pset 2 Labor

```
#=============================#
# ==== Labor Problem set 2 ====
#=============================#


#=====================================#
# ==== load packages and clear data ====
#=====================================#

  # clear data and consol
  rm(list = ls(pos = ".GlobalEnv"), pos = ".GlobalEnv")
  options(scipen = 999)
  cat("\f")

  # load packages
  library(data.table)
  library(lmtest)
  library(sandwich)
  library(broom)
  library(foreign)
  library(AER)
  library(ivpack)
  library(Matrix)
  library(knitr)
  library(kableExtra)
  library(xtable)
#======================#
# ==== data set up ====
#======================#

  # laod data
  dt <- data.table(
    read.dta("c://users/Nmath_000/Documents/MI_school/Second Year/621 Labor/Assignments/Assignment_2/MRU

  # create neede vars
  dt[, age_sq := age^2]
  dt[,age_cu := age^3]
  dt[,educ_sq := educ^2]
  dt[,educ_cu := educ^3]
  dt[,age_educ := age*educ]
  dt[,age_sq_educ := age_sq*educ]
  dt[,age_educ_sq := age*educ_sq]
  dt[, nonlab_i := (faminc - wage*hours - huswage*hushrs)/1000]


  # create a list to store main results
  r_l <- list()

#=============================#
# ==== a  replicate table 3 ====
```

```r
#============================#


  # get vars to mean ans sd
  var_l <- c("age", "educ", "kidslt6", "kidsge6",
             "husage", "huseduc", "wage", "huswage",
             "nonlab_i", "hours", "hushrs"  )

  tb1 <- melt.data.table(round(dt[,lapply(.SD,mean), .SDcols =var_l],2),
                         value.name = "Full Sample Mean")

  tb2 <- melt.data.table(round(dt[,lapply(.SD,sd), .SDcols =var_l],2),
                         value.name = "Full Sample Standard Deviation")

  tb3 <- melt.data.table(round(dt[inlf == 1,lapply(.SD,mean), .SDcols =var_l],2),
                         value.name = "Working Women Mean")

  tb4 <- melt.data.table(round(dt[inlf == 1,lapply(.SD,sd), .SDcols =var_l],2),
                         value.name = "Working Women Standard Deviation")

  # merge them all
  r_l[["a"]] <- Reduce(function(x, y) merge(x, y, by = "variable", all = T), list(tb1, tb2, tb3, tb4))

#=========================#
# ==== b baseline ols ====
#=========================#

  # run regression
  base_lm <- lm(hours ~ lwage + nwifeinc + kidslt6 + kidsge6  + age + educ , dt)

  # get robust standard errors.
  lm_robust <- coeftest(base_lm, vcov = vcovHC(base_lm, type="HC0"))

  r_l[["b"]] <- data.table(tidy(lm_robust))


#========================#
# ==== C IV estimate ====
#========================#

  iv_reg <- ivreg(hours ~ nwifeinc + kidslt6 + kidsge6 + age + educ + lwage |
                    nwifeinc + kidslt6 + kidsge6 + age + educ +
                    age_sq + age_cu + educ_sq + educ_cu +
                    age_educ + age_sq_educ + age_educ_sq +
                    unem + city + motheduc + fatheduc , data = dt)

  # robust the se
  iv_reg <- robust.se(iv_reg)

  r_l[["c"]] <- data.table(tidy(iv_reg, conf.int = TRUE))

#====================#
# ==== D repwage ====
```

```
#====================#

  # run regression
  base_lm <- lm(hours ~ lwage + nwifeinc + kidslt6 + kidsge6  + age + educ , dt[repwage >0,])

  # get robust standard errors.
  lm_robust <- coeftest(base_lm, vcov = vcovHC(base_lm, type="HC0"))

  r_l[["d"]] <- data.table(tidy(lm_robust))

#=======================#
# ==== E repwage IV ====
#=======================#

  iv_reg <- ivreg(hours ~  lwage + nwifeinc + kidslt6 + kidsge6 + age + educ  |
                    nwifeinc + kidslt6 + kidsge6 + age + educ + repwage , data = dt[repwage >0,])

  # robust the se
  iv_reg <- robust.se(iv_reg)

  r_l[["e"]] <- data.table(tidy(iv_reg, conf.int = TRUE))

#====================#
# ==== F probit ====
#====================#

  # create variable for working
  dt[hours > 0, working := 1]
  dt[hours <= 0, working := 0]

  # estimate a probit model
  myprobit <- glm(working ~nwifeinc + kidslt6 + kidsge6 + age + educ +
                    age_sq + age_cu + educ_sq + educ_cu +
                    age_educ + age_sq_educ + age_educ_sq +
                    unem + city + motheduc + fatheduc,
                  family = binomial(link = "probit"),
                  data = dt)

  r_l[["f"]] <- data.table(tidy(myprobit))

#======================#
# ==== G H heckman ====
#======================#


  dt[, IMR := dnorm(myprobit$linear.predictors)/pnorm(myprobit$linear.predictors)]

  # run regression
  samp_sel <- lm( lwage ~ age + educ +
                   age_sq + age_cu + educ_sq + educ_cu +
                   age_educ + age_sq_educ + age_educ_sq +
                   unem + city + motheduc + fatheduc + IMR , dt)
```

```r
  vars <- c("age", "educ", "age_sq", "age_cu",
            "educ_sq", "educ_cu", "age_educ", "age_sq_educ",
            "age_educ_sq", "unem",  "city", "motheduc",
            "fatheduc", "IMR")

  dt[,  lapply(.SD,  function(y) sum(length(which(is.na(y))))), .SDcols = vars]

  # NOTE: the standard errors are wrong since IMR is an estimate. See the worksheet for explination


  r_l[["h"]] <- data.table(tidy(samp_sel))


#======================#
# ==== I est hours ====
#======================#

  # get fitted values of lwage for full sample (including non workers ) without the IMR term
  B <- as.matrix(part_h_result[term != "IMR", estimate])

  X <- dt[, setdiff(part_h_result[, term] ,c("(Intercept)", "IMR")), with = FALSE]
  X[,  intercept := 1]
  setcolorder(X, c("intercept", setdiff(colnames(X), "intercept")))
  X <- as.matrix(X)

  fitted <- X%*%B

  # add it to data
  dt[, lwage_hat := fitted]


# estimate hours worked
hrs_reg <- lm( hours ~ lwage_hat + nwifeinc + age + educ + kidslt6 + kidsge6 + IMR , dt)

r_l[["i"]] <- data.table(tidy(hrs_reg))

#====================================#
# ==== J IV and sample correction ====
#====================================#

iv_reg <- ivreg(hours ~ nwifeinc + kidslt6 + kidsge6 + age + educ + IMR +lwage |
                  nwifeinc + kidslt6 + kidsge6 + age + educ + IMR +
                  age_sq + age_cu + educ_sq + educ_cu +
                  age_educ + age_sq_educ + age_educ_sq +
                  unem + city + motheduc + fatheduc , data = dt)

# robust the se
iv_reg <- robust.se(iv_reg)

r_l[["j"]] <- data.table(tidy(iv_reg, conf.int = TRUE))


#======================#
```

```r
# ==== save tables ====
#=====================#

names(r_l)

save_tex_tables <- function(letter = NULL){

  table <- r_l[[letter]]

  print(xtable(table, type = "latex"),
        file = paste0("C:/Users/Nmath_000/Documents/Code/courses/econ 621/assignment_2_tex/t_",
                      letter, ".tex"),
        include.rownames = FALSE)

}

lapply(names(r_l), save_tex_tables)
```