Non-Nested Likelihood Ratio Test:
A Bootstrapping Example

Charlie Gibbons
University of California, Berkeley

Fall 2015

Consider the data on the recreational trips to Lake Somerville in Texas in 1980 from the practice problems. Two models that we proposed for these data were the zero-inflated Poison

$$f(x, y \mid \mu, \lambda) = \left(\frac{\exp\{\mu\}}{1 + \exp\{\mu\}} + \frac{1}{1 + \exp\{\mu\}} e^{-\lambda} \right)^{1-y} \left(\frac{1}{1 + \exp\{\mu\}} \frac{e^{-\lambda}\lambda^x}{x!} \right)^y$$

and the negative binomial

$$f(x \mid r, p) = \frac{1}{x!\Gamma(r)b^r}\Gamma(x + r)p^{x+r}.$$

Since neither of these models is a special case of the other, standard hypothesis testing results cannot be applied here. Instead, we bootstrap the distribution of the likelihood ratio test statistic under the null and alternative hypotheses.

The first step is to find the MLE estimates for the parameters in each model. Recall our log likelihood functions:

```r
zip.ll <- function(theta, visits){
  yi     <- visits > 0

  lambda <- theta[1]
  mu     <- theta[2]
  alpha  <- exp(mu) / (1 + exp(mu))

  yi     <- visits > 0
  -sum((1 - yi) * log(alpha + (1 - alpha) * exp(- lambda)) +
       yi * (log(1 - alpha) + visits * log(lambda) - lambda -
       log(factorial(visits))))
}

zip.mle <- optim(c(0.5, 5), fn = zip.ll, visits = visits)

nb.ll <- function(theta, visits){
  r <- theta[1]
  p <- theta[2]

  -sum(-lgamma(r) - r * (log(p) - log(1 - p)) + lgamma(visits + r) +
       (visits + r) * log(p) - log(factorial(visits)))
}

nb.mle <- optim(c(0.5, 0.5), fn = nb.ll, visits = visits)
```

We also need to create a function that generates random draws from the ZIP distribution:

```
rzip <- function(n, theta){
  lambda <- theta[1]
  mu     <- theta[2]
  alpha  <- exp(mu) / (1 + exp(mu))

  draws <- rbinom(n = n, size = 1, prob = (1 - alpha))
  draws[draws == 1] <- rpois(n = sum(draws == 1), lambda = lambda)

  return(draws)
}
```

Now, let's write a function that does the bootstrapping.

```
LRTboot <- function(zip.mle, nb.mle, visits, BOOT = 999){
  sample.size <- length(visits)
  ### Calculate the sample test statistic
  sample.t <- zip.ll(zip.mle, visits) - nb.ll(nb.mle, visits)
  ## Perform the bootstraps
  boot.t   <- sapply(1:BOOT, function(x){
    ### Draw a sample from the null distribution
    boot.visits <- rzip(sample.size, zip.mle)

    ### Calculate the MLEs for the competing distributions
    boot.zip  <- optim(c(0.5, 5),   fn=zip.ll, visits = boot.visits)$value
    boot.nb   <- optim(c(0.5, 0.5), fn=nb.ll,  visits = boot.visits)$value

    ### Calculate the bootstrapped test statistic
    boot.t   <- boot.zip - boot.nb
    })

  ### Find the p-value
  ### Note: Since we minimized the negative LL function, we count the number
  ###       of bootstrapped statistics that are bigger than the sample value.
  (sum(boot.t >= sample.t) + 1) / (BOOT + 1)
}
```

To perform the bootstrapping:

```
LRTboot(zip.mle$par, nb.mle$par, visits)

Warning:  NaNs produced
Warning:  NaNs produced

[1] 0.001
```

We should check to ensure that this method actually works as expected by simulating draws from the ZIP distribution and checking for correct coverage.

```
zip.sim <- function(theta, n, SIMS){
  results <- sapply(1:SIMS, function(x){
    visits <- rzip(n, theta)
    zip.mle <- optim(c(0.5, 5), fn = zip.ll, visits = visits)
    nb.mle <- optim(c(0.5, 0.5), fn = nb.ll, visits = visits)
    LRTboot(zip.mle$par, nb.mle$par, visits)
  })
  mean(results <= 0.05)
}
```

```
sim.p <- zip.sim(c(6.1, 0.54), 659, 500)


sim.p

[1] 0.02
```