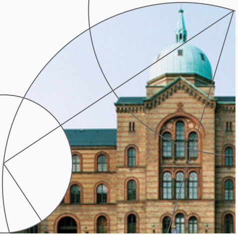
MLE with Panel Data
Simulated Maximum Likelihood

Advanced Microeconometrics

Anders Munk-Nielsen

2022



Plan for lectures: Helicopter

Part I: Linear methods. ✓

Part II: High-dimensional methods. ✓

Part III: M-estimation, theory ←

Part IV: M-estimation, examples ←

Where are we in the course?

Part	Topic	Parameterization non-linear	Estimation non-linear	Dimension $\dim(x)$	Numerical optimization	M-estimation (Part III)	Outcome (y_i)	Panel (c_i)
I	OLS	$\div$	$\div$	low	$\div$	✓	$\mathbb{R}$	✓
II	LASSO	$\div$	✓	high	✓	$\div$	$\mathbb{R}$	$\div$
IV	Probit	✓	✓	low	✓	✓	$\{0, 1\}$	$\div$
	Logit	✓	✓	low	✓	✓	$\{1, 2, ..., J\}$	$\div$
	Tobit	✓	✓	low	✓	✓	$[0; \infty)$	$\div$
	Simulated Likelihood	✓	✓	low	✓	✓	Any	✓
	Quantile Regression	$\div$	✓	(low)	✓	✓	$\mathbb{R}$	$\div$
	Non-parametric	✓	(✓)	∞	$\div$	$\div$	$\mathbb{R}$	$\div$

Goal for today

- **Panel data:** Can we say *more* when we have panel data?
- **End-goal:** Random coefficient models / simulation assisted estimation.
  - Hugely successful tool.
- **Intermediate step:** Understand random *intercept* models first.
  - Everything generalizes...
  - ... but more easily related to the well-known OLS with fixed vs. random effects.
- **Presentation:** Will show everything in a *linear model* for simplicity...
  - ... extension to probit, tobit, and logit models is “trivial”.

The short story: general case

	OLS	MLE
$c_i = 0$	Pooled OLS	Pooled ML
$c_i \perp\!\!\!\perp \mathbf{x}_i$	RE	RE
$c_i \not\perp\!\!\!\perp \mathbf{x}_i$	FE	$\div^*$

* No general fix-all, but in some cases,

- Correlated RE (Chamberlain or Mundlak),
- Dummy variables (subject to the *incidental parameters problem*),

The long story

	OLS	MLE
$c_i = 0$	Pooled OLS	Pooled ML
$c_i \perp\!\!\!\perp \mathbf{x}_i$	RE	RE, RC
$c_i \not\perp\!\!\!\perp \mathbf{x}_i$	FE	Correlated RE[1] Dummy variables[2] Transformations[3] ...

* No general fix-all, but in some cases,

1. Chamberlain or Mundlak versions,
2. Subject to the *incidental parameters problem*,
3. Highly model-dependent; examples include fixed effects logit (sufficient statistics approach),

Outline

Panel data is never a problem

- **Panel + non-linear:** Tricky to specify the full likelihood.
- **Question:** Does "pooling" work?
- **Sometimes**, the "cross-sectional" model is correctly specified
  $\Rightarrow$ then we can safely "ignore" the panel information.
  - **But:** Standard errors suffer.
- **Crucial issue:** Panel structure may break IIDness within i over t.

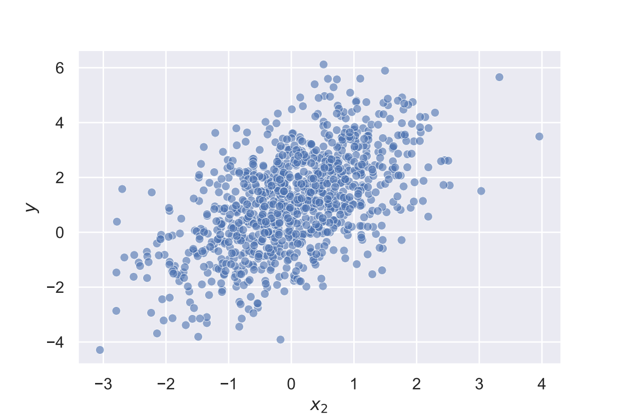
Example: OLS

Linear model

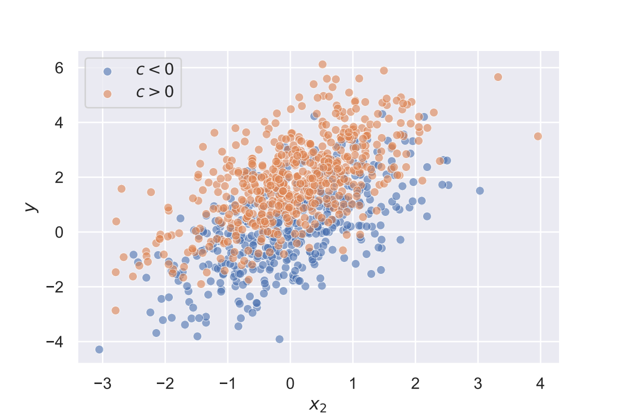
$$y_{it} = c_i + \mathbf{x}_{it}\beta + u_{it}, \quad c_i \perp\!\!\!\perp u_{it}, \mathbb{E}(u_{it}|\mathbf{x}_{it}, c_i) = 0.$$

- **Here:** Pooled OLS will be consistent.
- **But** a more *efficient* estimator is ______________.
- **Important:** The typical *standard errors* will be <u>wrong</u>!
  - Must use robust variance matrix.

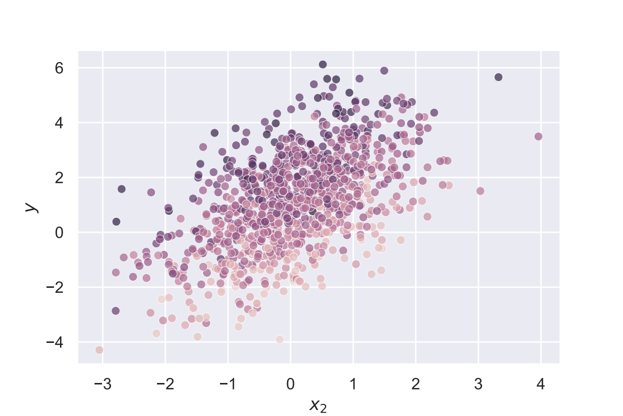
Data



Data, split on $c \gtrless 0$

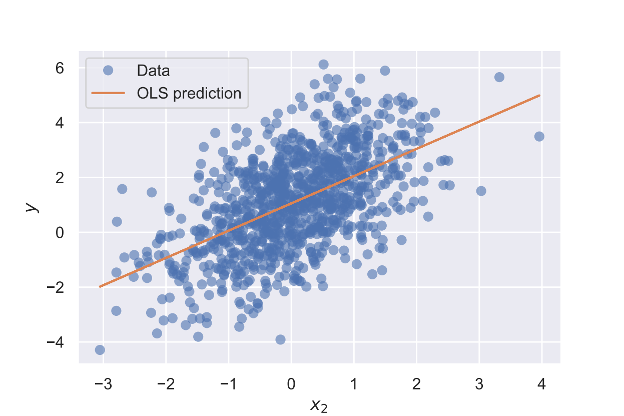


Data, colored by c

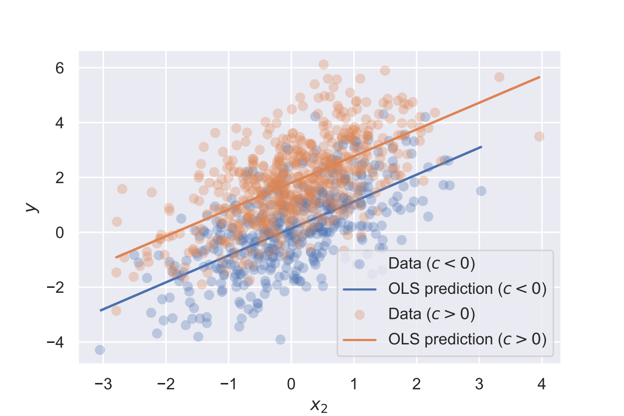


k

Pooled OLS, graphically



OLS, split by $c \gtrless 0$ (infeasible in practice)



Reminder: Regular ML

Linear model

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma_\varepsilon^2).$$

Criterion function

$$q(y_i, \mathbf{x}_i, \boldsymbol{\beta}, \sigma) = -\log\left[\frac{1}{\sigma}\phi\left(\frac{y_i - \mathbf{x}_i\boldsymbol{\beta}}{\sigma}\right)\right].$$

Pooled ML

Linear model

$$y_{it} = c_i + \mathbf{x}_{it}\beta + u_{it}, \quad c_i \sim \mathrm{IID}\mathcal{N}(0, \sigma_c^2), u_{it} \sim \mathrm{IID}\mathcal{N}(0, \sigma_u^2).$$

- **Claim:** A pooled ML estimator can give consistent estimates of β.

A different linear model

$$y_{it} = \mathbf{x}_{it}\beta + \varepsilon_{it}, \quad \varepsilon_{it} \sim \mathrm{IID}\mathcal{N}(0, \sigma_\varepsilon^2).$$

- **Criterion:**

$$f(y_{it}|\mathbf{x}_{it}; \beta, \sigma) = \frac{1}{\sigma}\phi\left(\frac{y_{it} - \mathbf{x}_{it}\beta}{\sigma}\right).$$

- **Mio. \$ question:** Why does this work?

Pooled ML continued

- **Recall:** Variance estimation with M-estimators requires IID over i!
  - otherwise, $\frac{1}{N}\hat{\mathbf{A}}^{-1}\hat{\mathbf{B}}^{-1}\hat{\mathbf{A}}^{-1}$ do not apply.
- **Solution:** Use a *robust* Sandwich variance estimator.
  - **Intuition:** Must take into account correlation within i-groups.

Pooled ML is not the full ML

Linear model

$$y_{it} = \mathbf{x}_{it}\beta + c_i + u_{it}, \quad c_i \sim \mathrm{IID}\mathcal{N}(0, \sigma_c^2), u_{it} \sim \mathrm{IID}\mathcal{N}(0, \sigma_u^2).$$

- **Full ML:** If we observed c_i (infeasible)

$$q(y_i, \mathbf{x}_i, \beta, \sigma_u; c_i) = \prod_{t=1}^{T} \frac{1}{\sigma_u}\phi\left(\frac{y_{it} - \mathbf{x}_{it}\beta - c_i}{\sigma_u}\right).$$

- **Pooled ML:**

$$q(y_i, \mathbf{x}_i, \beta, \sigma_\varepsilon) = \prod_{t=1}^{T} \frac{1}{\sigma_\varepsilon}\phi\left(\frac{y_{it} - \mathbf{x}_{it}\beta}{\sigma_\varepsilon}\right).$$

What works and what does not

True model

$$y_{it} = \mathbf{x}_{it}\beta + c_i + u_{it}, \quad c_i \sim \mathrm{IID}\mathcal{N}(0, \sigma_c^2), u_{it} \sim \mathrm{IID}\mathcal{N}(0, \sigma_u^2).$$

Criterion function

$$q(y_i, \mathbf{x}_i, \beta, \sigma) = -\log\left[\frac{1}{\sigma}\phi\left(\frac{y_i - \mathbf{x}_i\beta}{\sigma}\right)\right].$$

- **What works:** Consistency
  - since $u_{it} + c_i \sim \mathcal{N}(0, \sigma_u^2 + \sigma_c^2)$
- **What does not:** Standard errors
  - since data rows are no longer IID,
  - dependence over t for each individual i caused by c_i.

Outline

Agenda

1. The model: how to simulate data
2. "Integrating" out c_i: $f(y|\mathbf{x}) = \mathbb{E}_c[f(y|\mathbf{x}, c)]$
3. Adding panel data: $f(\mathbf{y}_i|\mathbf{x}_i, c_i) = \prod_t f(y_{it}|\mathbf{x}_{it}, c_i)$
4. Concrete models: Linear and Probit
5. Integration: how to compute "$\mathbb{E}_c(\cdot)$"

Random Effects Model

True model

$$y_{it} = c_i + \mathbf{x}_{it}\beta + u_{it}, \quad c_i \sim \mathrm{IID}\mathcal{N}(0, \sigma_c^2), u_{it} \sim \mathrm{IID}\mathcal{N}(0, \sigma_u^2).$$

Simulating data

```
def sim_data(theta, N, T):
    # unpack params
    beta = theta[:-2]
    sigma_u = theta[-2]
    sigma_c = theta[-1]
    NT = int(N*T) # no. rows in y and x

    # sim x
    oo = np.ones((NT,1))
    xx = np.random.normal(size=(NT,K-1))
    x = np.hstack([oo,xx])

    # draw unobserved terms
    c = sigma_c * np.random.normal(size=N)
    u = sigma_u * np.random.normal(size=NT)

    c_NT = np.kron(c, np.ones((T,))) # repeat c-terms T times
    y = x@beta + c_NT + u
    return y,x
```

Agenda

1. ~~The model: how to simulate data~~
2. "Integrating" out c_i: $f(y|\mathbf{x}) = \mathbb{E}_c[f(y|\mathbf{x}, c)]$
3. Adding panel data: $f(\mathbf{y}_i|\mathbf{x}_i, c_i) = \prod_t f(y_{it}|\mathbf{x}_{it}, c_i)$
4. Concrete models: Linear and Probit
5. Integration: how to compute "$\mathbb{E}_c(\cdot)$"

Fundamental Rule

Proposition

We must prove and understand that

$$f(y|\mathbf{x}) = \mathbb{E}_c\left[f(y|\mathbf{x}, c)\right].$$

Dice example

A dice is rolled, the outcome is A. The dice is either crooked ($A = 6$ with certainty) or fair with prob. $\frac{1}{2}$, but you don't know which. Then

$$\Pr(A = 6) = \frac{1}{2}\Pr(A = 6|c_i = \text{crooked dice}) + \frac{1}{2}\Pr(A = 6|c_i = \text{fair dice}).$$

$$= \frac{1}{2}1 + \frac{1}{2}\frac{1}{6} = \frac{7}{12}.$$

- We know $\Pr(A = a|c_i = c)$,
- We want to know $\Pr(A = a)$.
- The expectation, "$\mathbb{E}_c(\cdot)$" is simple, because c_i only takes two values.

Fundamental Rule

Proposition

We must prove and understand that

$$f(y|\mathbf{x}) = \mathbb{E}_c\left[f(y|\mathbf{x}, c)\right].$$

Probit with binary types

$$y_{it}^* = \mathbf{x}_{it}\beta + c_i + u_{it}, \quad u_{it} \sim \mathcal{N}(0,1),$$

$$c_i \in \{c^L, c^H\}, \quad \Pr(c_i = c^L) = \frac{1}{2}$$

$$y_{it} = \mathbf{1}\{y_{it}^* > 0\}.$$

Then

$$\begin{aligned}
\Pr(y_{it} = 1|\mathbf{x}_{it}) &= \frac{1}{2}\Pr(y_{it} = 1|\mathbf{x}_{it}, c_i = c^L) + \frac{1}{2}\Pr(y_{it} = 1|\mathbf{x}_{it}, c_i = c^H) \\
&= \frac{1}{2}\Phi(\mathbf{x}_{it}\beta + c^L) + \frac{1}{2}\left[1 - \Phi(\mathbf{x}_{it}\beta + c^H)\right].
\end{aligned}$$

Reminder: Definitions

- **Now:** Roll two dice, A, B
- **Marginal density:**

$$\Pr(A = 6) = \sum_{b=1}^{6} \Pr(A = 6, b).$$

  - **With continuous** outcomes for b, the sum becomes an integral.

- **Conditional density:** Think card draws,

$$\Pr(\heartsuit|\text{red}) = \frac{\Pr(\heartsuit \text{ and } \text{red})}{\Pr(\text{red})} = \frac{\frac{1}{4}}{\frac{1}{2}} = \frac{1}{2}.$$

  - with variables,

$$\Pr(A = a|B = b) = \frac{\Pr(A = a, B = b)}{\Pr(B = b)}.$$

Deriving the Rule

- **Definition:** [marginal density]

$$f(w) = \int_{-\infty}^{\infty} f(w, c)\,\mathrm{d}c.$$

- **Definition:** [conditional density]

$$f(w|c) = \frac{f(w, c)}{f(c)}.$$

- **Proposition:**

$$f(w) = \mathbb{E}_c[f(w|c)].$$

- **Proof:**

$$\begin{aligned} f(w) &= \int_{-\infty}^{\infty} f(w|c)f(c)\,\mathrm{d}c \\ &= \mathbb{E}_c[f(w|c)]. \end{aligned}$$

Inserting $w := y|\mathbf{x}$

- **Definition:** [marginal cond. density]

$$f(y|\mathbf{x}) = \int_{-\infty}^{\infty} f(y, c|\mathbf{x})\,\mathrm{d}c.$$

- **Definition:** [conditional density]

$$f(y|\mathbf{x}, c) = \frac{f(y, c|\mathbf{x})}{f(c|\mathbf{x})}.$$

- **Together:**

$$f(y|\mathbf{x}) =$$

- **Definition:** [expectation]

$$f(y|\mathbf{x}) =$$

Agenda

1. ~~The model: how to simulate data~~
2. ~~"Integrating" out c_i: $f(y|\mathbf{x}) = \mathbb{E}_c[f(y|\mathbf{x}, c)]$~~
3. Adding panel data: $f(\mathbf{y}_i|\mathbf{x}_i, c_i) = \prod_t f(y_{it}|\mathbf{x}_{it}, c_i)$
4. Integration: how to compute "$\mathbb{E}_c(\cdot)$"
5. Concrete models: Linear and Probit

- **Now:** Consider the joint distribution of $\mathbf{y}_i|\mathbf{x}_i$ ($\mathbf{y}_i \equiv (y_{i1}, ..., y_{iT})$, $\mathbf{x}_i \equiv (\mathbf{x}_{i1}, ..., \mathbf{x}_{iT})$)
- **Writing** our rule

$$f(\mathbf{y}_i|\mathbf{x}_i) = \mathbb{E}_c[f(\mathbf{y}_i|\mathbf{x}_i, c_i)|\mathbf{x}_i].$$

Assumption (Dynamic Completeness)

$$f(\mathbf{y}_i|\mathbf{x}_i, c_i) = \prod_{t=1}^{T} f(y_{it}|\mathbf{x}_{it}, c_i).$$

- **Intuition:** After conditioning on $\mathbf{x}_{it}$ and c_i, y_{it} is serially independent.
  - Rules out: e.g. lagged outcome as a regressor.

Expected Likelihood Under Dynamic Completeness

$$f(\mathbf{y}_i|\mathbf{x}_i) = \mathbb{E}_c\left[\prod_{t=1}^{T} f(y_{it}|\mathbf{x}_{it}, c_i)\middle|\mathbf{x}_i\right].$$

Implication

- **We have:**

$$
\begin{aligned}
f(\mathbf{y}_i|\mathbf{x}_i) &= \mathbb{E}_c\left[\prod_{t=1}^{T} f(y_{it}|\mathbf{x}_{it}, c_i)\middle|\mathbf{x}_i\right] \\
&\equiv \int_{-\infty}^{\infty}\left[\prod_{t=1}^{T} f(y_{it}|\mathbf{x}_{it}, c)\right] f(c|\mathbf{x}_i)\,\mathrm{d}c
\end{aligned}
$$

- **2 versions** for $f(c|\mathbf{x}_i)$
  1. **Random effects (RE):** $f(c|\mathbf{x}_i) = f(c)$.
  2. **Correlated RE:** E.g. $f(c|\mathbf{x}_i) = f(c - \bar{\mathbf{x}}_i\psi)$, where $\bar{\mathbf{x}}_i = T^{-1}\sum_{t=1}^{T}\mathbf{x}_{it}$ (Mundlak).
     ψ is then an additional vector to be estimated, $\theta = (\beta, \sigma_c, \psi)$.

- **Criterion function:**

$$
\ell_i(\theta) = \log \int_{-\infty}^{\infty}\left[\prod_{t=1}^{T} f(y_{it}|\mathbf{x}_{it}, c)\right] f(c|\mathbf{x}_i)\,\mathrm{d}c
$$

Discuss

In your own words, what is the relationship between $f(y|\mathbf{x})$ and $f(y|\mathbf{x}, c)$?

Criterion function (linear model)

Linear RE model

$$y_{it} = c_i + \mathbf{x}_{it}\beta + u_{it}, \quad c_i \sim \mathrm{IID}\mathcal{N}(0, \sigma_c^2), u_{it} \sim \mathrm{IID}\mathcal{N}(0, \sigma_u^2).$$

Criterion function

$$\ell_i(\beta, \sigma_u, \sigma_c) = \log \int_{-\infty}^{\infty} \left[\prod_{t=1}^{T} \underbrace{\frac{1}{\sigma_u} \phi\left(\frac{y_{it} - \mathbf{x}_{it}\beta - \sigma_c c}{\sigma_u} \right)}_{= f(y_{it} | \mathbf{x}_{it}, c_i = \sigma_c c)} \right] \phi(c) \, \mathrm{d}c$$

Criterion function (probit model)

Probit RE model (ch. 15.8.2)

$$
\begin{aligned}
y_{it}^* &= c_i + \mathbf{x}_{it}\beta + u_{it}, \quad c_i \sim \mathrm{IID}\mathcal{N}(0, \sigma_c^2), u_{it} \sim \mathrm{IID}\mathcal{N}(0, 1), \\
y_{it} &= \mathbf{1}\{y_{it}^* > 0\}.
\end{aligned}
$$

Criterion function

$$
\ell_i(\beta, \sigma_c) =
$$

$$
\log \int_{-\infty}^{\infty} \left\{ \prod_{t=1}^{T} \underbrace{[G(\mathbf{x}_{it}\beta + \sigma_c c)]^{y_{it}} [1 - G(\mathbf{x}_{it}\beta + \sigma_c c)]^{1-y_{it}}}_{f(y_{it}|\mathbf{x}_{it}, c_i = \sigma_c c)} \right\} \phi(c)\, \mathrm{d}c
$$

Discussion

Fact: We cannot identify $\sigma_u^2 \equiv \text{Var}(u_{it})$ and σ_c^2 *simultaneously.*

- **Normalization:** We normalize $\sigma_u := 1$ and estimate σ_c.
- **(Alternatively,** we could normalize $\sigma_c := 1$ and estimate σ_u)
- Intuitively, you could say that only the *relative* dispersion is identified.

Discussion

Why?

Agenda

1. ~~The model: how to simulate data~~
2. ~~"Integrating" out c_i: $f(y|\mathbf{x}) = \mathbb{E}_c[f(y|\mathbf{x}, c)]$~~
3. ~~Adding panel data: $f(\mathbf{y}_i|\mathbf{x}_i, c_i) = \prod_t f(y_{it}|\mathbf{x}_{it}, c_i)$~~
4. ~~Concrete models: Linear and Probit~~
5. Integration: how to compute "$\mathbb{E}_c(\cdot)$"
   Spoiler alert: We will replace it with an average!!!

- **Criterion function:**

$$q(\mathbf{y}_i, \mathbf{x}_i, \boldsymbol{\theta}) = -\log \mathbb{E}[f(\mathbf{y}_i|\mathbf{x}_i; \boldsymbol{\theta})].$$

- **Problem:** We have a model for $f(y_{it}|\mathbf{x}_{it}, c; \boldsymbol{\theta})$.
    - This is what we can *compute*.
    - In the dice example, this is $\Pr(x = 6|\text{crooked dice})$.
- **Shown:** We can write

$$f(\mathbf{y}_i|\mathbf{x}_i; \boldsymbol{\theta}) = \mathbb{E}_c \left[\prod_{t=1}^{T} f(y_{it}|\mathbf{x}_{it}, c; \boldsymbol{\theta}) \right].$$

- **Idea:** Replace $\mathbb{E}_c[\cdot]$ with an average.

Integration by Simulation

- **Recall:**

$$f(\mathbf{y}_i|\mathbf{x}_i;\boldsymbol{\theta}) = \mathbb{E}_c\left[\prod_{t=1}^{T} f(y_{it}|\mathbf{x}_{it}, c; \boldsymbol{\theta})\right].$$

  and suppose

$$c_i \sim \mathcal{N}(0, \sigma_c^2).$$

Integration by Simulation

For trial values, $(\beta, \sigma_u, \sigma_c)$,

1. Draw R values $\eta_{ir} \sim \mathcal{N}(0, 1)$,
2. Calculate $\varphi_{itr} := f(y_{it}|x_{it}, \sigma_c\eta_{ir}; \beta, \sigma_u)$, [i.e. setting $c_i := \sigma_c\eta_{ir}$]
3. Return

$$\ell_i(\theta) \cong R^{-1}\sum_{r=1}^{R}\prod_{t=1}^{T}\varphi_{itr}.$$

- **Note:** Make sure that η_{ir} does not change each time the function is evaluated.
- **Note:** R must rise faster than $\sqrt{N}$ for equivalence with ML

Criterion function

$$\mathcal{L}^{\mathrm{SML}}(\theta) = N^{-1} \sum_{i=1}^{N} \log \left[\frac{1}{R} \sum_{r=1}^{R} \prod_{t=1}^{T} f\left(y_{it} | \mathbf{x}_{it}, \sigma_c \eta_r; \theta\right) \right].$$

- **Note:** For each draw of η_r,
  - we simulate the full T path.
- **Intuition:** Suppose individual i is a high type.
  - Then a path with low η_r will *consistently* look unlikely.
    - Hence the time product will be tiny.
  - But a path with high η_r will *consistently* look reasonable.
    - Hence the time product will be much larger.
  - Hence, the higher paths will come out.
- **Type density:** Can compare the R time-paths; the largest is the best guess of what c-type individual i is.

```python
def loglikelihood(theta, y, x):
    N,T = y.shape
    N,T,K = x.shape

    beta = theta[:-2] # coefficients on x
    sigma_u = np.abs(theta[-2]) # idiosyncratic error
    sigma_c = np.abs(theta[-1]) # random effect dispersion

    c_draws = sigma_c * np.random.normal(size=(N,R)) # pre-draw c
    res_common = y - x@beta # (N,T): common over simulations
    f_itr = np.empty((N, T, R)) # preallocate
    for r in range(R):
        c_r = c_draws[:,r].reshape(N,1)
        res = res_common - c_r
        f_itr[:, :, r] = 1./sigma_u * norm.pdf(res/sigma_u)
    f_ir = np.product(f_itr, axis=1) # product over time
    f_i = np.mean(f_ir, axis=1) # # mean over simulations
    ll_i = np.log(f_i)
    return ll_i
```

Other ways of integrating

- **Binary support:** If $c_i \in \{c^L, c^H\}$ with $\Pr(c_i = c^L) \equiv \pi^L$, then

$$f(y_{it}|\mathbf{x}_{it}) = \pi^L f(y_{it}|\mathbf{x}_{it}, c_i = c^L) + (1 - \pi^L) f(y_{it}|\mathbf{x}_{it}, c_i = c^H).$$

  - Example: Only two types of people; a high and a low type.
  - Estimate π^L, c^L (c^L, c^H are often not identified jointly).
- **Discrete support:** If $c_i \in \{c^1, ..., c^K\}$, with $\Pr(c_i = c^k) \equiv \pi^k$

$$f(y_{it}|\mathbf{x}_{it}) = \sum_{k=1}^{K} \pi^k f(y_{it}|\mathbf{x}_{it}, c_i = c^k).$$

- **Normally distributed:** Quadrature can be used [next slide]

$$\int_{-\infty}^{\infty} g(c)\,\phi(c)\,\mathrm{d}c \cong \sum_{q=1}^{Q} w_q g(x_q).$$

- **General distribution:** Integration by simulation,

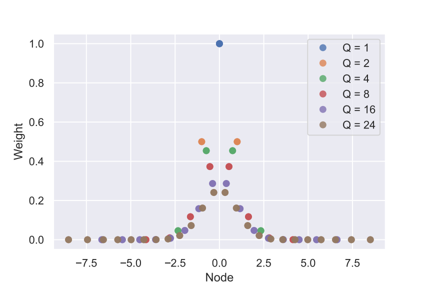
$$\int_{-\infty}^{\infty} g(c) f_c(c)\,\mathrm{d}c \cong \frac{1}{R} \sum_{r=1}^{R} g(c_r), \quad c_r \sim \mathrm{IID} F_c.$$

Quadrature

- **Approximate**

$$\int_{-\infty}^{\infty} g(c)\,\phi(c)\,\mathrm{d}c \cong \sum_{q=1}^{Q} w_q g(x_q).$$

- **Implementation:** `GaussHermite.m`.
  - **Nodes:** $\{x_q\}_{q=1}^{Q}$ are the nodes where we evaluate g,
  - **Weights:** $\{w_q\}_{q=1}^{Q}$ are the corresponding weights.



Quadrature

Integration by Quadrature

For trial values, (β, σ_c),

1. Read in $\{x_q, w_q\}_{q=1}^{Q}$ (e.g. from a table),
2. Calculate $\varphi_{itq} := f(y_{it}|x_{it}, \sigma_c x_q; \beta)$,
3. Return

$$\ell_i(\theta) \cong \sum_{q=1}^{Q} w_q \prod_{t=1}^{T} \varphi_{itq}.$$

Simulation or Quadrature?

- **Gauss-Hermite quadrature:**
  - Only works for Gaussian integrals, i.e. on the form $\int_{-\infty}^{\infty} g(c)\,\phi(c)\,\mathrm{d}c$.
  - Superior precision per evaluation.
- **Simulation:**
  - Intuitive.
  - Generalizes to non-Gaussian integrals, $\int_{-\infty}^{\infty} g(c)\,f(c)\,\mathrm{d}c$.
    [if we can draw from f]
  - Dimensionality advantage: Sometimes superior if multivariate f is "dense".
    - We avoid evaluating f many times in regions with a low density.

Agenda

1. ~~The model: how to simulate data~~
2. ~~“Integrating” out c_i: $f(y|\mathbf{x}) = \mathbb{E}_c[f(y|\mathbf{x}, c)]$~~
3. ~~Adding panel data: $f(\mathbf{y}_i|\mathbf{x}_i, c_i) = \prod_t f(y_{it}|\mathbf{x}_{it}, c_i)$~~
4. ~~Integration: how to compute “$\mathbb{E}_c(\cdot)$”~~
5. ~~Concrete models: Linear and Probit~~

Next up: A peak into a larger world.

Outline

Intercept vs. other coefficients

- **So far**, we have thought of

$$y_{it} = c_i + \beta_0 + \beta_1 x_{1it} + ... + \beta_K x_{Kit} + u_{it}.$$

  - **Intuition:** The intercept is really $\beta_{0i} \equiv \beta_0 + c_i$.
  - Alternatively, $\beta_{0i} \sim \mathcal{N}(\beta_0, \sigma_c^2)$.

- **Alternative**, perhaps more useful model,

$$y_{it} = \beta_0 + (\beta_1 + c_i) x_{1it} + ... + \beta_K x_{Kit} + u_{it}$$

  - **Intuition:** Now, $\beta_{1i} \sim \mathcal{N}(\beta_1, \sigma_c^2)$.

- **In general:** Random coefficients models let

$$y_{it} = \mathbf{x}_{it} \boldsymbol{\beta}_i + u_{it}, \quad \boldsymbol{\beta}_i \sim \mathcal{N}(\boldsymbol{\beta}, \boldsymbol{\Sigma}).$$

  - $\boldsymbol{\Sigma}$ $(K \times K)$ describes covariances between individual parameters.

Interactions

- **Note:** Above, β_i are *idiosyncratic* deviations from the common mean parameters.
- we can of course allow $y_{it} = \beta_0 + \mathbf{x}_{it}(\boldsymbol{\beta} + \boldsymbol{\gamma}\mathbf{z}_{it}) + u_{it}$.
  - Here, βs are also heterogenous,
  - but they vary systematically with some observables $\mathbf{z}_{it}$.
- **Example:** $\text{wage} = \beta_0 + (\beta_1 + \gamma \text{IQ})\text{programme} + u_{it}$.
  - Here, programme wage-payoff depends on participant IQ.

Random Coefficients Logit

Model

$$u_{ijt} = \mathbf{x}_j \boldsymbol{\beta}_i + \varepsilon_{ijt}, \quad \boldsymbol{\beta}_i \sim \mathrm{IID} F_\beta(\cdot), \varepsilon_{ijt} \sim \text{IID Extr.Val.1},$$
$$y_{it} = \mathrm{argmax}_j u_{ijt}.$$

- **Example:** Suppose j denotes cars and x_j includes
  - [horsepower, car price].

Discuss

- What are the interpretations of β_{i1} and β_{i2}?
- What would you expect about $\mathrm{Correlation}(\beta_{i2}, \mathrm{income}_i)$?
- Why might one expect $\mathrm{Corrrelation}(\beta_{i1}, \beta_{i2}) < 0$?

Outline

Dummy variables

- **Recall:** The *Least squares dummy variables* estimator (LSDV),

$$\text{regress } y_{it} \text{ on } x_{it}, \mathbf{1}\{i = 1\}, \mathbf{1}\{i = 2\}, ..., \mathbf{1}\{i = N\}.$$

- **Turns out:** Numerically identical estimates of coefficients on the x_{it}s.
- **Generally:** Estimating so many coefficients suffers from the *incidental parameters problem;*

Incidental parameters problem

Estimating N dummies results in a bias in the estimates of the K β-parameters. The bias goes away as $T \to \infty$.

- **Cool new stuff:** Estimate on subsets of T; explore how $\hat{\beta}$ changes; infer the magnitude of the bias.

Transformations

- **Recall:** When c_i is present,
- **OLS:** Transformations $\Rightarrow$ get rid of c_i.
  - FE and FD.
  - *Note:* only works when the *intercept* is random...
- **Nonlinear models:** $\Delta G(\mathbf{x}_{it}\beta + c_i)$ will not work.
- **Idea:** We want to write $G[\Delta(\mathbf{x}_{it}\beta + c_i)] = G[(\mathbf{x}_{it} - \mathbf{x}_{it-1})\beta]$.

Fixed Effects

- **Allowing** $c_i \not\perp\!\!\!\perp \mathbf{x}_{it}$ is hard.
- **Linear model:** we can just transform the outcome.
  - First-differences,
  - Fixed effects (within demeaning).
- **Non-linear model:** If

$$y_{it} = g(c_i + \mathbf{x}_{it}\beta) + u_{it},$$

  then differencing on both sides doesn't eliminate c_i (cause $\Delta g(\cdot) \neq g(\Delta \cdot)$)
- **Example:** Fixed effects binary logit (clever transformation): C&T ch. 23.4.3.
  - Only works for individuals with both $y_{it} = 0$ and $y_{is} = 1$ for some t, s (both outcomes occurring)
- **Example:** Quantile "plug-in" fixed effects (Canay, 2011)
  - Estimates linear FE and plugs into a 2nd stage quantile regression.