STAT 428 Homework 7

Yiming Gao (NetID: yimingg2)

2017/4/12

Contents

Exercise 1

Consider a mixture of a $N(\mu, 1)$ distribution and a $N(0, 1)$ distribution.

$$f(y; \tau, \mu) = \tau(\frac{1}{\sqrt{2\pi}}e^{-(y-\mu)^2/2}) + (1-\tau)(\frac{1}{\sqrt{2\pi}}e^{-y^2/2})$$

where τ is the unknown mixing parameter and μ is the unknown mean of the first subpopulation. Write an EM algorithm to estimate the parameters τ and μ.

E-step

Let $z_i = 1$ if the ith case was drawn from $N(\mu, 1)$ and let $z_i = 0$ if y_i was drawn from $N(0, 1)$. The E-step will involve finding the expected values of these given the observed data and current parameter values. The complete data likelihood can be written as

$$L(\tau, \mu; y, z) = \prod_{i=1}^{n}(\frac{\tau}{\sqrt{2\pi}}e^{-(y_i-\mu)^2/2})^{z_i}(\frac{1-\tau}{\sqrt{2\pi}}e^{-y_i^2/2})^{1-z_i}$$

The complete data log-likelihood is

$$l(\tau, \mu; y, z) = nlog(\frac{\tau}{\sqrt{2\pi}}) - \frac{1}{2}\sum_{i=1}^{n} z_i(y_i-\mu)^2 + nlog(\frac{1-\tau}{\sqrt{2\pi}}) - \frac{1}{2}\sum_{i=1}^{n} y_i^2(1-z_i)$$

Since $z_i = 1$ if the observation is drawn from the first distribution and $z_i = 0$ if y_i the observation is drawn from the second distribution. The unconditional $E(Z)$ is the probability that an observation comes from the first distribution, which is τ.

Suppose we have n observations on Y, $y_1, ..., y_n$. Given a provisional value of $\theta = (\tau, \mu)$, we can compute the conditional expected value $E(Z|y)$ for any realization of Y. That is

$$E(Z|y,\theta^t) = \frac{\tau^t p_1(y;\mu^t,1)}{f(y;\tau^t,\mu^t)}$$

where $p_1(y;\mu^t,1) = \frac{1}{\sqrt{2\pi}}e^{-(y-\mu^t)^2/2}$ is the normal pdf with parameters μ^t and 1. And $f(y;\tau^t,\mu^t)$ is the mixture pdf which is given above.

M-step

The M step is MLE of the parameters:

$$\tau^{t+1} = \frac{1}{n}\sum E(Z|y_i,\theta^t)$$

$$\mu^{t+1} = \frac{1}{n\tau^{t+1}}\sum E(Z|y_i,\theta^t)y_i$$

Let's generate some artificial data and try it out. We generate data from normal mixture with $\tau = 0.6, \mu = 1$.

```r
set.seed(48)
n = 500
# true value
tau = 0.6
mu = 1
y = ifelse(runif(n) < tau, rnorm(n, mu, 1), rnorm(n, 0, 1))
```

We will iterate 100 times.

```r
# Initialize
tau0 = 0.5
mu0 = 0.5

# create vectors to save parameters at all iterations
tauvals = rep(tau0, 100)
muvals = rep(mu0, 100)

# do 39 iterations of EM
for (i in 1:99) {
  # E step
  tmp = tauvals[i]*dnorm(y, muvals[i], 1)/(tauvals[i]*dnorm(y, muvals[i], 1)
                                           +(1-tauvals[i])*dnorm(y, 0, 1))
  # M step
  tauvals[i+1] = mean(tmp)
  muvals[i+1] = sum(tmp*y)/(n*tauvals[i+1])
}
```

We print out our last 30 estimates for two parameters.

```r
print(tauvals[71:100])
```

```
##  [1] 0.5986289 0.5987004 0.5987689 0.5988345 0.5988974 0.5989577 0.5990155
##  [8] 0.5990709 0.5991240 0.5991749 0.5992237 0.5992705 0.5993154 0.5993583
## [15] 0.5993996 0.5994391 0.5994769 0.5995132 0.5995481 0.5995814 0.5996134
## [22] 0.5996441 0.5996735 0.5997017 0.5997287 0.5997546 0.5997794 0.5998032
## [29] 0.5998261 0.5998480
```

```r
print(muvals[71:100])
```

```
##  [1] 0.9568857 0.9567932 0.9567046 0.9566196 0.9565381 0.9564601 0.9563853
##  [8] 0.9563136 0.9562449 0.9561790 0.9561159 0.9560553 0.9559973 0.9559417
## [15] 0.9558884 0.9558373 0.9557884 0.9557414 0.9556964 0.9556533 0.9556119
## [22] 0.9555723 0.9555343 0.9554978 0.9554629 0.9554294 0.9553973 0.9553665
## [29] 0.9553370 0.9553088
```

From the result, we can find out the estimates for (τ, μ) is $(0.5998, 0.9553)$, which is very close to the true value (0.6, 1).

Exercise 2

Generate a dataset with $\tau = 0.5, \mu = 1$ and $n = 100$. I've already simulated the data for illustration in exercise 1. Similarly, let's verify it again. Here we set the initial values for (τ_0, μ_0) is $(0.2, 0.5)$ and do 200 iterations.

```r
set.seed(48)
n = 100
# true value
tau = 0.5
mu = 1
y = ifelse(runif(n) < tau, rnorm(n, mu, 1), rnorm(n, 0, 1))

# Initialize
tau0 = 0.2
mu0 = 0.5
```

```
# create vectors to save parameters at all iterations
tauvals = rep(tau0, 200)
muvals = rep(mu0, 200)

# do 199 iterations of EM
for (i in 1:199) {
  # E step
  tmp = tauvals[i]*dnorm(y, muvals[i], 1)/(tauvals[i]*dnorm(y, muvals[i], 1)
                                               +(1-tauvals[i])*dnorm(y, 0, 1))

  # M step
  tauvals[i+1] = mean(tmp)
  muvals[i+1] = sum(tmp*y)/(n*tauvals[i+1])
}
```

Let's print out the last 50 values to see if the estimates converge.

```
print(tauvals[151: 200])
```

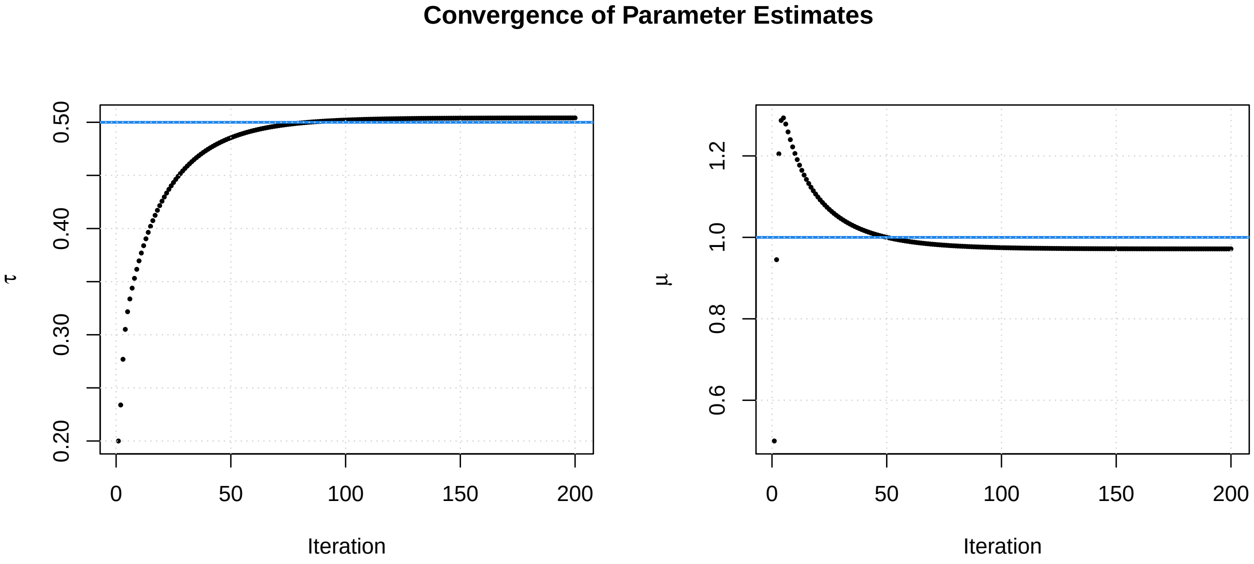
```
##  [1] 0.5039278 0.5039372 0.5039462 0.5039548 0.5039631 0.5039710 0.5039785
##  [8] 0.5039858 0.5039927 0.5039993 0.5040057 0.5040118 0.5040176 0.5040231
## [15] 0.5040285 0.5040336 0.5040385 0.5040432 0.5040476 0.5040519 0.5040560
## [22] 0.5040600 0.5040637 0.5040673 0.5040708 0.5040741 0.5040773 0.5040803
## [29] 0.5040832 0.5040860 0.5040886 0.5040912 0.5040936 0.5040959 0.5040982
## [36] 0.5041003 0.5041024 0.5041043 0.5041062 0.5041080 0.5041097 0.5041114
## [43] 0.5041129 0.5041144 0.5041159 0.5041173 0.5041186 0.5041199 0.5041211
## [50] 0.5041222
```

```
print(muvals[151 :200])
```

```
##  [1] 0.9720216 0.9720076 0.9719943 0.9719814 0.9719692 0.9719574 0.9719462
##  [8] 0.9719354 0.9719251 0.9719152 0.9719058 0.9718967 0.9718880 0.9718798
## [15] 0.9718718 0.9718642 0.9718569 0.9718500 0.9718433 0.9718369 0.9718308
## [22] 0.9718249 0.9718193 0.9718140 0.9718088 0.9718039 0.9717992 0.9717947
## [29] 0.9717904 0.9717862 0.9717823 0.9717785 0.9717749 0.9717714 0.9717681
## [36] 0.9717649 0.9717618 0.9717589 0.9717561 0.9717535 0.9717509 0.9717484
## [43] 0.9717461 0.9717439 0.9717417 0.9717396 0.9717377 0.9717358 0.9717340
## [50] 0.9717322
```

This converges very quickly to $(0.504, 0.972)$ after 200 iterations, which are close to our true parameter values.

Then we can visualize how the process happens.


Convergence of Parameter Estimates
τ
Iteration
μ
Iteration