BIOST_HW3

Shuyang Wu

3/9/2017

Q1

(a) Prior mean of sensitivity: $21.96/(21.96 + 5.49) = 0.8$ Prior mean of specificity: $4.1/(4.1 + 1.78) = 0.6972789$ Prior mean of prevalence: $(0+1)/2 = 0.5$

(b)

```
diag.one <- function(data=list(a=125, b= 37),
                     NPOST=5000, BURN=1000, THIN=5,
                     a.pi=1, b.pi=1,
                     a.S=21.96, b.S=5.49,
                     a.C=4.1, b.C=1.76,
                     pi0=.5, S0=.5, C0=.5){
  pi <- pi0
  S <- S0
  C <- C0
  post <- matrix(0, NPOST, 5) #5rows
  dimnames(post) <- list(NULL, c("pi", "S", "C", "PPV","NPV"))
  j <- 1
  for (i in (1:((NPOST*THIN) + BURN))){
    ## sampling from fc for latent Y1
    prob1 <- pi*S/(pi*S + (1-pi)*(1-C))
    Y1 <- rbinom(1, data$a, prob1)
    ## sampling from fc for latent Y2
    prob2 <- pi*(1-S)/(pi*(1-S) + (1-pi)*C)
    Y2 <- rbinom(1, data$b, prob2)
    ## sampling from fc for pi
    pi <- rbeta(1, Y1+Y2+a.pi, data$a+data$b-Y1-Y2+b.pi)
    ## sampling from fc for S
    S <- rbeta(1, Y1 + a.S, Y2 + b.S)
    ## sampling from fc for C
    C <- rbeta(1, data$b - Y2 + a.C, data$a - Y1 + b.C)
    ## PPV
    PPV<-Y1/data$a
    ##NPV
    NPV<-(data$b-Y2)/data$b
    if (i > BURN && (i %% THIN == 0)){
      post[j,] <- c(pi, S, C, PPV, NPV)
      j <- j+1
    }
  }
  return(post)
}

res<-diag.one()
apply(res,2,mean)  #posterior simulations
```

```
##         pi          S          C        PPV        NPV
## 0.7791488 0.8286576 0.5928616 0.8439632 0.4269568
```

```r
apply(res,2,sd)
```

```
##         pi          S          C        PPV        NPV
## 0.20158334 0.05058704 0.20894661 0.20853112 0.25076242
```

```r
apply(res,2,quantile,c(0.025,0.5,0.975)) #interval
```

```
##              pi         S         C   PPV       NPV
## 2.5%  0.1916344 0.7292724 0.2229913 0.208 0.0000000
## 50%   0.8393144 0.8290693 0.6053267 0.928 0.4189189
## 97.5% 0.9915505 0.9226912 0.9409186 1.000 0.9189189
```

Posterior Prevalence Sensitivity Specificity PPV NPV

Mean 0.8097315 0.8271123 0.6170737 0.8743792 0.3953297

std 0.18246162 0.04898285 0.20180800 0.18415189 0.24904023

2.5% 0.2314899 0.7346626 0.2339075 0.2638 0.0000000

median 0.8608360 0.8260054 0.6271441 0.9440 0.3783784

97.5% 0.9924523 0.9223433 0.9456956 1.0000 0.8918919

(c) Sample fraction of positive serologic test = 125/162 = 0.7716049 The mean prevalence calculated taken specificity and sensitivity into account is 0.81, larger than 0.77. Thus, the assumption that all positive results are true positives and all negative results are true negatives are unreasonale.

Q2

(a)

```r
library(tree)
library(treeMI)
library(mi)
```

```
## Loading required package: Matrix

## Loading required package: stats4

## mi (Version 1.0, packaged: 2015-04-16 14:03:10 UTC; goodrich)

## mi  Copyright (C) 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015 Trustees of Columbia University

## This program comes with ABSOLUTELY NO WARRANTY.

## This is free software, and you are welcome to redistribute it

## under the General Public License version 2 or later.

## Execute RShowDoc('COPYING') for details.
```

```r
library(mitools)

sb<-read.csv("~/Downloads/smallbone.csv")

#Percentage of missing values for each variable
colMeans(is.na(sb))
```

```
##         gr        age       race       etoh      smoke   dementia
## 0.00000000 0.00000000 0.00000000 0.10091743 0.12385321 0.03211009
##   Antiseiz     LevoT4   AntiChol    albumin        bmi       lhgb 
## 0.05275229 0.09174312 0.10779817 0.23853211 0.10321101 0.12155963
```

```
#Percentage of subjects with at least one missing variables
nnzero(rowSums(is.na(sb)))/nrow(sb)
```

```
## [1] 0.456422
```

(b) Complete-case analysis

```
model <- glm(gr~etoh+smoke+dementia+Antiseiz+LevoT4+AntiChol+albumin+bmi+lhgb,family=binomial(link='log
summary(model)
```

```
## 
## Call:
## glm(formula = gr ~ etoh + smoke + dementia + Antiseiz + LevoT4 + 
##     AntiChol + albumin + bmi + lhgb, family = binomial(link = "logit"), 
##     data = sb)
## 
## Deviance Residuals: 
##      Min        1Q    Median        3Q       Max  
## -2.47562  -0.60189   0.03018   0.66170   2.04497  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) 10.85508    2.97071   3.654 0.000258 ***
## etoh         1.39093    0.39078   3.559 0.000372 ***
## smoke        0.92920    0.40027   2.321 0.020264 *  
## dementia     2.50919    0.72369   3.467 0.000526 ***
## Antiseiz     3.31056    1.06405   3.111 0.001863 ** 
## LevoT4       2.01009    1.01515   1.980 0.047694 *  
## AntiChol    -1.91833    0.76767  -2.499 0.012458 *  
## albumin     -0.91116    0.35306  -2.581 0.009859 ** 
## bmi         -0.10416    0.03875  -2.688 0.007187 ** 
## lhgb        -2.59693    1.20162  -2.161 0.030681 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 328.55  on 236  degrees of freedom
## Residual deviance: 189.20  on 227  degrees of freedom
##   (199 observations deleted due to missingness)
## AIC: 209.2
## 
## Number of Fisher Scoring iterations: 6
```

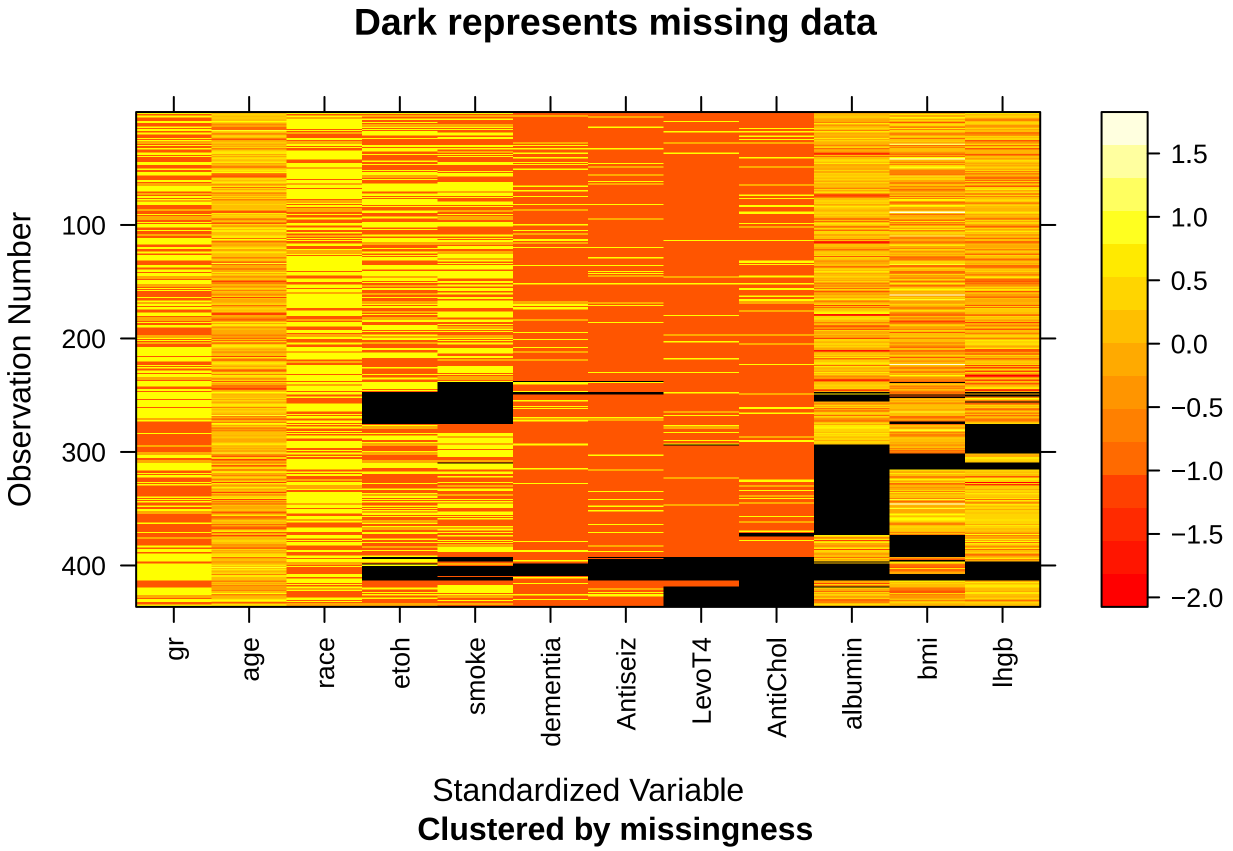
(c)Chained equations

```
##Multiple imputation using chained equations

sb<-missing_data.frame(sb)
```

```
## NOTE: In the following pairs of variables, the missingness pattern of the first is a subset of the s
##  Please verify whether they are in fact logically distinct variables.
##      [,1]       [,2]      
## [1,] "dementia" "Antiseiz"
```

```
#display missing data patterns
image(sb)
```



```
#display data types and other information
show(sb)
```

```
## Object of class missing_data.frame with 436 observations on 12 variables
##
## There are 38 missing data patterns
##
## Append '@patterns' to this missing_data.frame to access the corresponding pattern for every observati
##
##                type missing method  model
## gr           binary       0    <NA>   <NA>
## age      continuous       0    <NA>   <NA>
## race         binary       0    <NA>   <NA>
## etoh         binary      44     ppd  logit
## smoke        binary      54     ppd  logit
## dementia     binary      14     ppd  logit
## Antiseiz     binary      23     ppd  logit
## LevoT4       binary      40     ppd  logit
## AntiChol     binary      47     ppd  logit
## albumin  continuous     104     ppd linear
## bmi      continuous      45     ppd linear
## lhgb     continuous      53     ppd linear
##
##            family     link transformation
## gr           <NA>     <NA>           <NA>
## age          <NA>     <NA>    standardize
## race         <NA>     <NA>           <NA>
## etoh     binomial    logit           <NA>
## smoke    binomial    logit           <NA>
## dementia binomial    logit           <NA>
## Antiseiz binomial    logit           <NA>
## LevoT4   binomial    logit           <NA>
## AntiChol binomial    logit           <NA>
## albumin  gaussian identity    standardize
## bmi      gaussian identity    standardize
## lhgb     gaussian identity    standardize
```

```
#create multiply imputed data sets
IMP<-mi(sb)

#analysis
mi.fit=pool(gr~etoh+smoke+dementia+Antiseiz+LevoT4+AntiChol+albumin+bmi+lhgb,IMP,family=binomial(link="
display(mi.fit)
```

```
## bayesglm(formula = gr ~ etoh + smoke + dementia + Antiseiz +
##     LevoT4 + AntiChol + albumin + bmi + lhgb, data = IMP, family = binomial(link = "logit"))
##             coef.est coef.se
## (Intercept) 11.08     2.80
## etoh1        1.27     0.31
## smoke1       0.61     0.34
## dementia1    1.45     0.46
## Antiseiz1    2.31     0.56
## LevoT41      1.16     0.77
## AntiChol1   -0.90     0.43
## albumin     -0.88     0.24
## bmi         -0.12     0.03
## lhgb        -2.44     1.19
## n = 426, k = 10
## residual deviance = 387.5, null deviance = 604.4 (difference = 216.9)
```

```
summary(mi.fit)
```

```
##
## Call:
## pool(formula = gr ~ etoh + smoke + dementia + Antiseiz + LevoT4 +
##     AntiChol + albumin + bmi + lhgb, data = IMP, family = binomial(link = "logit"))
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.44267  -0.69750  -0.01373   0.75856   1.95597
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  11.0806     2.7961   3.963 7.40e-05 ***
## etoh1         1.2750     0.3072   4.151 3.32e-05 ***
## smoke1        0.6103     0.3395   1.798 0.072217 .
## dementia1     1.4468     0.4611   3.138 0.001703 **
## Antiseiz1     2.3143     0.5608   4.127 3.68e-05 ***
## LevoT41       1.1600     0.7731   1.500 0.133503
## AntiChol1    -0.8967     0.4310  -2.080 0.037483 *
## albumin      -0.8844     0.2443  -3.620 0.000295 ***
## bmi          -0.1179     0.0298  -3.956 7.62e-05 ***
## lhgb         -2.4390     1.1896  -2.050 0.040337 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 604.42  on 435  degrees of freedom
## Residual deviance: 387.48  on 426  degrees of freedom
## AIC: 407.48
##
## Number of Fisher Scoring iterations: 8.25
```

(d)

```
#Multiple imputation via sequential regression tree
g1=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g2=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g3=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g4=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g5=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g6=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g7=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g8=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g9=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g10=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g11=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)
g12=treeMI(data.frame(sb), ITER = 30, factorvar = c(1,0,1,1,1,1,1,1,1,0,0,0), minCut=5)


all<-imputationList(list(g1,g2,g3,g4,g5,g6,g7,g8,g9)) #combining the imputation
model1<-with(all,glm(gr~etoh+smoke+dementia+Antiseiz+LevoT4+AntiChol+albumin+bmi+lhgb,family=binomial))
summary(MIcombine(model1))
```

```
## Multiple imputation results:
##       with(all, glm(gr ~ etoh + smoke + dementia + Antiseiz + LevoT4 +
##     AntiChol + albumin + bmi + lhgb, family = binomial))
##       MIcombine.default(model1)
##                  results         se      (lower      upper) missInfo
## (Intercept) 12.82074489 2.29286120  8.31551081 17.32597896     13 %
## etoh1        1.19292217 0.28863063  0.62625642  1.75958792     11 %
## smoke1       0.57432216 0.30847151 -0.03409201  1.18273634     21 %
## dementia1    1.61269154 0.46000222  0.71079643  2.51458665      5 %
## Antiseiz1    2.50757055 0.60738611  1.31687384  3.69826725      4 %
## LevoT41      0.77970326 0.66210776 -0.53515777  2.09456429     31 %
## AntiChol1   -1.54722875 0.53391808 -2.59965944 -0.49479806     20 %
## albumin     -1.01784863 0.30256787 -1.62240723 -0.41329002     37 %
## bmi         -0.09973761 0.02815007 -0.15492956 -0.04454567      5 %
## lhgb        -3.05727575 0.92105042 -4.86872209 -1.24582942     16 %
```

Advantage of using MI with CART: Chained equation assumes that each variable has a joint distribution with the rest of the variables which is not always true and logical. Also, non-linear relationships can not be easily represented in the model. CART uses a tree structure to represent the dependencies among variables and thus solves the non-linearity problem as well. MI with CART generally shows smaller root mean squred error and bias.

(e) Comparing results from b,c,d complete chained CART

(Intercept) 10.85508 10.37845 12.53604075

etoh 1.39093 1.23003 1.09164551

smoke 0.92920 0.61175 0.55085822

dementia 2.50919 1.47472 1.52378169

Antiseiz 3.31056 2.37072 2.45534543

LevoT4 2.01009 0.89548 0.80360954

AntiChol -1.91833 -0.70482 -1.43958551

albumin -0.91116 -0.90528 -0.89038490

bmi -0.10416 -0.12159 -0.09883467

lhgb -2.59693 -2.07914 -3.11148691

The estimates using MI are generally smaller (closer to zero) than using complete case analysis. I think this could be due to ignoring all missing entries in complete case analysis, as now the imputed estimates accounted for the missing data, the estimates became less biased. The difference between estimates using CART and complete case is larger than that between estimates using chained equation and complete case. Based on the assumptions made in chained equation and CART, results from CART are probabily closer to reality.