat{\mu}_1 &= \frac{\sum_{i=1}^N (1 - \Delta_i^{(t)}) \cdot y_i}{\sum_{i=1}^N (1 - \Delta_i^{(t)})}, \\ \hat{\mu}_2 &= \frac{\sum_{i=1}^N \Delta_i^{(t)} \cdot y_i}{\sum_{i=1}^N \Delta_i^{(t)}}, \end{aligned}$$

   and generate $\mu_1^{(t)} \sim N(\hat{\mu}_1, \hat{\sigma}_1^2)$ and $\mu_2^{(t)} \sim N(\hat{\mu}_2, \hat{\sigma}_2^2)$.
3. Continue step 2 until the joint distribution of $(\boldsymbol{\Delta}^{(t)}, \mu_1^{(t)}, \mu_2^{(t)})$ doesn't change

I will skip Section 8.7 Bagging, 8.8 Averaging and 8.9 Bumping, and go to the Exercises . . .

Ex. 8.6

Consider the bone mineral density data of Figure 5.6.

(a) Fit a cubic smooth spline to the relative change in spinal BMD, as a function of age. Use cross-validation to estimate the optimal amount of smoothing. Construct pointwise 90% confidence bands for the underlying function.

(b) Compute the posterior mean and covariance for the true function via (8.28), and compare the posterior bands to those obtained in (a).

(c) Compute 100 bootstrap replicates of the fitted curves, as in the bottom left panel of Figure 8.2. Compare the results to those obtained in (a) and (b).

References

- T. Hastie, R. Tibshirani and J. Friedman. *The Elements of Statistical Learning*.
- B. Caffo. *Mathematical Biostatistics Boot Camp* on Coursera.
- M. Tipping. *Bayesian Inference: An Introduction to Principles and Practice in Machine Learning*.

The End