ST2005: Applied Probability II
Computing Laboratory 4

Likelihood function for success probability

Recall the example discussed in lectures where a coin (about which nothing is known) is tossed 10 times and 8 heads are observed. The task is to estimate *p* the probability of heads assuming that each toss of the coin is an independent Bernoulli trial with success probability *p*.

The following syntax returns the likelihood, L(p), the likelihood function for p.

```
BinomF <- function( p )
{
  dbinom( x=8, size=10, prob=p )
}
```

Now we make a vector of candidate values for p running from 0 to 1 separated by 0.01. You can use the seq function to set up the vector. Compute the L(p) for each candidate p in the vector. Return a list with named entries p and Lp giving the vector of candidate values and the corresponding likelihood.

```
LF <- function( )
{
   p <- seq( 0, 1, by=0.01 )
  Lp <- dbinom( x=8, size=10, prob=p )
  return( list( p=p, Lp=Lp ) )
}

Z <- LF( )
Z
```

Be careful that Z is a list!

You should try plotting the likelihood function to see what it looks like. For example, if we give vectors of x and y coordinates, using the function

```
plot( x, y, type=”l”, xlab=”x-axis”, ylab=”y-axis”, main=”Title”)
```

will plot give a plot where the points in increasing order of x values are connected by a line. The xlab and ylab and main arguments can be used to name the axes and give a title to the plot. When examining likelihood plots, to identify maxima, it is sometimes convenient to plot the normalized likelihood (all likelihoods divided by the maximum likelihood obtained). This means that at the MLE, the normalized likelihood plot will attain a value of 1. See if you can plot a normalized likelihood plot for this example.

Log-likelihood function for exponential rate

The following simulate 50 independent observations from an exponential distribution with mean 5.

```
ExpSamp <- function( ) {
  rexp( 50, rate=1/5 )
}

xsim <- ExpSamp ( )
```

The following function returns the log-likelihood function for the rate of the exponential distribution based on the data simulated in above.

```
LLExp <- function( x, lambda ) {
  N <- length(lambda)
  if( N > 1 )
  {
        ll <- numeric( N )
        for( k in 1:N ) ll[k] <- ExpSamp(x,lambda[k])
        return(ll)
  }
else
{
        return( sum( dexp( x, rate=lambda, log=TRUE ) ) )
  }
}
```

You could verify that your function is working by making a plot like that suggested above for the success probability.