################### Sample Code for Matching

#RHC Example

#install packages

install.packages("tableone")

install.packages("Matching")

#load packages

library(tableone)

library(Matching)

#read in data

load(url("http://biostat.mc.vanderbilt.edu/wiki/pub/Main/DataSets/rhc.sav"))

#view data

View(rhc)

#treatment variables is swang1

#x variables that we will use

#cat1: primary disease category

#age

#sex

#meanbp1: mean blood pressure

#create a data set with just these variables, for simplicity

ARF<-as.numeric(rhc$cat1=='ARF')

CHF<-as.numeric(rhc$cat1=='CHF')

Cirr<-as.numeric(rhc$cat1=='Cirrhosis')

colcan<-as.numeric(rhc$cat1=='Colon Cancer')

Coma<-as.numeric(rhc$cat1=='Coma')

COPD<-as.numeric(rhc$cat1=='COPD')

lungcan<-as.numeric(rhc$cat1=='Lung Cancer')

MOSF<-as.numeric(rhc$cat1=='MOSF w/Malignancy')

sepsis<-as.numeric(rhc$cat1=='MOSF w/Sepsis')

female<-as.numeric(rhc$sex=='Female')

died<-as.numeric(rhc$death=='Yes')

age<-rhc$age

treatment<-as.numeric(rhc$swang1=='RHC')

meanbp1<-rhc$meanbp1

#new dataset

mydata<-cbind(ARF,CHF,Cirr,colcan,Coma,lungcan,MOSF,sepsis,

age,female,meanbp1,treatment,died)

mydata<-data.frame(mydata)

#covariates we will use (shorter list than you would use in practice)

xvars<-c("ARF","CHF","Cirr","colcan","Coma","lungcan","MOSF","sepsis",

"age","female","meanbp1")

#look at a table 1

table1<- CreateTableOne(vars=xvars,strata="treatment", data=mydata, test=FALSE)

## include standardized mean difference (SMD)

print(table1,smd=TRUE)

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

#do greedy matching on Mahalanobis distance

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

greedymatch<-Match(Tr=treatment,M=1,X=mydata[xvars],replace=FALSE)

matched<-mydata[unlist(greedymatch[c("index.treated","index.control")]), ]

#get table 1 for matched data with standardized differences

matchedtab1<-CreateTableOne(vars=xvars, strata ="treatment",

data=matched, test = FALSE)

print(matchedtab1, smd = TRUE)

#outcome analysis

y\_trt<-matched$died[matched$treatment==1]

y\_con<-matched$died[matched$treatment==0]

#pairwise difference

diffy<-y\_trt-y\_con

#paired t-test

t.test(diffy)

#McNemar test

table(y\_trt,y\_con)

mcnemar.test(matrix(c(973,513,395,303),2,2))

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

#propensity score matching

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

#fit a propensity score model. logistic regression

psmodel<-glm(treatment~ARF+CHF+Cirr+colcan+Coma+lungcan+MOSF+

sepsis+age+female+meanbp1+aps,

family=binomial(),data=mydata)

#show coefficients etc

summary(psmodel)

#create propensity score

pscore<-psmodel$fitted.values

#do greedy matching on logit(PS) using Match with a caliper

logit <- function(p) {log(p)-log(1-p)}

psmatch<-Match(Tr=mydata$treatment,M=1,X=logit(pscore),replace=FALSE,caliper=.2)

matched<-mydata[unlist(psmatch[c("index.treated","index.control")]), ]

xvars<-c("ARF","CHF","Cirr","colcan","Coma","lungcan","MOSF","sepsis",

"age","female","meanbp1")

#get standardized differences

matchedtab1<-CreateTableOne(vars=xvars, strata ="treatment",

data=matched, test = FALSE)

print(matchedtab1, smd = TRUE)

#outcome analysis

y\_trt<-matched$died[matched$treatment==1]

y\_con<-matched$died[matched$treatment==0]

#pairwise difference

diffy<-y\_trt-y\_con

#paired t-test

t.test(diffy)

################### Sample Code for IPTW

#RHC Example

#install packages (if needed)

install.packages("tableone")

install.packages("ipw")

install.packages("sandwich")

install.packages("survey")

#load packages

library(tableone)

library(ipw)

library(sandwich) #for robust variance estimation

library(survey)

expit <- function(x) {1/(1+exp(-x)) }

logit <- function(p) {log(p)-log(1-p)}

#read in data

load(url("http://biostat.mc.vanderbilt.edu/wiki/pub/Main/DataSets/rhc.sav"))

#view data

View(rhc)

#treatment variables is swang1

#x variables that we will use

#cat1: primary disease category

#age

#sex

#meanbp1: mean blood pressure

#create a data set with just these variables, for simplicity

ARF<-as.numeric(rhc$cat1=='ARF')

CHF<-as.numeric(rhc$cat1=='CHF')

Cirr<-as.numeric(rhc$cat1=='Cirrhosis')

colcan<-as.numeric(rhc$cat1=='Colon Cancer')

Coma<-as.numeric(rhc$cat1=='Coma')

COPD<-as.numeric(rhc$cat1=='COPD')

lungcan<-as.numeric(rhc$cat1=='Lung Cancer')

MOSF<-as.numeric(rhc$cat1=='MOSF w/Malignancy')

sepsis<-as.numeric(rhc$cat1=='MOSF w/Sepsis')

female<-as.numeric(rhc$sex=='Female')

died<-as.integer(rhc$death=='Yes')

age<-rhc$age

treatment<-as.numeric(rhc$swang1=='RHC')

meanbp1<-rhc$meanbp1

#new dataset

mydata<-cbind(ARF,CHF,Cirr,colcan,Coma,lungcan,MOSF,sepsis,

age,female,meanbp1,treatment,died)

mydata<-data.frame(mydata)

#covariates we will use (shorter list than you would use in practice)

xvars<-c("age","female","meanbp1","ARF","CHF","Cirr","colcan",

"Coma","lungcan","MOSF","sepsis")

#look at a table 1

table1<- CreateTableOne(vars=xvars,strata="treatment", data=mydata, test=FALSE)

## include standardized mean difference (SMD)

print(table1,smd=TRUE)

#propensity score model

psmodel <- glm(treatment ~ age + female + meanbp1+ARF+CHF+Cirr+colcan+

Coma+lungcan+MOSF+sepsis,

family = binomial(link ="logit"))

## value of propensity score for each subject

ps <-predict(psmodel, type = "response")

#create weights

weight<-ifelse(treatment==1,1/(ps),1/(1-ps))

#apply weights to data

weighteddata<-svydesign(ids = ~ 1, data =mydata, weights = ~ weight)

#weighted table 1

weightedtable <-svyCreateTableOne(vars = xvars, strata = "treatment",

data = weighteddata, test = FALSE)

## Show table with SMD

print(weightedtable, smd = TRUE)

#to get a weighted mean for a single covariate directly:

mean(weight[treatment==1]\*age[treatment==1])/(mean(weight[treatment==1]))

#get causal risk difference

glm.obj<-glm(died~treatment,weights=weight,family=quasibinomial(link="identity"))

#summary(glm.obj)

betaiptw<-coef(glm.obj)

SE<-sqrt(diag(vcovHC(glm.obj, type="HC0")))

causalrd<-(betaiptw[2])

lcl<-(betaiptw[2]-1.96\*SE[2])

ucl<-(betaiptw[2]+1.96\*SE[2])

c(lcl,causalrd,ucl)

#get causal relative risk. Weighted GLM

glm.obj<-glm(died~treatment,weights=weight,family=quasibinomial(link=log))

#summary(glm.obj)

betaiptw<-coef(glm.obj)

#to properly account for weighting, use asymptotic (sandwich) variance

SE<-sqrt(diag(vcovHC(glm.obj, type="HC0")))

#get point estimate and CI for relative risk (need to exponentiate)

causalrr<-exp(betaiptw[2])

lcl<-exp(betaiptw[2]-1.96\*SE[2])

ucl<-exp(betaiptw[2]+1.96\*SE[2])

c(lcl,causalrr,ucl)

#truncate weights at 10

truncweight<-replace(weight,weight>10,10)

#get causal risk difference

glm.obj<-glm(died~treatment,weights=truncweight,family=quasibinomial(link="identity"))

#summary(glm.obj)

betaiptw<-coef(glm.obj)

SE<-sqrt(diag(vcovHC(glm.obj, type="HC0")))

causalrd<-(betaiptw[2])

lcl<-(betaiptw[2]-1.96\*SE[2])

ucl<-(betaiptw[2]+1.96\*SE[2])

c(lcl,causalrd,ucl)

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

#alternative: use ipw package

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

#first fit propensity score model to get weights

weightmodel<-ipwpoint(exposure= treatment, family = "binomial", link ="logit",

denominator= ~ age + female + meanbp1+ARF+CHF+Cirr+colcan+

Coma+lungcan+MOSF+sepsis, data=mydata)

#numeric summary of weights

summary(weightmodel$ipw.weights)

#plot of weights

ipwplot(weights = weightmodel$ipw.weights, logscale = FALSE,

main = "weights", xlim = c(0, 22))

mydata$wt<-weightmodel$ipw.weights

#fit a marginal structural model (risk difference)

msm <- (svyglm(died ~ treatment, design = svydesign(~ 1, weights = ~wt,

data =mydata)))

coef(msm)

confint(msm)

# fit propensity score model to get weights, but truncated

weightmodel<-ipwpoint(exposure= treatment, family = "binomial", link ="logit",

denominator= ~ age + female + meanbp1+ARF+CHF+Cirr+colcan+

Coma+lungcan+MOSF+sepsis, data=mydata,trunc=.01)

#numeric summary of weights

summary(weightmodel$weights.trun)

#plot of weights

ipwplot(weights = weightmodel$weights.trun, logscale = FALSE,

main = "weights", xlim = c(0, 22))

mydata$wt<-weightmodel$weights.trun

#fit a marginal structural model (risk difference)

msm <- (svyglm(died ~ treatment, design = svydesign(~ 1, weights = ~wt,

data =mydata)))

coef(msm)

confint(msm)

# Sample Code for IV analysis

#install package

install.packages("ivpack")

#load package

library(ivpack)

#read dataset

data(card.data)

#IV is nearc4 (near 4 year college)

#outcome is lwage (log of wage)

#'treatment' is educ (number of years of education)

#summary stats

mean(card.data$nearc4)

par(mfrow=c(1,2))

hist(card.data$lwage)

hist(card.data$educ)

#is the IV associated with the treatment? strenght of IV

mean(card.data$educ[card.data$nearc4==1])

mean(card.data$educ[card.data$nearc4==0])

#make education binary

educ12<-card.data$educ>12

#estimate proportion of 'compliers'

propcomp<-mean(educ12[card.data$nearc4==1])-

mean(educ12[card.data$nearc4==0])

propcomp

#intention to treat effect

itt<-mean(card.data$lwage[card.data$nearc4==1])-

mean(card.data$lwage[card.data$nearc4==0])

itt

#complier average causal effect

itt/propcomp

#two stage least squares

#stage 1: regress A on Z

s1<-lm(educ12~card.data$nearc4)

## get predicted value of A given Z for each subject

predtx <-predict(s1, type = "response")

table(predtx)

#stage 2: regress Y on predicted value of A

lm(card.data$lwage~predtx)

#2SLS using ivpack

ivmodel=ivreg(lwage ~ educ12, ~ nearc4, x=TRUE, data=card.data)

robust.se(ivmodel)

ivmodel=ivreg(lwage ~ educ12 + exper + reg661 + reg662 +

reg663 + reg664 + reg665+ reg666 + reg667 + reg668,

~ nearc4 + exper +

reg661+ reg662 + reg663 + reg664 + reg665 + reg666 +

reg667 + reg668, x=TRUE, data=card.data)