```
##########################################
###### Classical Measurement Error Regression Calibration Functionss
###### used for biomarker subset only
##########################################
RCalib<-function(W){
  #### W is the validation subset
  ### it has two columns indicating biomarker measurement for a particular nutrient
  ### at the first and second visits, sorted by id

  ### Within person mean
  Wsubjm<-apply(W,1,mean,na.rm=T)
  ### Population mean
  mW<-mZ<-mean(Wsubjm,na.rm=T)

  ### Variance of random error in W: var(Epsilon)
  relW<-!is.na(W[,1]) & !is.na(W[,2])
  varEps.hat<-mean(apply(W[relW,],1,var))

  ki<-ifelse(relW,2,1)
  ki<-ifelse(is.na(Wsubjm),NA,ki)
  varZ.hat<- var(Wsubjm,na.rm=T) - mean(varEps.hat/ki,na.rm=T)

#### ordinary Regression calibration: classical error model and no covariates
  Zstar<- mZ + varZ.hat/(varZ.hat + varEps.hat/ki)* (Wsubjm - mW)
  return(Zstar)
 }
```

```
###### Function for generating calibration equation and applying to the FFQ data ####
RC<-function(data,dat.val){
##  dat.val:  the validation set for generating the calibration equation, nested within the study
##  cohort
##  data:  the study cohort with FFQ information. Variable ‘innbs’ equals 1 for a subject also in
## the validation subset. A subject's nutrient consumption is imputed differently depending on
## whether or not one's biomarker value is observed.

### center intake variables for regression calibration
data$cage<-data$ffqage- mean(dat.val$age[dat.val$visit==1],na.rm=T)
data$cbmi<-data$ffqbmi-mean(dat.val$bmi[dat.val$visit==1],na.rm=T)
data$clogffqE<-data$logenrgy - mean(log(dat.val$ffqtee[dat.val$visit==1]),na.rm=T)
data$clogffqP<-data$logprot - mean(log(dat.val$ffqprot[dat.val$visit==1]),na.rm=T)
data$clogffqD<-data$logpep - mean(log(dat.val$ffqpep[dat.val$visit==1]),na.rm=T)

### Coefficients
dat.val$patno<-dat.val$comid
dat.val$cage<-dat.val$age- mean(dat.val$age[dat.val$visit==1],na.rm=T)
dat.val$cbmi<-dat.val$bmi-mean(dat.val$bmi[dat.val$visit==1],na.rm=T)
dat.val$clogffqE<-log(dat.val$ffqtee) - mean(log(dat.val$ffqtee[dat.val$visit==1]),na.rm=T)
dat.val$clogffqP<-log(dat.val$ffqprot) - mean(log(dat.val$ffqprot[dat.val$visit==1]),na.rm=T)
dat.val$clogffqD<-log(dat.val$ffqpep) - mean(log(dat.val$ffqpep[dat.val$visit==1]),na.rm=T)

#### (i) apply calibration equation to subjects not in the validation subset ###
##########################################
### Calibration equation for ENERGY
```

```
logfit<-gee(log(teechrt)~ clogffqE+cbmi+cage+black+hispanic+other+texpwk+income2+income3+income5+income6,id=patno,data=dat.val)

summary(logfit)

t4coeff<-coef(logfit)

### Impute ENERGY

V<-cbind(rep(1,length(data$ffqage)),data$clogffqE,

data$cbmi,data$cage,data$black,data$hispanic,data$other,data$texpwk,data$income2,data$income3,data$income5,data$income6)

data$zhat<-t(matrix(as.numeric(t4coeff), nrow=1) %*% t(V))


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

### Calibration equation for PROTEIN

logfit<-gee(log(biopraw)~ clogffqP*cbmi+cage+black+hispanic+other+educ3+educ5+income2+income3+income5+income6,id=patno,data=dat.val)

summary(logfit)

t4coeff<-coef(logfit)

### Impute PROTEIN

V<-cbind(rep(1,length(data$ffqage)),data$clogffqP,

data$cbmi,data$cage,data$black,data$hispanic,data$other,data$educ3,data$educ5,data$income2,data$income3,data$income5,data$income6,data$clogffqP*data$cbmi)

data$pzhat<-t(matrix(as.numeric(t4coeff), nrow=1) %*% t(V))


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

### Calibration equation for PERCENT ENERGY FROM PROTEIN

logfit<-gee(log(unpepC)~ clogffqD+cbmi+cage+smoke,id=patno,data=dat.val)

summary(logfit)

t4coeff<-coef(logfit)

### Impute PERCENT ENERGY FROM PROTEIN ###
```

```
V<-cbind(rep(1,length(data$ffqage)),data$clogffqD, data$cbmi,data$cage,data$smoke)

data$pepzhat<-t(matrix(as.numeric(t4coeff), nrow=1) %*% t(V))


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

##### (ii) Imputing nutrients for subjects with biomarker measurement based on classical measurement error model

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

tmp1<-dat.val[dat.val$visit==1,]

tmp2<-dat.val[dat.val$visit==3,]


#########

##Energy

#########

W<-cbind(tmp1$teechrt,rep(NA,nrow(tmp1)))

W[tmp1$relstudy==1 & tmp1$dmdisp=="Comparison",2]<-tmp2$teechrt[tmp2$relstudy==1 & tmp2$dmdisp=="Comparison"]

W[tmp1$relstudy==1 & tmp1$dmdisp!="Comparison",2]<-tmp2$teechrt[tmp2$relstudy==1 & tmp2$dmdisp!="Comparison"]

W<-log(W)

Zstar<-RCalib(W)

data$zhat[data$innbs==1]<-Zstar[tmp1$dmdisp=="Comparison"]


#########

#protein

#########

W<-cbind(tmp1$biopraw,rep(NA,nrow(tmp1)))

W[tmp1$relstudy==1 & tmp1$dmdisp=="Comparison",2]<-tmp2$biopraw[tmp2$relstudy==1 & tmp2$dmdisp=="Comparison"]
```

```
W[tmp1$relstudy==1 & tmp1$dmdisp!="Comparison",2]<-tmp2$biopraw[tmp2$relstudy==1 & tmp2$dmdisp!="Comparison"]
W<-log(W)
Zstar<-RCalib(W)
data$pzhat[data$innbs==1]<-Zstar[tmp1$dmdisp=="Comparison"]


#############
#protein density
#############
W<-cbind(tmp1$unpepC,rep(NA,nrow(tmp1)))
W[tmp1$relstudy==1 & tmp1$dmdisp=="Comparison",2]<-tmp2$unpepC[tmp2$relstudy==1 & tmp2$dmdisp=="Comparison"]
W[tmp1$relstudy==1 & tmp1$dmdisp!="Comparison",2]<-tmp2$unpepC[tmp2$relstudy==1 & tmp2$dmdisp!="Comparison"]
W<-log(W)
Zstar<-RCalib(W)
data$pepzhat[data$innbs==1]<-Zstar[tmp1$dmdisp=="Comparison"]
return(data)
}


removeNA<-function(x){
return(x[!is.na(x)])
}


#### Example of doing stratified bootstrap #################
### id.matrix contains sample id and strata information for the whole study cohort
### data is the cohort data
### dat.val is the nutritional biomarker data, observed at visit 1 or at both visit 1 and visit 3
```

```
> names(id.matrix)

[1] "strata" "comid"  "innbs"


### resampling with strata  ##################################################

comid<-tapply(id.matrix$comid,id.matrix$strata,sample,replace=T)

comid<-unlist(comid)


#### resampling cohort data #####################

data.boot<-data[removeNA(match(comid,data$comid)),]

### resampling biomarker data ###################

dat.val.boot<-dat.val[c(removeNA(match(comid,dat.val$comid*(dat.val$visit==1))),

removeNA(match(comid,dat.val$comid*(dat.val$visit==3)))),]


######################## order data.boot and dat.val.boot #########################3

oo<-order(dat.val.boot$dmdisp,dat.val.boot$relstudy,dat.val.boot$comid)

dat.val.boot<-dat.val.boot[oo,]

oo<-order(data.boot$innbs,data.boot$relstudy,data.boot$comid)

data.boot<-data.boot[oo,]


################# perform regression calibration ###################

data.boot<-RC(data.boot,dat.val.boot)

##### fit cox regression model ##########################

fit<-coxph( Surv(randyr, Event) ~ zhat+texpwk+factor(ethnr)+factor(tabedu)+

factor(premmi)+factor(smoke)+factor(htn)+factor(diabtrt)+factor(statin)+factor(asa80mg)+facto
r(tothstat)+ strata(trial)+strata(ffqagecat),data=data.boot)
```