--

Skip this for now! Not necessary!!!

library("SuperLearner")

SL.library<- c("SL.glm", "SL.step", "SL.glm.interaction")

# data frame X with baseline covariates and exposure

X<-subset(ObsData, select=c(A.binary, W1, W2))

# create data frames with A=1 and A=0

X1 <- X0<-X

X1$A.binary<- 1

X0$A.binary<- 0

# create newdata by stacking

newdata<- rbind(X,X1,X0)

# call superlearner

Qinit<- SuperLearner(Y=ObsData$Y, X=X, newX=newdata, SL.library=SL.library)

Qinit

> Qinit

Call:

SuperLearner(Y = ObsData$Y, X = X, newX = newdata,

SL.library = SL.library)

Risk Coef

SL.glm\_All 1.285358e-01 0.0000000000

SL.step\_All 1.285358e-01 0.0002581873

SL.glm.interaction\_All 9.384916e-05 0.9997418127

# pred prob of survival given A,W

QbarAW <- Qinit$SL.predict[1:n]

# predicted probability of survival for each subject given A=1 and W

Qbar1W <- Qinit$SL.predict[(n+1): (2\*n)]

# predicted probability of survival for each subject given A=0 and W

Qbar0W <- Qinit$SL.predict[(2\*n+1): (3\*n)]

> # note the simple substitution estimator would be

> PsiHat.SS<-mean(Qbar1W - Qbar0W)

> PsiHat.SS

[1] 7.001514

##########

# 2. Estimate g\_0(A|W) with SuperLearner

############

# creating data frame with only baseline cov

W<- subset(ObsData, select= c(W1,W2))

> # call superlearner

> gHatSL<- SuperLearner(Y=ObsData$A.binary, X=W, family = "binomial", SL.library=SL.library)

> gHatSL

Call:

SuperLearner(Y = ObsData$A.binary, X = W, family = "binomial",

SL.library = SL.library)

Risk Coef

SL.glm\_All 0.1071645 1

SL.step\_All 0.1071645 0

SL.glm.interaction\_All 0.1080595 0

gHat1W<- gHatSL$SL.predict

gHat0W<- 1- gHat1W

> summary(gHat1W)

V1

Min. :0.02377

1st Qu.:0.17860

Median :0.66002

Mean :0.56897

3rd Qu.:0.97494

Max. :0.97494

> summary(gHat0W)

V1

Min. :0.02506

1st Qu.:0.02506

Median :0.33998

Mean :0.43103

3rd Qu.:0.82140

Max. :0.97623

> PsiHat.IPTW<- mean(as.numeric(ObsData$A.binary==1)\*ObsData$Y/gHat1W) -mean(as.numeric(ObsData$A.binary==0)\*ObsData$Y/gHat0W)

> PsiHat.IPTW

[1] 6.563232

>

H.AW<- as.numeric(ObsData$A.binary==1)/gHat1W - as.numeric(ObsData$A.binary==0)/gHat0W

# also want to evaluate the clever covariates at A=1 and A=0 for all subjects

H.1W<- 1/gHat1W

H.0W<- -1/gHat0W

logitUpdate<- glm(ObsData$Y ~ -1 +offset(qlogis(QbarAW)) + H.AW)

eps<- logitUpdate$coef

eps

library("SuperLearner")

SL.library<- c("SL.glm", "SL.step", "SL.glm.interaction")

# data frame X with baseline covariates and exposure

X<-subset(ObsData, select=c(A.binary, W1, W2))

# create data frames with A=1 and A=0

X1 <- X0<-X

X1$A.binary<- 1

X0$A.binary<- 0

# create newdata by stacking

newdata<- rbind(X,X1,X0)

# call superlearner

Qinit<- SuperLearner(Y=ObsData$Y, X=X, newX=newdata, SL.library=SL.library)

Qinit

> Qinit

Call:

SuperLearner(Y = ObsData$Y, X = X, newX = newdata,

SL.library = SL.library)

Risk Coef

SL.glm\_All 1.285358e-01 0.0000000000

SL.step\_All 1.285358e-01 0.0002581873

SL.glm.interaction\_All 9.384916e-05 0.9997418127

# pred prob of survival given A,W

QbarAW <- Qinit$SL.predict[1:n]

# predicted probability of survival for each subject given A=1 and W

Qbar1W <- Qinit$SL.predict[(n+1): (2\*n)]

# predicted probability of survival for each subject given A=0 and W

Qbar0W <- Qinit$SL.predict[(2\*n+1): (3\*n)]