Newton–Raphson for Meng (1997)

Brendon M. Lapham

July 13, 2013

1 Introduction

The details can be found in Meng (1997). The log-likelihood for the zero-inflated Poisson is

$$\ell(\lambda|Y_{\rm obs}) = n_{\rm obs}[\bar{x}_{\rm obs}\log\lambda - \lambda - \log(1-e^{-\lambda})],$$

where $n_{\rm obs}(=55)$ and $x_{\rm obs}(=1.563636)$ are found from $Y_{\rm obs}$, see Table 1 in Meng (1997). To find the MLE of λ, we want to solve

$$\frac{\mathrm{d}\ell(\lambda|Y_{\rm obs})}{\mathrm{d}\lambda} = 0,$$

subject to the constraint $\lambda > 0$. This can be done by using the Newton–Raphson algorithm and Meng (1997) does this. However, Meng (1997) shows that Newton–Raphson can "fail" for a starting value of $\lambda = 0.4$, and the reason for this is because he fails to impose a constraint on λ. I provide R code here which will find the MLE of λ using Newton–Raphson and imposing the constraint. The derivatives required for Newton–Raphson are found symbolically using the `D` routine in R.

2 The R code

The constraint on λ is imposed by a transformation. Specifically, I set

$$\lambda = e^{\theta},$$

where $\theta \in \mathbb{R}$. Writing the log-likelihood in terms of θ, I have

$$\ell(\theta|Y_{\rm obs}) = n_{\rm obs}[\bar{x}_{\rm obs}\theta - e^{\theta} - \log(1-e^{-e^{\theta}})].$$

Now let

$$g(\theta) = \frac{\mathrm{d}\ell(\theta|Y_{\rm obs})}{\mathrm{d}\theta}.$$

We wish to solve $g(\theta) = 0$ and the general formula for the Newton–Raphson iteration to solve this is given by

$$\theta^{(t+1)} = \theta^{(t)} - \frac{g(\theta^{(t)})}{g'(\theta^{(t)})},$$

where $g'(\theta)$ is the derivative of g. The following R code finds $g(\theta)$ and $g'(\theta)$.

```
ll.expr.c <- expression(n*(xbar*theta-exp(theta)-
                        log(1-exp(-exp(theta)))))

dll.expr.c <- D(ll.expr.c,"theta")
ddll.expr.c <- D(dll.expr.c,"theta")

g <- function(theta) eval(dll.expr.c)
dg <- function(theta) eval(ddll.expr.c)
```

The following is a function which implements the Newton–Raphson algorithm.

```
newtRap = function(start,tol,f,df){
  sol=c(start)
  check=1
  while(check>tol){
  xn = sol[length(sol)] - f(sol[length(sol)])/df(sol[length(sol)])
  sol=c(sol,xn)
  check=abs(sol[length(sol)]-sol[length(sol)-1])
}
  return(sol[length(sol)])
}
```

The following code finds the MLE for λ for the data in Meng (1997).

```
# The "data"
n <- 55
xbar <- 1.56363636363636

# Finding the MLE
theta0 <- log(0.4)
theta.est <- newtRap(theta0,10^-6,g,dg)
exp(theta.est)
```

The MLE of lambda is 0.97218. Figure 1 plots the log-likelihood function and the path the Newton–Raphson iterates.

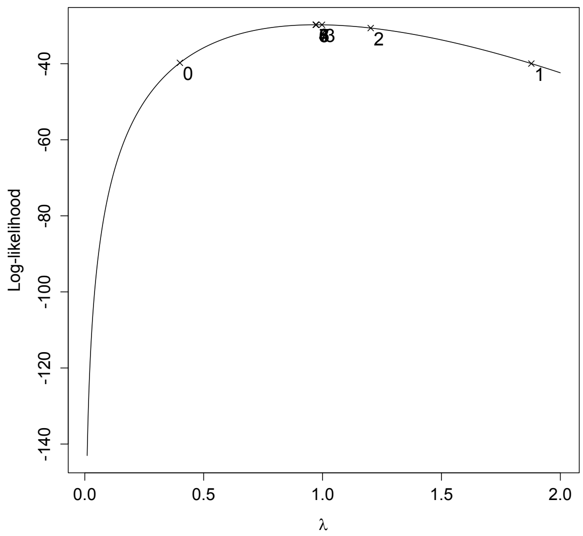

Figure 1: Rapid convergence of Newton–Raphson with constraints.

3 What if we ignore the constraint?

```
ll.expr <- expression(n*(xbar*log(lambda)-lambda-log(1-exp(-lambda))))

dll.expr <- D(ll.expr,"lambda")
ddll.expr <- D(dll.expr,"lambda")

f <- function(lambda) eval(dll.expr)
df <- function(lambda) eval(ddll.expr)

# The "data"
n <- 55
xbar <- 1.56363636363636

# Finding the MLE
lambda0 <- 0.4
newtRap(lambda0,10^-6,f,df)
# Seems to work
# What about a more extreme starting value
lambda0 <- 10^-10
newtRap(lambda0,10^-15,f,df)
# Seems to work
# What about a more extreme starting value
lambda0 <- 10^-15
newtRap(lambda0,10^-15,f,df)
```

References

Meng, X. L. (1997). The EM algorithm and medical studies: a historical linik, *Statistical Methods in Medical Research* **6**: 3–23.