```
R version 4.3.1 (2023-06-16) -- "Beagle Scouts"
Copyright (C) 2023 The R Foundation for Statistical Computing
Platform: aarch64-apple-darwin20 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.79 (8238) aarch64-apple-darwin20]

[History restored from /Users/u1572955/.Rapp.history]

> setwd('/Users/u1572955/Dropbox/mexican-citizen-forecasting-2024/
replication')
> # Citizen forecasts of Mexican presidential elections, 2000-2024
> # Andreas Murr
> # Creates Tables 4-9
> # Writes functions invz(), cor.pos(), cor.pos.mul()
> # Calls hlm.r
>
> # clear working memory
>
> rm(list = ls())
>
> # write functions
>
> ## summary statistics
> describe = function(x){
+   out = t(apply(x, 2, function(y){c(mean(y), sd(y), min(y), max(y))}))
+   colnames(out) = c("Mean", "SD", "Min", "Max")
+   noquote(format(round(out, 2), digits = 2))
+ }
>
> ## correlations
>
> # the inverse of the z transform
> invz = function(z){
+   # Lindley, D. V., and Scott, W. F., New Cambridge Elementary
```

```
Statistical Tables, Cambridge: Cambridge University Press (1995) [1st edn
(1984)].  Table 17
+   # Neave, H. R., Statistics Tables for Mathematicians, Engineers,
Economists and the Behavioural and Management Sciences, London: George
Allen & Unwin (1978).  Table 6.3
+   r = (exp(2 * z) - 1) / (exp(2 * z) + 1)
+   return(r)
+ }
>
> # the posterior of bivariate correlation
> cor.pos = function(x, y, alpha = 0.05){
+   n = length(x)
+   r = cor(x, y)
+   z = atanh(r)
+   z + c(-1, 1) * qnorm(1 - alpha / 2) * sqrt(1 / n)
+   invz(z + c(-1, 1) * qnorm(1 - alpha / 2) * sqrt(1 / n))
+ }
>
> # posterior of multivariate correlations
> cor.pos.mul = function(X, alpha = .05, interval = TRUE){
+   # compute correlation
+   R = cor(X)
+   # create matrix for output
+   k = ncol(X)
+   O = matrix("", nrow = (k - 1) * 2, ncol = k - 1)
+   # compute correlation and interval (standard error)
+   i = 1
+   j = i + 1
+   while (i < k){
+     while (j <= k){
+       X.new = na.omit(X[,c(i,j)])
+       x = X.new[,1]
+       y = X.new[,2]
+       r = cor(x, y)
+       int = cor.pos(x, y, alpha = alpha)
+       print.r = formatC(r, format = "f", digits = 2)
+       if (interval == FALSE){
+         se = (int[2] - int[1]) / (qnorm(1 - alpha / 2) * 2)
+         print.i = paste("(", formatC(se, format = "f", digits = 2), ")",
sep = "")
+       } else{
+         print.i = paste("[", paste(formatC(int, format = "f", digits =
2), collapse = "; "), "]", sep = "")
+       }
+       # lower triangle
+       O[(j-1) + (j-1) - 1, i] = print.r
+       O[(j-1) + (j-1),i] = print.i
+       j = j + 1
+     }
```

```
+     i = i + 1
+     j = i + 1
+   }
+   # return results
+   noquote(O)
+ }
>
> # =============
> # = load data =
> # =============
>
> load("study-3-data.rdata")
>
> # ===========
> # = table 4 =
> # ===========
>
> d00 = describe(sel00[,-c(1:3,13)])
> d06 = describe(sel06[,-c(1:3,13)])
> d12 = describe(sel12[,-c(1:3,13)])
> d18 = describe(sel18[,-c(1:3,13)])
>
> D = matrix(NA, nrow = nrow(d00)*4, ncol = 6)
> for (i in 1:nrow(d00)){
+   D[(1+(i-1)*4),1] = rownames(d00)[i]
+   D[(1+(i-1)*4):(i*4),2] = c("2000", "2006", "2012", "2018")
+   D[(1+(i-1)*4):(i*4),3:6] = rbind(d00[i,], d06[i,], d12[i,], d18[i,])
+ }
> noquote(D)
      [,1]     [,2] [,3]  [,4]  [,5]  [,6]
 [1,] female   2000  0.49  0.50  0.00  1.00
 [2,] <NA>     2006  0.53  0.50  0.00  1.00
 [3,] <NA>     2012  0.50  0.50  0.00  1.00
 [4,] <NA>     2018  0.52  0.50  0.00  1.00
 [5,] age      2000 35.29 12.58 17.00 89.00
 [6,] <NA>     2006 39.20 14.22 17.00 86.00
 [7,] <NA>     2012 40.19 13.93 18.00 81.00
 [8,] <NA>     2018 39.12 13.20 17.00 75.00
 [9,] educ     2000  2.29  1.08  1.00  4.00
[10,] <NA>     2006  2.24  1.04  1.00  4.00
[11,] <NA>     2012  2.26  0.91  1.00  4.00
[12,] <NA>     2018  2.42  0.78  1.00  4.00
[13,] interest 2000  1.11  0.61  0.00  2.00
[14,] <NA>     2006  1.18  0.62  0.00  2.00
[15,] <NA>     2012  1.13  0.58  0.00  2.00
[16,] <NA>     2018  0.97  0.66  0.00  2.00
[17,] pid.pan  2000  0.29  0.45  0.00  1.00
[18,] <NA>     2006  0.34  0.47  0.00  1.00
[19,] <NA>     2012  0.19  0.39  0.00  1.00
```

```
[20,] <NA>     2018  0.19  0.39  0.00  1.00
[21,] pid.prd  2000  0.15  0.35  0.00  1.00
[22,] <NA>     2006  0.31  0.46  0.00  1.00
[23,] <NA>     2012  0.21  0.40  0.00  1.00
[24,] <NA>     2018  0.30  0.46  0.00  1.00
[25,] pid.pri  2000  0.31  0.46  0.00  1.00
[26,] <NA>     2006  0.21  0.41  0.00  1.00
[27,] <NA>     2012  0.35  0.48  0.00  1.00
[28,] <NA>     2018  0.17  0.38  0.00  1.00
[29,] time     2000  0.27  0.45  0.00  1.78
[30,] <NA>     2006  0.72  0.81  0.00  2.78
[31,] <NA>     2012  1.19  1.21  0.00  3.78
[32,] <NA>     2018  1.55  1.39  0.00  4.78
[33,] clean2   2000  1.19  0.74  0.00  2.00
[34,] <NA>     2006  1.53  0.63  0.00  2.00
[35,] <NA>     2012  1.27  0.78  0.00  2.00
[36,] <NA>     2018  1.06  0.71  0.00  2.00
>
> # ===========
> # = table 5 =
> # ===========
>
> C00 = sel00[,-c(1:3,5)]
> C06 = sel06[,-c(1:3,5)]
> C12 = sel12[,-c(1:3,5)]
> C00$pid.mrn = NA
> C06$pid.mrn = NA
> C12$pid.mrn = NA
> C18 = cbind(sel18[,c(4,6:8)], NA, sel18[,c(10:13, 9)])
> names(C18)[5] = "pid.prd"
> C = rbind(C00, C06, C12, C18)
> C = C[,c(1,9,2:3,10,4:6,7:8)]
>
> C.out = cor.pos.mul(C, interval = FALSE)
> colnames(C.out) = colnames(C)[-ncol(C)]
> rownames(C.out) = rep("", nrow(C.out))
> rownames(C.out)[seq(1, nrow(C.out), 2)] = colnames(C)[-1]
> C.out
         female age100 educ   interest pid.mrn pid.pan pid.prd pid.pri
age100   -0.04
         (0.01)
educ     -0.08  -0.38
         (0.01) (0.01)
interest -0.04  0.01   0.16
         (0.01) (0.01) (0.01)
pid.mrn  0.01   -0.14  0.04   0.07
         (0.03) (0.03) (0.03) (0.03)
pid.pan  0.02   -0.07  0.06   0.05     -0.32
         (0.01) (0.01) (0.01) (0.01)   (0.03)
```

```
pid.prd  -0.02  0.01   -0.04  0.05      NA     -0.33
         (0.01) (0.01) (0.01) (0.01)   ( NA)   (0.01)
pid.pri  0.02   0.08   -0.06  0.08     -0.30   -0.37   -0.33
         (0.01) (0.01) (0.01) (0.01)   (0.03)  (0.01)  (0.01)
time     0.00   0.06   0.02   -0.06    0.12    -0.01   0.03    -0.08
         (0.01) (0.01) (0.01) (0.01)   (0.03)  (0.01)  (0.01)  (0.01)
clean2   0.02   0.03   -0.00  0.22     -0.08   0.10    -0.06   0.21
         (0.01) (0.01) (0.01) (0.01)   (0.03)  (0.01)  (0.01)  (0.01)
         time
age100

educ

interest

pid.mrn

pid.pan

pid.prd

pid.pri

time

clean2   -0.00
         (0.01)
>
> # ================================================
> # = fit bayesian heteroskedastic linear model =
> # ================================================
>
> # create model matrix
>
> y00.pan = sel00$err.pan
> y06.pan = sel06$err.pan
> y12.pan = sel12$err.pan
> y18.pan = sel18$err.pan
>
> y00.prd = sel00$err.prd
> y06.prd = sel06$err.prd
> y12.prd = sel12$err.prd
> y18.mrn = sel18$err.mrn
>
> y00.pri = sel00$err.pri
> y06.pri = sel06$err.pri
> y12.pri = sel12$err.pri
> y18.pri = sel18$err.pri
>
```

```
> X00.pan = with(sel00, cbind(1, female, age100, educ, interest, pid.pan,
clean2, time))
> X06.pan = with(sel06, cbind(1, female, age100, educ, interest, pid.pan,
clean2, time))
> X12.pan = with(sel12, cbind(1, female, age100, educ, interest, pid.pan,
clean2, time))
> X18.pan = with(sel18, cbind(1, female, age100, educ, interest, pid.pan,
clean2, time))
>
> X00.prd = with(sel00, cbind(1, female, age100, educ, interest, pid.prd,
clean2, time))
> X06.prd = with(sel06, cbind(1, female, age100, educ, interest, pid.prd,
clean2, time))
> X12.prd = with(sel12, cbind(1, female, age100, educ, interest, pid.prd,
clean2, time))
> X18.mrn = with(sel18, cbind(1, female, age100, educ, interest, pid.mrn,
clean2, time))
>
> X00.pri = with(sel00, cbind(1, female, age100, educ, interest, pid.pri,
clean2, time))
> X06.pri = with(sel06, cbind(1, female, age100, educ, interest, pid.pri,
clean2, time))
> X12.pri = with(sel12, cbind(1, female, age100, educ, interest, pid.pri,
clean2, time))
> X18.pri = with(sel18, cbind(1, female, age100, educ, interest, pid.pri,
clean2, time))
>
> Z00.pan = X00.pan
> Z06.pan = X06.pan
> Z12.pan = X12.pan
> Z18.pan = X18.pan
>
> Z00.prd = X00.prd
> Z06.prd = X06.prd
> Z12.prd = X12.prd
> Z18.mrn = X18.mrn
>
> Z00.pri = X00.pri
> Z06.pri = X06.pri
> Z12.pri = X12.pri
> Z18.pri = X18.pri
>
> # compute heteroskedastic linear model
>
> source("hlm.r")
>
> ## fit models
>
>   set.seed(512438)
```

```
>   # pan
>   fit00.pan = hlm(y00.pan, X00.pan, Z00.pan, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
>   fit06.pan = hlm(y06.pan, X06.pan, Z06.pan, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
>   fit12.pan = hlm(y12.pan, X12.pan, Z12.pan, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
>   fit18.pan = hlm(y18.pan, X18.pan, Z18.pan, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
```

```
>   # prd / mrn
>   fit00.prd = hlm(y00.prd, X00.prd, Z00.prd, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
>   fit06.prd = hlm(y06.prd, X06.prd, Z06.prd, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
>   fit12.prd = hlm(y12.prd, X12.prd, Z12.prd, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
>   fit18.mrn = hlm(y18.mrn, X18.mrn, Z18.mrn, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
```

```
>   # pri
>   fit00.pri = hlm(y00.pri, X00.pri, Z00.pri, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
>   fit06.pri = hlm(y06.pri, X06.pri, Z06.pri, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
>   fit12.pri = hlm(y12.pri, X12.pri, Z12.pri, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
>   fit18.pri = hlm(y18.pri, X18.pri, Z18.pri, n.iter = 1000, n.prog =
100)
iteration: 100
iteration: 200
iteration: 300
iteration: 400
iteration: 500
iteration: 600
iteration: 700
iteration: 800
iteration: 900
iteration: 1000
```

```
>
> ## check convergence
>
>   # pan
>   plot.hlm.iter(fit00.pan)
>   plot.hlm.iter(fit00.pan, n.burn = 100, store = 1)
>   plot.hlm.iter(fit06.pan)
>   plot.hlm.iter(fit06.pan, n.burn = 100, store = 1)
>   plot.hlm.iter(fit12.pan)
>   plot.hlm.iter(fit12.pan, n.burn = 100, store = 1)
>   plot.hlm.iter(fit18.pan)
>   plot.hlm.iter(fit18.pan, n.burn = 100, store = 1)
>   # prd / mrn
>   plot.hlm.iter(fit00.prd)
>   plot.hlm.iter(fit00.prd, n.burn = 100, store = 1)
>   plot.hlm.iter(fit06.prd)
>   plot.hlm.iter(fit06.prd, n.burn = 100, store = 1)
>   plot.hlm.iter(fit12.prd)
>   plot.hlm.iter(fit12.prd, n.burn = 100, store = 1)
>   plot.hlm.iter(fit18.mrn)
>   plot.hlm.iter(fit18.mrn, n.burn = 100, store = 1)
>   # pri
>   plot.hlm.iter(fit00.pri)
>   plot.hlm.iter(fit00.pri, n.burn = 100, store = 1)
>   plot.hlm.iter(fit06.pri)
>   plot.hlm.iter(fit06.pri, n.burn = 100, store = 1)
>   plot.hlm.iter(fit12.pri)
>   plot.hlm.iter(fit12.pri, n.burn = 100, store = 1)
>   plot.hlm.iter(fit18.pri)
>   plot.hlm.iter(fit18.pri, n.burn = 100, store = 1)
>
> # ===========
> # = table 6 =
> # ===========
>
> r = matrix(NA, nrow = (ncol(X00.pan) - 1) * 2, ncol = 3)
> rownames(r) = rep("", nrow(r))
> j = 0
> for (i in 2:ncol(X00.pan)){
+   # create range the variable
+   r00 = range(X00.pan[,i])
+   r06 = range(X06.pan[,i])
+   r12 = range(X12.pan[,i])
+   r18 = range(X18.pan[,i])
+   lo = max(c(r00[1], r06[1], r12[1], r18[1]))
+   hi = min(c(r00[2], r06[2], r12[2], r18[2]))
+   # create two copies of data
+   ## pan
+   X00.pan.lo = X00.pan.hi = X00.pan
```

```
+    X06.pan.lo = X06.pan.hi = X06.pan
+    X12.pan.lo = X12.pan.hi = X12.pan
+    X18.pan.lo = X18.pan.hi = X18.pan
+    ## prd / mrn
+    X00.prd.lo = X00.prd.hi = X00.prd
+    X06.prd.lo = X06.prd.hi = X06.prd
+    X12.prd.lo = X12.prd.hi = X12.prd
+    X18.mrn.lo = X18.mrn.hi = X18.mrn
+    ## pri
+    X00.pri.lo = X00.pri.hi = X00.pri
+    X06.pri.lo = X06.pri.hi = X06.pri
+    X12.pri.lo = X12.pri.hi = X12.pri
+    X18.pri.lo = X18.pri.hi = X18.pri
+    # set predictor to low value
+    X00.pan.lo[,i] = lo
+    X06.pan.lo[,i] = lo
+    X12.pan.lo[,i] = lo
+    X18.pan.lo[,i] = lo
+    X00.prd.lo[,i] = lo
+    X06.prd.lo[,i] = lo
+    X12.prd.lo[,i] = lo
+    X18.mrn.lo[,i] = lo
+    X00.pri.lo[,i] = lo
+    X06.pri.lo[,i] = lo
+    X12.pri.lo[,i] = lo
+    X18.pri.lo[,i] = lo
+    # set predictor to high value
+    X00.pan.hi[,i] = hi
+    X06.pan.hi[,i] = hi
+    X12.pan.hi[,i] = hi
+    X18.pan.hi[,i] = hi
+    X00.prd.hi[,i] = hi
+    X06.prd.hi[,i] = hi
+    X12.prd.hi[,i] = hi
+    X18.mrn.hi[,i] = hi
+    X00.pri.hi[,i] = hi
+    X06.pri.hi[,i] = hi
+    X12.pri.hi[,i] = hi
+    X18.pri.hi[,i] = hi
+    # compute bias squared
+    ## for low values
+    bias.sq.00.pan.lo = (X00.pan.lo%*%t(fit00.pan[[1]]))^2
+    bias.sq.06.pan.lo = (X06.pan.lo%*%t(fit06.pan[[1]]))^2
+    bias.sq.12.pan.lo = (X12.pan.lo%*%t(fit12.pan[[1]]))^2
+    bias.sq.18.pan.lo = (X18.pan.lo%*%t(fit18.pan[[1]]))^2
+    bias.sq.00.prd.lo = (X00.prd.lo%*%t(fit00.prd[[1]]))^2
+    bias.sq.06.prd.lo = (X06.prd.lo%*%t(fit06.prd[[1]]))^2
+    bias.sq.12.prd.lo = (X12.prd.lo%*%t(fit12.prd[[1]]))^2
+    bias.sq.18.mrn.lo = (X18.mrn.lo%*%t(fit18.mrn[[1]]))^2
```

```
+   bias.sq.00.pri.lo = (X00.pri.lo%*%t(fit00.pri[[1]]))^2
+   bias.sq.06.pri.lo = (X06.pri.lo%*%t(fit06.pri[[1]]))^2
+   bias.sq.12.pri.lo = (X12.pri.lo%*%t(fit12.pri[[1]]))^2
+   bias.sq.18.pri.lo = (X18.pri.lo%*%t(fit18.pri[[1]]))^2
+   ## for high values
+   bias.sq.00.pan.hi = (X00.pan.hi%*%t(fit00.pan[[1]]))^2
+   bias.sq.06.pan.hi = (X06.pan.hi%*%t(fit06.pan[[1]]))^2
+   bias.sq.12.pan.hi = (X12.pan.hi%*%t(fit12.pan[[1]]))^2
+   bias.sq.18.pan.hi = (X18.pan.hi%*%t(fit18.pan[[1]]))^2
+   bias.sq.00.prd.hi = (X00.prd.hi%*%t(fit00.prd[[1]]))^2
+   bias.sq.06.prd.hi = (X06.prd.hi%*%t(fit06.prd[[1]]))^2
+   bias.sq.12.prd.hi = (X12.prd.hi%*%t(fit12.prd[[1]]))^2
+   bias.sq.18.mrn.hi = (X18.mrn.hi%*%t(fit18.mrn[[1]]))^2
+   bias.sq.00.pri.hi = (X00.pri.hi%*%t(fit00.pri[[1]]))^2
+   bias.sq.06.pri.hi = (X06.pri.hi%*%t(fit06.pri[[1]]))^2
+   bias.sq.12.pri.hi = (X12.pri.hi%*%t(fit12.pri[[1]]))^2
+   bias.sq.18.pri.hi = (X18.pri.hi%*%t(fit18.pri[[1]]))^2
+   ## compute difference
+   d.bias.sq.00.pan = bias.sq.00.pan.hi - bias.sq.00.pan.lo
+   d.bias.sq.06.pan = bias.sq.06.pan.hi - bias.sq.06.pan.lo
+   d.bias.sq.12.pan = bias.sq.12.pan.hi - bias.sq.12.pan.lo
+   d.bias.sq.18.pan = bias.sq.18.pan.hi - bias.sq.18.pan.lo
+   d.bias.sq.00.prd = bias.sq.00.prd.hi - bias.sq.00.prd.lo
+   d.bias.sq.06.prd = bias.sq.06.prd.hi - bias.sq.06.prd.lo
+   d.bias.sq.12.prd = bias.sq.12.prd.hi - bias.sq.12.prd.lo
+   d.bias.sq.18.mrn = bias.sq.18.mrn.hi - bias.sq.18.mrn.lo
+   d.bias.sq.00.pri = bias.sq.00.pri.hi - bias.sq.00.pri.lo
+   d.bias.sq.06.pri = bias.sq.06.pri.hi - bias.sq.06.pri.lo
+   d.bias.sq.12.pri = bias.sq.12.pri.hi - bias.sq.12.pri.lo
+   d.bias.sq.18.pri = bias.sq.18.pri.hi - bias.sq.18.pri.lo
+   ## compute individual mean in each election for each parties
+   m01 = mean(apply(d.bias.sq.00.pan, 1, mean))
+   m02 = mean(apply(d.bias.sq.06.pan, 1, mean))
+   m03 = mean(apply(d.bias.sq.12.pan, 1, mean))
+   m04 = mean(apply(d.bias.sq.18.pan, 1, mean))
+   m05 = mean(apply(d.bias.sq.00.prd, 1, mean))
+   m06 = mean(apply(d.bias.sq.06.prd, 1, mean))
+   m07 = mean(apply(d.bias.sq.12.prd, 1, mean))
+   m08 = mean(apply(d.bias.sq.18.mrn, 1, mean))
+   m09 = mean(apply(d.bias.sq.00.prd, 1, mean))
+   m10 = mean(apply(d.bias.sq.06.pri, 1, mean))
+   m11 = mean(apply(d.bias.sq.12.pri, 1, mean))
+   m12 = mean(apply(d.bias.sq.18.pri, 1, mean))
+   ## compute variance of mean
+   ## var(x.bar) = var(1/n sum_i^n x_i)
+   ## = 1/n^2 sum_i^n var(x_i) = 1 / n^2 sum_i^n sigma_i^2
+   v01 = sum(apply(d.bias.sq.00.pan, 1, var)) / nrow(d.bias.sq.00.pan)^2
+   v02 = sum(apply(d.bias.sq.06.pan, 1, var)) / nrow(d.bias.sq.06.pan)^2
+   v03 = sum(apply(d.bias.sq.12.pan, 1, var)) / nrow(d.bias.sq.12.pan)^2
```

```
+   v04 = sum(apply(d.bias.sq.18.pan, 1, var)) / nrow(d.bias.sq.18.pan)^2
+   v05 = sum(apply(d.bias.sq.00.prd, 1, var)) / nrow(d.bias.sq.00.prd)^2
+   v06 = sum(apply(d.bias.sq.06.prd, 1, var)) / nrow(d.bias.sq.06.prd)^2
+   v07 = sum(apply(d.bias.sq.12.prd, 1, var)) / nrow(d.bias.sq.12.prd)^2
+   v08 = sum(apply(d.bias.sq.18.mrn, 1, var)) / nrow(d.bias.sq.18.mrn)^2
+   v09 = sum(apply(d.bias.sq.00.prd, 1, var)) / nrow(d.bias.sq.00.prd)^2
+   v10 = sum(apply(d.bias.sq.06.pri, 1, var)) / nrow(d.bias.sq.06.pri)^2
+   v11 = sum(apply(d.bias.sq.12.pri, 1, var)) / nrow(d.bias.sq.12.pri)^2
+   v12 = sum(apply(d.bias.sq.18.pri, 1, var)) / nrow(d.bias.sq.18.pri)^2
+   ## save results
+   bias.sq.m = (m01 + m02 + m03 + m04 + m05 + m06 + m07 + m08 + m09 + m10
+ m11 + m12) / 12
+   bias.sq.s = sqrt((v01 + v02 + v03 + v04 + v05 + v06 + v07 + v08 + v09
+ v10 + v11 + v12) / 12^2)
+   # compute variance
+   ## for low values
+   var.00.pan.lo = exp(X00.pan.lo%*%t(fit00.pan[[2]]))
+   var.06.pan.lo = exp(X06.pan.lo%*%t(fit06.pan[[2]]))
+   var.12.pan.lo = exp(X12.pan.lo%*%t(fit12.pan[[2]]))
+   var.18.pan.lo = exp(X18.pan.lo%*%t(fit18.pan[[2]]))
+   var.00.prd.lo = exp(X00.prd.lo%*%t(fit00.prd[[2]]))
+   var.06.prd.lo = exp(X06.prd.lo%*%t(fit06.prd[[2]]))
+   var.12.prd.lo = exp(X12.prd.lo%*%t(fit12.prd[[2]]))
+   var.18.mrn.lo = exp(X18.mrn.lo%*%t(fit18.mrn[[2]]))
+   var.00.pri.lo = exp(X00.pri.lo%*%t(fit00.pri[[2]]))
+   var.06.pri.lo = exp(X06.pri.lo%*%t(fit06.pri[[2]]))
+   var.12.pri.lo = exp(X12.pri.lo%*%t(fit12.pri[[2]]))
+   var.18.pri.lo = exp(X18.pri.lo%*%t(fit18.pri[[2]]))
+   ## for high values
+   var.00.pan.hi = exp(X00.pan.hi%*%t(fit00.pan[[2]]))
+   var.06.pan.hi = exp(X06.pan.hi%*%t(fit06.pan[[2]]))
+   var.12.pan.hi = exp(X12.pan.hi%*%t(fit12.pan[[2]]))
+   var.18.pan.hi = exp(X18.pan.hi%*%t(fit18.pan[[2]]))
+   var.00.prd.hi = exp(X00.prd.hi%*%t(fit00.prd[[2]]))
+   var.06.prd.hi = exp(X06.prd.hi%*%t(fit06.prd[[2]]))
+   var.12.prd.hi = exp(X12.prd.hi%*%t(fit12.prd[[2]]))
+   var.18.mrn.hi = exp(X18.mrn.hi%*%t(fit18.mrn[[2]]))
+   var.00.pri.hi = exp(X00.pri.hi%*%t(fit00.pri[[2]]))
+   var.06.pri.hi = exp(X06.pri.hi%*%t(fit06.pri[[2]]))
+   var.12.pri.hi = exp(X12.pri.hi%*%t(fit12.pri[[2]]))
+   var.18.pri.hi = exp(X18.pri.hi%*%t(fit18.pri[[2]]))
+   ## compute difference
+   d.var.00.pan = var.00.pan.hi - var.00.pan.lo
+   d.var.06.pan = var.06.pan.hi - var.06.pan.lo
+   d.var.12.pan = var.12.pan.hi - var.12.pan.lo
+   d.var.18.pan = var.18.pan.hi - var.18.pan.lo
+   d.var.00.prd = var.00.prd.hi - var.00.prd.lo
+   d.var.06.prd = var.06.prd.hi - var.06.prd.lo
+   d.var.12.prd = var.12.prd.hi - var.12.prd.lo
```

```
+   d.var.18.mrn = var.18.mrn.hi - var.18.mrn.lo
+   d.var.00.pri = var.00.pri.hi - var.00.pri.lo
+   d.var.06.pri = var.06.pri.hi - var.06.pri.lo
+   d.var.12.pri = var.12.pri.hi - var.12.pri.lo
+   d.var.18.pri = var.18.pri.hi - var.18.pri.lo
+   ## compute individual mean in each election for each parties
+   m01 = mean(apply(d.var.00.pan, 1, mean))
+   m02 = mean(apply(d.var.06.pan, 1, mean))
+   m03 = mean(apply(d.var.12.pan, 1, mean))
+   m04 = mean(apply(d.var.18.pan, 1, mean))
+   m05 = mean(apply(d.var.00.prd, 1, mean))
+   m06 = mean(apply(d.var.06.prd, 1, mean))
+   m07 = mean(apply(d.var.12.prd, 1, mean))
+   m08 = mean(apply(d.var.18.mrn, 1, mean))
+   m09 = mean(apply(d.var.00.prd, 1, mean))
+   m10 = mean(apply(d.var.06.pri, 1, mean))
+   m11 = mean(apply(d.var.12.pri, 1, mean))
+   m12 = mean(apply(d.var.18.pri, 1, mean))
+   ## compute variance of mean
+   ## var(x.bar) = var(1/n sum_i^n x_i)
+   ## = 1/n^2 sum_i^n var(x_i) = 1 / n^2 sum_i^n sigma_i^2
+   v01 = sum(apply(d.var.00.pan, 1, var)) / nrow(d.var.00.pan)^2
+   v02 = sum(apply(d.var.06.pan, 1, var)) / nrow(d.var.06.pan)^2
+   v03 = sum(apply(d.var.12.pan, 1, var)) / nrow(d.var.12.pan)^2
+   v04 = sum(apply(d.var.18.pan, 1, var)) / nrow(d.var.18.pan)^2
+   v05 = sum(apply(d.var.00.prd, 1, var)) / nrow(d.var.00.prd)^2
+   v06 = sum(apply(d.var.06.prd, 1, var)) / nrow(d.var.06.prd)^2
+   v07 = sum(apply(d.var.12.prd, 1, var)) / nrow(d.var.12.prd)^2
+   v08 = sum(apply(d.var.18.mrn, 1, var)) / nrow(d.var.18.mrn)^2
+   v09 = sum(apply(d.var.00.prd, 1, var)) / nrow(d.var.00.prd)^2
+   v10 = sum(apply(d.var.06.pri, 1, var)) / nrow(d.var.06.pri)^2
+   v11 = sum(apply(d.var.12.pri, 1, var)) / nrow(d.var.12.pri)^2
+   v12 = sum(apply(d.var.18.pri, 1, var)) / nrow(d.var.18.pri)^2
+   ## save results
+   var.sq.m = (m01 + m02 + m03 + m04 + m05 + m06 + m07 + m08 + m09 + m10
+ m11 + m12) / 12
+   var.sq.s = sqrt((v01 + v02 + v03 + v04 + v05 + v06 + v07 + v08 + v09 +
v10 + v11 + v12) / 12^2)
+   # compute difference in mean squared error
+   ## is difference in bias.sq + difference in variance
+   m01 = mean(apply(d.bias.sq.00.pan + d.var.00.pan, 1, mean))
+   m02 = mean(apply(d.bias.sq.06.pan + d.var.06.pan, 1, mean))
+   m03 = mean(apply(d.bias.sq.12.pan + d.var.12.pan, 1, mean))
+   m04 = mean(apply(d.bias.sq.18.pan + d.var.18.pan, 1, mean))
+   m05 = mean(apply(d.bias.sq.00.prd + d.var.00.prd, 1, mean))
+   m06 = mean(apply(d.bias.sq.06.prd + d.var.06.prd, 1, mean))
+   m07 = mean(apply(d.bias.sq.12.prd + d.var.12.prd, 1, mean))
+   m08 = mean(apply(d.bias.sq.18.mrn + d.var.18.mrn, 1, mean))
+   m09 = mean(apply(d.bias.sq.00.prd + d.var.00.prd, 1, mean))
```

```
+   m10 = mean(apply(d.bias.sq.06.pri + d.var.06.pri, 1, mean))
+   m11 = mean(apply(d.bias.sq.12.pri + d.var.12.pri, 1, mean))
+   m12 = mean(apply(d.bias.sq.18.pri + d.var.18.pri, 1, mean))
+   ## compute variance
+   v01 = sum(apply(d.bias.sq.00.pan + d.var.00.pan, 1, var)) /
nrow(d.bias.sq.00.pan)^2
+   v02 = sum(apply(d.bias.sq.06.pan + d.var.06.pan, 1, var)) /
nrow(d.bias.sq.06.pan)^2
+   v03 = sum(apply(d.bias.sq.12.pan + d.var.12.pan, 1, var)) /
nrow(d.bias.sq.12.pan)^2
+   v04 = sum(apply(d.bias.sq.18.pan + d.var.18.pan, 1, var)) /
nrow(d.bias.sq.18.pan)^2
+   v05 = sum(apply(d.bias.sq.00.prd + d.var.00.prd, 1, var)) /
nrow(d.bias.sq.00.prd)^2
+   v06 = sum(apply(d.bias.sq.06.prd + d.var.06.prd, 1, var)) /
nrow(d.bias.sq.06.prd)^2
+   v07 = sum(apply(d.bias.sq.12.prd + d.var.12.prd, 1, var)) /
nrow(d.bias.sq.12.prd)^2
+   v08 = sum(apply(d.bias.sq.18.mrn + d.var.18.mrn, 1, var)) /
nrow(d.bias.sq.18.mrn)^2
+   v09 = sum(apply(d.bias.sq.00.prd + d.var.00.prd, 1, var)) /
nrow(d.bias.sq.00.prd)^2
+   v10 = sum(apply(d.bias.sq.06.pri + d.var.06.pri, 1, var)) /
nrow(d.bias.sq.06.pri)^2
+   v11 = sum(apply(d.bias.sq.12.pri + d.var.12.pri, 1, var)) /
nrow(d.bias.sq.12.pri)^2
+   v12 = sum(apply(d.bias.sq.18.pri + d.var.18.pri, 1, var)) /
nrow(d.bias.sq.18.pri)^2
+   ## save results
+   mse.sq.m = (m01 + m02 + m03 + m04 + m05 + m06 + m07 + m08 + m09 + m10
+ m11 + m12) / 12
+   mse.sq.s = sqrt((v01 + v02 + v03 + v04 + v05 + v06 + v07 + v08 + v09 +
v10 + v11 + v12) / 12^2)
+   ## return results
+   r[(i-1) + j,] = c(bias.sq.m, var.sq.m, mse.sq.m)
+   r[i + j,] = c(bias.sq.s, var.sq.s, mse.sq.s)
+   rownames(r)[(i-1) + j] = paste(colnames(X00.pan)[i], " (", lo, " to ",
hi, ")", sep="")
+   j = j + 1
+ }
>
> colnames(r) = c("bias.sq", "variance", "mse")
> rownames(r) = rep("", nrow(r))
> rownames(r)[seq(1, nrow(r), 2)] = colnames(X00.pan)[-1]
>
> p = r
> for (i in seq(1, nrow(p), 2)){
+   p[i,] = paste(formatC(r[i,], format = "f", digits = 1))
+ }
```

```
> for (i in seq(2, nrow(p), 2)){
+   p[i,] = paste("(", formatC(r[i,], format = "f", digits = 1), ")", sep
= "")
+ }
> noquote(p)
         bias.sq variance mse
female   12.7    5.3      18.0
         (0.1)   (0.5)    (0.5)
age100   7.8     5.5      13.3
         (0.3)   (0.7)    (0.8)
educ     -43.1   -38.7    -81.8
         (0.2)   (0.3)    (0.4)
interest 18.2    6.0      24.3
         (0.2)   (1.2)    (1.2)
pid.pan  285.5   145.3    430.8
         (0.3)   (6.3)    (6.3)
clean2   2.3     -35.5    -33.2
         (0.2)   (1.3)    (1.3)
time     -29.6   -17.6    -47.3
         (0.1)   (0.9)    (0.9)
>
> # return changes in predictor vairables (1st column)
>
> l00 = apply(X00.pan[,-1], 2, min)
> h00 = apply(X00.pan[,-1], 2, max)
> l06 = apply(X06.pan[,-1], 2, min)
> h06 = apply(X06.pan[,-1], 2, max)
> l12 = apply(X12.pan[,-1], 2, min)
> h12 = apply(X12.pan[,-1], 2, max)
> l18 = apply(X18.pan[,-1], 2, min)
> h18 = apply(X18.pan[,-1], 2, max)
>
> apply(rbind(l00, l06, l12, l18), 2, max)
  female   age100     educ interest  pid.pan   clean2     time
    0.00     0.18     1.00     0.00     0.00     0.00     0.00
> apply(rbind(h00, h06, h12, h18), 2, min)
  female   age100     educ interest  pid.pan   clean2     time
1.000000 0.750000 4.000000 2.000000 1.000000 2.000000 1.779452
>
> # ==============
> # = tables 7-9 =
> # ==============
>
> # tabulate summaries of posterior
>
>   # name intercept
>   colnames(X00.pan)[1] = "Intercept"
>   colnames(X06.pan)[1] = "Intercept"
>   colnames(X12.pan)[1] = "Intercept"
```

```
>   colnames(X18.pan)[1] = "Intercept"
>   colnames(X00.prd)[1] = "Intercept"
>   colnames(X06.prd)[1] = "Intercept"
>   colnames(X12.prd)[1] = "Intercept"
>   colnames(X18.mrn)[1] = "Intercept"
>   colnames(X00.pri)[1] = "Intercept"
>   colnames(X06.pri)[1] = "Intercept"
>   colnames(X12.pri)[1] = "Intercept"
>   colnames(X18.pri)[1] = "Intercept"
>
>   # table 7
>   noquote(cbind(
+     summary.hlm(fit00.pan, y00.pan, X00.pan, n.burn = 100, store = 1),
+     summary.hlm(fit06.pan, y06.pan, X06.pan, n.burn = 100, store = 1),
+     summary.hlm(fit12.pan, y12.pan, X12.pan, n.burn = 100, store = 1),
+     summary.hlm(fit18.pan, y18.pan, X18.pan, n.burn = 100, store = 1)
+   ))
          Mean            Variance       Mean            Variance
Intercept 17.29           4.91           15.38           5.03
          (1.09)          (0.11)         (1.60)          (0.15)
          [15.22; 19.40] [4.72; 5.10]   [12.28; 18.42] [4.77; 5.32]
female    2.52            0.30           0.97            0.08
          (0.47)          (0.04)         (0.57)          (0.06)
          [1.60; 3.53]   [0.21; 0.38]   [-0.13; 2.07]  [-0.03; 0.19]
age100    3.04            -0.07          -3.39           -0.23
          (1.94)          (0.18)         (2.01)          (0.21)
          [-0.75; 6.78]  [-0.42; 0.28]  [-7.08; 0.62]  [-0.61; 0.17]
educ      -0.92           -0.08          -0.81           -0.04
          (0.23)          (0.02)         (0.30)          (0.03)
          [-1.37; -0.46] [-0.12; -0.03] [-1.38; -0.23] [-0.10; 0.02]
interest  0.14            0.05           0.09            -0.09
          (0.38)          (0.03)         (0.47)          (0.04)
          [-0.55; 0.90]  [-0.02; 0.12]  [-0.78; 1.00]  [-0.17; -0.01]
pid.pan   -2.40           0.29           5.15            1.08
          (0.54)          (0.05)         (0.80)          (0.06)
          [-3.47; -1.41] [0.20; 0.38]   [3.59; 6.78]   [0.97; 1.19]
clean2    0.46            -0.11          -0.02           -0.10
          (0.32)          (0.03)         (0.46)          (0.04)
          [-0.12; 1.13]  [-0.16; -0.05] [-0.89; 0.86]  [-0.18; -0.01]
time      -3.38           -0.23          -0.37           -0.21
          (0.48)          (0.04)         (0.32)          (0.04)
          [-4.32; -2.47] [-0.31; -0.14] [-1.01; 0.26]  [-0.28; -0.14]
          n = 2206        ll = -8423.10  n = 1483        ll = -5705.79
          Mean            Variance       Mean            Variance
Intercept 7.79            3.72           9.69            4.02
          (1.20)          (0.20)         (1.31)          (0.21)
          [5.42; 10.11]  [3.34; 4.10]   [7.09; 12.17]  [3.59; 4.43]
female    -0.76           -0.08          -0.03           -0.14
          (0.44)          (0.07)         (0.48)          (0.07)
```

```
          [-1.66; 0.11]  [-0.20; 0.07]  [-0.96; 0.89]  [-0.30; -0.00]
age100    -0.33          -0.35          2.06           2.46
          (1.66)         (0.29)         (1.98)         (0.30)
          [-3.47; 2.89]  [-0.85; 0.18]  [-1.92; 6.11]  [1.91; 3.05]
educ      -0.56          -0.09          -1.12          -0.36
          (0.26)         (0.04)         (0.31)         (0.05)
          [-1.10; -0.02] [-0.18; -0.01] [-1.74; -0.52] [-0.46; -0.25]
interest  0.99           0.19           -0.77          0.08
          (0.38)         (0.06)         (0.38)         (0.07)
          [0.24; 1.73]   [0.08; 0.30]   [-1.47; -0.02] [-0.05; 0.20]
pid.pan   11.01          1.60           13.83          0.55
          (0.99)         (0.09)         (0.71)         (0.09)
          [8.97; 12.95]  [1.43; 1.77]   [12.41; 15.19] [0.39; 0.75]
clean2    0.06           -0.04          0.11           -0.12
          (0.29)         (0.04)         (0.37)         (0.05)
          [-0.50; 0.64]  [-0.13; 0.06]  [-0.62; 0.85]  [-0.22; -0.02]
time      0.53           0.12           0.12           -0.08
          (0.19)         (0.03)         (0.15)         (0.03)
          [0.17; 0.91]   [0.07; 0.17]   [-0.20; 0.41]  [-0.13; -0.02]
          n = 972        ll = -3291.95  n = 889        ll = -3019.30
> # table 8
> noquote(cbind(
+   summary.hlm(fit00.prd, y00.prd, X00.prd, n.burn = 100, store = 1),
+   summary.hlm(fit06.prd, y06.prd, X06.prd, n.burn = 100, store = 1),
+   summary.hlm(fit12.prd, y12.prd, X12.prd, n.burn = 100, store = 1),
+   summary.hlm(fit18.mrn, y18.mrn, X18.mrn, n.burn = 100, store = 1)
+ ))
          Mean           Variance       Mean           Variance
Intercept 9.95           4.75           15.29          5.13
          (0.88)         (0.11)         (1.60)         (0.15)
          [8.22; 11.65]  [4.54; 4.96]   [12.11; 18.38] [4.83; 5.44]
female    0.26           0.10           1.17           0.02
          (0.33)         (0.05)         (0.60)         (0.06)
          [-0.38; 0.86]  [0.00; 0.18]   [0.03; 2.33]   [-0.08; 0.13]
age100    2.41           0.34           -2.70          -0.03
          (1.43)         (0.18)         (2.14)         (0.19)
          [-0.57; 5.19]  [-0.04; 0.71]  [-6.87; 1.66]  [-0.43; 0.30]
educ      -0.27          -0.11          -0.61          -0.06
          (0.17)         (0.02)         (0.30)         (0.03)
          [-0.62; 0.07]  [-0.16; -0.08] [-1.20; -0.07] [-0.13; 0.00]
interest  0.16           -0.00          0.48           0.05
          (0.29)         (0.04)         (0.47)         (0.04)
          [-0.48; 0.70]  [-0.07; 0.08]  [-0.40; 1.47]  [-0.04; 0.13]
pid.prd   23.30          1.73           3.43           0.93
          (1.11)         (0.06)         (0.79)         (0.06)
          [21.20; 25.40] [1.62; 1.85]   [1.93; 5.00]   [0.83; 1.04]
clean2    -0.93          -0.38          -0.05          -0.25
          (0.25)         (0.03)         (0.49)         (0.04)
          [-1.41; -0.42] [-0.44; -0.32] [-0.97; 0.89]  [-0.34; -0.17]
```

```
time      -1.46          -0.43          -1.16          -0.14
          (0.32)         (0.05)         (0.34)         (0.04)
          [-2.08; -0.84] [-0.53; -0.34] [-1.82; -0.51] [-0.21; -0.07]
          n = 2206       ll = -7908.43  n = 1483       ll = -5692.34
          Mean           Variance       Mean           Variance
Intercept 10.67          4.24           17.75          5.30
          (1.57)         (0.21)         (1.69)         (0.23)
          [7.56; 13.64]  [3.77; 4.63]   [14.46; 20.97] [4.82; 5.70]
female    -0.42          -0.08          0.32           0.06
          (0.52)         (0.07)         (0.55)         (0.07)
          [-1.45; 0.59]  [-0.21; 0.07]  [-0.76; 1.43]  [-0.07; 0.20]
age100    -2.42          -0.15          5.50           0.52
          (2.13)         (0.29)         (2.36)         (0.31)
          [-6.47; 1.93]  [-0.78; 0.37]  [0.59; 9.75]   [-0.08; 1.18]
educ      -0.48          -0.08          -0.16          -0.33
          (0.33)         (0.04)         (0.37)         (0.05)
          [-1.09; 0.19]  [-0.16; 0.01]  [-0.89; 0.53]  [-0.43; -0.23]
interest  0.47           0.07           1.05           0.07
          (0.50)         (0.06)         (0.46)         (0.07)
          [-0.45; 1.48]  [-0.05; 0.19]  [0.19; 1.97]   [-0.05; 0.20]
pid.prd   2.49           0.66           -11.22         -0.61
          (0.87)         (0.08)         (0.56)         (0.08)
          [0.92; 4.18]   [0.50; 0.83]   [-12.41; -10.10] [-0.76; -0.44]
clean2    0.85           -0.05          1.15           -0.29
          (0.37)         (0.04)         (0.44)         (0.06)
          [0.15; 1.58]   [-0.13; 0.03]  [0.34; 1.99]   [-0.41; -0.17]
time      0.79           0.14           -0.57          -0.02
          (0.24)         (0.03)         (0.19)         (0.03)
          [0.32; 1.24]   [0.07; 0.19]   [-0.93; -0.21] [-0.07; 0.04]
          n = 972        ll = -3451.19  n = 889        ll = -3158.47
> # table 9
> noquote(cbind(
+   summary.hlm(fit00.pri, y00.pri, X00.pri, n.burn = 100, store = 1),
+   summary.hlm(fit06.pri, y06.pri, X06.pri, n.burn = 100, store = 1),
+   summary.hlm(fit12.pri, y12.pri, X12.pri, n.burn = 100, store = 1),
+   summary.hlm(fit18.pri, y18.pri, X18.pri, n.burn = 100, store = 1)
+ ))
          Mean           Variance       Mean           Variance
Intercept 15.82          4.78           11.46          4.42
          (1.14)         (0.11)         (1.17)         (0.15)
          [13.59; 18.07] [4.56; 5.01]   [9.09; 13.78]  [4.13; 4.70]
female    1.91           0.25           0.84           0.18
          (0.49)         (0.04)         (0.44)         (0.06)
          [0.99; 2.82]   [0.17; 0.34]   [0.00; 1.76]   [0.07; 0.29]
age100    1.57           0.19           -1.44          -0.40
          (1.93)         (0.18)         (1.60)         (0.20)
          [-2.27; 5.39]  [-0.14; 0.56]  [-4.58; 1.80]  [-0.76; 0.00]
educ      -0.95          -0.09          -0.48          -0.17
          (0.24)         (0.02)         (0.22)         (0.03)
```

```
           [-1.42; -0.46] [-0.14; -0.05] [-0.89; -0.05] [-0.23; -0.11]
interest   -0.46          -0.08          -0.18          -0.01
           (0.40)         (0.04)         (0.34)         (0.04)
           [-1.22; 0.32]  [-0.16; -0.02] [-0.85; 0.48]  [-0.09; 0.07]
pid.pri    6.71           0.61           12.93          1.71
           (0.62)         (0.05)         (1.02)         (0.06)
           [5.43; 7.86]   [0.50; 0.69]   [10.90; 14.86] [1.58; 1.84]
clean2     -0.22          0.01           -0.30          0.00
           (0.34)         (0.03)         (0.34)         (0.04)
           [-0.87; 0.44]  [-0.05; 0.07]  [-0.98; 0.33]  [-0.08; 0.09]
time       -2.55          -0.22          -0.12          0.08
           (0.47)         (0.05)         (0.28)         (0.03)
           [-3.51; -1.61] [-0.32; -0.12] [-0.67; 0.41]  [0.00; 0.14]
           n = 2206       ll = -8431.92  n = 1483       ll = -5343.60
           Mean           Variance       Mean           Variance
Intercept  13.50          4.26           8.32           4.81
           (1.60)         (0.19)         (1.22)         (0.20)
           [10.34; 16.47] [3.90; 4.63]   [5.84; 10.69]  [4.41; 5.24]
female     -0.02          0.09           -0.52          -0.39
           (0.55)         (0.07)         (0.42)         (0.07)
           [-1.11; 1.06]  [-0.05; 0.22]  [-1.40; 0.27]  [-0.53; -0.23]
age100     -2.93          -0.91          1.22           0.02
           (2.23)         (0.27)         (1.66)         (0.30)
           [-7.17; 1.49]  [-1.42; -0.44] [-2.08; 4.42]  [-0.60; 0.54]
educ       -1.10          -0.10          -0.15          -0.44
           (0.35)         (0.04)         (0.27)         (0.05)
           [-1.74; -0.42] [-0.19; -0.03] [-0.67; 0.36]  [-0.54; -0.35]
interest   1.19           0.13           0.28           0.05
           (0.51)         (0.06)         (0.34)         (0.06)
           [0.16; 2.15]   [0.01; 0.25]   [-0.34; 0.94]  [-0.06; 0.19]
pid.pri    0.65           0.52           15.64          1.08
           (0.68)         (0.07)         (0.79)         (0.10)
           [-0.69; 2.02]  [0.38; 0.66]   [14.02; 17.13] [0.91; 1.24]
clean2     0.10           0.09           -0.42          -0.20
           (0.36)         (0.04)         (0.32)         (0.05)
           [-0.59; 0.80]  [0.00; 0.17]   [-1.02; 0.22]  [-0.31; -0.09]
time       0.21           0.19           -0.18          0.09
           (0.26)         (0.03)         (0.15)         (0.02)
           [-0.32; 0.72]  [0.13; 0.25]   [-0.48; 0.10]  [0.05; 0.14]
           n = 972        ll = -3507.07  n = 889        ll = -2905.21
>
> # ==================
> # = end source code =
> # ==================
>
```