3.8.1 Maximum likelihood computation

In many situations the log-likelihood $l_N(\theta)$ is particularly well behaved in being continuous with a single maximum away from the extremes of the range of variation of θ. Then $\hat{\theta}_{\text{ml}}$ is obtained simply as the solution of

$$\frac{\partial l_N(\theta)}{\partial \theta} = 0$$

subject to

$$\frac{\partial^2 l_N(\theta)}{\partial \theta^2}|_{\hat{\theta}_{\text{ml}}} < 0$$

to ensure that the identified stationary point is a maximum.

An example is the Gaussian case where it is possible to derive analytically the expression of the maximum likelihood estimators of the mean and variance of $\mathbf{z}$. Let D_N be a random sample from the r.v. $\mathbf{z} \sim \mathcal{N}(\mu, \sigma^2)$. According to (3.8.19), the likelihood of the N samples is given by

$$L_N(\mu, \sigma^2) = \prod_{i=1}^{N} p_{\mathbf{z}}(z_i, \mu, \sigma^2) = \prod_{i=1}^{N} \left(\frac{1}{\sqrt{2\pi}\sigma} \right) \exp\left[\frac{-(z_i - \mu^2}{2\sigma^2} \right]$$

and the log-likelihood is

$$\begin{aligned} l_N(\mu, \sigma^2) &= \log L_N(\mu, \sigma^2) \\ &= \log \left[\prod_{i=1}^{N} p_{\mathbf{z}}(z_i, \mu, \sigma^2) \right] \\ &= \sum_{i=1}^{N} \log p_{\mathbf{z}}(z_i, \mu, \sigma^2) \\ &= -\frac{\sum_{i=1}^{N} (z_i - \mu)^2}{2\sigma^2} + N \log\left(\frac{1}{\sqrt{2\pi}\sigma} \right) \end{aligned}$$

Note that, for a given σ, maximizing the log-likelihood is equivalent to minimize the sum of squares of the difference between z_i and the mean. Taking the derivatives with respect to μ and σ^2 and setting them equal to zero, we obtain

$$\begin{aligned} \hat{\mu}_{\text{ml}} &= \frac{\sum_{i=1}^{N} z_i}{N} \\ &= \hat{\mu} \\ \hat{\sigma}^2_{\text{ml}} &= \frac{\sum_{i=1}^{N} (z_i - \hat{\mu}_{\text{ml}})^2}{N} \\ &\neq \hat{\sigma}^2 \end{aligned}$$

Book information

About this book
Feedback on this book

Buy a printed copy

Gianluca Bontempi
Souhaib Ben Taieb

Statistical foundations of machine learning

Note that the m.l. estimator of the mean coincides with the sample average but that the m.l. estimator of the variance differs from the sample variance in terms of the denominator.

Exercise

- Let $\mathbf{z} \sim \mathcal{U}(0, M)$ and $F_{\mathbf{z}} \to D_N = z_1, \dots, z_N$. Find the maximum likelihood estimator of M.
- Let $\mathbf{z}$ have a Poisson distribution, i.e.

$$p_{\mathbf{z}}(z, \lambda) = \frac{e^{-\lambda}\lambda^z}{z!}$$

If $F_{\mathbf{z}}(z, \lambda) \to D_N = z_1, \dots, z_N$, find the m.l.e. of λ

-

In case of generic distributions $F_{\mathbf{z}}$ computational difficulties may arise. For example no explicit solution might exist for $\partial l_N(\theta)/\partial\theta = 0$. Iterative numerical methods must be used in this case. The computational cost becomes heavier if we consider a vector of parameters instead of a scalar θ or when there are several relative maxima of the function l_N.

Another complex situation occurs when $l_N(\theta)$ is discontinuous, or have a discontinuous first derivative, or a maximum at an extremal point.

R script

Suppose we know the analytical form of a one dimensional function $f(x) : I \to \mathbb{R}$ but not the analytical expression of its extreme points. In this case numerical optimization methods can be applied. The implementation of some continuous optimization routines is available in the R statistical tool.

Consider for example the function $f(x) = (x - 1/3)^2$ and $I = [0, 1]$. The value of the point x where f takes a minimum value can be approximated numerically by this set of R commands

R code

```
f  <- function (x,a) (x-a)^2
xmin  <- optimize(f, c(0, 1), tol = 0.0001, a = 1/3)
xmin
```

These routines may be applied to solve the problem of maximum likelihood estimation which is nothing more than a particular case of optimization problem. Let D_N be a random sample from the r.v. $\mathbf{z} \sim \mathcal{N}(\mu, \sigma^2)$. The minus log-likelihood function of the N samples can be written in R by

R code

```
eml  <- function(m,D,var) {
  N <- length(D)
  Lik <-1
  for (i in 1:N)
  Lik <- Lik*dnorm(D[i],m,sqrt(var))
  -log(Lik)
  }
```

and the numerical minimization of $-l_N(\mu, s^2)$ for a given $\sigma = s$ in the interval $I = [-10, 10]$ can be written in R as

R code

```
xmin <- optimize( eml,c(-10,10),D=DN,var=s)
```

In order to run the above code and compute numerically the m.l. solution we invite the reader to run the following R script.

R code

```
# Script: shows the use of maximum likelihood for parameter estimation

eml <- function(m,D,var) {   ## empirical likelihood function (1 argument)

         N<- length(D)
         Lik<-1
         for (i in 1:N)
           {
          Lik<-Lik*dnorm(D[i],m,sqrt(var))

        }
         -log(Lik)
     }

eml2 <- function(m,D) {   ## empirical likelihood function (2 arguments)
         N<- length(D)
         Lik<-1
         for (i in 1:N)
          Lik<-Lik*dnorm(D[i],m[1],sqrt(max(0,m[2])))
         -log(Lik)
     }

N <-10

DN<-rnorm(N) # data generation

xmin<-optimize( eml,c(-10,10),D=DN,var=1,lower=-1,upper=1)
# maximization of log likelihood function (1 argument)
xmin

xmin2<-optim( c(-10,10),eml2, D=DN)
# maximization of log likelihood function (2 arguments)

mean(DN)
var(DN)
```

•

‹ 3.8 The principle of maximum likelihood up 3.8.2 Properties of m.l. estimators ›

About - Contact - Help - Twitter - Terms of Service - Privacy Policy