Hypothesis Testing for Lognormal Data

Ryan Lafferty

Introduction

In environmental applications we will find it necessary to model data whose logarithm has an approximate normal distribution. In this report we will demonstrate how to test goodness of fit using a chi square test statistic and then we will demonstrate how to perform hypothesis tests for the mean μ and for the proportion P of possible samples exceeding a threshold.

Generating the Data

We generated the the five datasets for our analysis in the following manner. First, we define an R function called `getData` which takes a number δ as input and returns a list of $n = 500$ values. The function will first randomly generate 500 data points from a standard normal distribution. Then it will add noise from a $U[-\delta, \delta]$ and exponentiate to get a noisy lognormal dataset. The code for this function is as follows. We have included n as an optional second argument in case we want to adjust the size of our sample.

```
getData <- function(delta,numvalues = 500){
  x0 <- rnorm(numvalues,0,1)
  noise <- runif(numvalues,-delta,delta)
  y = exp(x0 + noise)
  return(y)
}
```

Testing for Goodness of Fit

Next we define a function `testLN` that will carry out our chi square test for goodness of fit to test for lognormality. For simplicity, we will take the logarithm of the data and test it for normality. First we will partition our data into k cells, $(-\infty, c_1), (c_1, c_2), ..., (c_{k-1}, \infty)$. Recall the formula for the chi square statistic:

$$\chi^2 = \sum_{i=1}^{k} \frac{(f_i - n\hat{\pi}_i)^2}{n\hat{\pi}_i},$$

where f_i is the observed frequency in the i-th cell, $n = 500$ is the sample size, and $\hat{\pi}_i$ is the probability that a normal variable with mean $\hat{\mu}$ and standard deviation $\hat{\sigma}$ will fall in the i-th cell. Note that both population parameters μ and σ are unknown to us so we have to estimate them based on the data. In our code we will compute π_i for each cell as the difference of the values of the normal cdf evaluated at the cell boundaries.

We wish to test the null hypothesis that the log of the data is normal vs. the alternative hypothesis that the log of the data is not normal. We will reject H_0 if χ^2 exceeds the critical value $\chi_{\alpha,k-1-d}$ where α is the desired level of Type I error, $k - 1 - d$ is the degrees of freedom and $d = 2$ is the number of population parameters we had to estimate. We can compute the critical value using the `qchisq` command in R.

There is not a unique way to partition the real line into cells, but we chose here to evenly space the cell boundaries c_i, $i = 1, ..., k$ between the minimum and maximum values of the logarithm of our data. Given below is the R implementation of this procedure, which will return the value 0 if the null hypothesis is accepted, and 1 if the null hypothesis is rejected.

```
testLN <- function(alpha,data,k=10){
  logdata <- log(data)
  n <- length(logdata)
  cells <- seq(min(logdata),max(logdata),length.out = k-1)

  #compute observed frequencies
  f <- c(length(logdata[logdata<cells[1]]))
  for(i in 1:(k-2)) {
    f <- append(f,length(logdata[logdata>cells[i] & logdata < cells[i+1]]))
  }
  f <- append(f,length(logdata[logdata>cells[k-1]]))

  #use mean(logdata) and var(logdata) to estimate parameters in pi_i = P(X in cell i)
  muhat <- mean(logdata)
  sighat <- sqrt(var(logdata))

  #compute expected frequencies
  pi <- c(pnorm(cells[1],muhat,sighat))
  for(i in 1:(k-2)) {
    pi <- append(pi, pnorm(cells[i+1],muhat,sighat)-pnorm(cells[i],muhat,sighat))
  }
  pi <- append(pi, 1-pnorm(cells[k-1],muhat,sighat))

  #compute chi square statistic
  summands = (f-n*pi)^2/(n*pi)
  chsq <- sum(summands)

  #compute critical value using degrees of freedom = k - 1 - d = k - 3
  critval <- qchisq(1-alpha,k-3)

  #return 0 if null is accepted, 1 if null is rejected
  retval <- 0
  if(chsq > critval) retval <- 1
  return(retval)
}
```

Now that we have defined our test procedure, we can generate some example data and see how the test performs. First we generate some data using the `getData` function.

```
myData <- getData(0)
```

Since `myData` has no added noise in this case, we should expect the goodness of fit test to accept the null hypothesis since we know it was generated from a lognormal distribution. We have:

```
if(testLN(.05, myData) == 1) print("Reject null") else print("Accept null",quote = "FALSE")
```

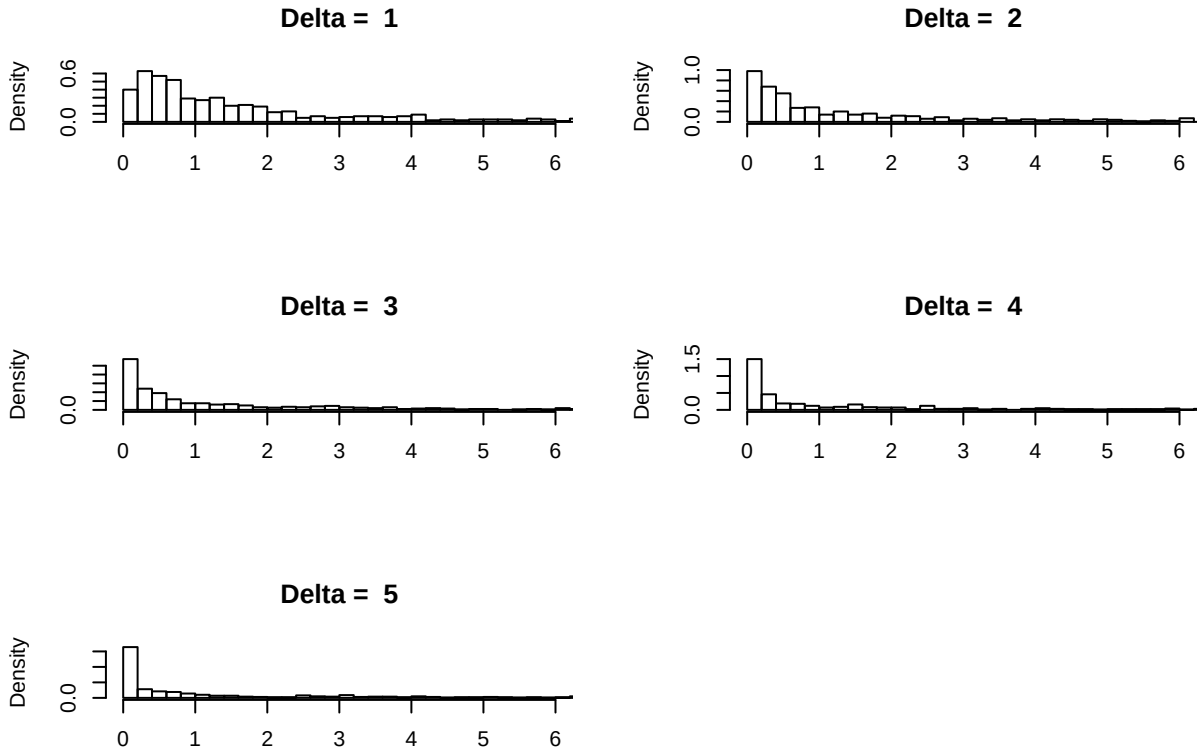
```
## [1] Accept null
```

Next we try the goodness of fit test with δ taking values $1, 2, 3, 4, 5$. We define `datasets` as a list of vectors where `datasets[i]` is generated from the distribution with $\delta = i$ for $i = 1, ..., 5$.

```
datasets <- list(getData(1),getData(2),getData(3),getData(4),getData(5))
```

Below we show a histogram for each of the five datasets to demonstrate how the increase in delta causes the LN curve to deviate.

```
par(c(2, 1, 1, 2), mfrow = c(3,2))
for(i in 1:5) {
  hist(datasets[[i]],main = paste("Delta = ",i),
       xlim = c(0,6),breaks = seq(0,10000,.2),prob = TRUE, xlab = "")}
```



Let us try testing each dataset with $k = 5, 10, 15$ and 20.

```
## [1]                    k=5          k=10         k=15         k=20
## [1] Delta =  0 :       Accept       Accept       Accept       Accept
## [1] Delta =  1 :       Accept       Accept       Accept       Accept
## [1] Delta =  2 :       Reject       Reject       Accept       Accept
## [1] Delta =  3 :       Reject       Accept       Reject       Reject
## [1] Delta =  4 :       Reject       Reject       Reject       Reject
## [1] Delta =  5 :       Reject       Reject       Reject       Reject
```

Testing for the Mean

We will assume that the central limit theorem applies. This assumption will hold in our case since we are using a large sample ($n = 500$). We want to test the null hypothesis $H_0 : \mu \leq \mu_0$ against the alternative hypothesis $H_1 : \mu > \mu_0$. Define the test statistic

$$T = \frac{\sqrt{n}(\bar{X} - \mu_0)}{s}$$

where s is the sample variance. This statistic will have an approximately standard normal distribution since the sample size is very large. We will reject the null hypothesis whenever T exceeds the critical value Z_α,

where Z_α is such that $P(Z > Z_\alpha) = \alpha$ when Z is a standard normal random variable. We can determine this critical value by using the R command `qnorm`. Below we define a function that will test for the mean.

```
testMean <- function(alpha, data, mu0){
  n <- length(data)
  xbar <- mean(data)
  s <- sqrt(var(data))

  #compute test statistic
  T <- sqrt(n)*(xbar - mu0)/s

  #compute critical value
  Za <- qnorm(1-alpha)

  #return 0 if H0 is accepted, return 1 if H0 is rejected
  if(T>Za) return(1) else return(0)
}
```

Let us try to perform a hypothesis test. Consider the dataset `myData` which was generated as the exponential of an ordinary standard normal variable (with no noise). Recall that if $Y \sim N(\eta, \tau^2)$ and $X = e^Y \sim LN(\mu, \sigma^2)$ then we have $\mu = e^{\eta + \frac{\tau^2}{2}}$. Therefore we should expect the mean in our case to be $e^{\frac{1}{2}} \approx 1.65$. Thus let us try testing if the mean is greater than 1. We have

```
if(testMean(.05,myData, 1)==0) print("Accept",quote=FALSE) else print("Reject",quote=FALSE)
```

```
## [1] Reject
```

On the other hand, if we test if the mean is greater than 2, we have

```
if(testMean(.05,myData, 2)==0) print("Accept",quote=FALSE) else print("Reject",quote=FALSE)
```

```
## [1] Accept
```

Therefore the procedure works as expected. Note that we have relied on the assumption that the sample size of our test is very large. If this assumption does not hold, it may be necessary to use a t test or even a more complicated test derived from the lognormal distribution.

We will demonstrate the test for $\mu = 1$ on three more datasets ($\delta = 1, 2, 3$) as follows:

```
## [1] Delta =  1 :   Reject
## [1] Delta =  2 :   Reject
## [1] Delta =  3 :   Reject
```

Testing the Proportion of Observations Exceeding a Threshold

Here we define P to be the chance that a randomly selected observation will exceed c where c is a predetermined constant. Equivalently, P is the percentage of possible observations whose value exceeds c. We want to test whether P is greater than some value P_0. We define the test statistic

$$T = \frac{\sqrt{n}(p - P_0)}{\sqrt{P_0(1 - P_0)}},$$

where p is the observed percentage. Then we will reject $H_0 : P > P_0$ whenever T is greater than the critical value Z_α (the same critical value as before).

To do this we will define a function `testP` which will return 0 if the null is accepted and 1 if the null is rejected:

```r
testP <- function(alpha, data, c, P0){
  n <- length(data)
  p <- length(data[data>c])/n

  #compute test statistic
  T <- sqrt(n)*(p-P0)/sqrt(P0*(1-P0))

  #compute critical value
  Za <- qnorm(1-alpha)

  #return 0 if H0 is accepted, return 1 if H0 is rejected
  if(T>Za) return(1) else return(0)
}
```

Note that we are again assuming that the sample is very large and that the central limit theorem applies. As an example, let us test whether the proportion of values exceeding $c = 3$ in our data is greater than .1.

We compute:

```r
if(testP(.05, myData, 3,.1)==0) print("Accept",quote=FALSE) else
  print("Reject", quote=FALSE)
```

```
## [1] Reject
```

In this case the null was rejected which suggests that at least of the samples have a value exceeding 3. We will demonstrate the test for $c = 3, P_0 = .1$ on three more datasets ($\delta = 1, 2, 3$) as follows:

```
## [1] Delta =  1 :   Reject
## [1] Delta =  2 :   Reject
## [1] Delta =  3 :   Reject
```