Econometrics II TA Session #8

Hiroki Kato

1 Empirical Application of Panel Data Model: Earnings Equation

1.1 Backgruond

A researcher wants to estimate the effect of full-time work experience on wages. He uses a *balanced* panel of 595 individuals from 1976 to 1982, taken from the Panel Study of Income Dynamics (PSID). The *balanced* panel data means that we can observe all individuals every year.

```
dt <- read.csv("./data/wages.csv")
head(dt, 14)
```

```
##    exp wks bluecol ind south smsa married  sex union ed black   lwage id time
## 1    3  32      no   0   yes   no     yes male    no  9    no 5.56068  1    1
## 2    4  43      no   0   yes   no     yes male    no  9    no 5.72031  1    2
## 3    5  40      no   0   yes   no     yes male    no  9    no 5.99645  1    3
## 4    6  39      no   0   yes   no     yes male    no  9    no 5.99645  1    4
## 5    7  42      no   1   yes   no     yes male    no  9    no 6.06146  1    5
## 6    8  35      no   1   yes   no     yes male    no  9    no 6.17379  1    6
## 7    9  32      no   1   yes   no     yes male    no  9    no 6.24417  1    7
## 8   30  34     yes   0    no   no     yes male    no 11    no 6.16331  2    1
## 9   31  27     yes   0    no   no     yes male    no 11    no 6.21461  2    2
## 10  32  33     yes   1    no   no     yes male   yes 11    no 6.26340  2    3
## 11  33  30     yes   1    no   no     yes male    no 11    no 6.54391  2    4
## 12  34  30     yes   1    no   no     yes male    no 11    no 6.69703  2    5
## 13  35  37     yes   1    no   no     yes male    no 11    no 6.79122  2    6
## 14  36  30     yes   1    no   no     yes male    no 11    no 6.81564  2    7
```

The variable `id` and `time` indicate individual and time indexs. We use these two variables to apply panel data models. Additionally, we use the following variables:

- `exp`: years of full-time work experience
- `sqexp`: squared value of `exp`
- `lwage`: logarithm of wage

```r
dt <- dt[,c("id", "time", "exp", "lwage")]
dt$sqexp <- dt$exp^2
summary(dt)
```

```
##        id           time         exp           lwage           sqexp
##  Min.   :  1   Min.   :1   Min.   : 1.00   Min.   :4.605   Min.   :   1.0
##  1st Qu.:149   1st Qu.:2   1st Qu.:11.00   1st Qu.:6.395   1st Qu.: 121.0
##  Median :298   Median :4   Median :18.00   Median :6.685   Median : 324.0
##  Mean   :298   Mean   :4   Mean   :19.85   Mean   :6.676   Mean   : 514.4
##  3rd Qu.:447   3rd Qu.:6   3rd Qu.:29.00   3rd Qu.:6.953   3rd Qu.: 841.0
##  Max.   :595   Max.   :7   Max.   :51.00   Max.   :8.537   Max.   :2601.0
```

To examine the effect of labor experience on wages, we want to estimate the following linear panel data model:

$$\text{lwage}_{it} = \beta_1 \cdot \text{exp}_{it} + \beta_2 \cdot \text{sqexp}_{it} + u_{it}.$$

We can define the regression equation as the `formula` object in `R`. To exclude the intercept, we must specify `-1` in the rhs of regression equation. Thus, in `R`, we define the linear panel data model as follows:

```r
model <- lwage ~ -1 + exp + sqexp
```

1.2 Pooled OLSE

We want to estimate the above regression equation by the OLS method. We will discuss assumptions for implementation. Let $\mathbf{X}_{it}$ be a $1 \times K$ (stochastic) explanatory vector. This vector contains `exp` and `sqexp`. Let Y_{it} be a random variable of outcome, that is, `lwage`. Then, the linear panel data model can be rewritten as follows:

$$Y_{it} = \mathbf{X}_{it}\beta + u_{it}, \quad t = 1, \dots, T, \quad i = 1, \dots, n.$$

Using notations $\underline{\mathbf{X}}_i = (\mathbf{X}'_{i1}, \dots, \mathbf{X}'_{iT})'$ and $\underline{Y}_i = (Y_{i1}, \dots, Y_{iT})'$, and $\underline{u}_i = (u_{i1}, \dots, u_{iT})'$, we can reformulate this model as follows:

$$\underline{Y}_i = \underline{\mathbf{X}}_i\beta + \underline{u}_i, \quad \forall i.$$

Now, we assume

1. (contempraneous) exogneity assumption: $E[\mathbf{X}'_{it}u_{it}] = 0$, $\forall i, t$.
    - This assumption,implies that u_{it} and $\mathbf{X}_{it}$ are orthogonal in the conditional mean sence, $E[u_{it}|\mathbf{X}_{it}] = 0$. However, this assumption does not imply u_{it} is uncorrelated with the explanatory variables in all time periods (strictly exogeneity), that is, $E[u_{it}|\mathbf{X}_{i1}, \dots, \mathbf{X}_{iT}] = 0$. This assumption palces no restriction on the relationship between $\mathbf{X}_{is}$ and u_{it} for $s \neq t$.

2. $E[\underline{\mathbf{X}}_i'\underline{\mathbf{X}}_i] \succ 0$.

Under these two assumptions, the true parameter is given by

$$\beta = E[\underline{\mathbf{X}}_i'\underline{\mathbf{X}}_i]^{-1}E[\underline{\mathbf{X}}_i'\underline{Y}_i].$$

Hence, the OLSE (pooled OLSE) is given by

$$\hat{\beta} = \left(\frac{1}{n}\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\underline{\mathbf{X}}_i\right)^{-1}\left(\frac{1}{n}\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\underline{Y}_i\right) = \left(\frac{1}{n}\sum_{i=1}^{n}\sum_{t=1}^{T}\mathbf{X}_{it}'\mathbf{X}_{it}\right)^{-1}\left(\frac{1}{n}\sum_{i=1}^{n}\sum_{t=1}^{T}\mathbf{X}_{it}'Y_{it}\right).$$

Using the full matrix notation, the OLS estimator is

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'Y),$$

where $\mathbf{X} = (\underline{\mathbf{X}}_1, \dots, \underline{\mathbf{X}}_n)'$ and $Y = (\underline{Y}_1, \dots, \underline{Y}_n)'$.

In `R` programming, the `lm` function provides the pooled OLSE in the context of panel data model. Another way is the `plm` function in the package `plm`. When you want to estimate pooled OLS by the `plm` function, you need to specify `model = "pooling"`. Moreover, you should specify individual and time index using `index` augment. This augment passes `index = c("individual index", "time index")`.

```
bols1 <- lm(model, data = dt)

library(plm)
bols2 <- plm(model, data = dt, model = "pooling", index = c("id", "time"))
```

The pooled OLS estimator is consistent and asymptotically normally distributed.

$$\sqrt{n}(\hat{\beta} - \beta) \sim N_{\mathbb{R}^K}(0, A^{-1}BA^{-1}),$$

where $A = E[\underline{\mathbf{X}}_i'\underline{\mathbf{X}}_i]$ and $B = E[\underline{\mathbf{X}}_i'\underline{u}_i\underline{u}_i'\underline{\mathbf{X}}_i]$. The consistent estimator of A and B is given by

$$\hat{A} = \frac{1}{n}\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\underline{\mathbf{X}}_i,$$

$$\hat{B} = \frac{1}{n}\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\hat{\underline{u}}_i\hat{\underline{u}}_i'\underline{\mathbf{X}}_i.$$

Thus, estimator of asymptotic variance of the pooled OLSE is

$$\widehat{Asyvar}(\hat{\beta}) = \left(\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\underline{\mathbf{X}}_i\right)^{-1}\left(\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\underline{u}_i\underline{u}_i'\underline{\mathbf{X}}_i\right)\left(\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\underline{\mathbf{X}}_i\right)^{-1}.$$

Using the full matrix notations, we can reformulate

$$\widehat{Asyvar}(\hat{\beta}) = (\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'U\mathbf{X})(\mathbf{X}'\mathbf{X})^{-1},$$

where

$$U = \begin{pmatrix} \hat{\underline{u}}_1\hat{\underline{u}}_1' & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \hat{\underline{u}}_2\hat{\underline{u}}_2' & \cdots & \mathbf{0} \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \hat{\underline{u}}_n\hat{\underline{u}}_n' \end{pmatrix}.$$

The standard errors calculated by this matrix is called *robust standard errors clustered by individuals*.

In `R`, the `lm` and `plm` function provide the standard errors based on $\widehat{Asyvar}(\hat{\beta}) = \hat{\sigma}^2(X'X)^{-1}$, where $\hat{\sigma}^2 = \hat{u}\hat{u}'/(nT - K)$ and $\hat{u} = Y - X\hat{\beta}$. There are two ways to obtain cluster robust standard errors. The first way is to caclulate by yourself. The second way is to use the `coeftest` function in the package `lmtest`. When you use this function, we should use the `plm` function to estimate the pooled OLSE, and the `vcovHC` function (the package `sandwich`) in the `vcov` augment of `coeftest` fucntion.

```
# Setup
N <- length(unique(dt$id)); T <- length(unique(dt$time))
X <- model.matrix(bols1); k <- ncol(X)

# Inference
uhat <- bols1$residuals
uhatset <- matrix(0, nrow = nrow(X), ncol = nrow(X))

i_from <- 1; j_from <- 1
for (i in 1:max(dt$id)) {
  x <- as.numeric(rownames(dt))[dt$id == i]
  usq <- uhat[x] %*% t(uhat[x])
  i_to <- i_from + nrow(usq) - 1
  j_to <- j_from + ncol(usq) - 1
  uhatset[i_from:i_to, j_from:j_to] <- usq
  i_from <- i_to + 1; j_from <- j_to + 1
}

Ahat <- t(X) %*% X
Bhat <- t(X) %*% uhatset %*% X
vcovols <- solve(Ahat) %*% Bhat %*% solve(Ahat)
seols <- sqrt(diag(vcovols))

# Easy way
library(lmtest)
library(sandwich)
easy_cluster <- coeftest(
  bols2, vcov = vcovHC(bols2, type = "HC0", cluster = "group"))
```

The result is shown in the first column of Table 1. The partial effect of experience represents the percent change of wages. Thus,

$$(\text{\% Change of Wage}) = 64.6 - 2 \cdot 1.3 \cdot \text{exp}.$$

For example, wages increase by 12.99% at a mathematical mean of labor experience (`exp`). Moreover, this result implies diminishing marginal returns of labor experience.

1.3 Feasible GLSE

Adding and assumption of the conditional variance of $\underline{u}_i$ allows for using the Generalized Ordinary Squares method. To implement, we assume

1. $E[\underline{X}_i \otimes \underline{u}_i] = 0$. A sufficient condition to satisfy this relationship is $E[\underline{u}_i|\underline{X}_i] = 0$. This assumption implies $E[\underline{X}_i'\Omega^{-1}\underline{u}_i] = 0$ where $\Omega = E[\underline{u}_i\underline{u}_i']$ is $T \times T$ matrix.
2. $\Omega \succ 0$ and $E[\underline{X}_i'\Omega^{-1}\underline{X}_i'] \succ 0$.

The GLS estimator is given by

$$\hat{\beta}_{GLS} = \left(\frac{1}{n} \sum_{i=1}^{n} \underline{\mathbf{X}}_i'\Omega^{-1}\underline{\mathbf{X}}_i \right)^{-1} \left(\frac{1}{n} \sum_{i=1}^{n} \underline{\mathbf{X}}_i'\Omega^{-1}\underline{Y}_i \right).$$

Under two assumptions, this estimator is weakly consistent.

In the feasible GLS method, we replace the unknown Ω with a consistent estimator. Here, we consider the two-step FGLS: obtain the OLS estimator and residuals; replace Ω by it. Then, the unknown Ω is replaced by

$$\hat{\Omega} = \frac{1}{n} \sum_{i=1}^{n} \hat{\underline{u}}_i\hat{\underline{u}}_i',$$

where $\hat{\underline{u}}_i = \underline{Y}_i - \underline{\mathbf{X}}_i\hat{\beta}_{OLS}$.

Thus, the FGLS estimator is

$$\hat{\beta}_{FGLS} = \left(\frac{1}{n} \sum_{i=1}^{n} \underline{\mathbf{X}}_i'\hat{\Omega}^{-1}\underline{\mathbf{X}}_i \right)^{-1} \left(\frac{1}{n} \sum_{i=1}^{n} \underline{\mathbf{X}}_i'\hat{\Omega}^{-1}\underline{Y}_i \right).$$

Using the full matrix notations,

$$\hat{\beta}_{FGLS} = \{\mathbf{X}'(I_n \otimes \hat{\Omega}^{-1})\mathbf{X}\}^{-1}\{\mathbf{X}'(I_n \otimes \hat{\Omega}^{-1})Y\}.$$

In the `R` programming, there are two ways to obtain the FGLS estimator. The first way is to caclulate by yourself. The second way is to use the `pggls` function in the package `plm`. When you use the `pggls` function, you should specify individual and time indexs using `index` augment, and type in `model = "pooling"`.

```r
# Setup
X <- model.matrix(model, dt); k <- ncol(X)
y <- dt$lwage
N <- length(unique(dt$id)); T <- length(unique(dt$time))

# Estimator of Omega
uhat <- bols1$residuals

Omega_sum <- matrix(0, ncol = T, nrow = T)
for (i in 1:N) {
  x <- as.numeric(rownames(dt))[dt$id == i]
  Omega_sum <- uhat[x] %*% t(uhat[x]) + Omega_sum
}
Omega <- Omega_sum/N

# FGLS estimator
kroOmega <- diag(N) %x% solve(Omega)
bfgls <- solve(t(X) %*% kroOmega %*% X) %*% (t(X) %*% kroOmega %*% y)

# Easy way!!!
easy_fgls <- pggls(
  model, data = dt, index = c("id", "time"), model = "pooling")
```

The asymptotic distribution of the FGLS estimator is given by

$$\sqrt{n}(\hat{\beta}_{FGLS} - \beta) \sim N_{\mathbb{R}^K}(0, A^{-1}BA^{-1}),$$

where $A = E[\underline{\mathbf{X}}_i'\Omega^{-1}\underline{\mathbf{X}}_i]$ and $B = E[\underline{\mathbf{X}}_i'\Omega^{-1}\underline{u}_i\underline{u}_i'\Omega^{-1}\underline{\mathbf{X}}_i]$. The consistent estimator of A and B is

$$\hat{A} = \frac{1}{n}\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\hat{\Omega}^{-1}\underline{\mathbf{X}}_i,$$

$$\hat{B} = \frac{1}{n}\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\hat{\Omega}^{-1}\underline{\hat{u}}_i^{FLGS}\underline{\hat{u}}_i^{FGLS'}\hat{\Omega}^{-1}\underline{\mathbf{X}}_i,$$

where $\underline{u}_i^{FLGS} = \underline{Y}_i - \underline{\mathbf{X}}_i\hat{\beta}_{FGLS}$. Thus, estimator of asymptotic variance of the FGLS estimator is

$$\widehat{Asyvar}(\hat{\beta}_{FGLS}) = \left(\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\hat{\Omega}^{-1}\underline{\mathbf{X}}_i\right)^{-1}\left(\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\hat{\Omega}^{-1}\underline{u}_i^{FLGS}\underline{u}_i^{FGLS'}\hat{\Omega}^{-1}\underline{\mathbf{X}}_i\right)\left(\sum_{i=1}^{n}\underline{\mathbf{X}}_i'\hat{\Omega}^{-1}\underline{\mathbf{X}}_i\right)^{-1}.$$

Using the full matrix notations,

$$\widehat{Asyvar}(\hat{\beta}_{FGLS}) = \{\mathbf{X}'(I_n \otimes \hat{\Omega}^{-1})\mathbf{X}\}^{-1}\{\mathbf{X}'(I_n \otimes \hat{\Omega}^{-1})U(I_n \otimes \hat{\Omega}^{-1})\mathbf{X}\}^{-1}\{\mathbf{X}'(I_n \otimes \hat{\Omega}^{-1})\mathbf{X}\}^{-1},$$

where

$$U = \begin{pmatrix} \hat{\underline{u}}_1^{FLGS} \hat{\underline{u}}_1^{FGLS\prime} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \hat{\underline{u}}_2^{FLGS} \hat{\underline{u}}_2^{FGLS\prime} & \cdots & \mathbf{0} \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \hat{\underline{u}}_n^{FLGS} \hat{\underline{u}}_n^{FGLS\prime} \end{pmatrix}.$$

In the `R` programming, you need to caclculate by yourself. The `pggls` function provides the FGLS estimator. However, this function calculates standard errors, assuming *system homoskedasticity*, that is, $E[\underline{\mathbf{X}}_i'\Omega^{-1}\underline{u}_i\underline{u}_i'\Omega^{-1}\underline{\mathbf{X}}_i] = E[\underline{\mathbf{X}}_i'\Omega^{-1}\underline{\mathbf{X}}_i]$. If you can rationale this assumption, the `bggls` fucntion is the easiest way to carry out statistical inference.

```
ufgls <- y - X %*% bfgls
uhatset <- matrix(0, nrow = nrow(X), ncol = nrow(X))
i_from <- 1; j_from <- 1
for (i in 1:max(dt$id)) {
  x <- as.numeric(rownames(dt))[dt$id == i]
  usq <- uhat[x] %*% t(uhat[x])
  i_to <- i_from + nrow(usq) - 1
  j_to <- j_from + ncol(usq) - 1
  uhatset[i_from:i_to, j_from:j_to] <- usq
  i_from <- i_to + 1; j_from <- j_to + 1
}

Ahat <- t(X) %*% kroOmega %*% X
Bhat <- t(X) %*% kroOmega %*% uhatset %*% kroOmega %*% X
vcovfgls <- solve(Ahat) %*% Bhat %*% solve(Ahat)
sefgls <- sqrt(diag(vcovfgls))
```

The result is shown in the second column of Table 1. The partial effect of experience represents the percent change of wages. Thus,

$$(\text{\% Change of Wage}) = 52.9 - 2 \cdot 0.9 \cdot \text{exp}.$$

For example, wages increase by 17.17% at a mathematical mean of labor experience (`exp`).

1.4 Fixed Effect Model

To examine the effect of labor experience on wages, we introduce unobserved heterogeneity such as ability. The unobserved effects model is given by

$$\text{lwage}_{it} = \beta_1 \cdot \text{exp}_{it} + \beta_2 \cdot \text{sqexp}_{it} + c_i + u_{it},$$

where c_i is unobserved component which is constant over time, u_{it} is the idiosyncratic error term. The fixed effect model treats c_i as a parameter to be estimated for each cross section unit i.

We generalize the unobserved effects model as follows:

$$Y_{it} = \mathbf{X}_{it}\beta + c_i + u_{it}, \quad t = 1, \dots, T, \quad i = 1, \dots, n.$$

Using notations $\underline{\mathbf{X}}_i = (\mathbf{X}'_{i1}, \dots, \mathbf{X}'_{iT})'$ and $\underline{Y}_i = (Y_{i1}, \dots, Y_{iT})'$, and $\underline{u}_i = (u_{i1}, \dots, u_{iT})'$, we can reformulate this model as follows:

$$\underline{Y}_i = \underline{\mathbf{X}}_i\beta + \iota c_i + \underline{u}_i, \quad \forall i,$$

where $\iota = (1, \dots, 1)'$ is $T \times 1$ vector.

To implement the fixed effect model, we assume the following three assumptions:

1. Strict exogeneity: $E[u_{it}|\mathbf{X}_{i1}, \dots, \mathbf{X}_{iT}, c_i] = 0$.
2. Full rank: $rank(\sum_t E[\ddot{\mathbf{X}}'_{it}\ddot{\mathbf{X}}_{it}]) = rank(E[\underline{\ddot{\mathbf{X}}}'_i\underline{\ddot{\mathbf{X}}}_i]) = K$ where $\ddot{\mathbf{X}}_{it} = \mathbf{X}_{it} - T^{-1}\sum_t \mathbf{X}_{it}$.
3. homoskedasticity: $E[\underline{u}_i\underline{u}'_i|\mathbf{X}_{i1}, \dots, \mathbf{X}_{iT}, c_i] = \sigma_u^2 I_T$.

To obtain the FE estimator, we consider the within transformation first. Averaging the unobserved effects model for individual i and time t over time yields

$$\bar{Y}_i = \bar{\mathbf{X}}_i\beta + c_i + \bar{u}_i,$$

where $\bar{Y}_i = T^{-1}\sum_t Y_{it}$, $\bar{\mathbf{X}}_i = T^{-1}\sum_t \mathbf{X}_{it}$, and $\bar{u}_i = T^{-1}\sum_t u_{it}$. Subtracting this equation from the original one for each t yields

$$Y_{it} - \bar{Y}_i = (\mathbf{X}_{it} - \bar{\mathbf{X}}_i)\beta + (u_{it} - \bar{u}_i) \Leftrightarrow \ddot{Y}_{it} = \ddot{\mathbf{X}}_{it}\beta + \ddot{u}_{it}.$$

Note that $E[\ddot{u}_{it}|\ddot{\mathbf{X}}_{i1}, \dots, \ddot{\mathbf{X}}_{iT}] = 0$ under the first assumption. Using the T system of equation, the within transformation is

$$Q_T\underline{Y}_i = Q_T\underline{\mathbf{X}}_i\beta + Q_T\underline{u}_i \Leftrightarrow \underline{\ddot{Y}}_i = \underline{\ddot{\mathbf{X}}}_i\beta + \underline{\ddot{u}}_i.$$

where $Q_T = I_T - \iota(\iota'\iota)^{-1}\iota$ is *time-demeaning matrix*, and $Q_T\iota = 0$. Using the matrix notations, the within transformation is

$$(I_n \otimes Q_t)Y = (I_n \otimes Q_t)X\beta + (I_n \otimes Q_t)u \Leftrightarrow \ddot{Y} = \ddot{X}\beta + \ddot{u}.$$

Before showing the FE estimator, I will show $Q_T\underline{Y}_i = Y_{it} - T^{-1}\sum_t Y_{it}$, using `R`. As an illustration, we caclulate time-demeaned outcome variable for $i = 1$, $\ddot{Y}_{1t}$. R snippet is as follows:

```r
# extract outcome variables for i = 1
i <- as.numeric(rownames(dt))[dt$id == 1]
y1 <- dt$lwage[i]

# deviation from mean
Ydev1 <- y1 - mean(y1)
print("Deviation from mean across time"); Ydev1
```

```
## [1] "Deviation from mean across time"

## [1] -0.40407857 -0.24444857  0.03169143  0.03169143  0.09670143  0.20903143
## [7]  0.27941143
```

```
# time demean-matrix
T <- length(y1)
vec1 <- rep(1, T)
Qt <- diag(T) - vec1 %*% solve(t(vec1) %*% vec1) %*% t(vec1)
Ydev2 <- Qt %*% y1
print("Time-demeaning matrix"); Ydev2
```

```
## [1] "Time-demeaning matrix"

##             [,1]
## [1,] -0.40407857
## [2,] -0.24444857
## [3,]  0.03169143
## [4,]  0.03169143
## [5,]  0.09670143
## [6,]  0.20903143
## [7,]  0.27941143
```

The FE estimator is given by

$$\hat{\beta}_{FE} = \left(\frac{1}{n} \sum_{i=1}^{n} \ddot{\underline{\mathbf{X}}}_i' \ddot{\underline{\mathbf{X}}}_i \right)^{-1} \left(\frac{1}{n} \sum_{i=1}^{n} \ddot{\underline{\mathbf{X}}}_i' \ddot{\underline{Y}}_i \right) = (\ddot{X}'\ddot{X})^{-1}(\ddot{X}'\ddot{Y}).$$

In the `R` programming, there are two ways to obtain the FE estimator. The first way is to caclulate by yourself. The second way is to use the `plm` function. When you use the `plm` function, you need to specify `model = "within"` to implement the FE model.

```
# Setup
X <- model.matrix(model, dt); k <- ncol(X)
y <- dt$lwage
N <- length(unique(dt$id)); T <- length(unique(dt$time))

# FE estimator
i <- rep(1, T)
Qt <- diag(T) - i %*% solve(t(i) %*% i) %*% t(i)
Ydev <- diag(N) %x% Qt %*% y
Xdev <- diag(N) %x% Qt %*% X
bfe <- solve(t(Xdev) %*% Xdev) %*% t(Xdev) %*% Ydev

# Awesome way !!!
plmfe <- plm(model, data = dt, index = c("id", "time"), model = "within")
```

Under the third assumption, asymptotic distribution of the FE estimator is given by

$$\sqrt{n}(\hat{\beta}_{FE} - \beta) \sim N_{\mathbb{R}^K}(0, \sigma_u^2 E[\underline{\ddot{\mathbf{X}}}_i' \underline{\ddot{\mathbf{X}}}_i]).$$

The consistent estimator of the asymptotic variance of the FE estimator is

$$\widehat{Asyvar}(\hat{\beta}_{FE}) = \hat{\sigma}_u^2 \left(\sum_{i=1}^n \underline{\ddot{\mathbf{X}}}_i' \underline{\ddot{\mathbf{X}}}_i \right)^{-1} = \hat{\sigma}_u^2 (\ddot{X}'\ddot{X})^{-1},$$

where $\hat{\sigma}_u^2 = \frac{1}{n(T-1)-K} \sum_i \sum_t \hat{\ddot{u}}_{it}$, and $\hat{\ddot{u}}_{it} = \ddot{Y}_{it} - \ddot{\mathbf{X}}_{it}\hat{\beta}_{FE}$.

In the R programming, the `plm` function also returns standard errors, $\hat{\sigma}_u^2(\ddot{X}'\ddot{X})^{-1}$. Of course, you can compute the standard errors manually. The sample code is as follows:

```
uhat <- Ydev - Xdev %*% bfe
sigmahat <- sum(uhat^2)/(N*(T-1)-k)
vcovfe <- sigmahat * solve(t(Xdev) %*% Xdev)
sefe <- sqrt(diag(vcovfe))
```

The result is shown in the third column in Table 1. The partial effect of experience represents the percent change of wages. Thus,

$$(\% \text{ Change of Wage}) = 11.4 - 2 \cdot 0.04 \cdot \text{exp}.$$

For example, wages increase by 9.812% at a mathematical mean of labor experience (`exp`).

1.5 Random Effect Model

Again, consider the unobserved effects model:

$$Y_{it} = \mathbf{X}_{it}\beta + c_i + u_{it}, \quad t = 1, \dots, T, \quad i = 1, \dots, n.$$

The random effect model treats c_i as a random variable. Thus, the variable c_i is put into the error term. We reformulate the model as follows:

$$Y_{it} = \mathbf{X}_{it}\beta + v_{it},$$

where $v_{it} = c_i + u_{it}$. Using notations $\underline{\mathbf{X}}_i = (\mathbf{X}_{i1}', \dots, \mathbf{X}_{iT}')'$ and $\underline{Y}_i = (Y_{i1}, \dots, Y_{iT})'$, and $\underline{u}_i = (u_{i1}, \dots, u_{iT})'$, we can reformulate this model as follows:

$$\underline{Y}_i = \underline{\mathbf{X}}_i\beta + \underline{v}_i,$$

where $\underline{v}_i = \iota c_i + \underline{u}_i$, and $\iota = (1, \dots, 1)'$ is $T \times 1$ vector.

To implement the RE model, we assume

1. Strict exogeneity: $E[u_{it}|\mathbf{X}_{i1}, \dots, \mathbf{X}_{iT}, c_i] = 0$.
2. Orthogonality between c_i and $\mathbf{X}_{it}$: $E[c_i|\mathbf{X}_{i1}, \dots, \mathbf{X}_{iT}] = 0$.
3. Full rank: $rank(E[\underline{\mathbf{X}}_i'\Omega^{-1}\underline{\mathbf{X}}_i]) = K$.

4. $E[\underline{u}_i\underline{u}_i'|\mathbf{X}_{i1},\dots,\mathbf{X}_{iT},c_i] = \sigma_u^2 I_T$, and $E[c_i^2|\mathbf{X}_{i1},\dots,\mathbf{X}_{iT}] = \sigma_c^2$.

Using the FGLS method through the introduction of Σ, we can obtain the FGLS-type RE estimator as follows:

$$\hat{\beta}_{RE} = \left(\frac{1}{n}\sum_{i=1}^n \underline{\mathbf{X}}_i'\hat{\Omega}^{-1}\underline{\mathbf{X}}_i\right)^{-1}\left(\frac{1}{n}\sum_{i=1}^n \underline{\mathbf{X}}_i'\hat{\Omega}^{-1}\underline{Y}_i\right),$$

where

$$\hat{\Omega} = \hat{\sigma}_u^2 I_T + \hat{\sigma}_c^2 \iota\iota' = \begin{pmatrix} \hat{\sigma}_c^2+\hat{\sigma}_u^2 & \hat{\sigma}_c^2 & \cdots & \hat{\sigma}_c^2 \\ \hat{\sigma}_c^2 & \hat{\sigma}_c^2+\hat{\sigma}_u^2 & \cdots & \hat{\sigma}_c^2 \\ \vdots & \vdots & \cdots & \vdots \\ \hat{\sigma}_c^2 & \hat{\sigma}_c^2 & \cdots & \hat{\sigma}_c^2+\hat{\sigma}_u^2 \end{pmatrix}.$$

The estimator $\hat{\sigma}_u^2$ and $\hat{\sigma}_c^2$ can be obtained by

$$\hat{\sigma}_u^2 = \hat{\sigma}_v^2 - \hat{\sigma}_c^2,$$

$$\hat{\sigma}_v^2 = \frac{1}{nT-K}\sum_{i=1}^n\sum_{t=1}^T \hat{v}_{it}^2,$$

$$\hat{\sigma}_c^2 = \frac{1}{nT(T-1)/2-K}\sum_{i=1}^n\sum_{t=1}^{T-1}\sum_{s=t+1}^{T}\hat{v}_{it}\hat{v}_{is},$$

$$\hat{v}_{it} = Y_{it} - X_{it}\hat{\beta}_{OLS}.$$

In the R programming, the plm function provides the random effect model. However, the procedure is not the feasible GLS method, but the OLS method on a dataset in which all variables are subject to quasi-demeaning [1]. The two procedures generate the same RE estimator. Moreover, the idiosyncratic error and the unobserved component are obtained by differenct approach. To implement the RE model described above, we compute manually.

```
# Setup
X <- model.matrix(model, dt)
y <- dt$lwage
k <- ncol(X)
N <- length(unique(dt$id))
T <- length(unique(dt$time))

# estimator of Omega
pols <- lm(model, dt)
```

[1] The RE estimator by the quasi-demeaning method is simple. First, we calculate quasi-demeaned variables as in $\tilde{Y}_{it} = Y_{it} - \theta\bar{Y}_i$ where $\theta = 1 - (\sigma_u^2/(\sigma_u^2 + T\sigma_c^2))^{1/2}$. Using the matrix notations, $\underline{\tilde{Y}}_i = \tilde{Q}_T\underline{Y}_i$ where $\tilde{Q}_T$ is the quasi-demeaning matrix, which is given by $\tilde{Q}_T = I_T - \theta\iota(\iota'\iota)^{-1}\iota$. After transforming all variables, we estimate $\underline{\tilde{Y}}_i = \underline{\tilde{X}}_i\beta + \underline{\tilde{u}}_i$ by OLS method. The variance-covariance matrix is $\hat{\sigma}(\sum_i \underline{\tilde{X}}_i\underline{\tilde{X}}_i')^{-1}$ where $\hat{\sigma} = \underline{\tilde{Y}}_i - \underline{\tilde{X}}_i\hat{\beta}$. See http://ricardo.ecn.wfu.edu/~cottrell/gretl/random-effects.pdf in detail.

```
vhat <- pols$residuals
sigmav <- sum(vhat^2)/(N*T - k)

sumuc <- matrix(0, nrow = N, ncol = T-1)
for (i in 1:N) {
  for (t in 1:T-1) {
    it <- as.numeric(rownames(dt))[dt$id == i & dt$time == t]
    is <- as.numeric(rownames(dt))[dt$id == i & dt$time > t]
    sumuc[i,t] <- vhat[it] * sum(vhat[is])
  }
}
sigmac <- sum(colSums(sumuc))/((N*T*(T-1))/2-k)
sigmau <- sigmav - sigmac

i <- rep(1, T)
Omega <- sigmau * diag(T) + sigmac * i %*% t(i)
kroOmega <- diag(N) %x% solve(Omega)

# Random effect
bre <- solve(t(X) %*% kroOmega %*% X) %*% t(X) %*% kroOmega %*% y
```

A consistent estimator of asymptotic variance of the RE estimator is given by

$$\widehat{Asyvar}(\hat{\beta}_{RE}) = \left(\underline{\mathbf{X}}_i' \hat{\Omega}^{-1} \underline{\mathbf{X}}_i\right)^{-1}.$$

In the R programming, the `plm` function returns standard errors calculated by variance-covariance matrix of OLS on a quasi-demeaned data. To obtain the FGLS-type standard errors, we compuate manually. The sample code is as follows:

```
vcovre <- solve(t(X) %*% kroOmega %*% X)
sere <- sqrt(diag(vcovre))
```

The result is shown in the fourth column in Table 1. The partial effect of experience represents the percent change of wages. Thus,

$$(\text{\% Change of Wage}) = 39.5 - 2 \cdot 0.6 \cdot \text{exp}.$$

For example, wages increase by 15.68% at a mathematical mean of labor experience (`exp`).

1.6 Hausman Test

The Hausman test provides empirical evidence on choosing between FE and RE model. The null hypothesis of this test is $\mathbf{X}_{it}$ and c_i are independent. If we can reject the null hypothesis, then the FE model is preferred. If we cannot reject the null hypothesis, then the RE model should be used.

Table 1: Effect of Experience on Wages (Standard errors are in parentheses)

	Dependent variable:			
	lwage			
	Pooled OLS	FGLS	Fixed Effect	Random Effect
	(1)	(2)	(3)	(4)
exp	0.646	0.529	0.114	0.395
	(0.011)	(0.010)	(0.002)	(0.006)
sqexp	−0.013	−0.009	−0.0004	−0.006
	(0.0004)	(0.0004)	(0.0001)	(0.0002)
Observations	4,165	4,165	4,165	4,165

The test statistics is

$$\hat{H} = (\hat{\beta}_{RE} - \hat{\beta}_{FE})'\{\widehat{Var}(\hat{\beta}_{RE}) - \widehat{Var}(\hat{\beta}_{FE})\}^{-1}(\hat{\beta}_{RE} - \hat{\beta}_{FE}).$$

The limiting distribution of this test statistics is $\hat{H} \rightarrow \chi^2(K)$.

In the `R` programming, the manual computation is very easy. Alternatively, the `phtest` function in the package `plm` provides the Hausman test. To use the `phtest`, we need to estimate the FE and RE model by the `plm` function.

```
delta <- bre - bfe
diffv <- vcovre - vcovfe
H <- t(delta) %*% solve(diffv) %*% delta
qtchi <- qchisq(0.99, nrow(delta))
paste("The test statistics of Hausman test is ", round(H, 3))
```

```
## [1] "The test statistics of Hausman test is  3999.537"
```

```
paste("The 1% quantile value of chi-sq dist is", round(qtchi, 3))
```

```
## [1] "The 1% quantile value of chi-sq dist is 9.21"
```

In this empirical application, we can reject the null hypothesis at 1% significance level. This implies that we should use the FE model in this application beucase observed covariates and unobserved component are not independent.