Sample Code - Guide

- "Sample Code - Mean Models" - Provides an example of the code used to generate the means of each outcome and each survey using imputation and accounting for survey designs (PROC Surveyreg). It then outputs the data necessary to run the metan() program in Stata and to generate linear trend and relative change (% change) tables.

- "Linear Trend and Relative Change Analyses.R" - Provides entire code used for linear trend statistics (2-stage regression) and for relative change tables with 95% CI (calculated using delta method). Structure and head() of data are provided in the code as an example.

- "Sample Code - Forest Plots.do" - Provides a sample of the code used to generate the forest plots using metan() in Stata.

"Sample Code - Connected Plots.do" - Provides a sample of the code used to generate the connected plots that correspond with the forest plots.

Sample Code - Mean Models

```
%macro sr_others(dat=, var=,byvar=,stravar=, clustvar=,wvar=, eligible=) ;
proc sort data=&dat._mrged2; by _imputation_; run;

proc surveyreg data=&dat._mrged2;
strata &stravar.;
cluster &clustvar.;
weight &wvar.;
model &var. = ageyrs gender &byvar. / solution clparm i;
domain &eligible.;
ods output ParameterEstimates = b1
           invxpx=c1;
by _imputation_;
run;

data b11;
set b1;
where &eligible.=1;
run;

data c11;
set c1;
where domain ne "";
run;

proc mianalyze parms=b11 xpxi=c11 ;
modeleffects Intercept ageyrs gender &byvar;
ods output parameterestimates=b2;
run;

data b3;
set b2  (keep = parm estimate stderr);
where parm in ("Intercept", "inc20", "risk_p", "nocollege") ;
run;

data b31 ; set b3 (keep = parm estimate); where parm="Intercept"; rename
estimate=intercept; junk=1; run;
data b32; set b3; where parm ne "Intercept"; junk=1; run;
proc sort data=b31; by junk; run;
proc sort data=b32; by junk; run;
data b4 (drop = junk); merge b31 b32; by junk; run;

data out1; set out1 b4; run;
%mend sr_others;

%sr_others(dat=nhanes1,var=bmi,byvar=nocollege,stravar=N1BM0194,clustvar=N1BM0196,wv
ar=N1BM0176, eligible=blackadult) ;
```

Linear Trend and Relative Change Analyses.R

```
# This code takes the output of the mean models and tests linear trend (using
# 2 stage regression) and calculates the relative change as well as the variance
# of the relative change (using multivariate delta method) to get the 95%
# confidence interval of that change.

# These are imported with the general structure and head() as follows:
# > head(ad.perc, n=3)
# study  race MD_bmi_beta0 MD_bmi_beta MD_bmi_SE MD_waistcm_beta0
# 1     NHESI white        22.90    -0.03734    0.2731            91.23
# 2   NHANESI white        24.61     0.28004    0.2893               NA
# 3 NHANIESII white        24.19     0.66193    0.2681               NA
# MD_waistcm_beta MD_waistcm_SE MD_sbp_beta0 MD_sbp_beta MD_sbp_SE
# 1          0.7109        0.7062        93.74       4.054     1.304
# 2              NA            NA       102.07       5.733     1.123
# 3              NA            NA       107.45       3.478     1.037
# MD_serum_beta0 MD_serum_beta MD_serum_SE MD_menarche_beta0
# 1          157.1        -7.977       2.807             12.11
# 2          165.5        -7.963       2.958             11.63
# 3          156.2        -6.815       2.521             11.84
# MD_menarche_beta MD_menarche_SE
# 1          0.26221         0.1780
# 2          0.23952         0.1593
# 3          0.09116         0.1070

# > str(ad.perc)
# 'data.frame':    12 obs. of  17 variables:
#     $ study           : Factor w/ 6 levels "NHANES0508","NHANES9904",..: 6 3 5 4
2 1 6 3 5 4 ...
# $ race            : Factor w/ 2 levels "black","white": 2 2 2 2 2 2 1 1 1 1 ...
# $ MD_bmi_beta0    : num  22.9 24.6 24.2 25 25.9 ...
# $ MD_bmi_beta     : num  -0.0373 0.28 0.6619 0.8477 0.959 ...
# $ MD_bmi_SE       : num  0.273 0.289 0.268 0.359 0.346 ...
# $ MD_waistcm_beta0 : num  91.2 NA NA 92 97.1 ...
# $ MD_waistcm_beta  : num  0.711 NA NA 3.017 3.552 ...
# $ MD_waistcm_SE    : num  0.706 NA NA 0.925 0.867 ...
# $ MD_sbp_beta0    : num  93.7 102.1 107.4 102.4 98.5 ...
# $ MD_sbp_beta     : num  4.05 5.73 3.48 2.21 3.16 ...
# $ MD_sbp_SE       : num  1.304 1.123 1.037 0.846 0.71 ...
# $ MD_serum_beta0   : num  157 166 156 159 178 ...
# $ MD_serum_beta    : num  -7.977 -7.963 -6.815 -0.402 3.661 ...
# $ MD_serum_SE      : num  2.81 2.96 2.52 2.86 1.72 ...
# $ MD_menarche_beta0: num  12.1 11.6 11.8 12.5 12.7 ...
# $ MD_menarche_beta : num  0.2622 0.2395 0.0912 -0.0748 0.0904 ...
# $ MD_menarche_SE   : num  0.178 0.1593 0.107 0.122 0.0955 ...

#   Make sure you have the xlsx package.
require(xlsx); require(lmtest)

### Import data
ad.perc <- read.xlsx("betas_and_SEs_Dec10.xlsx", 1, endRow=13)
ad.edu <- read.xlsx("betas_and_SEs_Dec10.xlsx", 2, endRow=13)
kids.perc <- read.xlsx("betas_and_SEs_Dec10.xlsx", 3, endRow=15)

## Rename the columns--remove MD_
dimnames(ad.perc)[[2]][3:17] <- substring(names(ad.perc[3:17]), 4)
dimnames(ad.edu)[[2]][3:17] <- substring(names(ad.edu[3:17]), 4)
dimnames(kids.perc)[[2]][3:8] <- substring(names(kids.perc[3:8]), 4)

## Add years
ad.midpoint <- c(1961, 1972.5, 1978, 1991.5, 2002, 2007)
kids.midpoint <- c(1964.5, 1968, 1972.5, 1978, 1991.5, 2002, 2007)
ad.edu$year <- ad.perc$year <- ad.midpoint
kids.perc$year <- kids.midpoint
```

Linear Trend and Relative Change Analyses.R

```
############## RELATIVE CHANGE ANALYSES ##########################

### Import var-cov matrices. Example head() and structure:
# > head(vc.edu, n=2)
# Model       Group  Variable Intercept nocollege study
# 1   BMI black adult Intercept    1.7073   -0.7572 nhes1
# 2   BMI black adult nocollege   -0.7572    0.6480 nhes1

# > str(vc.edu)
# 'data.frame':    112 obs. of  6 variables:
#     $ Model    : Factor w/ 5 levels "age at menarche",..: 2 2 2 2 2 2 2 2 2 2 ...
# $ Group    : Factor w/ 2 levels "black adult",..: 1 1 1 1 1 1 1 1 1 1 ...
# $ Variable : Factor w/ 2 levels "Intercept","nocollege": 1 2 1 2 1 2 1 2 1 2 ...
# $ Intercept: num  1.7073 -0.7572 1.4272 -0.1235 0.0559 ...
# $ nocollege: num  -0.7572 0.648 -0.1235 0.2397 -0.0197 ...
# $ study    : Factor w/ 6 levels "nhanes05_08",..: 6 6 2 2 3 3 4 4 5 5 ...

vc.edu  <- read.xlsx("covariances_dec10.xlsx", 1, endRow=113)
vc.edu[vc.edu=="."] <- NA
vc.perc <- read.xlsx("covariances_dec10.xlsx", 2, endRow=113)
vc.perc[vc.perc=="."] <- NA
vc.kids <- read.xlsx("covariances_dec10.xlsx", 3, endRow=49)
vc.kids[vc.kids=="."] <- NA

## Change factor variables to numeric.
vc.edu$Intercept <- as.numeric(as.character(vc.edu$Intercept))
vc.edu$nocollege <- as.numeric(as.character(vc.edu$nocollege))
vc.perc$Intercept <- as.numeric(as.character(vc.perc$Intercept))
vc.perc$inc20 <- as.numeric(as.character(vc.perc$inc20))
vc.kids$Intercept <- as.numeric(as.character(vc.kids$Intercept))
vc.kids$inc20 <- as.numeric(as.character(vc.kids$inc20))

## Modify data so that varB0, varB1, and covarB0B1 are all on one line.
vc.edu$varB1 <- rep(vc.edu[c(FALSE, TRUE), 5], each=2)
vc.edu <- vc.edu[c(TRUE, FALSE), -3]
vc.perc$varB1 <- rep(vc.perc[c(FALSE, TRUE), 5], each=2)
vc.perc <- vc.perc[c(TRUE, FALSE), -3]
vc.kids$varB1 <- rep(vc.kids[c(FALSE, TRUE), 5], each=2)
vc.kids <- vc.kids[c(TRUE, FALSE), -3]

## Rename columns so they make sense
dimnames(vc.edu)[[2]][1:4] <- c("outcome", "race", "varB0", "covarB0B1")
dimnames(vc.perc)[[2]][1:4] <- c("outcome", "race", "varB0", "covarB0B1")
dimnames(vc.kids)[[2]][1:4] <- c("outcome", "race", "varB0", "covarB0B1")

## Change strings so we can match them later. Note the misspelling of NHANESII
## was in the original data. Keep it for simplicity.
levels(vc.edu$outcome) <- c("menarche", "bmi", "sbp", "serum", "waistcm")
levels(vc.perc$outcome) <- c("menarche", "bmi", "sbp", "serum", "waistcm")
levels(vc.kids$outcome) <- c("menarche", "height")
levels(vc.edu$race) <- c("black", "white")
levels(vc.perc$race) <- c("black", "white")
levels(vc.perc$study) <- c("NHANES0508", "NHANESI", "NHANIESII", "NHANESIII",
                           "NHANES9904", "NHESI")
levels(vc.edu$study) <- c("NHANES0508", "NHANESI", "NHANIESII", "NHANESIII",
                          "NHANES9904", "NHESI")
levels(vc.kids$outcome) <- c("menarche", "height")
levels(vc.kids$race) <- c("black", "white")
levels(vc.kids$study) <- c("NHANES0508", "NHANESI", "NHANIESII", "NHANESIII",
                           "NHANES9904", "NHESII", "NHESIII")

### Reshape the data into something usable.
```

Linear Trend and Relative Change Analyses.R (cont.)

```
ad.edu.beta0 <- stack(ad.edu[, c(3, 6, 9, 12, 15)])
ad.edu.beta1 <- stack(ad.edu[, c(4, 7, 10, 13, 16)])
ad.edu2 <- data.frame(study=rep(ad.edu$study, 5), race=rep(ad.edu$race, 5),
                      year=rep(ad.edu$year, 5),
                      outcome=rep(c("bmi", "waistcm", "sbp", "serum", "menarche"),
each=12))
ad.edu2$beta0 <- ad.edu.beta0$values
ad.edu2$beta1 <- ad.edu.beta1$values

ad.perc.beta0 <- stack(ad.perc[, c(3, 6, 9, 12, 15)])
ad.perc.beta1 <- stack(ad.perc[, c(4, 7, 10, 13, 16)])
ad.perc2 <- data.frame(study=rep(ad.perc$study, 5), race=rep(ad.perc$race, 5),
                      year=rep(ad.perc$year, 5),
                      outcome=rep(c("bmi", "waistcm", "sbp", "serum", "menarche"),
each=12))
ad.perc2$beta0 <- ad.perc.beta0$values
ad.perc2$beta1 <- ad.perc.beta1$values


kids.beta0 <- stack(kids.perc[, c(3, 6)])
kids.beta1 <- stack(kids.perc[, c(4, 7)])
kids.perc2 <- data.frame(study=rep(kids.perc$study, 2), race=rep(kids.perc$race, 2),

                         year=rep(kids.perc$year, 2),
                         outcome=rep(c("height", "menarche"), each=14))
kids.perc2$beta0 <- kids.beta0$values
kids.perc2$beta1 <- kids.beta1$values

### Merge them
merged.edu <- merge(ad.edu2, vc.edu, by=c("outcome", "race", "study"), all=TRUE)
merged.perc <- merge(ad.perc2, vc.perc, by=c("outcome", "race", "study"), all=TRUE)
merged.kids <- merge(kids.perc2, vc.kids, by=c("outcome", "race", "study"),
all=TRUE)

merged.edu$percchange <- merged.edu$beta1 / merged.edu$beta0
merged.perc$percchange <- merged.perc$beta1 / merged.perc$beta0
merged.kids$percchange <- merged.kids$beta1 / merged.kids$beta0

## Now use the delta method
deltamethod <- function(x) {
    beta0 <- as.numeric(x[5])
    beta1 <- as.numeric(x[6])

    phi <- c(-1 * (beta1 / beta0^2), 1/beta0)
    sigma <- matrix(c(as.numeric(x[7]), as.numeric(x[8]),
                      as.numeric(x[8]), as.numeric(x[9])), nrow=2)

    varPERCchange <- t(phi) %*% sigma %*% phi

    return(varPERCchange)
}

## Apply it
merged.edu$vardelta <- apply(merged.edu, 1, deltamethod)
merged.perc$vardelta <- apply(merged.perc, 1, deltamethod)
merged.kids$vardelta <- apply(merged.kids, 1, deltamethod)

## Calculate 95% CI with it
merged.edu$lower <- merged.edu$percchange - 1.96 * sqrt(merged.edu$vardelta)
merged.edu$upper <- merged.edu$percchange + 1.96 * sqrt(merged.edu$vardelta)
merged.perc$lower <- merged.perc$percchange - 1.96 * sqrt(merged.perc$vardelta)
merged.perc$upper <- merged.perc$percchange + 1.96 * sqrt(merged.perc$vardelta)
merged.kids$lower <- merged.kids$percchange - 1.96 * sqrt(merged.kids$vardelta)
```

```
merged.kids$upper <- merged.kids$percchange + 1.96 * sqrt(merged.kids$vardelta)

## Export tables
write.csv(merged.edu, "Relative Change - Education.csv", row.names=FALSE, na="")
write.csv(merged.perc, "Relative Change - Percentile.csv", row.names=FALSE, na="")
write.csv(merged.kids, "Relative Change - Kids.csv", row.names=FALSE, na="")

############# LINEAR TREND ANALYSES ########################
### Run regressions
###     THIS IS FOR EDUCATION
# Make a little holder for results
results.ed <- data.frame(race=c(rep("White", 10), rep("Black", 10)))
results.ed$model <- rep(c("Linear (Edu)", "Quadratic (Edu)"), 10)
results.ed$outcome <- rep(c("BMI", "Waist", "SBP", "Serum", "Menarche"), each=2)
results.ed$int <- results.ed$time <- results.ed$timese <- results.ed$time2 <-
    results.ed$time2se <- results.ed$timep <- results.ed$time2p <-
    results.ed$LR <- results.ed$adjR2 <- NA

# Run through regressions and collect results for WHITES
index <- -1
for (y in c(4, 7, 10, 13, 16)){
    index <- index + 2
    regress1 <- lm(data=ad.edu[1:6, ], ad.edu[1:6, y] ~ year,
                   weights=(1/ad.edu[1:6, y+1]))
    regress2 <- lm(data=ad.edu[1:6, ], ad.edu[1:6, y] ~ year + I(year^2),
                   weights=(1/ad.edu[1:6, y+1]))
    results.ed$time[index] <- summary(regress1)$coefficients[2]
    results.ed$timese[index] <- summary(regress1)$coefficients[4]
    results.ed$timep[index] <- summary(regress1)$coefficients[8]
    results.ed$adjR2[index] <- summary(regress1)$adj.r.squared
    results.ed$int[index] <- summary(regress1)$coefficients[1]
    results.ed$int[index+1] <- summary(regress2)$coefficients[1]
    results.ed$time[index+1] <- summary(regress2)$coefficients[2]
    results.ed$timese[index+1] <- summary(regress2)$coefficients[5]
    results.ed$timep[index+1] <- summary(regress2)$coefficients[11]
    results.ed$time2[index+1] <- summary(regress2)$coefficients[3]
    results.ed$time2se[index+1] <- summary(regress2)$coefficients[6]
    results.ed$time2p[index+1] <- summary(regress2)$coefficients[12]
    results.ed$LR[index+1] <- lrtest(regress1, regress2)[2, 5]
    results.ed$adjR2[index+1] <- summary(regress2)$adj.r.squared
}

# Now regressions for BLACKS
index <- 9
for (y in c(4, 7, 10, 13, 16)){
    index <- index + 2
    regress1 <- lm(data=ad.edu[7:12, ], ad.edu[7:12, y] ~ year,
                   weights=(1/ad.edu[7:12, y+1]))
    regress2 <- lm(data=ad.edu[7:12, ], ad.edu[7:12, y] ~ year + I(year^2),
                   weights=(1/ad.edu[7:12, y+1]))
    results.ed$time[index] <- summary(regress1)$coefficients[2]
    results.ed$timese[index] <- summary(regress1)$coefficients[4]
    results.ed$timep[index] <- summary(regress1)$coefficients[8]
    results.ed$adjR2[index] <- summary(regress1)$adj.r.squared
    results.ed$int[index] <- summary(regress1)$coefficients[1]
    results.ed$int[index+1] <- summary(regress2)$coefficients[1]
    results.ed$time[index+1] <- summary(regress2)$coefficients[2]
    results.ed$timese[index+1] <- summary(regress2)$coefficients[5]
    results.ed$timep[index+1] <- summary(regress2)$coefficients[11]
    results.ed$time2[index+1] <- summary(regress2)$coefficients[3]
    results.ed$time2se[index+1] <- summary(regress2)$coefficients[6]
    results.ed$time2p[index+1] <- summary(regress2)$coefficients[12]
    results.ed$LR[index+1] <- lrtest(regress1, regress2)[2, 5]
```

```
    results.ed$adjR2[index+1] <- summary(regress2)$adj.r.squared
}

###     THIS IS FOR PERCENTILE POVERTY
# Make a little holder for results
results.perc <- data.frame(race=c(rep("White", 10), rep("Black", 10)))
results.perc$model <- rep(c("Linear (%)", "Quadratic (%)"), 10)
results.perc$outcome <- rep(c("BMI", "Waist", "SBP", "Serum", "Menarche"), each=2)
results.perc$int <- results.perc$time <- results.perc$timese <-
    results.perc$time2 <- results.perc$time2se <- results.perc$timep <-
    results.perc$time2p <- results.perc$LR <- results.perc$adjR2 <- NA

# Run through regressions and collect results for WHITES
index <- -1
for (y in c(4, 7, 10, 13, 16)){
    index <- index + 2
    regress1 <- lm(data=ad.perc[1:6, ], ad.perc[1:6, y] ~ year,
                   weights=(1/ad.perc[1:6, y+1]))
    regress2 <- lm(data=ad.perc[1:6, ], ad.perc[1:6, y] ~ year + I(year^2),
                   weights=(1/ad.perc[1:6, y+1]))
    results.perc$time[index] <- summary(regress1)$coefficients[2]
    results.perc$timese[index] <- summary(regress1)$coefficients[4]
    results.perc$timep[index] <- summary(regress1)$coefficients[8]
    results.perc$adjR2[index] <- summary(regress1)$adj.r.squared
    results.perc$int[index] <- summary(regress1)$coefficients[1]
    results.perc$int[index+1] <- summary(regress2)$coefficients[1]
    results.perc$time[index+1] <- summary(regress2)$coefficients[2]
    results.perc$timese[index+1] <- summary(regress2)$coefficients[5]
    results.perc$timep[index+1] <- summary(regress2)$coefficients[11]
    results.perc$time2[index+1] <- summary(regress2)$coefficients[3]
    results.perc$time2se[index+1] <- summary(regress2)$coefficients[6]
    results.perc$time2p[index+1] <- summary(regress2)$coefficients[12]
    results.perc$LR[index+1] <- lrtest(regress1, regress2)[2, 5]
    results.perc$adjR2[index+1] <- summary(regress2)$adj.r.squared
}

# Now regressions for BLACKS
index <- 9
for (y in c(4, 7, 10, 13, 16)){
    index <- index + 2
    regress1 <- lm(data=ad.perc[7:12, ], ad.perc[7:12, y] ~ year,
                   weights=(1/ad.perc[7:12, y+1]))
    regress2 <- lm(data=ad.perc[7:12, ], ad.perc[7:12, y] ~ year + I(year^2),
                   weights=(1/ad.perc[7:12, y+1]))
    results.perc$time[index] <- summary(regress1)$coefficients[2]
    results.perc$timese[index] <- summary(regress1)$coefficients[4]
    results.perc$timep[index] <- summary(regress1)$coefficients[8]
    results.perc$adjR2[index] <- summary(regress1)$adj.r.squared
    results.perc$int[index] <- summary(regress1)$coefficients[1]
    results.perc$int[index+1] <- summary(regress2)$coefficients[1]
    results.perc$time[index+1] <- summary(regress2)$coefficients[2]
    results.perc$timese[index+1] <- summary(regress2)$coefficients[5]
    results.perc$timep[index+1] <- summary(regress2)$coefficients[11]
    results.perc$time2[index+1] <- summary(regress2)$coefficients[3]
    results.perc$time2se[index+1] <- summary(regress2)$coefficients[6]
    results.perc$time2p[index+1] <- summary(regress2)$coefficients[12]
    results.perc$LR[index+1] <- lrtest(regress1, regress2)[2, 5]
    results.perc$adjR2[index+1] <- summary(regress2)$adj.r.squared
}

###     THIS IS FOR CHILDREN
# Make a little holder for results
results.kids <- data.frame(race=c(rep("White", 4), rep("Black", 4)))
```

Linear Trend and Relative Change Analyses.R (cont.)

```
results.kids$model <- rep(c("Linear", "Quadratic"), 4)
results.kids$outcome <- rep(c("(Children) Height 6", "(Children) Menarche"), each=2)
results.kids$int <- results.kids$time <- results.kids$timese <-
    results.kids$time2 <- results.kids$time2se <- results.kids$timep <-
    results.kids$time2p <- results.kids$LR <- results.kids$adjR2 <- NA

# Run through regressions and collect results for WHITE KIDS
index <- -1
for (y in c(4, 7)){
    index <- index + 2
    regress1 <- lm(data=kids.perc[1:7, ], kids.perc[1:7, y] ~ year,
                   weights=(1/kids.perc[1:7, y+1]))
    regress2 <- lm(data=kids.perc[1:7, ], kids.perc[1:7, y] ~ year + I(year^2),
                   weights=(1/kids.perc[1:7, y+1]))
    results.kids$time[index] <- summary(regress1)$coefficients[2]
    results.kids$timese[index] <- summary(regress1)$coefficients[4]
    results.kids$timep[index] <- summary(regress1)$coefficients[8]
    results.kids$adjR2[index] <- summary(regress1)$adj.r.squared
    results.kids$int[index] <- summary(regress1)$coefficients[1]
    results.kids$int[index+1] <- summary(regress2)$coefficients[1]
    results.kids$time[index+1] <- summary(regress2)$coefficients[2]
    results.kids$timese[index+1] <- summary(regress2)$coefficients[5]
    results.kids$timep[index+1] <- summary(regress2)$coefficients[11]
    results.kids$time2[index+1] <- summary(regress2)$coefficients[3]
    results.kids$time2se[index+1] <- summary(regress2)$coefficients[6]
    results.kids$time2p[index+1] <- summary(regress2)$coefficients[12]
    results.kids$LR[index+1] <- lrtest(regress1, regress2)[2, 5]
    results.kids$adjR2[index+1] <- summary(regress2)$adj.r.squared
}

# Now regressions for BLACK KIDS
index <- 3
for (y in c(4, 7)){
    index <- index + 2
    regress1 <- lm(data=kids.perc[8:14, ], kids.perc[8:14, y] ~ year,
                   weights=(1/kids.perc[8:14, y+1]))
    regress2 <- lm(data=kids.perc[8:14, ], kids.perc[8:14, y] ~ year + I(year^2),
                   weights=(1/kids.perc[8:14, y+1]))
    results.kids$time[index] <- summary(regress1)$coefficients[2]
    results.kids$timese[index] <- summary(regress1)$coefficients[4]
    results.kids$timep[index] <- summary(regress1)$coefficients[8]
    results.kids$adjR2[index] <- summary(regress1)$adj.r.squared
    results.kids$int[index] <- summary(regress1)$coefficients[1]
    results.kids$int[index+1] <- summary(regress2)$coefficients[1]
    results.kids$time[index+1] <- summary(regress2)$coefficients[2]
    results.kids$timese[index+1] <- summary(regress2)$coefficients[5]
    results.kids$timep[index+1] <- summary(regress2)$coefficients[11]
    results.kids$time2[index+1] <- summary(regress2)$coefficients[3]
    results.kids$time2se[index+1] <- summary(regress2)$coefficients[6]
    results.kids$time2p[index+1] <- summary(regress2)$coefficients[12]
    results.kids$LR[index+1] <- lrtest(regress1, regress2)[2, 5]
    results.kids$adjR2[index+1] <- summary(regress2)$adj.r.squared
}

### EXPORT (Quadratic was just for explatory purposes. Use linear trends.)
results.all <- rbind(results.ed, results.perc, results.kids)
results.all <- results.all[, c(1:3, 12:10, 7, 9:8, 6:4)]
write.csv(results.all, "Trend Analysis Results (All).csv", row.names=FALSE, na="")
results.linear <- results.all[is.na(results.all$time2), -c(8:11) ]
write.csv(results.linear, "Trend Analysis Results (Linear).csv", row.names=FALSE,
na="")
```

Sample Code - Forest Plots.do

```
// This is sample code using the metan() package in Stata to generate the
// forest plots. Since each forest plot used the same base code and only
// change the relevant variables per model/race/outcome, only one example
// is provided.

// Variables:
// MD_bmi is the calculated mean BMI
// MD_bmi_lci and MD_bmi_uci are lower and upper bounds of the 95% CI.
// min_bmi and max_bmi are the lowest and highest 95% CI bounds to keep all  axes
across outcomes the same.
metan MD_bmi MD_bmi_lci MD_bmi_uci if race=="white" & ses==0, random label(namevar =
study) nowt effect({&beta}) textsize(152) xtitle("Body Mass Index
(kg/m{superscript:2})") xlabel (min_bmi, 0, max_bmi) nowarning name(edbmiW,replace)
saving(edbmiW, replace) scale(1.25) graphregion(margin(small))
plotregion(lwidth(none))
```

Sample Code - Connected Plots.do

```
// The data are structured like so:
/*
  obs:           480
 vars:            12                          19 Jan 2013 13:07
 size:        14,400
-------------------------------------------------------------------------------
-------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
-------------------------------------------------
tablerow        float  %9.0g                  Row Number in Table 2
data            byte   %18.0g      Data       Data Source
outcome         byte   %25.0g      Outcome    Outcome of Interest
race            byte   %13.0g      Race       Race
inc             byte   %8.0g       Income     Income (%ile)
pov             byte   %24.0g      Poverty    At risk of poverty
edu             byte   %19.0g      Education
                                              Education
mean            float  %9.0g
sd              float  %9.0g
lower           float  %9.0g
upper           float  %9.0g
midpoint        float  %9.0g
-------------------------------------------------------------------------------
-------------------------------------------------
Sorted by:  tablerow  data
*/


// GROUP A SQ 1, BMI, Black Income
twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(medium)
lpattern(shortdash))            ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(medium)
lpattern(longdash))             ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(medium)
lpattern(solid))        ///
                if outcome==1 & race==2,                ///
                ytitle("BMI (kg/m{superscript:2})", size(medlarge) margin(tiny))
ylabel(25(2.5)32.5, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6
"80th") cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule))
/* title("Black by income percentile", size(medium)) */ name(A1, replace)
gr export "~/Income - BMI - Black2.eps", replace mag(150)

// GROUP A SQ 2,BMI, White Income
twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
```

Sample Code - Connected Plots.do (cont.)

```
        ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==1 & race==1,                ///
                ytitle("BMI (kg/m{superscript:2})", size(medlarge) margin(tiny))
ylabel(25(2.5)32.5, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6 "80th") cols(3)
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White
by income percentile", size(medium)) */ name(A2, replace)
gr export "Income - BMI - White.eps", replace

// GROUP A SQ 3, BMI, Black Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
        ///
                if outcome==1 & race==2,                ///
                ytitle("BMI (kg/m{superscript:2})", size(medlarge) margin(tiny))
ylabel(25(2.5)32.5, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("Black
by risk of poverty", size(medium)) */ name(A3, replace)
gr export "Poverty - BMI - Black.eps", replace

// GROUP A SQ 4, BMI, White Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==1 & race==1,                ///
                ytitle("BMI (kg/m{superscript:2})", size(medlarge) margin(tiny))
ylabel(25(2.5)32.5, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White
by risk of poverty", size(medium)) */ name(A4, replace)
gr export "Poverty - BMI - White.eps", replace

// GROUP A SQ 5, BMI, Black Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
```

Sample Code - Connected Plots.do (cont.)

```
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(med)
lpattern(longdash))             ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
///
                if outcome==1 & race==2,                ///
                ytitle("BMI (kg/m{superscript:2})", size(medlarge) margin(tiny))
ylabel(23.5(2)31.5, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y
College" 6 ">=4y College") cols(3) size(large) rowgap(minuscule) colgap(minuscule)
keygap(minuscule)) /* title("Black by education level", size(medium)) */ name(A5,
replace)
gr export "Education - BMI - Black.eps", replace

// GROUP A SQ 6, BMI, White Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==1 & race==1,                ///
                ytitle("BMI (kg/m{superscript:2})", size(medlarge) margin(tiny))
ylabel(23.5(2)31.5, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y College" 6 ">=4y
College") cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule))
/* title("White by education level", size(medium)) */ name(A6, replace)
gr export "Education - BMI - White.eps", replace

/* graph combine A1 A2 A3 A4 A5 A6, colfirst imargin(small) ycommon xcommon
title("Body Mass Index (BMI) of US-born black and white adults", size(medium))
name(ComA, replace) xsize(8.5) ysize(11) scale(.65)
gr export "Table 2 - Page 1 - BMI.eps", replace */



// GROUP B SQ 1, WC, Black Income
twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(med)
lpattern(longdash))             ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
                ///
                if outcome==2 & race==2,                                ///
```

Sample Code - Connected Plots.do (cont.)

```
                /* ytitle("Waist Circumference (cm)", size(medlarge) margin(tiny))
*/ ylabel(80(7.5)102.5, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6
"80th") cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule))
/* title("Black by income percentile", size(medium)) */ name(B1, replace)
gr export "Income - WC - Black.eps", replace

// GROUP B SQ 2, WC, White Income
twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==2 & race==1,                ///
                /* ytitle("Waist Circumference (cm)", size(medlarge) margin(tiny))
*/ ylabel(80(7.5)102.5, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6 "80th") cols(3)
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White
by income percentile", size(medium)) */ name(B2, replace)
gr export "Income - WC - White.eps", replace

// GROUP B SQ 3, WC, Black Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
        ///
                if outcome==2 & race==2,                ///
                /* ytitle("Waist Circumference (cm)"), size(medlarge) margin(tiny))
*/ ylabel(80(7.5)102.5, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("Black
by risk of poverty", size(medium)) */ name(B3, replace)
gr export "Poverty - WC - Black.eps", replace

// GROUP B SQ 4, WC, White Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==2 & race==1,                ///
```

Sample Code - Connected Plots.do (cont.)

```
                /* ytitle("Waist Circumference (cm)"), size(medlarge) margin(tiny))
*/ ylabel(80(7.5)102.5, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White
by risk of poverty", size(medium)) */ name(B4, replace)
gr export "Poverty - WC - White.eps", replace

// GROUP B SQ 5, WC, Black Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(med)
lpattern(longdash))             ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
///
                if outcome==2 & race==2,                ///
                /* ytitle("Waist Circumference (cm)"), size(medlarge) margin(tiny))
*/ ylabel(76(9)103, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y
College" 6 ">=4y College") cols(3) size(large) rowgap(minuscule) colgap(minuscule)
keygap(minuscule)) /* title("Black by education level", size(medium)) */ name(B5,
replace)
gr export "Education - WC - Black.eps", replace

// GROUP B SQ 6, WC, White Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==2 & race==1,                ///
                /* ytitle("Waist Circumference (cm)"), size(medlarge) margin(tiny))
*/ ylabel(76(9)103, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y College" 6 ">=4y
College") cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule))
/* title("White by education level", size(medium)) */ name(B6, replace)
gr export "Education - WC - White.eps", replace

/* graph combine B1 B2 B3 B4 B5 B6, colfirst imargin(small) ycommon xcommon
title("Waist circumference of US-born black and white adults", size(medium))
name(ComB, replace) xsize(8.5) ysize(11) scale(.65)
gr export "Table 2 - Page 2 - Waist Circumference.eps", replace  */


// GROUP C SQ 1, SC, Black Income
```

Sample Code - Connected Plots.do (cont.)

```
       twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(med)
lpattern(longdash))             ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
                ///
                if outcome==3 & race==2,                                ///

                /* ytitle("Serum Cholesterol (mg/dL)", size(medlarge) margin(tiny))
*/ ylabel(190(10)240, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6
"80th") cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule))
/* title("Black by income percentile", size(medium)) */ name(C1, replace)
gr export "Income - SC - Black.eps", replace

// GROUP C SQ 2, SC, White Income
twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==3 & race==1,                ///
                /* ytitle("Serum Cholesterol (mg/dL)", size(medlarge) margin(tiny))
*/ ylabel(190(10)240, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6 "80th") cols(3)
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White
by income percentile", size(medium)) */ name(C2, replace)
gr export "Income - SC - White.eps", replace

// GROUP C SQ 3, SC, Black Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
        ///
                if outcome==3 & race==2,                ///
                /* ytitle("Serum Cholesterol (mg/dL)"), size(medlarge) margin(tiny))
*/ ylabel(190(10)240, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
```

Sample Code - Connected Plots.do (cont.)

```
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("Black
by risk of poverty", size(medium)) */ name(C3, replace)
gr export "Poverty - SC - Black.eps", replace

// GROUP C SQ 4, SC, White Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==3 & race==1,                ///
                /* ytitle("Serum Cholesterol (mg/dL)"), size(medlarge) margin(tiny))
*/ ylabel(190(10)240, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White
by risk of poverty", size(medium)) */ name(C4, replace)
gr export "Poverty - SC - White.eps", replace

// GROUP C SQ 5, SC, Black Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(med)
lpattern(longdash))             ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
///
                if outcome==3 & race==2,                ///
                /* ytitle("Serum Cholesterol (mg/dL)"), size(medlarge) margin(tiny))
*/ ylabel(187.5(15)232.5, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y
College" 6 ">=4y College") cols(3) size(large) rowgap(minuscule) colgap(minuscule)
keygap(minuscule)) /* title("Black by education level", size(medium)) */ name(C5,
replace)
gr export "Education - SC - Black.eps", replace

// GROUP C SQ 6, SC, White Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///
```

Sample Code - Connected Plots.do (cont.)

```
                if outcome==3 & race==1,                ///
                /* ytitle("Serum Cholesterol (mg/dL)"), size(medlarge) margin(tiny))
*/ ylabel(187.5(15)232.5, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y College" 6 ">=4y
College") cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule))
/* title("White by education level", size(medium)) */ name(C6, replace)
gr export "Education - SC - White.eps", replace

/* graph combine C1 C2 C3 C4 C5 C6, colfirst imargin(small) ycommon xcommon
title("Serum cholesterol of US-born black and white adults", size(medium))
name(ComC, replace) xsize(8.5) ysize(11) scale(.65)
gr export "Table 2 - Page 3 - Serum Cholesterol.eps", replace */


// GROUP D SQ 1, SC, Black Income
twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(med)
lpattern(longdash))             ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
                ///
                if outcome==4 & race==2,                                ///

                /* ytitle("Age (years)", size(medlarge) margin(tiny)) */
ylabel(11(1)14, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6
"80th") cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule))
/* title("Black by income percentile", size(medium)) */ name(D1, replace)
gr export "Income - AoM - Black.eps", replace

// GROUP D SQ 2, SC, White Income
twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==4 & race==1,                ///
                /* ytitle("Age (years)", size(medlarge) margin(tiny)) */
ylabel(11(1)14, labsize(medium)) xtitle("Survey Midpoint", size(medlarge)) xlabel(,
labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6 "80th") cols(3) size(large)
rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White by income
percentile", size(medium)) */ name(D2, replace)
gr export "Income - AoM - White.eps", replace
```

Sample Code - Connected Plots.do (cont.)

```
// GROUP D SQ 3, SC, Black Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
        ///
                if outcome==4 & race==2,                ///
                /* ytitle("Age (years)"), size(medlarge) margin(tiny)) */
ylabel(11.5(1)13.55, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("Black
by risk of poverty", size(medium)) */ name(D3, replace)
gr export "Poverty - AoM - Black.eps", replace

// GROUP D SQ 4, SC, White Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==4 & race==1,                ///
                /* ytitle("Age (years)"), size(medlarge) margin(tiny)) */
ylabel(11.5(1)13.55, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White
by risk of poverty", size(medium)) */ name(D4, replace)
gr export "Poverty - AoM - White.eps", replace

// GROUP D SQ 5, SC, Black Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(med)
lpattern(longdash))             ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
///
                if outcome==4 & race==2,                ///
                /* ytitle("Age (years)"), size(medlarge) margin(tiny)) */
ylabel(11(1)14, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y
College" 6 ">=4y College") cols(3) size(large) rowgap(minuscule) colgap(minuscule)
keygap(minuscule)) /* title("Black by education level", size(medium)) */ name(D5,
replace)
gr export "Education - AoM - Black.eps", replace
```

Sample Code - Connected Plots.do (cont.)

```
// GROUP D SQ 6, SC, White Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==4 & race==1,                ///
                /* ytitle("Age (years)"), size(medlarge) margin(tiny)) */
ylabel(11(1)14, labsize(medium)) xtitle("Survey Midpoint", size(medlarge)) xlabel(,
labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y College" 6 ">=4y College")
cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /*
title("White by education level", size(medium)) */ name(D6, replace)
gr export "Education - AoM - White.eps", replace

/* graph combine D1 D2 D3 D4 D5 D6, colfirst imargin(small) ycommon xcommon
title("Age of menarche of US-born black and white females", size(medium)) name(DomD,
replace) xsize(8.5) ysize(11) scale(.65)
gr export "Table 2 - Page 4 - Age of Menarche.eps", replace */

// GROUP E SQ 1, SC, Black Income
twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(med)
lpattern(longdash))             ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
                ///
                if outcome==5 & race==2,                                        ///

                /* ytitle("SBP (mmHg)", size(medlarge) margin(tiny)) */
ylabel(120(10)150, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6
"80th") cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule))
/* title("Black by income percentile", size(medium)) */ name(E1, replace)
gr export "Income - SBP - Black.eps", replace

// GROUP E SQ 2, SC, White Income
twoway  (rcap lower upper midpoint if inc == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if inc == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if inc == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if inc == 1, sort mcolor(black)
```

Sample Code - Connected Plots.do (cont.)

```
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if inc == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if inc == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==5 & race==1,                ///
                /* ytitle("SBP (mmHg)", size(medlarge) margin(tiny)) */
ylabel(120(10)150, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(4 "20th" 5 "50th" 6 "80th") cols(3)
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White
by income percentile", size(medium)) */ name(E2, replace)
gr export "Income - SBP - White.eps", replace

// GROUP E SQ 3, SC, Black Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
lpattern(shortdash))            ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
        ///
                if outcome==5 & race==2,                ///
                /* ytitle("SBP (mmHg)"), size(medlarge) margin(tiny)) */
ylabel(115(10)146, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("Black
by risk of poverty", size(medium)) */ name(E3, replace)
gr export "Poverty - SBP - Black.eps", replace

// GROUP E SQ 4, SC, White Poverty
twoway  (rcap lower upper midpoint if pov == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if pov == 2, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if pov == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if pov == 2, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==5 & race==1,                ///
                /* ytitle("SBP (mmHg)"), size(medlarge) margin(tiny)) */
ylabel(115(10)146, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(3 "At risk" 4 "2x above risk")
size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule)) /* title("White
by risk of poverty", size(medium)) */ name(E4, replace)
gr export "Poverty - SBP - White.eps", replace

// GROUP E SQ 5, SC, Black Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond_hollow) lcolor(black) lwidth(med)
```

Sample Code - Connected Plots.do (cont.)

```
lpattern(shortdash))              ///
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle_hollow) lcolor(black) lwidth(med)
lpattern(longdash))               ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare_hollow) lcolor(black) lwidth(med) lpattern(solid))
///
                if outcome==5 & race==2,                ///
                /* ytitle("SBP (mmHg)"), size(medlarge) margin(tiny)) */
ylabel(115(10)146, labsize(medium)) xtitle(off) xtitle("Survey Midpoint",
size(medlarge)) xlabel(, labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y
College" 6 ">=4y College") cols(3) size(large) rowgap(minuscule) colgap(minuscule)
keygap(minuscule)) /* title("Black by education level", size(medium)) */ name(E5,
replace)
gr export "Education - SBP - Black.eps", replace

// GROUP E SQ 6, SC, White Education
twoway  (rcap lower upper midpoint if edu == 1, sort lwidth(medium) msize(medlarge))
///
                (rcap lower upper midpoint if edu == 2, sort lwidth(medium)
msize(medlarge))        ///
                (rcap lower upper midpoint if edu == 3, sort lwidth(medium)
msize(medlarge))        ///
                (connected mean midpoint if edu == 1, sort mcolor(black)
msize(medlarge) msymbol(smdiamond) lcolor(gs8) lwidth(medium) lpattern(shortdash))
        ///
                (connected mean midpoint if edu == 2, sort mcolor(black)
msize(medlarge) msymbol(smtriangle) lcolor(gs8) lwidth(medium) lpattern(longdash))
        ///
                (connected mean midpoint if edu == 3, sort mcolor(black)
msize(medlarge) msymbol(smsquare) lcolor(gs8) lwidth(medium) lpattern(solid))   ///

                if outcome==5 & race==1,                ///
                /* ytitle("SBP (mmHg)"), size(medlarge) margin(tiny)) */
ylabel(115(10)146, labsize(medium)) xtitle("Survey Midpoint", size(medlarge))
xlabel(, labels labsize(medium)) legend(order(4 "<HS" 5 "Some 4y College" 6 ">=4y
College") cols(3) size(large) rowgap(minuscule) colgap(minuscule) keygap(minuscule))
/* title("White by education level", size(medium)) */ name(E6, replace)
gr export "Education - SBP - White.eps", replace

/* graph combine E1 E2 E3 E4 E5 E6, colfirst imargin(small) ycommon xcommon
title("Systolic blood pressure (SBP) of US black and white adults", size(medium))
name(EomD, replace) xsize(8.5) ysize(11) scale(.65)
gr export "Table 2 - Page 5 - SBP.eps", replace */
```