Wald Tests with R

```
> # Wald test example
>
> # Re-generate Gamma data
> set.seed(3201); alpha=2; beta=3
> D <- round(rgamma(50,shape=alpha, scale=beta),2); D
 [1] 20.87 13.74  5.13  2.76  4.73  2.66 11.74  0.75 22.07 10.49  7.26  5.82 13.08
[14]  1.79  4.57  1.40  1.13  6.84  3.21  0.38 11.24  1.72  4.69  1.96  7.87  8.49
[27]  5.31  3.40  5.24  1.64  7.17  9.60  6.97 10.87  5.23  5.53 15.80  6.40 11.25
[40]  4.91 12.05  5.44 12.62  1.81  2.70  3.03  4.09 12.29  3.23 10.94
> momalpha <- mean(D)^2/var(D); momalpha
[1] 1.899754
> mombeta <- var(D)/mean(D); mombeta
[1] 3.620574
>
> gmll2 <- function(theta,datta)
+      { gmll2 <- -sum(dgamma(datta,shape=theta[1],scale=theta[2],log=T))
+        gmll2
+      } # End of gmll2
>
> # Maximum likelihood estimation
> gamama = nlm(gmll2,c(momalpha,mombeta),hessian=T,datta=D)
> thetahat = gamama$estimate; thetahat
[1] 1.805930 3.808674
> kov = solve(gamama$hessian) # Inverse of (estimated) observed info
> kov
           [,1]       [,2]
[1,]  0.1111796 -0.2345578
[2,] -0.2345578  0.6555641
>
> # Wald test of H0: alpha = beta
> # LR test gave G2 = 4.2776, p = 0.039
>
> source("http://www.utstat.utoronto.ca/~brunner/appliedf13/code_n_data/lecture/Wtest.txt")
> Wtest

function(L,Tn,Vn,h=0) # H0: L theta = h
# Note Vn is the estimated asymptotic covariance matrix of Tn,
# so it's Sigma-hat divided by n. For Wald tests based on numerical
# MLEs, Tn = theta-hat, and Vn is the inverse of the Hessian.
     {
     Wtest = numeric(3)
     names(Wtest) = c("W","df","p-value")
     r = dim(L)[1]
     W = t(L%*%Tn-h) %*% solve(L%*%Vn%*%t(L)) %*%
          (L%*%Tn-h)
     W = as.numeric(W)
     pval = 1-pchisq(W,r)
     Wtest[1] = W; Wtest[2] = r; Wtest[3] = pval
     Wtest
     }
>
> LL = rbind(c(1,-1)); LL
     [,1] [,2]
[1,]    1   -1
> Wtest(LL,thetahat,kov)
         W         df    p-value
3.24550195 1.00000000 0.07161975
```