

Grouped Fixed Effects estimators in gretl: the GFE package

Riccardo (Jack) Lucchetti Alessandro Pionati
Francesco Valentini

version 0.1

Abstract

The package provides functions for computing of the Grouped Fixed Effects estimator for panel data models with a grouped structure of unobserved heterogeneity proposed by [Bonhomme and Manresa \(2015a\)](#).

Contents

1	Introduction	2
1.1	Parameter estimation	2
1.2	Number of groups	3
2	Basic usage	4
3	Replication of Bonhomme and Manresa (2015a)	6
4	List of public functions	9
4.1	The function <code>gfe_estimate</code>	9
4.2	The function <code>gfe_printout</code>	11
4.3	The <code>group_fx</code> function	12
4.4	The <code>group_plot</code> function	12
	References	13
	Changelog	13

1 Introduction

This package provides functions to estimate linear models for panel data with “grouped fixed effects” (GFE), as proposed in [Bonhomme and Manresa \(2015a\)](#). In these models, units (countries, firms, individuals) are assumed to belong to a small number of latent groups that share common time patterns. The key idea is to assume the unobserved heterogeneity to be structured in a finite number of latent groups, say G , allowing units within the same group to share common time-varying unobserved components.

We consider the following linear panel data model:¹

$$y_{it} = x'_{it}\beta + \alpha_{g_i,t} + \varepsilon_{it}, \quad (1)$$

where $i = 1, \dots, n$ and $t = 1, \dots, T$; y_{it} is the outcome variable for unit i at time t , x_{it} is a vector of observed covariates, β is the parameter vector of interest. The distinctive element of the model is $\alpha_{g_i,t}$ which captures unobserved heterogeneity, where g_i denotes the group membership of unit i , so that $\alpha_{g_i,t}$ is a group-specific time-varying effect. This structure allows for rich patterns of unobserved heterogeneity, e.g., group-specific trajectories, while avoiding the curse of dimensionality associated with unrestricted unit-time fixed effects. Finally, ε_{it} is an idiosyncratic error term.

A nested specification of the model in Equation (1) allows for group-specific intercepts that are constant over time, that is

$$y_{it} = x'_{it}\beta + \alpha_{g_i} + \varepsilon_{it}. \quad (2)$$

Our package also provides dedicated functions to estimate this restricted version of the model.

1.1 Parameter estimation

The GFE estimator jointly determines: (i) the group assignments $\{g_i\}$; (ii) the parameter vector β , and (iii) the group-specific effects $\{\alpha_{g_i,t}\}$ by minimizing the sum of squared residuals (SSR):

$$\min_{\beta, \{\alpha_{g_i,t}\}, \{g_i\}} \sum_{i=1}^N \sum_{t=1}^T (y_{it} - x'_{it}\beta - \alpha_{g_i,t})^2. \quad (3)$$

This optimization problem is non-convex due to the discrete nature of group assignments. [Bonhomme and Manresa \(2015a\)](#) propose iterative algorithms that alternate between estimating β and the group-time effects conditional on group membership, and updating group assignments conditional on parameter estimates.

¹Note: the panel must be balanced.

Under suitable regularity conditions, the authors show that the GFE estimator is consistent as both the cross-sectional dimension N and the time dimension T grow, when the number of groups G is fixed. In particular, the large- T framework is necessary to guarantee the misclassification of group assignment to be asymptotically negligible.

[Bonhomme and Manresa \(2015a\)](#) discuss two alternative computational strategies for estimating the GFE model parameters, referred to as Algorithms 1 and 2.

Algorithm 1 consists of two steps. In the Update Step, OLS estimates the model parameters for the current group assignments and computes the corresponding SSR. In the Assignment Step, each unit is reassigned to the group that minimizes its SSR. These two steps are repeated alternately until convergence. The entire procedure is then repeated independently for several random initializations, and the solution with the lowest SSR is selected.

Algorithm 2 is designed to be more robust to local minima of the objective function. This is achieved by perturbing the solution obtained in the Assignment Step, thereby exploring a local neighborhood of the objective function.

Algorithm 2 is the default choice, but Algorithm 1 is also available. Since Algorithm 2 is computationally much more demanding, we offer the option of running it in parallel processes using gretl's MPI infrastructure: in this case, the algorithm is initialised from multiple random points of the group assignments.²

1.2 Number of groups

According to [Bonhomme and Manresa \(2015a\)](#), the number of groups G can be selected by minimizing the following Bayesian Information Criterion (BIC), subject to the assumption of a maximum value for G , denoted $G_{\max}$:

$$\text{BIC}(G) = \frac{1}{NT} \sum_{i=1}^N \sum_{t=1}^T (y_{it} - x'_{it} \hat{\beta}(G) - \hat{\alpha}_{\hat{g}_{it}}(G))^2 + \hat{\sigma}^2 \frac{GT + N + K}{NT} \ln(NT), \quad (4)$$

where $\hat{\sigma}^2$ denotes a consistent estimator of the variance of ε_{it} , K is the number of regressors, and $\hat{\beta}(G)$ and $\hat{\alpha}_{\hat{g}_{it}}(G)$ are the estimated parameters under G groups.

²It should be noted that MPI parallelisation is *not* enabled with a standard, off-the-shelf installation of gretl on Windows or MacOS. Although MPI is not strictly necessary for running this package, its usage is strongly recommended. You may want to have a look at the "Gretl + MPI" pdf manual, available under the Help menu of the gretl program, especially Section 9.

With reference to the estimator of σ^2 , we first estimate β , α , and the group assignments $\{g_i\}_{i=1}^N$ using the grouped fixed effects estimator with $G_{\max}$ groups, and then compute the residual variance as

$$\hat{\sigma}^2 = \frac{1}{NT - G_{\max}T - N - K} \sum_{i=1}^N \sum_{t=1}^T (y_{it} - x'_{it}\hat{\beta} - \hat{\alpha}_{g_{it}})^2. \quad (5)$$

2 Basic usage

As a simple example, we estimate a GFE model of the employment elasticity with respect to several production inputs using the dataset in [Arelano and Bond \(1991\)](#), which comprises observations on a panel of 140 U.K. manufacturing companies over the period 1976-1984.

First, we load the package and open the dataset supplied with gretl, and set the random seed for replicability:

```
include GFE.gfn
open abdata.gdt --quiet
set seed 8273647
```

Next we select the variables used in the estimation. The dependent variable is the log of the employment (n), while the regressors are the logarithms of the real product wage (w), gross capital (k), and industry output (ys).

```
list X = w k ys
list ALL = n X
```

The current implementation of the package requires a balanced panel, so we restrict the sample to firms with complete observations over the selected time period.

```
smpl (YEAR > 1976) --restrict
series d = ok(ALL)
smpl pmin(d) == 1 --restrict
```

We then specify six groups and create an options bundle requesting verbose output.

```
nGrp = 6
opts = _(verbose = 1)
```

To illustrate the available estimation methods, we estimate the same model using the two algorithms implemented in the package. We begin with the default choice, the parallelized Algorithm 2.

```
mod = gfe_estimate(n, X, nGrp, opts)
gfe_printout(mod)
```

which, through the `gfe_printout` function, gives us the following output

```
=====
Group fixed effects estimator (algorithm 2)
19 random starts, 10 processes
dependent variable: n
Using 33 units and 8 time periods (264 total observations)
6 time-varying groups (sandwich s.e.)
=====
```

	coefficient	std. error	z	p-value	
w	-0.326559	0.0442799	-7.375	1.64e-13	***
k	0.658681	0.00931556	70.71	0.0000	***
ys	0.0694762	0.260335	0.2669	0.7896	

SSR = 5.52703, Log-likelihood = 135.752
Elapsed time = 1.97608 seconds

The header reports the estimation algorithm, the number of random starting values, the number of processes used, information on the sample, the number of groups, whether the group effects are time varying, and the type of variance-covariance estimator.

Below the coefficients, we have the resulting value for the objective function, the log-likelihood and the time employed for the computation. We then proceed to estimate the same GFE model using Algorithm 1. We override default options by initialising Algorithm 1 over 2048 random starts.

```
opts.algorithm = 1
opts.randstarts = 2048
mod = gfe_estimate(n, X, nGrp, opts)
gfe_printout(mod)
```

We have the following output

```
=====
Group fixed effects estimator (algorithm 1)
method = 2048 random starts
dependent variable: n
Using 33 units and 8 time periods (264 total observations)
6 time-varying groups (sandwich s.e.)
=====
```

	coefficient	std. error	z	p-value	
w	-0.400593	0.0449276	-8.916	4.82e-19	***
k	0.650138	0.0112953	57.56	0.0000	***
ys	0.214517	0.281436	0.7622	0.4459	

SSR = 5.59956, Log-likelihood = 134.031
Elapsed time = 1.1954 seconds

where we see that Algorithm 1 has been employed. Note that the estimation converges to a solution with a slightly higher value of the objective function.

3 Replication of Bonhomme and Manresa (2015a)

This section illustrates the use of the package to replicates the empirical application on income and democracy in Bonhomme and Manresa (2015a),³ Section 4, using the data of Acemoglu et al. (2008).⁴

In particular, it reproduces the model-selection exercise reported in Table S.XI of Bonhomme and Manresa (2015b), based on the Bayesian Information Criterion described in Section 1.2. Data are relative to $N = 90$ countries, observed over $T = 7$ five-year intervals. The equation below describes a dynamic panel regression where the outcome is a measure of democracy, measured by the Freedom House index, on its lagged value and lagged log GDP per capita, allowing for unrestricted group-specific time-varying patterns of unobserved heterogeneity, so that

$$\text{democracy}_{it} = \beta_1 \text{democracy}_{i,t-1} + \beta_2 \log(\text{GDPpc}_{i,t-1}) + \alpha_{g_i,t} + \varepsilon_{it}. \quad (6)$$

First, we load the GFE package by

```
include GFE.gfn
open Acemoglu_etal.gdt --quiet
```

and open the dataset. We assign the dependent variable and the list of regressors via standard gretl commands

```
list X = lag_dem lag_income
```

We specify the maximum number of latent groups considered in the BIC model-selection procedure. Following Bonhomme and Manresa (2015b), we set $G_{max} = 15$. Then we assign inputs to the bundle opts.

```
G = 15
opts = _(tvar=1, algorithm=2, verbose=1, randstarts=24, BICselect=1)
```

where we select time-varying grouped fixed effects, parallelized Algorithm 2, a verbose output and 24 random starts. The final input BICselect=1 specifies that we want the model selection across $G_{max} = 15$ groups. Then, we estimate Model (6):

³The original replication material is publicly available at <https://www.econometricsociety.org/publications/econometrica/2015/05/01/Grouped-Patterns-of-Heterogeneity-in-Panel-Data>.

⁴The full script is available in the package examples directory under the name BonMan_replication.inp.

```
bundle mod = gfe_estimate(fhpolrigaug, X, G, opts)
gfe_printout(mod, TRUE)
```

and get the following output:

```
=====
      BIC model selection (Gmax = 15)
=====
93.3% done
-----
  G      BIC      SSR
-----
  1 0.05188    24.3
  2 0.04575    19.85
  3 0.04154    16.6
  4 0.03886    14.32
  5 0.03706    12.59
  6 0.03568    11.13
  7 0.03492    10.06
  8 0.03458     9.251
  9 0.03421     8.427
 10 0.03407     7.749 *
 11 0.03417     7.218
 12 0.03442     6.781
 13 0.03474     6.391
 14 0.03505     5.996
 15 0.03551     5.688
-----
estimation done (obj = 7.74906)
```

where the `gfe_estimate` function reports the BIC and the corresponding value of the objective function (SSR) for each candidate number of groups. The minimum BIC is attained at $G = 10$, indicated by the asterisk, and this specification is therefore selected for estimation.

The second part of the output reports the estimates for the selected specification with $G = 10$ groups, using the `gfe_printout` function. The header summarizes the estimation settings, including the algorithm used, the number of random starting values, the number of computational processes, and the dimensions of the panel.

The coefficient table reproduces the estimates reported in Table S.XI of [Bonhomme and Manresa \(2015b\)](#). Standard errors are computed using the large-N, large-T sandwich estimator of [Bonhomme and Manresa \(2015a\)](#), clustered at the country level. The reported elapsed time refers to the complete model-selection procedure over all candidate values of G . Finally, the output reports the number of countries assigned to each estimated group. This last table is displayed only when the optional second argument of `gfe_printout` is set to `TRUE`.

```

=====
Group fixed effects estimator (algorithm 2)
24 random starts, 10 processes
dependent variable: fhpolrigaug
Using 90 units and 7 time periods (630 total observations)
10 time-varying groups (sandwich s.e.)
=====

```

	coefficient	std. error	z	p-value
lag_dem	0.277179	0.0487607	5.684	1.31e-08 ***
lag_income	0.0752583	0.00803139	9.371	7.22e-21 ***

```

SSR = 7.74906, Log-likelihood = 491.485
Elapsed time = 36.9484 seconds

```

Number of units per group:

```

Group 1: 3
Group 2: 29
Group 3: 3
Group 4: 11
Group 5: 7
Group 6: 3
Group 7: 18
Group 8: 2
Group 9: 6
Group 10: 8

```

We can also visualize the estimated group-specific trajectories $\hat{a}_{g,t}$ by using

```
group_plot(mod)
```

which produces Figure 1. Finally, we compute the cumulative income coefficient via the delta method.

```

function matrix CumIncomeCoef(matrix theta, matrix V)
    coef = theta[2]/(1 - theta[1])
    Jacobian = 1/(1 - theta[1]) * {coef; 1}
    se_coef = sqrt(qform(Jacobian', V))
    return coef ~ se_coef
end function
ci = CumIncomeCoef(mod.coef, mod.vcv)
modprint ci "Cum.Income"

```

to get the following output

	coefficient	std. error	z	p-value
Cum.Income	0.104117	0.00878266	11.85	2.03e-32 ***

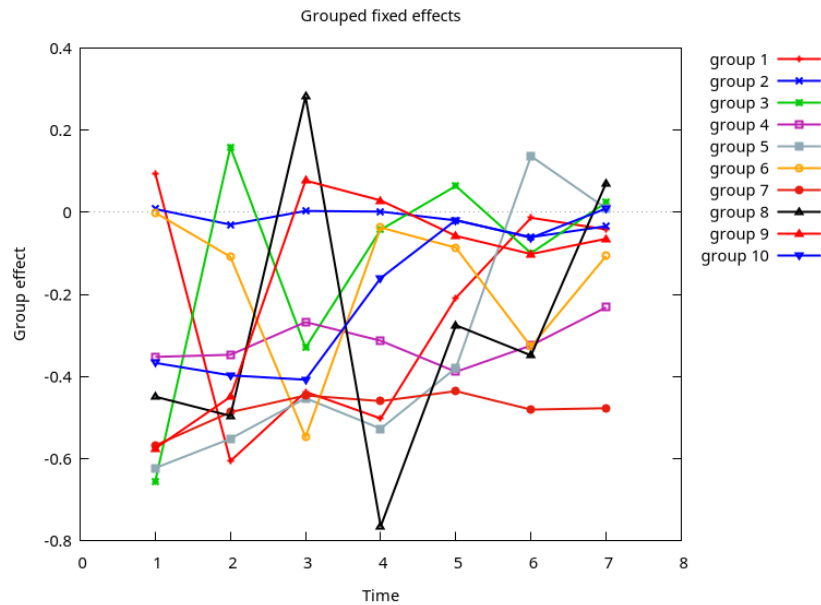


Figure 1: Estimated group trajectories

4 List of public functions

The package provides 4 public functions:

- `gfe_estimate`: estimates a Grouped Fixed Effects (GFE) model, providing various algorithmic choices for group assignment and estimation;
- `gfe_printout`: prints a formatted summary table of the estimated GFE model results to the console;
- `group_fx`: returns a series representing the estimated group-specific effects mapped over time for each cross-sectional unit;
- `group_plot`: generates a graphical representation of the estimated group fixed effects.

4.1 The function `gfe_estimate`

```
gfe_estimate(series y, list X, scalar G, bundle opt_in[null])
```

Return type : bundle

y : dependent variable;

X : list of regressors;

G : number of groups (or maximum number of groups if BICselect=1);

opt_in : an optional bundle containing various configuration parameters,
null is the default.

The function requires a strictly balanced panel dataset; if the panel is unbalanced, the execution will halt with an error.

The behavior of the estimation is largely governed by the optional bundle opt_in. The key keys that can be passed inside opt_in include:

- algorithm: a scalar defining the estimation routine. algorithm=1 uses Algorithm 1; algorithm=2 (default) employs local search via Algorithm 2.
- tvar: a boolean indicating whether time-varying unit heterogeneity is considered. It is set to 1 as a default.
- parallel: a scalar determining the number of MPI processes/cores to use. By default, it automatically detects and uses the available MPI cores via \$sysinfo. Set to 0 to disable parallelization.
- method: a string defining the group initialization routine. Valid options are:
 - "random": draws for a discrete uniform distribution (default);
 - "given": uses the user-specified matrix provided in inigrp.
- randstarts: number of random starts; if set to NA (the default), the number depends on the algorithm chosen: for algorithm 1, it is 1024 if $G < 8$, 4096 otherwise. For algorithm 2, it is the integer part of $8\sqrt{G}$.
- inigrp: a matrix containing the initial group assignments. If algorithm==2, this is automatically optimized and updated internally.
- BICselect: a boolean controlling the BIC-based model selection. If set to 1, the function evaluates models from 1 to G groups and returns the one that minimizes the BIC criterion. 0 is the default.
- verbose: an integer (0 to 2) controlling the verbosity of the output. 1 is the default.

The output of the function is a bundle containing the following top-level keys:

- `opts`: a bundle. The full set of options actually used for the estimation (includes `algorithm`, `ngrp`, `tvar`, etc.).
- `N`: a scalar. The number of cross-sectional units (N).
- `T`: a scalar. The number of time periods (T).
- `nreg`: a scalar. The number of regressors (k), excluding the constant.
- `lnl`: a scalar. The log-likelihood of the estimated model.
- `obj`: a scalar. The value of the minimized objective function (Sum of Squared Residuals, SSR).
- `drop`: a scalar. A boolean indicator for dropped observations or missing values/empty groups management.
- `etime`: a scalar. The total elapsed time for the estimation, in seconds.
- `coeff`: a $k \times 1$ matrix. The estimated coefficients for the regressors.
- `stderr`: a $k \times 1$ matrix. The estimated standard errors for the coefficients (clustered sandwich standard errors).
- `vcv`: a $k \times k$ matrix. The covariance matrix of the estimated parameters.
- `a`: a matrix. The estimated group fixed effects/parameters.
- `grp`: a $N \times 1$ matrix. The final group assignment vector for each cross-sectional unit.
- `initgrp`: a $N \times 1$ matrix. The initial group assignment vector.
- `parnames`: an array of strings. The names of the covariates.
- `X`: a list. The list of regressors used in the model.
- `depvarname`: a string. The name of the dependent variable.

4.2 The function `gfe_printout`

```
gfe_printout(bundle mod, bool showfreq[0])
```

Return type : `void`

`mod` : the estimation model bundle returned by `gfe_estimate`;

`showfreq` : a boolean flag controlling whether to print the frequency distribution of units across groups, 0 is the default.

The function prints a comprehensive summary of the estimation results contained in the model bundle. The printed output provides a detailed summary containing the estimation algorithm specifications, general panel dimensions, and whether the estimated group effects are time-varying or time-invariant. It also displays the coefficients table with their respective standard errors (Large- N , Large- T sandwich in this version, bootstrap coming soon), the sum of squared residuals, log-likelihood, and total execution time.

If the optional argument `showfreq` is set to 1, the function will additionally print a table showing the number of cross-sectional units assigned to each group.

4.3 The `group_fx` function

```
group_fx(const bundle mod)
```

Return type : `series`

`mod` : the bundle containing the estimated model, returned by the `gfe_estimate` function.

This function stores the estimated group-specific effects $\hat{\alpha}_{g,i,t}$ for each cross-sectional unit over time, using the final group assignments and the estimated parameters stored in the `mod` bundle. If the model includes time-varying elements (`tvar=1`), the function reconstructs the dynamic time paths specific to each group and maps them to the corresponding units. If the model is time-invariant, it simply expands the constant group effects across the time dimension of the panel.

The output of the function is:

- A series of the same length as the dataset, representing the estimated unobserved heterogeneity (group specific effects) evaluated for each unit i at time t .

4.4 The `group_plot` function

```
group_plot(const bundle mod, string dest)
```

Return type : `none`

`mod` : the bundle containing the estimated model, returned by the `gfe_estimate` function.

`dest` : a string for the plot destination.

This function generates a graphical representation of the estimated group fixed effects. The plot layout automatically adapts to the model specifications: if the group effects are time-invariant, it produces a bar chart displaying the static coefficients; if the effects are time-varying, it plots the dynamic trajectories over time for each group.

References

- Acemoglu, D., Johnson, S., Robinson, J. A., and Yared, P. (2008). Income and democracy. *American economic review*, 98(3):808–842.
- Arellano, M. and Bond, S. (1991). Some tests of specification for panel data: Monte carlo evidence and an application to employment equations. *The review of economic studies*, 58(2):277–297.
- Bonhomme, S. and Manresa, E. (2015a). Grouped patterns of heterogeneity in panel data. *Econometrica*, 83(3):1147–1184.
- Bonhomme, S. and Manresa, E. (2015b). "supplement to 'grouped patterns of heterogeneity in panel data'". *Econometrica*, 83(3):1147–1184.

Changelog

- v0.1: initial release