Please submit your homework with codes (hard copy) in class and upload the corresponding codes to the Blackboard. Problems marked with * will be graded in detail and they are worth 50% of the total score. Remaining problems, worth the remaining 50% of the total score, will be given full mark if reasonable amount of work is shown.

For this homework, use R for programming parts unless otherwise specified.

1 EM Algorithms

1. * Consider the multinomial distribution with four outcomes, that is, the multinomial with pdf

$$p(x_1, x_2, x_3, x_4) = \frac{n!}{x_1!x_2!x_3!x_4!} p_1^{x_1} p_2^{x_2} p_3^{x_3} p_4^{x_4}, \quad \sum_{i=1}^{4} x_i = n, \quad \sum_{i=1}^{4} p_i = 1.$$

Suppose the probabilities are related by a single parameter $0 \le \theta \le 1$:

$$\begin{aligned} p_1 &= \frac{1}{2} + \frac{1}{4}\theta \\ p_2 &= \frac{1}{4} - \frac{1}{4}\theta \\ p_3 &= \frac{1}{4} - \frac{1}{4}\theta \\ p_4 &= \frac{1}{4}\theta. \end{aligned}$$

Given an observation $\boldsymbol{x} = (x_1, x_2, x_3, x_4)$, the log-likelihood is

$$l(\theta) = x_1 \log(2 + \theta) + (x_2 + x_3) \log(1 - \theta) + x_4 \log \theta + c. \tag{1}$$

To use the EM algorithm on this problem, consider a multinomial with five classes formed from the original multinomial by splitting the first class into two with probabilities $1/2$ and $\theta/4$. The original variable x_1 is now split into $x_1 = x_{11} + x_{12}$. Under this reformulation, we now have a MLE of θ by considering $x_{12} + x_4$ to be a realization of a binomial with $n = x_{12} + x_4 + x_2 + x_3$ and $p = \theta$. However, we do not know x_{12}, and the complete data log-likelihood is

$$l_c(\theta) = (x_{12} + x_4) \log \theta + (x_2 + x_3) \log(1 - \theta). \tag{2}$$

(a) Suppose $\boldsymbol{x} = (125, 18, 20, 34)$. Find the MLE of θ by maximizing (1).

(b) Using (2), develop an EM algorithm for estimating θ. Note: you should be able to combine the E-Step and the M-Step together; i.e., $\hat{\theta}^{(t+1)}$ can be expressed in terms of $\hat{\theta}^{(t)}$.

(c) Compare your answers obtained in (a) and (b).

2. Consider an iid sample drawn from a bivariate normal distribution with mean vector $\mu = (\mu_1, \mu_2)$ and covariance matrix

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{pmatrix}.$$

Suppose through some random accident that the first p observations are missing their first component, the next q observations are missing their second component, and the last r observations are complete. Design an EM algorithm for estimating the five mean and variance parameters, taking the original data before the accidental loss as complete data.

2 Genetic Algorithms

For this question you will develop automatic procedures for fitting piecewise constant regression. Loosely speaking, your task is to use the circles in Figure 1 to estimate the true function which is also displayed in the same figure.

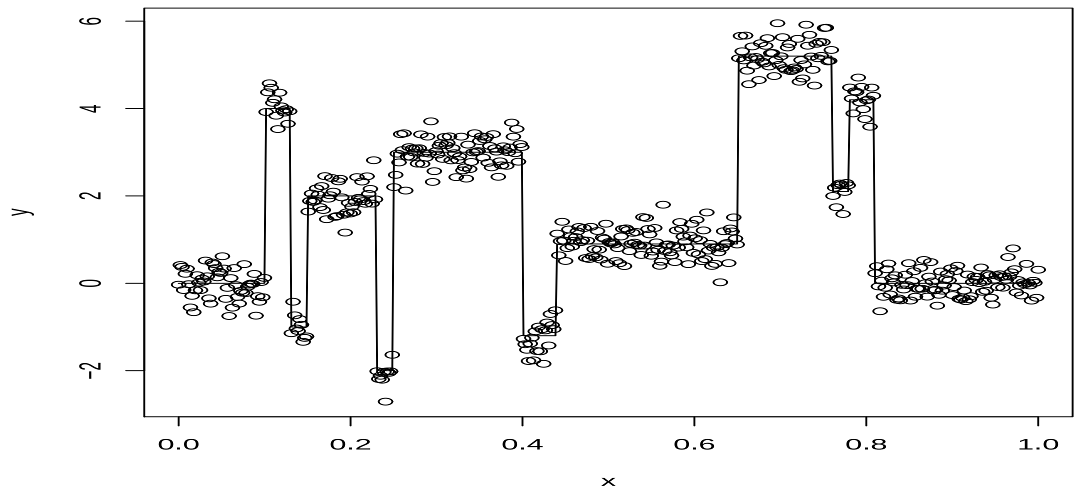

Figure 1: *The circles are the observations* $\{(x_i, y_i)\}_{i=1}^n$ *while the solid line is the true regression function* $f(x)$. *Your task is to estimate* $f(x)$ *given* $\{(x_i, y_i)\}_{i=1}^n$. *This figure is generated by the R-codes listed in Section 2.4.*

2.1 Problem Statement

Suppose n pairs of noisy measurements (x_i, y_i) are observed, with

$$y_i = f(x_i) + e_i, \quad x_1 < \ldots < x_n, \quad e_i \sim \text{ iid } N(0, \sigma^2), \quad i = 1, \ldots, n.$$

The aim is to estimate f. It is known that f is a piecewise constant function, but other details, such as the number of pieces, are unknown.

Let the (unknown) number of pieces be B, and the different pieces are joined at breakpoints $b_1, b_2, \ldots, b_{B-1}$. Without loss of generality, let $b_0 = 0 = x_1$ and $b_B = x_n + \delta = 1$ for a small $\delta > 0$, and assume $b_0 < b_1 < \ldots < b_B$. Let I_E be the indicator function for the event E; that is, $I_E = 1$ if E is true and $I_E = 0$ otherwise. Then our regression model for f is

$$f(x) = f_1 I_{\{b_0 \le x < b_1\}} + f_2 I_{\{b_1 \le x < b_2\}} + \ldots + f_B I_{\{b_{B-1} \le x < b_B\}}, \tag{3}$$

where f_j is the function value (or the "height") of the j-th piece of $f(x)$. To estimate $f(x)$ with the regression model (3), we need to estimate B, $b_1, \ldots, b_{B-1}$, and $f_1, \ldots, f_B$. For clarity, we collect all these parameters in a vector $\boldsymbol{\theta} = (B, b_1, \ldots, b_{B-1}, f_1, \ldots, f_B)$ and denote the corresponding estimates as $\hat{\boldsymbol{\theta}} = (\hat{B}, \hat{b}_1, \ldots, \hat{b}_{\hat{B}-1}, \hat{f}_1, \ldots, \hat{f}_{\hat{B}})$. Unfortunately, for the estimation of $\boldsymbol{\theta}$, the least-squares principle does not work here, nor maximum likelihood (why?). Thus we need to switch to some other methods.

Before proceeding further, we remark that once B and $b_1, \ldots, b_{B-1}$ are estimated, $f_1, \ldots, f_B$ can be uniquely estimated by

$$\hat{f}_j = \frac{1}{\hat{n}_j} \sum_{\hat{b}_{j-1} \le x_i < \hat{b}_j} y_i,$$

where $\hat{n}_j$ is the number of x_i that are inside the interval $[\hat{b}_{j-1}, \hat{b}_j)$. In other words, f_j is estimated by the average of all the y_i's that are in the estimated j-th piece $[\hat{b}_{j-1}, \hat{b}_j)$.

2.2 Model Selection Methods

Now we presents two methods for estimating a "best" fitting model $\hat{\boldsymbol{\theta}}$: the minimum description length (MDL) principle and the Akaike information criterion (AIC).

Minimum Description Length Principle: The MDL principle *defines* the best fitting model as the one that produces the shortest code length of the data; see [2] and references given therein. We will skip the details and state the result that, for our problem, the best $\hat{f}$ (equivalently $\hat{\boldsymbol{\theta}}$) is estimated as the minimizer of

$$\text{MDL}(\hat{f}) = \hat{B} \log n + \frac{1}{2} \sum_{j=1}^{\hat{B}} \log \hat{n}_j + \frac{n}{2} \log \left[\frac{1}{n} \sum_{i=1}^{n} \left\{ y_i - \hat{f}(x_i) \right\}^2 \right].$$

Akaike Information Criterion: With AIC the best fitting model is chosen as the one that minimizes an estimator of the Kullback–Leibler (KL) distance measure between a fitted model and the "true" model (e.g., see [1]). If p is the number of parameters that need to be estimated in a fitted model, then under mild regularity conditions one can show that such a KL distance estimator is $-2 \times$ "maximized log likelihood" $+ 2p$. For our piecewise constant function fitting problem this distance estimator amounts to

$$\mathrm{AIC}(\hat{f}) = n \log\left[\frac{1}{n}\sum_{i=1}^{n}\left\{y_i - \hat{f}(x_i)\right\}^2\right] + \gamma p\Big|_{\gamma=2}.$$

However, it is known that for similar problems $\gamma = \log n$ is a better choice than $\gamma = 2$. Therefore in here we shall select the $\hat{f}$ that minimizes $\mathrm{AIC}(\hat{f})$ with $\gamma = \log n$ and $p = 2\hat{B}$.

2.3 What is Your Task?

Your task is to implement a genetic algorithm for fitting the piecewise constant regression model (3). You will need to implement both $\mathrm{MDL}(\hat{f})$ and $\mathrm{AIC}(\hat{f})$. Write an *R* function that takes two input arguments, the noisy data and an indicator specifying if MDL or AIC should be used. As outputs, your *R* function should plot the noisy data set as well as the fitting piecewise constant function on the screen.

2.4 *R*-Codes for Generating Figure 1

```
truefunction<-function(x){
  t <- c(0.1, 0.13, 0.15, 0.23, 0.25, 0.4, 0.44, 0.65, 0.76, 0.78, 0.81)
  h <- c(4, -5, 3, -4, 5, -4.2, 2.1, 4.3, -3.1, 2.1, -4.2)
  temp <- 0
  for(i in 1:11) {
    temp <- temp + h[i]/2 * (1 + sign(x - t[i]))
  }
  return(temp)
}
n<-512
x<-(0:(n-1))/n
f<-truefunction(x)
set.seed(0401)
y<-f+rnorm(f)/3
plot(x,y)
lines(x,f)
```

References

[1] K. P. Burnham and D. R. Anderson. *Model Selection and Inference: A Practical Information-Theoretic Approach*. Springer-Verlag New York Inc., 1998.

[2] J. Rissanen. *Stochastic Complexity in Statistical Inquiry*. World Scientific, Singapore, 1989.