#### R CODE FOR MISCLASSIFICATION-ADJUSTED ROC #####

#### FUNCTIONS FOR MISCLASSIFICATION ROC ####

## Implementation of the misclassification-adjusted logistic regression model

## described in Neuhaus (Biometrika 1999)

Misclassify\_logistic\_IWLS <- function(X, y, gamma0, gamma1, maxIter=300, tol=1E-6){

# X is the covariate matrix

# y is the response vector

# maxIter is the maximum number of iterations

# tol is a convergence criterion

# gamma0 = P(Y=1 | T=0)

# gamma1 = P(Y=0 | T=1)

b <- bLast <- glm(y~X, family="binomial")$coef

X <- cbind(1, X) # add constant

w=0

it <- 1 # iteration index

while (it <= maxIter){

eta <- X %\*% b

mu <- (1-gamma0-gamma1)\*(1/(1 + exp(-eta)))+gamma0

nu <- as.vector(mu\*(1 - mu))

w <- nu

z <- log((mu-gamma0)/(1-gamma1-mu)) + (y - mu)/nu

b <- lsfit(X, z, w, intercept=FALSE)$coef

if (max(abs(b - bLast)/(abs(bLast) + 0.01\*tol)) < tol) {break}

bLast <- b

it <- it + 1 # increment index

}

if(it > maxIter) {warning("maximum iterations exceeded"); list(coefficients=0, var=0, iterations=it)}

else{

Vb <- solve(t(X) %\*% diag(w) %\*% X)

list(coefficients=b, var=Vb, iterations=it)

}

}

# function to create misclassified version of simulated data

misclassify <- function(v,gamma0,gamma1) {

p00 = 1-gamma0

p11 = 1-gamma1

miss\_v=rep(-9, length(v))

if(length(p00)==1) {p00 = rep(p00, length(v)); p11 = rep(p11, length(v))}

for(i in 1:length(v)) {

new\_v=v[i]

if(v[i]==0) {if(runif(1) > p00[i]) {new\_v=1}}

else{{if(runif(1) > p11[i]) {new\_v=0}}}

miss\_v[i]=new\_v

}

return(miss\_v)

}

# this expects the data columns and the betas to be in the same order,

# following the intercept beta

predict <- function(data, beta\_v) {

n\_rows = NULL

if(is.vector(data)) {n\_rows=length(data)}

else{n\_rows=dim(data)[1]}

xb=apply(as.matrix(cbind(rep(1,n\_rows),data)) %\*% diag(beta\_v), 1, sum)

pred = exp(xb) / (1+exp(xb))

}

# Standard ROC Analysis

ROC <- function(pheno, score) {

tp=NULL

fp=NULL

auc=0

n.pts=100

cut = seq(max(score), min(score), length=n.pts)

for(c in cut) {

tp = c(tp, sum(score>c & pheno==1, na.rm=T)/sum(pheno==1 & !is.na(score), na.rm=T))

fp = c(fp, sum(score>c & pheno==0, na.rm=T)/sum(pheno==0 & !is.na(score), na.rm=T))

if(c<max(score)) {auc = auc + ((tail(fp,n=1)-tail(fp,n=2)[1])\*tail(tp,n=2)[1]) + .5\*((tail(fp,n=1)-tail(fp,n=2)[1])\*((tail(tp,n=1)-tail(tp,n=2)[1])))}

}

list(tp=tp, fp=fp, auc=auc)

}

# Misclassification-adjusted ROC procedure

mis\_ROC <- function(y, score, gamma0, gamma1) {

# Compute conidtional predictive probability

Pr\_T = ((gamma1-y\*(2\*gamma1-1)) \* score) / ((1-y) + ((-1)^(1-y))\*((1-gamma1-gamma0)\*score+gamma0))

tp=NULL

fp=NULL

auc=0

n.pts=100

cut = seq(max(score), min(score), length=n.pts)

t=cbind(Pr\_T, 1-Pr\_T, score)

for(c in cut) {

tp = c(tp, sum(subset(t, score>c)[,1]) /sum(Pr\_T))

fp = c(fp, sum(subset(t, score>c)[,2]) /sum(1-Pr\_T))

if(c<max(score)) {auc = auc + ((tail(fp,n=1)-tail(fp,n=2)[1])\*tail(tp,n=2)[1]) + .5\*((tail(fp,n=1)-tail(fp,n=2)[1])\*((tail(tp,n=1)-tail(tp,n=2)[1])))}

}

list(tp=tp, fp=fp, auc=auc, Pr\_T=Pr\_T)

}

### Return the probability from a logitistic model

logit.pred = function(beta,x) {

xb= beta[1] + x %\*% beta[-1]

return(exp(xb)/(1+exp(xb)))

}

#### END OF FUNCTIONS #######################

### Example usage ###

N=5000 #sample size

# Model effect size parameters

B0 = -1

B1 = 1.5

B2 = -1

# misclassification parameters

g0=.2 #gamma0

g1=.3 # gamma1

# Simulate Data

X1 = rnorm(N, 0, 1)

X2 = rnorm(N, 1, 2)

p=exp(B0 + B1\*X1 + B2\*X2) /(1 + exp(B0 + B1\*X1 + B2\*X2))

T = rbinom(N, 1, p)

Y = misclassify(T,g0,g1)

# True Outcome Analysis

true.beta = glm(T~X1+X2, family="binomial")$coef

true.pred = logit.pred(true.beta, cbind(X1,X2))

mis\_ROC(T, true.pred, 0, 0)$auc

# Misclassified Outcome

mis.beta = glm(Y~X1+X2, family="binomial")$coef

mis.pred = logit.pred(mis.beta, cbind(X1,X2))

mis\_ROC(Y, mis.pred, 0, 0)$auc

# Misclassification-Adjusted ROC

cor.beta = Misclassify\_logistic\_IWLS(cbind(X1,X2), Y, g0, g1, maxIter=300)$coefficients

cor.pred = logit.pred(cor.beta, cbind(X1,X2))

mis\_ROC(Y, cor.pred, g0, g1)$auc