```
# A simple demonstration of the tobit model

# clear
rm(list = ls())

# the dgp
# set sample size
N=500
# draw N random x's
x=rnorm(N,0,5)
# define parameter vector (for true model)
parameters=rbind(.5,2)
# generate ystar (the true y)
ystar=cbind(1,x)%*%parameters+rnorm(N)
# generate y (the observed y)
y=pmax(0,ystar)

#generate x matrix
X=cbind(1,x)


#obtain ols estimates
betaols=solve(t(X)%*%X)%*%t(X)%*%(y)

#obtain tobit estimates

# define tobit log-likelihood funcion
tobitlik<-function(beta,x,y){
l=numeric(length(y))
betatobit=beta[1:ncol(x),1]
sigma=beta[nrow(beta),1]
l1=(-0.5*(log(2*pi)+log(sigma^2)+((y-x%*%betatobit)^2)/(sigma^2) ))
l2=log(1-pnorm(x%*%betatobit/sigma))
l[y>0]=l1[y>0]
l[y==0]=l2[y==0]
return(l)
}


# install maxLik package
# install.packages("maxLik")

# load the maxLik package
library("maxLik")

# maximum likelihood estimation
estim=maxBHHH(tobitlik,x=X,y=y,start=rbind(betaols,1))
# save estimates
betatobit=estim$estimate
betatobit=betatobit[1:ncol(X)]
```

```
# Obtain estimtes using Powell's estimator (CLAD)

# define objective function
clad<-function(beta,x,y){
-sum(abs(y-pmax(0,x%*%beta)))
}

# optimize
estim=maxNM(clad,x=X,y=y,start=betaols)
# save estimates
betapowell=estim$estimate

# Create output plot
plot(x,ystar,pch="*",cex=0.5,col="blue",main="
     A demonstration of the Tobit estimator",xlab="x",ylab="y")
abline(betaols)
abline(betatobit,col="red",lty=2)
abline(betapowell,col="yellow",lty=3)
points(x,y,col="green",cex=0.5)
legend("topleft",c("ystar","y","OLS","TOBIT","POWELL"),pch=c("*","o","","",""),
       lty=c(0,0,1,2,3),col=c("blue","green","black","red","yellow"),cex=0.7)

#THIS IS IT!
```