Problem Set 2

Prof. Conlon

Due: 3/1/19

% math bold

Packages to Install

The packages used this week are

- estimatr (Tidyverse version of lm function)

Problem 1 (Analytical Exercise)

Consider the estimation of the individual effects model:

$$y_{it} = x_{it}'\beta + \alpha_i + \epsilon_{it},\ \mathbb{E}\left[\epsilon|x_{it}, \alpha_i\right] = 0$$

where $i = \{1, ..., n\}$ and $t = \{1, ..., T\}$.
This exercises ask you to relate the (random effects) GLS estimator $\hat{\beta}_{GLS} = (X_*'X_*)^{-1}X_*'y_*$ to the "within" (fixed-effects) estimator $\hat{\beta}_{FE} = (\dot{X}'\dot{X})\dot{X}'\dot{y}$ and the "between" estimator $\hat{\beta}_{BW} = (\bar{X}'\bar{X})^{-1}\bar{X}'\bar{y}$ where $w = \{x, y\}$:

$$\begin{aligned}
\bar{w}_i &:= \frac{1}{T}\sum_{i=1}^{T} w_i \\
\dot{w}_i &:= w_{it} - \bar{w}_i \\
w_{it,*} &:= w_{it} - (1-\lambda)\bar{w}_i \\
\lambda^2 &= \frac{Var(\epsilon)}{T\,Var(\alpha_i) + Var(\epsilon_{it})}
\end{aligned}$$

1. Express the GLS estimator in terms of $\bar{X}$, $\dot{X}$, $\bar{y}$, $\dot{y}$, λ, and T. Notation:

$$\dot{X}_{it} := X_{it} - \frac{1}{T}\sum_{t=1}^{T} X_{it}$$

$$X_{it}^* := X_{it} - (1-\lambda)\frac{1}{T}\sum_{t=1}^{T} X_{it}$$

where

$$\lambda^2 := \frac{\mathrm{Var}(e_{it})}{T\mathrm{Var}(\alpha_i) + \mathrm{Var}(e_{it})}$$

and

$$\bar{X}_i := \frac{1}{T}\sum_{t=1}^{T} X_{it}$$

Note also

$$X_{it}^* = \dot{X}_{it} + \lambda\bar{X}_i$$

which can be written in stacked form as

$$X^* = \dot{X} + \lambda\mathbf{I}\bar{X}$$

Then

$$\begin{aligned}\hat{\beta}_{\text{GLS}} &= (X^{*\prime}X^*)^{-1}X^{*\prime}Y^* \\ &= ([\dot{X}+\lambda\mathbf{I}\bar{X}]'[\dot{X}+\lambda\mathbf{I}\bar{X}])^{-1}[\dot{X}+\lambda\mathbf{I}\bar{X}]'Y^* \\ &= (\dot{X}'\dot{X}+\bar{X}'\mathbf{I}'\lambda^2\mathbf{I}\bar{X})^{-1}[\dot{X}+\lambda\mathbf{I}\bar{X}]'Y^*\end{aligned}$$

where the cross-product terms $\dot{X}'\lambda\mathbf{I}\bar{X}$ cancel as $\dot{X}$ represents the orthogonal component of a projection of X onto its column-by-column time-average; thus

$$\dot{X}'\mathbf{I}\bar{X} = \mathbf{0}$$

2. Show that there is a matrix R depending on $\bar{X}$, $\dot{X}$, λ and T such that the GLS estimator is a weighted average of the "within" and "between" estimators:

$$\hat{\beta}_{GLS} = R\hat{\beta}_{FE} + (I-R)\hat{B}_W$$

From part (a), we have

$$\hat{\beta}_{\text{GLS}} = (\dot{X}'\dot{X}+\bar{X}'\mathbf{I}'\lambda^2\mathbf{I}\bar{X})^{-1}[\dot{X}+\lambda\mathbf{I}\bar{X}]'(\dot{Y}+\lambda\mathbf{I}\bar{Y})$$

As

$$\dot{X}'\lambda\mathbf{I}\bar{Y} = \mathbf{0}$$

and

$$\lambda\bar{X}'\mathbf{I}'\dot{Y} = \mathbf{0}$$

our expression reduces to

$$\hat{\beta}_{\text{GLS}} = (\dot{X}'\dot{X}+\bar{X}'\mathbf{I}'\lambda^2\mathbf{I}\bar{X})^{-1}[\dot{X}'\dot{Y}+\lambda^2T\bar{X}'\bar{Y}]$$

Let

$$R := (\dot{X}'\dot{X}+\lambda^2T\bar{X}'\bar{X})^{-1}(\dot{X}'\dot{X})$$

and note that

$$(\dot{X}'\dot{X}+\lambda^2T\bar{X}'\bar{X})^{-1}(\dot{X}'\dot{X}) + (\dot{X}'\dot{X}+\lambda^2T\bar{X}'\bar{X})^{-1}\lambda^2T(\bar{X}'\bar{X}) = I$$

Then

$$\begin{aligned}\hat{\beta}_{\text{GLS}} &= R(\dot{X}'\dot{X})^{-1}\dot{X}'\dot{Y} + (I-R)(\bar{X}'\bar{X})^{-1}\bar{X}'\bar{Y} \\ &= R\hat{\beta}_{\text{FE}} + (I-R)\hat{\beta}_{\text{BW}}\end{aligned}$$

3. What happens to the relative weights on the "within" and "between" estimators as we increase the sample size, i.e. $T\to\infty$?
As $T\to\infty$, $R\xrightarrow{p} I$. To see this, note that

$$\lambda^2 \to 0 \quad \text{as } T\to\infty$$

and, under some finite moment assumptions, for a fixed n and a given i,

$$\bar{X}_i = \frac{1}{T}\sum_{t=1}^{T} X_{it} \xrightarrow{p} \mathbb{E}_i[X_{it}]$$

and

$$\bar{X}_i'\bar{X}_i \xrightarrow{p} (\mathbb{E}_i[X_{it}])'\mathbb{E}_i[X_{it}]$$

Similarly for $\dot{X}_{it}$. Thus the second term in the denominator of R converges to zero, and we have

$$R \xrightarrow{p} I$$

4. Suppose that the random effects assumption $\mathbb{E}\big[\alpha_i|x_{i1,...,x_{iT}}\big] = 0$ does not hold. Characterize the bias of the estimators $\hat{\beta}_{FE}$, $\hat{\beta}_W$. (Note: An estimator $\hat{\beta}$ is unbiased if $\mathbb{E}\big[\hat{\beta}\big] = \beta$) Our estimators are

$$\hat{\beta}_{\text{FE}} = (\dot{X}'\dot{X})^{-1}\dot{X}'\dot{Y}$$
$$\hat{\beta}_{\text{BW}} = (\bar{X}'\bar{X})^{-1}\bar{X}'\bar{Y}$$

Note that

$$\begin{aligned}\dot{Y} &= Y - \bar{Y} \\ &= X\beta + \alpha\iota_T + e - [\bar{X}\beta + \alpha\iota_T + \bar{e}] \\ &= \dot{X}\beta + \dot{e}\end{aligned}$$

and

$$\bar{Y} = \bar{X}\beta + \alpha\iota_T + \bar{e}$$

Then

$$\hat{\beta}_{\text{FE}} = (\dot{X}'\dot{X})^{-1}\dot{X}'[\dot{X}\beta + \dot{e}]$$
$$\hat{\beta}_{\text{BW}} = (\bar{X}'\bar{X})^{-1}\bar{X}'[\bar{X}\beta + \alpha\iota_T + \bar{e}]$$

or

$$\hat{\beta}_{\text{FE}} = \beta + (\dot{X}'\dot{X})^{-1}\dot{X}'\dot{e}$$
$$\hat{\beta}_{\text{BW}} = \beta + (\bar{X}'\bar{X})^{-1}\bar{X}'\alpha\iota_T + (\bar{X}'\bar{X})^{-1}\bar{X}'\bar{e}$$

For the first term, the mean-independence assumption, $\mathbb{E}\big[[\big]e_{it}|X_{it}, \alpha_i] = 0$, yields

$$\mathbb{E}\big[[\big]\hat{\beta}_{\text{FE}}|X_{it}, \alpha_i] = \beta + (\dot{X}'\dot{X})^{-1}\dot{X}'\mathbb{E}\big[[\big]\dot{e}|X_{it}, \alpha_i] = \beta$$

For the second estimator,

$$\mathbb{E}\big[[\big]\hat{\beta}_{\text{BW}}|X_{it}, \alpha_i] = \beta + \mathbb{E}\big[[\big](\bar{X}'\bar{X})^{-1}\bar{X}'\alpha\iota_T|X_{it}, \alpha_i] + \mathbb{E}\big[[\big](\bar{X}'\bar{X})^{-1}\bar{X}'\bar{e}|X_{it}, \alpha_i] = \beta + (\bar{X}'\bar{X})^{-1}\bar{X}'\mathbb{E}\big[[\big]\alpha|X]\iota_T$$

5. Use your result from (d) to give a formula for the bias of our random effects estimator $\hat{\beta}_{GLS}$. What happens to the bias as $T \to \infty$.
The bias vanishes as $T \to \infty$, as $R \xrightarrow{p} I$.

Problem 2 (Coding Exercise)

We observe N observations of the random variable X_i where each X_i is drawn from the Weibull distribution:

$$X_i \sim W(\gamma)$$

The probability density function for the Weibull is the following:

$$f(x;\gamma) = \gamma x^{\gamma-1}\exp(-(x^{\gamma})) \;\; ; x \geq 0, \gamma > 0$$

1. Assume our N observations are independent and identically distributed, what is the log-likelihood function?

$$\mathcal{L}(\gamma|x_1, ..., x_N) = \sum_{i=1}^{N} log(\gamma) + log(x_i)(\gamma - 1) - (x_i^{\gamma})$$

2. Calculate the gradient (or first derivative) of your log-likelihood function.

$$\frac{\partial L}{\partial \gamma} = \sum_{i=1}^{N} \frac{1}{\gamma} + log(x_i) - log(x_i)x_i^{\gamma}$$

3. Using the first order condition, what is the MLE estimator for γ?

$$\frac{\partial L}{\partial \gamma} = 0$$

   This is an implicit function in γ, we will need to solve it numerically.

4. Verify that the second order condition guarantees a unique global solution.

$$\frac{\partial^2 L}{\partial \gamma^2} = \sum_{i=1}^{N} \frac{-1}{\gamma^2} - log(x_i)^2 x_i^{\gamma}$$

   This is clearly less than 0 if $\gamma > 0$ and $x_i > 0$.

5. In R, I want you to write a function called mle_weibull that takes two arguments (X, γ), where X is a vector of data and γ is a scalar. The function returns the value of the log-likelihood function you derived in the last part.

```
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
# Calculate maximum likelihood estimate for 1-parameter
#  Weibull distribution
#
# Inputs :
#  X      : vector of data
#  gamma  : numeric value of parameter guess
#
# Outputs :
#  likeli : value of likelihood for data X evaluated
#           at gamma
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#

mle_weibull = function(gamma,X,noise=10^(-10)){
  # Calculate number of observations
  N = length(X)

  # Calculate three parts of the sum
  first_term = N*log(gamma+noise)
  second_term = sum(log(X)*(gamma-1))
  third_term = sum(X^gamma)

  # Likelihood is sum of three terms
  likeli = first_term + second_term - third_term

  return(-likeli)
```

```
}

#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
# Calculate gradient of log-likelihood for 1-parameter
#  Weibull distribution
#
# Inputs :
#  X      : vector of data
#  gamma  : numeric value of parameter guess
#
# Outputs :
#  gradient : value of gradient of log-likelihood
#              for data X evaluated at gamma
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#

gr_weibull = function(gamma,X,noise=10^(-10)){
  # Calculate number of observations
  N = length(X)

  # Calculate three parts of the sum
  first_term = N/(gamma+noise)
  second_term = sum(log(X))
  third_term = sum(log(X)*X^(gamma))

  # Gradient is sum of three terms
  gradient = first_term + second_term - third_term

  return(-gradient)
}
```

1. Optimiziation routines can either be given a first derivative (or gradient) or the optimization routines calculate numerical derivatives. We will be using the R function *optim*, which accepts the first derivative as an argument *gr*.

a. We first want you to run *optim* without supplying a first derivative (leaving gr out of the function). Note, to run optim you will need to supply your data X as an additional parameter at the end of the function. We have provided you with simulated data in the file 'prob_4_simulation.rda' located in the data folder.

b. We now want you to create a new function called gradient, which takes the same two arguments as your likelihood function. Now calculate the MLE using optim with the gradient.

c. Compare both the number of iterations until convergence and your estimated γ values from both runs.

```
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
# ~ Use optim to calculate the MLE estimator of 1-parameter Weibull ~ #
#
# ~~~~~~~~~~~~~~        True Value: Gamma = 2          ~~~~~~~~~~~~~~ #
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#

# Load dataset
load(paste(data_dir,"prob_4_simulation.rda",sep=''))

# Set initial guess for gamma
# I am setting initial gamma guess far away from the ture value (gamma = 2)
#  so that I can show the improvement in performance by with the gradient
```

```
init_gamma = 50

# Run optim using simulated data
run_mle_noGR =optim(init_gamma,mle_weibull,X=sim,lower=c(1),method=c('L-BFGS-B'))
run_mle_GR   =optim(init_gamma,fn=mle_weibull,gr=gr_weibull,X=sim,lower=c(1),method=c('L-BFGS-B'))

cat(paste('Numerical Gradient -> MLE Estimate : ',run_mle_noGR$par,' Iterations : ',run_mle_noGR$counts
cat(paste('Analytical Gradient -> MLE Estimate : ',run_mle_GR$par,' Iterations : ',run_mle_GR$counts['f
```

```
Numerical Gradient -> MLE Estimate :  1.98230716340946  Iterations :  70
Analytical Gradient -> MLE Estimate :  1.98230709769759  Iterations :  70
```