

```

# Example code for binary logit MLE: Model Fitting
# Voting behavior example
# Chris Adolph
#
# Estimation by ML using optim() on reduced dataset
#   Model 1: Age, HS, College
#   Model 2: Age, HS, College, Married
#   Model 3: Age, Age^2, HS, College, Married
#
# Likelihood ratio test
# Akaike Information Criterion
# Bayesian Information Criterion
# Average vs Predicted Plots
# ROC plots
# Residual vs Leverage Plots


# Load libraries
library(MASS)
library(nlme)
library(verification)
source("avp.r")          # Average vs Predicted plotting code


# Load data
file <- "nes00a.csv";    # Reduced to only shared obs
data <- read.csv(file,header=TRUE);
attach(data)


# Construct variables and model objects
y <- vote00
x <- cbind(age,hsdeg,coldeg)


# Likelihood function for logit
llk.logit <- function(param,y,x) {
  os <- rep(1,length(x[,1]))
  x <- cbind(os,x)
  b <- param[ 1 : ncol(x) ]
  xb <- x%*%b
  sum( y*log(1+exp(-xb)) + (1-y)*log(1+exp(xb)));
  # optim is a minimizer, so min -ln L(param|y)
}


# Fit logit model using optim

```

```

ls.result <- lm(y~x) # use ls estimates as starting values
stval <- ls.result$coefficients # initial guesses
logit.result <- optim(stval,llk.logit,method="BFGS",hessian=T,y=y,x=x)
# call minimizer procedure
pe.1 <- logit.result$par # point estimates
vc.1 <- solve(logit.result$hessian) # var-cov matrix
se.1 <- sqrt(diag(vc.1)) # standard errors
ll.1 <- -logit.result$value # likelihood at maximum

# Fit logit model with added covariate: married
x <- cbind(age,hsdeg,coldeg,marriedo)
ls.result <- lm(y~x) # use ls estimates as starting values
stval <- ls.result$coefficients # initial guesses
logit.result <- optim(stval,llk.logit,method="BFGS",hessian=T,y=y,x=x)
# call minimizer procedure
pe.2 <- logit.result$par # point estimates
vc.2 <- solve(logit.result$hessian) # var-cov matrix
se.2 <- sqrt(diag(vc.2)) # standard errors
ll.2 <- -logit.result$value # likelihood at maximum

# GOF of added variable

# LR test
lr.test <- 2*(ll.2 - ll.1)
lr.test.p <- pchisq(lr.test,df=1,lower.tail=FALSE)

# BIC
bic.test <- - 2*(ll.2 - ll.1) + 1*log(nrow(x))

# AIC
aic.test <- 2*(ll.2 - ll.1) - 1*2

# Deviance
deviance <- -2*(ll.2)

# Act v Pred plot
avp(y,
    x=cbind(rep(1,nrow(x)),x),
    beta=pe.2,
    fnform="logit",
    cutpoints=seq(0,1,0.1),
    usr=c(0,1,0,1),

```

```

sizefactor=.25,
color = "blue",
output = list(outfile="nes_logit_AvP_0.eps",high=6,wide=5.5,epsi=FALSE),
lab = list(x = .7, y=0.175, str="Model with married", col="blue", cex=1),
ylab = "Actual Voting Rate, by bins of predicted",
xlab = "Predicted Voting Rate, binned",
closeplot=T)

# Act v Pred plot
avp(y,
    x=cbind(rep(1,nrow(x)),x),
    beta=pe.2,
    fnform="logit",
    cutpoints=seq(0,1,0.1),
    usr=c(0,1,0,1),
    sizefactor=.25,
    color = "blue",
    output = list(outfile="nes_logit_AvP.eps",high=6,wide=5.5,epsi=FALSE),
    lab = list(x = .7, y=0.175, str="Model with married", col="blue", cex=1),
    ylab = "Actual Voting Rate, by bins of predicted",
    xlab = "Predicted Voting Rate, binned",
    closeplot=F)

x <- cbind(age,hsdeg,coldeg)
avp(y,
    x=cbind(rep(1,nrow(x)),x),
    beta=pe.1,
    fnform="logit",
    cutpoints=seq(0,1,0.1),
    usr=c(0,1,0,1),
    sizefactor=.25,
    color = "red",
    output = list(outfile="nes_logit_AvP.eps",high=6,wide=5.5,epsi=FALSE),
    lab = list(x = .7, y=0.1, str="Model w/o married", col="red", cex=1),
    addtoplot=T,
    closeplot=T)

# ROC
yhat.1 <- 1/(1+exp(-cbind(rep(1,nrow(x)),x)%*%pe.1))
postscript("roc_m1.eps",paper="letter",pointsize = 14,width=6,height=5.5,horizontal = FALSE, c
roc.plot(y,cbind(yhat.1))
dev.off()

```

```

yhat.2 <- 1/(1+exp(-cbind(rep(1,nrow(x)),x,marriedo)%*%pe.2))
postscript("roc_m1m2.eps",paper="letter",pointsizesize = 14,width=6,height=5.5,horizontal = FALSE,
roc.plot(y,cbind(yhat.1,yhat.2))
dev.off()

```

```

### Residuals
glm.result <- glm(y~age + hsdeg + coldeg + marriedo,family="binomial")
summary.glm(glm.result)
hatscore <- hatvalues(glm.result)/mean(hatvalues(glm.result))
rstu <- rstudent(glm.result)

```

```

#x11()
#plot(hatscore,rstu)

```

```

usr <- c(0,10,-3,3)

```

```

postscript("nes_resid1.eps",paper="letter",pointsizesize = 14,width=5.5,height=5,horizontal = FALSE,
plot.new()
par(usr=usr,tcl=-0.1,mgp=c(2,0.35,0))
axis(2,labels = T, tick = T, line = 0, outer = F, font = NA, fg = "black",las=1);
par(usr=usr,tcl=-0.1,mgp=c(2,0.15,0))
axis(1,at=c(0,1,2,3,4,5,6,7,8,9,10),labels = T, tick = T, line = 0, outer = F, font = NA, fg = "black",las=1);

title(xlab="Standardized hat-values",ylab="Studentized residuals")
points(hatscore,rstu,col = "red")
lines(c(usr[1],usr[2]),c(-2,-2),lty="dashed")
lines(c(usr[1],usr[2]),c(2,2),lty="dashed")
lines(c(3,3),c(usr[3],usr[4]),lty="dashed")

```

```

dev.off()

```

```

# Fit logit model with added covariate: married & age^2
x <- cbind(age,hsdeg,coldeg,marriedo,age^2)
ls.result <- lm(y~x) # use ls estimates as starting values
stval <- ls.result$coefficients # initial guesses
logit.result <- optim(stval,llk.logit,method="BFGS",hessian=T,y=y,x=x)
# call minimizer procedure
pe.3 <- logit.result$par # point estimates
vc.3 <- solve(logit.result$hessian) # var-cov matrix
se.3 <- sqrt(diag(vc.2)) # standard errors
ll.3 <- -logit.result$value # likelihood at maximum

```

```

### Residuals
age2 <- age*age
glm.result <- glm(y~age + hsdeg + coldeg + marriedo + age2,family="binomial")
summary.glm(glm.result)
hatscore2 <- hatvalues(glm.result)/mean(hatvalues(glm.result))
rstu2 <- rstudent(glm.result)
#x11()
#plot(hatscore2,rstu2)

usr <- c(0,10,-3,3)

postscript("nes_resid2.eps",paper="letter",pointsize = 14,width=5.5,height=5,horizontal = FALSE)
plot.new()
par(usr=usr,tcl=-0.1,mgp=c(2,0.35,0))
axis(2,labels = T, tick = T, line = 0, outer = F, font = NA, fg = "black",las=1);
par(usr=usr,tcl=-0.1,mgp=c(2,0.15,0))
axis(1,at=c(0,1,2,3,4,5,6,7,8,9,10),labels = T, tick = T, line = 0, outer = F, font = NA, fg = "black",las=1);

title(xlab="Standardized hat-values",ylab="Studentized residuals")
points(hatscore,rstu,col = "red")
points(hatscore2,rstu2, col = "blue")
lines(c(usr[1],usr[2]),c(-2,-2),lty="dashed")
lines(c(usr[1],usr[2]),c(2,2),lty="dashed")
lines(c(3,3),c(usr[3],usr[4]),lty="dashed")

dev.off()

```