Filtering procedures for dynamic multinomial probit models

Augusto Fasano

University of Torino and Collegio Carlo Alberto

SISBAYES Workshop, 4 September 2025.

Joint work with:

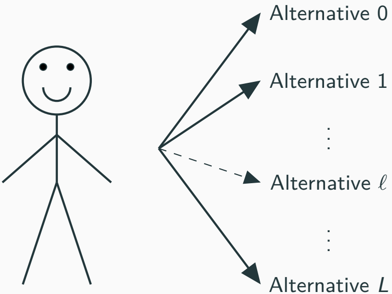
Lorenzo Rimella

Giovanni Rebaudo

Introduction

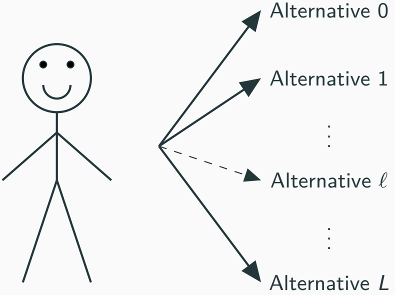
A brief overview on discrete-choice models

Discrete-choice models: A discrete choice model specifies the probability that a person chooses a particular alternative.



A brief overview on discrete-choice models

Discrete-choice models: A discrete choice model specifies the probability that a person chooses a particular alternative.

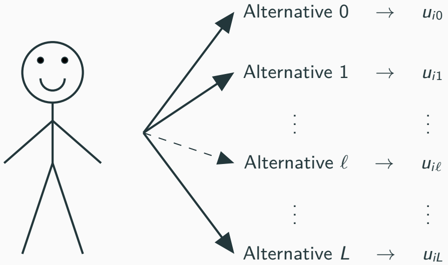


The **probability of choosing a specific alternative** is expressed as a **function of observed variables** that relate to:

- the person
- the alternatives

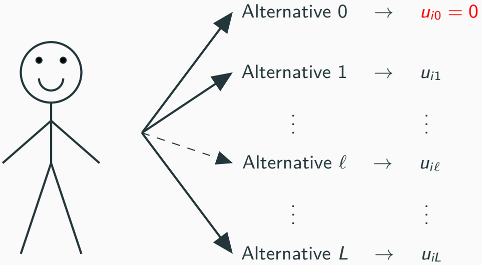
How is the choice made?

- Individual i obtains a utility $u_{i\ell}$ by choosing alternative ℓ, for $\ell = 0, 1, \dots, L$.
- **The behavior of the person is utility-maximizing**: individual i chooses the alternative that provides the highest utility.



How is the choice made?

- Individual i obtains a utility $u_{i\ell}$ by choosing alternative ℓ, for $\ell = 0, 1, \ldots, L$.
- **The behavior of the person is utility-maximizing**: individual i chooses the alternative that provides the highest utility.
- Only differences in utilities count. We set $u_{i0} = 0$ for identifiability.



How are utilities defined?

Calling $\mathbf{u}_i = (u_{i1}, \ldots, u_{iL})^\intercal$, in the multinomial probit model (MNP) we have

$$\mathbf{u}_i = \mathbf{X}_i \beta + \varepsilon_i,$$

where

- $\mathbf{X}_i$: $L \times p$ matrix which can contain
  - intercept terms,
  - covariates that vary across agents (e.g., age)
  - and/or covariates that vary across alternatives (e.g., prices)
- β: p-dimensional vector of parameters to be estimated.
- $\varepsilon_i \sim N_L(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$ is a covariance matrix satisfying appropriate constraints for identifiability (e.g., $\sigma_{11} = 1$ or $\mathrm{tr}(\mathbf{\Sigma}) = L$).
- **Errors ε_{ij}, ε_{ik}, $j \neq k$, are not independent**.

Calling $\mathbf{x}_{i\ell}^\intercal$ the ℓ-th row of $\mathbf{X}_i$, we thus have $u_{i\ell} = \mathbf{x}_{i\ell}^\intercal \beta + \varepsilon_{i\ell}$.

The likelihood of the MNP

Let us consider $\boldsymbol{\Sigma}$ fixed (for the moment) and call $\mathbf{x}_{i\ell}^{\mathsf{T}}$ the ℓ-th row of $\mathbf{X}_i$.

The likelihood of observation y_i is

$$\begin{aligned}
\Pr[y_i = \ell \mid \boldsymbol{\beta}] &= \Pr[u_{i\ell} > u_{ik} \ \forall k \neq \ell \mid \boldsymbol{\beta}] \\
&= \Pr[\cap_{k\neq\ell}\{\mathbf{x}_{i\ell}^{\mathsf{T}}\boldsymbol{\beta} + \varepsilon_{i\ell} > \mathbf{x}_{ik}^{\mathsf{T}}\boldsymbol{\beta} + \varepsilon_{ik}\} \mid \boldsymbol{\beta}] \\
&= \Pr[\cap_{k\neq\ell}\{\varepsilon_{ik} - \varepsilon_{i\ell} < (\mathbf{x}_{i\ell}^{\mathsf{T}} - \mathbf{x}_{ik})^{\mathsf{T}}\boldsymbol{\beta}\} \mid \boldsymbol{\beta}] \\
&= \Pr[\mathbf{V}_{[-\ell]}\boldsymbol{\varepsilon}_i < \mathbf{X}_{[i,-\ell]}\boldsymbol{\beta} \mid \boldsymbol{\beta}],
\end{aligned}$$

for appropriate matrices $\mathbf{V}_{[-\ell]}$ and $\mathbf{X}_{[i,-\ell]}$, with $\varepsilon_{i0} = 0$ and $\mathbf{x}_{i0} = \mathbf{0}_p^{\mathsf{T}}$.

The likelihood of the MNP

Let us consider $\mathbf{\Sigma}$ fixed (for the moment) and call $\mathbf{x}_{i\ell}^\intercal$ the ℓ-th row of $\mathbf{X}_i$.

The likelihood of observation y_i is

$$\begin{aligned}
\Pr[y_i = \ell \mid \beta] &= \Pr[u_{i\ell} > u_{ik} \ \forall k \neq \ell \mid \beta] \\
&= \Pr[\cap_{k\neq\ell}\{\mathbf{x}_{i\ell}^\intercal\beta + \varepsilon_{i\ell} > \mathbf{x}_{ik}^\intercal\beta + \varepsilon_{ik}\} \mid \beta] \\
&= \Pr[\cap_{k\neq\ell}\{\varepsilon_{ik} - \varepsilon_{i\ell} < (\mathbf{x}_{i\ell}^\intercal - \mathbf{x}_{ik})^\intercal\beta\} \mid \beta] \\
&= \Pr[\mathbf{V}_{[-\ell]}\boldsymbol{\varepsilon}_i < \mathbf{X}_{[i,-\ell]}\beta \mid \beta],
\end{aligned}$$

for appropriate matrices $\mathbf{V}_{[-\ell]}$ and $\mathbf{X}_{[i,-\ell]}$, with $\varepsilon_{i0} = 0$ and $\mathbf{x}_{i0} = \mathbf{0}_p^\intercal$.

MNP likelihood of a single observation

Thus, the likelihood of the i-th observation in the MNP model is

$$\Pr[y_i = \ell \mid \beta] = \Phi_L(\mathbf{X}_{[i,-\ell]}\beta; \mathbf{S}_\ell), \qquad \mathbf{S}_\ell = \mathbf{V}_{[-\ell]}\mathbf{\Sigma}\mathbf{V}_{[-\ell]}^\intercal,$$

where $\Phi_L(\mathbf{x}; \mathbf{\Sigma})$ denotes the cdf of a $N_L(\mathbf{0}, \mathbf{\Sigma})$ random v., evaluated at $\mathbf{x}$.

(Bayesian) Inference in the multinomial probit models

The likelihood of the MNP

MNP likelihood of a sample

The likelihood of a sample $\mathbf{y}_{1:n} = (y_1, \ldots, y_n)$ is thus given by

$$p(\mathbf{y}_{1:n} \mid \beta) = \prod_{i=1}^{n} \Phi_L(\mathbf{X}_{[i,-y_i]}\beta; \mathbf{S}_{y_i}) = \Phi_{nL}(\mathbf{X}\beta; \mathbf{\Lambda}),$$

with

$$\mathbf{X} = [\mathbf{X}_{[1,-y_1]}^\intercal, \ldots, \mathbf{X}_{[n,-y_n]}^\intercal]^\intercal, \qquad \mathbf{\Lambda} = \text{block-diag}(\mathbf{S}_{y_1}, \ldots, \mathbf{S}_{y_n})$$

How can we make inference on β?

The likelihood of the MNP

MNP likelihood of a sample

The likelihood of a sample $\mathbf{y}_{1:n} = (y_1, \dots, y_n)$ is thus given by

$$p(\mathbf{y}_{1:n} \mid \beta) = \prod_{i=1}^{n} \Phi_L(\mathbf{X}_{[i,-y_i]}\beta; \mathbf{S}_{y_i}) = \Phi_{nL}(\mathbf{X}\beta; \mathbf{\Lambda}),$$

with

$$\mathbf{X} = [\mathbf{X}_{[1,-y_1]}^\intercal, \dots, \mathbf{X}_{[n,-y_n]}^\intercal]^\intercal, \qquad \mathbf{\Lambda} = \text{block-diag}(\mathbf{S}_{y_1}, \dots, \mathbf{S}_{y_n})$$

How can we make inference on β?

We focus on the Bayesian framework, with multivariate Gaussian prior

$$\beta \sim \mathcal{N}_p(\boldsymbol{\mu}, \mathbf{\Omega}) \qquad (\text{usually } \boldsymbol{\mu} = \mathbf{0}).$$

Thus

$$p(\beta \mid \mathbf{y}_{1:n}) \propto d\mathcal{N}_p(\beta; \boldsymbol{\mu}, \mathbf{\Omega}) \cdot \Phi_{nL}(\mathbf{X}\beta; \mathbf{\Lambda}).$$

The likelihood of the MNP

MNP likelihood of a sample

The likelihood of a sample $\mathbf{y}_{1:n} = (y_1, \dots, y_n)$ is thus given by

$$p(\mathbf{y}_{1:n} \mid \beta) = \prod_{i=1}^{n} \Phi_L(\mathbf{X}_{[i,-y_i]}\beta; \mathbf{S}_{y_i}) = \Phi_{nL}(\mathbf{X}\beta; \mathbf{\Lambda}),$$

with

$$\mathbf{X} = [\mathbf{X}_{[1,-y_1]}^\intercal, \dots, \mathbf{X}_{[n,-y_n]}^\intercal]^\intercal, \qquad \mathbf{\Lambda} = \text{block-diag}(\mathbf{S}_{y_1}, \dots, \mathbf{S}_{y_n})$$

How can we make inference on β?

We focus on the Bayesian framework, with multivariate Gaussian prior

$$\beta \sim \mathcal{N}_p(\boldsymbol{\mu}, \mathbf{\Omega}) \qquad (\text{usually } \boldsymbol{\mu} = \mathbf{0}).$$

Thus

$$p(\beta \mid \mathbf{y}_{1:n}) \propto d\mathcal{N}_p(\beta; \boldsymbol{\mu}, \mathbf{\Omega}) \cdot \Phi_{nL}(\mathbf{X}\beta; \mathbf{\Lambda}).$$

Is this the kernel of some known family of distributions?

The unified skew-normal (SUN) distribution: a short recap

Short recap: a $\mathrm{SUN}_{p,m}\left(\boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\Delta}, \gamma, \boldsymbol{\Gamma}\right)$ distributed random variable has density

$$p(\beta) = d\mathcal{N}_p(\beta; \boldsymbol{\mu}, \boldsymbol{\Omega}) \frac{\Phi_m\left(\gamma + \boldsymbol{\Delta}^\intercal \bar{\boldsymbol{\Omega}}^{-1}\omega(\beta - \xi); \boldsymbol{\Gamma} - \boldsymbol{\Delta}^\intercal \bar{\boldsymbol{\Omega}}^{-1}\boldsymbol{\Delta}\right)}{\Phi_m(\gamma; \boldsymbol{\Gamma})},$$

where $\bar{\boldsymbol{\Omega}}$ and ω correlation and scale matrices associated to $\boldsymbol{\Omega} = \omega\bar{\boldsymbol{\Omega}}\omega$;
Moments may be quite involved, but we can develop an **i.i.d. sampler**.

Important fact: SUN additive representation

If $\beta \sim \mathrm{SUN}_{p,m}$, then it can be characterized probabilistically as a linear combination of:

- a p-variate Gaussian term;
- an m-variate truncated Gaussian component.

The unified skew-normal (SUN) distribution: a short recap

Short recap: a $\mathrm{SUN}_{p,m}(\boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\Delta}, \gamma, \boldsymbol{\Gamma})$ distributed random variable has density

$$p(\beta) = d\mathcal{N}_p(\beta; \boldsymbol{\mu}, \boldsymbol{\Omega}) \frac{\Phi_m\left(\gamma + \boldsymbol{\Delta}^\intercal \bar{\boldsymbol{\Omega}}^{-1} \omega(\beta - \xi); \boldsymbol{\Gamma} - \boldsymbol{\Delta}^\intercal \bar{\boldsymbol{\Omega}}^{-1} \boldsymbol{\Delta}\right)}{\Phi_m(\gamma; \boldsymbol{\Gamma})},$$

where $\bar{\boldsymbol{\Omega}}$ and ω correlation and scale matrices associated to $\boldsymbol{\Omega} = \omega \bar{\boldsymbol{\Omega}} \omega$;
Moments may be quite involved, but we can develop an **i.i.d. sampler**.

Important fact: SUN additive representation

If $\beta \sim \mathrm{SUN}_{p,m}$, then it can be characterized probabilistically as a linear combination of:

- a p-variate Gaussian term;
- an m-variate truncated Gaussian component.

In the MNP, we have $p(\beta \mid \mathbf{y}_{1:n}) \propto d\mathcal{N}_p(\beta; \boldsymbol{\mu}, \boldsymbol{\Omega}) \cdot \Phi_{nL}(\mathbf{X}\beta; \boldsymbol{\Lambda})$
$\implies p(\beta \mid \mathbf{y}_{1:n})$ is the kernel of a $\mathrm{SUN}_{p,nL}$ (F., Durante, JMLR, 2022).

The unified skew-normal (SUN) distribution: a short recap

Short recap: a $\mathrm{SUN}_{p,m}(\boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\Delta}, \gamma, \boldsymbol{\Gamma})$ distributed random variable has density

$$p(\beta) = d\mathcal{N}_p(\beta; \boldsymbol{\mu}, \boldsymbol{\Omega}) \frac{\Phi_m\left(\gamma + \boldsymbol{\Delta}^\intercal \bar{\boldsymbol{\Omega}}^{-1} \omega(\beta - \xi); \boldsymbol{\Gamma} - \boldsymbol{\Delta}^\intercal \bar{\boldsymbol{\Omega}}^{-1} \boldsymbol{\Delta}\right)}{\Phi_m(\gamma; \boldsymbol{\Gamma})},$$

where $\bar{\boldsymbol{\Omega}}$ and ω correlation and scale matrices associated to $\boldsymbol{\Omega} = \omega \bar{\boldsymbol{\Omega}} \omega$;
Moments may be quite involved, but we can develop an **i.i.d. sampler**.

Important fact: SUN additive representation

If $\beta \sim \mathrm{SUN}_{p,m}$, then it can be characterized probabilistically as a linear combination of:

- a p-variate Gaussian term;
- an m-variate truncated Gaussian component.

In the MNP, we have $p(\beta \mid \mathbf{y}_{1:n}) \propto d\mathcal{N}_p(\beta; \boldsymbol{\mu}, \boldsymbol{\Omega}) \cdot \Phi_{nL}(\mathbf{X}\beta; \boldsymbol{\Lambda})$
$\Longrightarrow$ $p(\beta \mid \mathbf{y}_{1:n})$ is the kernel of a $\mathrm{SUN}_{p,nL}$ (F., Durante, JMLR, 2022).
$\Longrightarrow$ we get an i.i.d. sampler for $p(\beta \mid \mathbf{y}_{1:n})$ **well-suited for high-dimensional scenarios**, but it becomes **infeasible as nL gets larger**.

The unified skew-normal (SUN) distribution: a short recap

Short recap: a $\mathrm{SUN}_{p,m}(\boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\Delta}, \gamma, \boldsymbol{\Gamma})$ distributed random variable has density

$$p(\beta) = d\mathcal{N}_p(\beta; \boldsymbol{\mu}, \boldsymbol{\Omega}) \frac{\Phi_m\left(\gamma + \boldsymbol{\Delta}^\intercal \bar{\boldsymbol{\Omega}}^{-1} \omega(\beta - \xi); \boldsymbol{\Gamma} - \boldsymbol{\Delta}^\intercal \bar{\boldsymbol{\Omega}}^{-1} \boldsymbol{\Delta}\right)}{\Phi_m(\gamma; \boldsymbol{\Gamma})},$$

where $\bar{\boldsymbol{\Omega}}$ and ω correlation and scale matrices associated to $\boldsymbol{\Omega} = \omega\bar{\boldsymbol{\Omega}}\omega$;
Moments may be quite involved, but we can develop an **i.i.d. sampler**.

Important fact: SUN additive representation

If $\beta \sim \mathrm{SUN}_{p,m}$, then it can be characterized probabilistically as a linear combination of:

- a p-variate Gaussian term;
- an m-variate truncated Gaussian component.

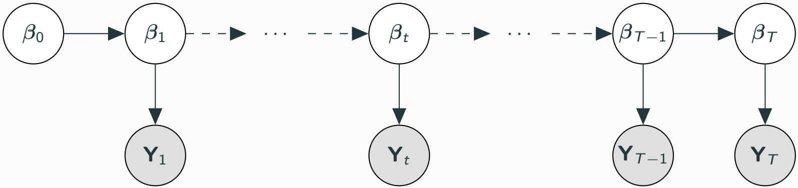
In the MNP, we have $p(\beta \mid \mathbf{y}_{1:n}) \propto d\mathcal{N}_p(\beta; \boldsymbol{\mu}, \boldsymbol{\Omega}) \cdot \Phi_{nL}(\mathbf{X}\beta; \boldsymbol{\Lambda})$
$\implies$ $p(\beta \mid \mathbf{y}_{1:n})$ is the kernel of a $\mathrm{SUN}_{p,nL}$ (F., Durante, JMLR, 2022).
$\implies$ we get an i.i.d. sampler for $p(\beta \mid \mathbf{y}_{1:n})$ **well-suited for high-dimensional scenarios**, but it becomes **infeasible as nL gets larger**.
$\implies$ **Variational Bayes** can help to overcome this issues.

The dynamic multinomial probit model

What can we say about **dynamic versions** of the MNP model that account for time-dependent observations?

What can we say about **dynamic versions** of the MNP model that account for time-dependent observations?

Dynamic multinomial probit model:



$\mathbf{Y}_t = (y_{t,1}, \dots, y_{t,n_t})$ represents the n_t categorical observations sampled at time t.

$$y_{t,i} \mid \beta_t \overset{\text{ind}}{\sim} \text{MNP}(\beta_t; \mathbf{X}_{t,i}), \quad i = 1, \dots, n_t \qquad \text{(observed)}$$
$$\beta_t = \mathbf{G}_t \beta_{t-1} + \boldsymbol{\eta}_t \qquad \text{(latent)}$$

with $\beta_0 \sim \mathcal{N}_p(\mathbf{a}_0, \mathbf{P}_0)$ independent of $\boldsymbol{\eta}_t \overset{\text{iid}}{\sim} \mathcal{N}_p(\mathbf{0}, \mathbf{W})$.

The filtering distribution of the dynamic multinomial probit

Methodological question:

Can we develop **online procedures** for the **filtering** distribution $p(\beta_t \mid \mathbf{Y}_{1:t})$?

The filtering distribution of the dynamic multinomial probit

Methodological question:

Can we develop **online procedures** for the **filtering** distribution $p(\beta_t \mid \mathbf{Y}_{1:t})$?

Rimella, F. and Rebaudo (2025+)

Calling $N_t = \sum_{s=1}^{t} n_s$ the total number of observations up to time t, under the dynamic probit model:

1. the **filtering distribution** $p(\beta_t \mid \mathbf{Y}_{1:t})$ is $\mathrm{SUN}_{p,N_t L}$;
2. the **state predictive distribution** $p(\beta_{t+1} \mid \mathbf{Y}_{1:t})$ is $\mathrm{SUN}_{p,N_t L}$;

The filtering distribution of the dynamic multinomial probit

Methodological question:

Can we develop **online procedures** for the **filtering** distribution $p(\beta_t \mid \mathbf{Y}_{1:t})$?

Rimella, F. and Rebaudo (2025+)

Calling $N_t = \sum_{s=1}^{t} n_s$ the total number of observations up to time t, under the dynamic probit model:

1. the **filtering distribution** $p(\beta_t \mid \mathbf{Y}_{1:t})$ is $\mathrm{SUN}_{p,N_t L}$;
2. the **state predictive distribution** $p(\beta_{t+1} \mid \mathbf{Y}_{1:t})$ is $\mathrm{SUN}_{p,N_t L}$;

I.i.d. sampling in principle feasible thanks to the SUN additive representation. Need to linearly combine:

- a sample from a p-variate Gaussian;
- a sample from a $N_t L$-variate truncated Gaussian.

The truncated multivariate component is the computationally-impractical part.

The filtering distribution of the dynamic multinomial probit

Methodological question:

Can we develop **online procedures** for the **filtering** distribution $p(\beta_t \mid \mathbf{Y}_{1:t})$?

Rimella, F. and Rebaudo (2025+)

Calling $N_t = \sum_{s=1}^{t} n_s$ the total number of observations up to time t, under the dynamic probit model:

1. the **filtering distribution** $p(\beta_t \mid \mathbf{Y}_{1:t})$ is $\mathrm{SUN}_{p,N_t L}$;
2. the **state predictive distribution** $p(\beta_{t+1} \mid \mathbf{Y}_{1:t})$ is $\mathrm{SUN}_{p,N_t L}$;

I.i.d. sampling in principle feasible thanks to the SUN additive representation. Need to linearly combine:

- a sample from a p-variate Gaussian;
- a sample from a $N_t L$-variate truncated Gaussian.

The truncated multivariate component is the computationally-impractical part.

Can we get a procedure that scales linearly in t?

Computational methods

Basic idea: at each time t, obtain a sample from the **target distribution** $p(\beta_{1:t} \mid \mathbf{Y}_{1:t})$, exploiting the samples obtained at previous iterations $t-1$.

Option 1: Particle filters

Basic idea: at each time t, obtain a sample from the **target distribution** $p(\boldsymbol{\beta}_{1:t} \mid \mathbf{Y}_{1:t})$, exploiting the samples obtained at previous iterations $t-1$.

The performance of particle filters relies on the **proposal** $\pi(\boldsymbol{\beta}_{t|t} \mid \boldsymbol{\beta}^{(r)}_{1:t-1|1:t-1}, \mathbf{Y}_{1:t})$ and the form of the **resampling weights** $w_t^{(r)} = w_t(\bar{\boldsymbol{\beta}}^{(r)}_{1:t|t})$, $r = 1, \ldots, R$.

Option 1: Particle filters

> **Basic idea**: at each time t, obtain a sample from the **target distribution** $p(\boldsymbol{\beta}_{1:t} \mid \mathbf{Y}_{1:t})$, exploiting the samples obtained at previous iterations $t-1$.

The performance of particle filters relies on the **proposal** $\pi(\boldsymbol{\beta}_{t|t} \mid \boldsymbol{\beta}^{(r)}_{1:t-1|1:t-1}, \mathbf{Y}_{1:t})$ and the form of the **resampling weights** $w_t^{(r)} = w_t(\bar{\boldsymbol{\beta}}^{(r)}_{1:t|t})$, $r = 1, \ldots, R$.

In the simplest case (**bootstrap** particle filter):

- $\pi(\boldsymbol{\beta}_t \mid \boldsymbol{\beta}_{1:t-1}, \mathbf{Y}_{1:t}) = p(\boldsymbol{\beta}_t \mid \boldsymbol{\beta}_{t-1})$
- $w_t^{(r)} \propto p(\mathbf{Y}_t \mid \bar{\boldsymbol{\beta}}_t^{(r)}) = \Phi_L(\mathbf{X}_{[y_{t,1}]}\bar{\boldsymbol{\beta}}^{(r)}_{t|t}; \mathbf{S}_{y_{t,1}}) \cdots \Phi_L(\mathbf{X}_{[y_{t,n_t}]}\bar{\boldsymbol{\beta}}^{(r)}_{t|t}; \mathbf{S}_{y_{t,n_t}})$

At each step t, we would need to compute $R \times n_t$ L-dimensional Gaussian cdfs.

Option 1: Particle filters

Basic idea: at each time t, obtain a sample from the **target distribution** $p(\boldsymbol{\beta}_{1:t} \mid \mathbf{Y}_{1:t})$, exploiting the samples obtained at previous iterations $t-1$.

The performance of particle filters relies on the **proposal** $\pi(\boldsymbol{\beta}_{t|t} \mid \boldsymbol{\beta}^{(r)}_{1:t-1|1:t-1}, \mathbf{Y}_{1:t})$ and the form of the **resampling weights** $w_t^{(r)} = w_t(\bar{\boldsymbol{\beta}}^{(r)}_{1:t|t})$, $r = 1, \dots, R$.

In the simplest case (**bootstrap** particle filter):

- $\pi(\boldsymbol{\beta}_t \mid \boldsymbol{\beta}_{1:t-1}, \mathbf{Y}_{1:t}) = p(\boldsymbol{\beta}_t \mid \boldsymbol{\beta}_{t-1})$
- $w_t^{(r)} \propto p(\mathbf{Y}_t \mid \bar{\boldsymbol{\beta}}_t^{(r)}) = \Phi_L(\mathbf{X}_{[y_{t,1}]}\bar{\boldsymbol{\beta}}^{(r)}_{t|t}; \mathbf{S}_{y_{t,1}}) \cdots \Phi_L(\mathbf{X}_{[y_{t,n_t}]}\bar{\boldsymbol{\beta}}^{(r)}_{t|t}; \mathbf{S}_{y_{t,n_t}})$

At each step t, we would need to compute $R \times n_t$ L-dimensional Gaussian cdfs.

$\implies$ computationally demanding.

Option 1: Particle filters

Basic idea: at each time t, obtain a sample from the **target distribution** $p(\beta_{1:t} \mid \mathbf{Y}_{1:t})$, exploiting the samples obtained at previous iterations $t-1$.

The performance of particle filters relies on the **proposal** $\pi(\beta_{t|t} \mid \beta^{(r)}_{1:t-1|1:t-1}, \mathbf{Y}_{1:t})$ and the form of the **resampling weights** $w_t^{(r)} = w_t(\bar{\beta}^{(r)}_{1:t|t})$, $r = 1, \ldots, R$.

In the simplest case (**bootstrap** particle filter):

- $\pi(\beta_t \mid \beta_{1:t-1}, \mathbf{Y}_{1:t}) = p(\beta_t \mid \beta_{t-1})$
- $w_t^{(r)} \propto p(\mathbf{Y}_t \mid \bar{\beta}_t^{(r)}) = \Phi_L(\mathbf{X}_{[y_{t,1}]}\bar{\beta}^{(r)}_{t|t}; \mathbf{S}_{y_{t,1}}) \cdots \Phi_L(\mathbf{X}_{[y_{t,n_t}]}\bar{\beta}^{(r)}_{t|t}; \mathbf{S}_{y_{t,n_t}})$

At each step t, we would need to compute $R \times n_t$ L-dimensional Gaussian cdfs.

$\implies$ computationally demanding.

Other choices for the proposal would be possible (optimal proposal as in Doucet et al., 2000), but the weigths would still have the same problem.

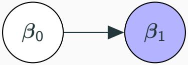
Option 2: Assumed density filtering (ADF)

Basic idea: use sequential **simple** SUN approximations $h_t(\boldsymbol{\beta}_t)$ of the **filtering distribution** $p(\boldsymbol{\beta}_t \mid \mathbf{Y}_{1:t})$, by sequentially approximating the filtering distribution of the previous time with a Gaussian density.

Option 2: Assumed density filtering (ADF)

At time $t = 0$: $p(\beta_0) = d\mathcal{N}_p(\beta_0; \mathbf{a}_0, \mathbf{P}_0)$.

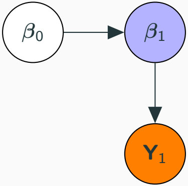
Option 2: Assumed density filtering (ADF)



At time $t = 0$: $p(\beta_0) = d\mathcal{N}_p(\beta_0; \mathbf{a}_0, \mathbf{P}_0)$.

This gives a prior distribution for β_1: $p(\beta_1) = d\mathcal{N}_p(\beta_1; \mathbf{G}\mathbf{a}_0, \mathbf{G}\mathbf{P}_0\mathbf{G}^\intercal + \mathbf{W})$.

Option 2: Assumed density filtering (ADF)

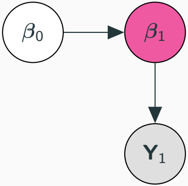


At time $t = 0$: $p(\beta_0) = d\mathcal{N}_p(\beta_0; \mathbf{a}_0, \mathbf{P}_0)$.

This gives a prior distribution for β_1: $p(\beta_1) = d\mathcal{N}_p(\beta_1; \mathbf{G}\mathbf{a}_0, \mathbf{G}\mathbf{P}_0\mathbf{G}^\intercal + \mathbf{W})$.

At time $t = 1$: $p(\beta_1 \mid \mathbf{Y}_1) \propto p(\beta_1)p(\mathbf{Y}_1 \mid \beta_1)$

Option 2: Assumed density filtering (ADF)

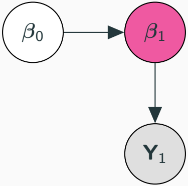


At time $t = 0$: $p(\beta_0) = d\mathcal{N}_p(\beta_0; \mathbf{a}_0, \mathbf{P}_0)$.

This gives a prior distribution for β_1: $p(\beta_1) = d\mathcal{N}_p(\beta_1; \mathbf{G}\mathbf{a}_0, \mathbf{G}\mathbf{P}_0\mathbf{G}^\intercal + \mathbf{W})$.

At time $t = 1$: $p(\beta_1 \mid \mathbf{Y}_1) \propto p(\beta_1)\boldsymbol{\Phi}_{n_1 L}(\mathbf{X}_{[\mathbf{Y}_1]}\beta_1; \boldsymbol{\Lambda}_{[\mathbf{Y}_1]}) = d\mathrm{SUN}_{p, n_1 L}$

Option 2: Assumed density filtering (ADF)



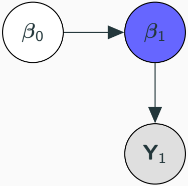
At time $t = 0$: $p(\beta_0) = d\mathcal{N}_p(\beta_0; \mathbf{a}_0, \mathbf{P}_0)$.

This gives a prior distribution for β_1: $p(\beta_1) = d\mathcal{N}_p(\beta_1; \mathbf{G}\mathbf{a}_0, \mathbf{G}\mathbf{P}_0\mathbf{G}^\intercal + \mathbf{W})$.

At time $t = 1$: $p(\beta_1 \mid \mathbf{Y}_1) \propto p(\beta_1)\boldsymbol{\Phi}_{n_1 L}(\mathbf{X}_{[\mathbf{Y}_1]}\beta_1; \boldsymbol{\Lambda}_{[\mathbf{Y}_1]}) = d\mathrm{SUN}_{p,n_1 L}$

Idea: Approximate now $p(\beta_1 \mid \mathbf{Y}_1)$ with a Gaussian density $q_1(\beta_1)$ and repeat the process at next time.

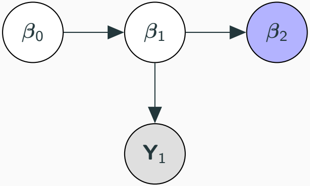
Option 2: Assumed density filtering (ADF)



At time $t = 2$:

Approximate $p(\beta_1 \mid \mathbf{Y}_1)$ with $q_1(\beta_1) = d\mathcal{N}_p(\beta_1; \boldsymbol{\mu}_{1|1}, \boldsymbol{\Omega}_{1|1})$ matching the first two moments.

Option 2: Assumed density filtering (ADF)



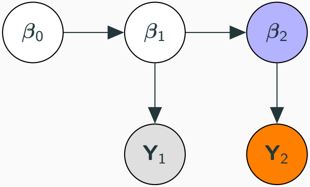
At time $t = 2$:

Approximate $p(\beta_1 \mid \mathbf{Y}_1)$ with $q_1(\beta_1) = d\mathcal{N}_p(\beta_1; \boldsymbol{\mu}_{1|1}, \boldsymbol{\Omega}_{1|1})$ matching the first two moments.

This induces a **Gaussian predictive** distribution for β_2:

$$q_1(\beta_2) = d\mathcal{N}_p(\beta_2; \mathbf{G}\boldsymbol{\mu}_{1|1}, \mathbf{G}\boldsymbol{\Omega}_{1|1}\mathbf{G}^\intercal + \mathbf{W}) \approx p(\beta_2 \mid \mathbf{Y}_1).$$

Option 2: Assumed density filtering (ADF)



At time $t = 2$:

Approximate $p(\beta_1 \mid \mathbf{Y}_1)$ with $q_1(\beta_1) = d\mathcal{N}_p(\beta_1; \boldsymbol{\mu}_{1|1}, \boldsymbol{\Omega}_{1|1})$ matching the first two moments.

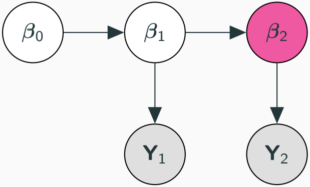
This induces a **Gaussian predictive** distribution for β_2:

$$q_1(\beta_2) = d\mathcal{N}_p(\beta_2; \mathbf{G}\boldsymbol{\mu}_{1|1}, \mathbf{G}\boldsymbol{\Omega}_{1|1}\mathbf{G}^\intercal + \mathbf{W}) \approx p(\beta_2 \mid \mathbf{Y}_1).$$

Plugging this in $p(\beta_2 \mid \mathbf{Y}_{1:2}) \propto p(\beta_2 \mid \mathbf{Y}_1)p(\mathbf{Y}_2 \mid \beta_2)$, we get the approximation

$$p(\beta_2 \mid \mathbf{Y}_{1:2}) \approx h_2(\beta_2) \propto q_1(\beta_2)p(\mathbf{Y}_2 \mid \beta_2)$$

Option 2: Assumed density filtering (ADF)



At time $t = 2$:

Approximate $p(\beta_1 \mid \mathbf{Y}_1)$ with $q_1(\beta_1) = d\mathcal{N}_p(\beta_1; \boldsymbol{\mu}_{1|1}, \boldsymbol{\Omega}_{1|1})$ matching the first two moments.

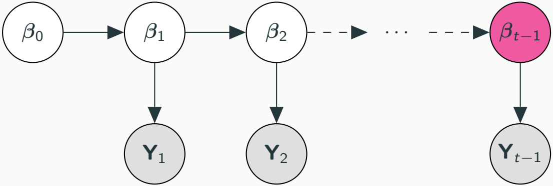
This induces a **Gaussian predictive** distribution for β_2:

$$q_1(\beta_2) = d\mathcal{N}_p(\beta_2; \mathbf{G}\boldsymbol{\mu}_{1|1}, \mathbf{G}\boldsymbol{\Omega}_{1|1}\mathbf{G}^\intercal + \mathbf{W}) \approx p(\beta_2 \mid \mathbf{Y}_1).$$

Plugging this in $p(\beta_2 \mid \mathbf{Y}_{1:2}) \propto p(\beta_2 \mid \mathbf{Y}_1)p(\mathbf{Y}_2 \mid \beta_2)$, we get the approximation

$$p(\beta_2 \mid \mathbf{Y}_{1:2}) \approx h_2(\beta_2) \propto q_1(\beta_2)p(\mathbf{Y}_2 \mid \beta_2) = d\mathrm{SUN}_{p,n_2L}.$$

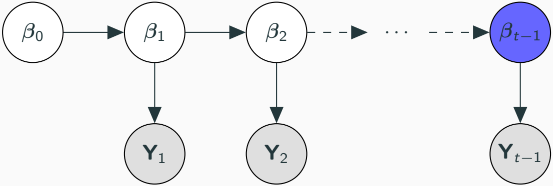
Option 2: Assumed density filtering (ADF)



At generic time t:

From previous time $t-1$, we have the **SUN** filtering approximation $h_{t-1}(\beta_{t-1})$.

Option 2: Assumed density filtering (ADF)

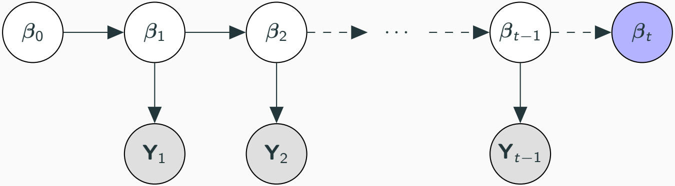


At generic time t:

From previous time $t-1$, we have the **SUN** filtering approximation $h_{t-1}(\beta_{t-1})$.

Then, take the **Gaussian** approximation $q_{t-1}(\beta_t) \approx p(\beta_{t-1} \mid \mathbf{Y}_{1:t-1})$, matching the first two moments of $h_{t-1}(\beta_{t-1})$.

Option 2: Assumed density filtering (ADF)



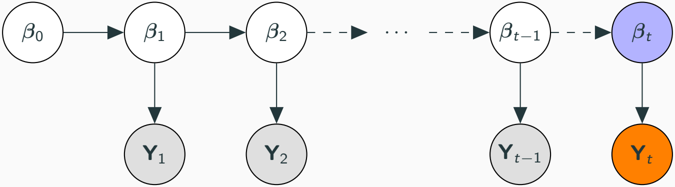
At generic time t:

From previous time $t-1$, we have the **SUN** filtering approximation $h_{t-1}(\beta_{t-1})$.

Then, take the **Gaussian** approximation $q_{t-1}(\beta_t) \approx p(\beta_{t-1} \mid \mathbf{Y}_{1:t-1})$, matching the first two moments of $h_{t-1}(\beta_{t-1})$.

This induces a **Gaussian predictive** distribution for β_t: $q_{t-1}(\beta_t) \approx p(\beta_t \mid \mathbf{Y}_{1:t-1})$.

Option 2: Assumed density filtering (ADF)



At generic time t:

From previous time $t-1$, we have the **SUN** filtering approximation $h_{t-1}(\beta_{t-1})$.

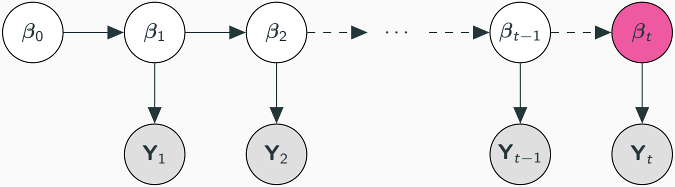
Then, take the **Gaussian** approximation $q_{t-1}(\beta_t) \approx p(\beta_{t-1} \mid \mathbf{Y}_{1:t-1})$, matching the first two moments of $h_{t-1}(\beta_{t-1})$.

This induces a **Gaussian predictive** distribution for β_t: $q_{t-1}(\beta_t) \approx p(\beta_t \mid \mathbf{Y}_{1:t-1})$.

Plugging this in $p(\beta_t \mid \mathbf{Y}_{1:t}) \propto p(\beta_t \mid \mathbf{Y}_{1:t-1})p(\mathbf{Y}_t \mid \beta_t)$, we get

$p(\beta_t \mid \mathbf{Y}_{1:t}) \approx h_t(\beta_t) \propto q_{t-1}(\beta_t)p(\mathbf{Y}_t \mid \beta_t)$

Option 2: Assumed density filtering (ADF)



At generic time t:

From previous time $t-1$, we have the **SUN** filtering approximation $h_{t-1}(\beta_{t-1})$.

Then, take the **Gaussian** approximation $q_{t-1}(\beta_t) \approx p(\beta_{t-1} \mid \mathbf{Y}_{1:t-1})$, matching the first two moments of $h_{t-1}(\beta_{t-1})$.

This induces a **Gaussian predictive** distribution for β_t: $q_{t-1}(\beta_t) \approx p(\beta_t \mid \mathbf{Y}_{1:t-1})$.

Plugging this in $p(\beta_t \mid \mathbf{Y}_{1:t}) \propto p(\beta_t \mid \mathbf{Y}_{1:t-1})p(\mathbf{Y}_t \mid \beta_t)$, we get

$p(\beta_t \mid \mathbf{Y}_{1:t}) \approx h_t(\beta_t) \propto q_{t-1}(\beta_t)p(\mathbf{Y}_t \mid \beta_t) = d\mathrm{SUN}_{p,n_t L}$

Wrap up of methods

Computational bottlenecks of the various methods seen so far:

- i.i.d. sampler: sampling from $N_t \times L$-variate truncated Gaussian, $N_t = \sum_{s=1}^{t} n_s$;
- particle filter: computation of $R \times n_t$ cdfs of L-variate Gaussians at each time;
- assumed density filtering (ADF): sampling from an $n_t \times L$-variate truncated Gaussian at each time t (arising from the $\mathrm{SUN}_{p, n_t \times L}$ approximated filtering distribution of time t).

Wrap up of methods

Computational bottlenecks of the various methods seen so far:

- i.i.d. sampler: sampling from $N_t \times L$-variate truncated Gaussian, $N_t = \sum_{s=1}^{t} n_s$;
- particle filter: computation of $R \times n_t$ cdfs of L-variate Gaussians at each time;
- assumed density filtering (ADF): sampling from an $n_t \times L$-variate truncated Gaussian at each time t (arising from the $\mathrm{SUN}_{p,n_t \times L}$ approximated filtering distribution of time t).

Can we get something more efficient? Can we use the spirit of ADF but avoid the $n_t \times L$-variate truncated Gaussian?

Option 3: (Sequential) Expectation Propagation

Basic idea: see the **filtering distribution**

$$p(\beta_t \mid \mathbf{Y}_{1:t}) \propto p(\beta_t \mid \mathbf{Y}_{1:t-1}) \prod_{i=1}^{n_t} p(y_{t,i} \mid \beta_t)$$

as the posterior distribution in a model where

- $p(\beta_t \mid \mathbf{Y}_{1:t-1})$ is the prior,
- $p(y_{t,i} \mid \beta_t)$ is the likelihood of observation $y_{i,t}$, $i = 1, \dots, n_t$.

Approximate this posterior via expectation propagation (EP) with a **Gaussian approximation** $q_t(\beta_t) = d\mathcal{N}_p(\beta_t; \boldsymbol{\mu}_{t|t}, \boldsymbol{\Omega}_{t|t})$.

Option 3: (Sequential) Expectation Propagation

In EP, we approximate $p(\beta_t \mid \mathbf{Y}_{1:t}) \propto p(\beta_t \mid \mathbf{Y}_{1:t-1}) \prod_{i=1}^{n_t} p(y_{t,i} \mid \beta_t)$ with

$$q_t(\beta_t) \propto q_{t-1}(\beta_t) \prod_{i=1}^{n_t} q_{t,i}(\beta_t), \quad \text{where}$$

- $q_{t-1}(\beta_t) = d\mathcal{N}_p(\beta_t; \mathbf{G}\boldsymbol{\mu}_{t-1|t-1}, \mathbf{G}\boldsymbol{\Omega}_{t-1|t-1}\mathbf{G}^\intercal + \mathbf{W}) \approx p(\beta_t \mid \mathbf{Y}_{1:t-1})$,
- $q_{t,i}(\beta_t) \propto \exp\{-\frac{1}{2}\beta_t^\intercal \mathbf{Q}_{t,i}\beta_t + \beta_t^\intercal \mathbf{r}_{t,i}\} \approx p(y_{t,i} \mid \beta_t)$ for $i = 1, \ldots, n_t$.

Since we have Gaussian prior and Gaussian likelihood

$$q_t(\beta_t) = d\mathcal{N}_p(\beta_t; \boldsymbol{\mu}_{t|t}, \boldsymbol{\Omega}_{t|t}),$$

where $\boldsymbol{\mu}_{t|t}$ and $\boldsymbol{\Omega}_{t|t}$ are determined by $\mathbf{r}_{t,i}$ and $\mathbf{Q}_{t,i}$, $i = 1, \ldots, n_t$.

Option 3: (Sequential) Expectation Propagation

In EP, we approximate $p(\beta_t \mid \mathbf{Y}_{1:t}) \propto p(\beta_t \mid \mathbf{Y}_{1:t-1}) \prod_{i=1}^{n_t} p(y_{t,i} \mid \beta_t)$ with

$$q_t(\beta_t) \propto q_{t-1}(\beta_t) \prod_{i=1}^{n_t} q_{t,i}(\beta_t), \quad \text{where}$$

- $q_{t-1}(\beta_t) = d\mathcal{N}_p(\beta_t; \mathbf{G}\boldsymbol{\mu}_{t-1|t-1}, \mathbf{G}\boldsymbol{\Omega}_{t-1|t-1}\mathbf{G}^\intercal + \mathbf{W}) \approx p(\beta_t \mid \mathbf{Y}_{1:t-1})$,
- $q_{t,i}(\beta_t) \propto \exp\{-\frac{1}{2}\beta_t^\intercal \mathbf{Q}_{t,i}\beta_t + \beta_t^\intercal \mathbf{r}_{t,i}\} \approx p(y_{t,i} \mid \beta_t)$ for $i = 1, \ldots, n_t$.

Since we have Gaussian prior and Gaussian likelihood

$$q_t(\beta_t) = d\mathcal{N}_p(\beta_t; \boldsymbol{\mu}_{t|t}, \boldsymbol{\Omega}_{t|t}),$$

where $\boldsymbol{\mu}_{t|t}$ and $\boldsymbol{\Omega}_{t|t}$ are determined by $\mathbf{r}_{t,i}$ and $\mathbf{Q}_{t,i}$, $i = 1, \ldots, n_t$.

The quantities $\mathbf{r}_{t,i}$ and $\mathbf{Q}_{t,i}$, $i = 1, \ldots, n_t$ are fixed in an "optimal" way via EP **moment matching conditions**, available in **closed form**.

Marginal likelihood via EP

EP can also be used to get an approximation of the **marginal likelihood**

$$p(\mathbf{Y}_{1:T}) = p(\mathbf{Y}_1)p(\mathbf{Y}_2 \mid \mathbf{Y}_1)p(\mathbf{Y}_3 \mid \mathbf{Y}_{1:2}) \cdots p(\mathbf{Y}_T \mid \mathbf{Y}_{1:T-1})$$

since at each time it can give an approximation of $p(\mathbf{Y}_t \mid \mathbf{Y}_{1:t-1})$.

Expectation-maximization (EM) can be used to **estimate** the desired **hyper-parameters** (e.g., $\boldsymbol{\Sigma}$) by maximizing the marginal likelihood.

Experiments and results

EM experiments

We checked the ability of the proposed EM procedure to maximize the log-marginal likelihood and obtain meaningful estimates for $\mathbf{\Sigma}$.

We ran the following experiment generating synthetic data as follows:

- we took $L = 3$ and $T = 10$,
- 15 individuals on average at each time step,
- 2 individual-specific covariates $+$ 3 choice-specific covariates ($+$ choice-specific intercepts).

EM experiments

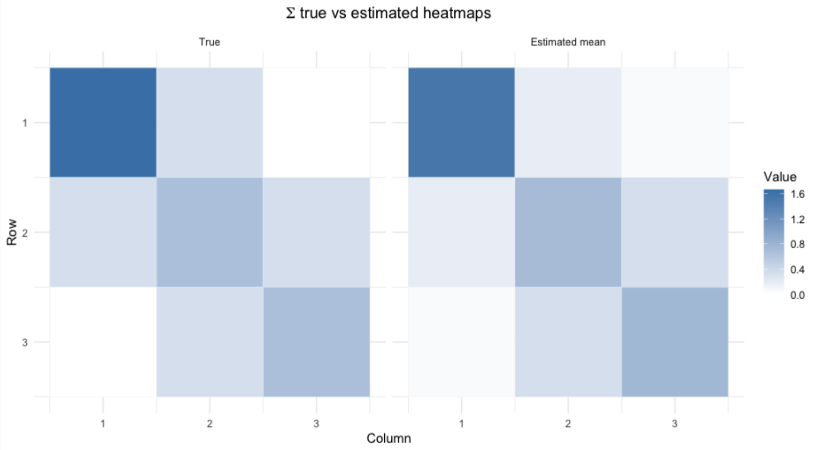
We checked the ability of the proposed EM procedure to maximize the log-marginal likelihood and obtain meaningful estimates for $\mathbf{\Sigma}$.

We ran the following experiment generating synthetic data as follows:

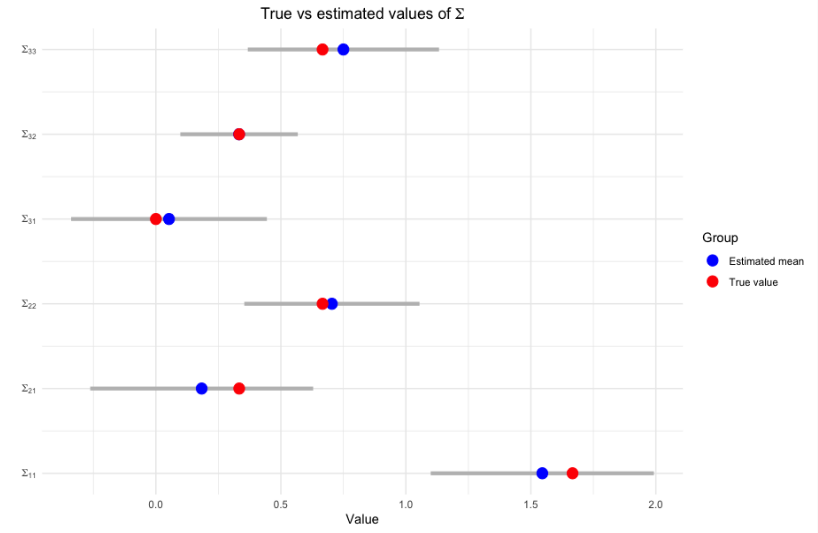
- we took $L = 3$ and $T = 10$,
- 15 individuals on average at each time step,
- 2 individual-specific covariates + 3 choice-specific covariates (+ choice-specific intercepts).
- Keeping the above covariates fixed, we generated 50 datasets from the model:
  - each time generating a different trajectory of the parameters $\beta_{0:T}$,
  - then, given the generated $\beta_{0:T}$, we generated observations $\mathbf{Y}_{1:T}$.

Is the EM effective on average in the estimation of $\mathbf{\Sigma}$?

EM experiments



EM experiments



Filter experiments

We then compared the performance of the algorithms in a setting where:

- $T = 20$,
- $n_t \sim \text{Pois}(10)$ for each t,
- $L = 3$,
- there are 2 individual-specific characteristics,
- there are 3 choice-specific characteristics.

That is,

$$u_{t,i,\ell} = \alpha_{t,0,\ell} + \boldsymbol{\xi}_{t,i}^{\intercal}\boldsymbol{\alpha}_{t,\ell} + \boldsymbol{\zeta}_{t,i,\ell}^{\intercal}\boldsymbol{\gamma}_t + \varepsilon_{t,i,\ell}, \quad \ell = 1, \ldots, 3,$$

where $\boldsymbol{\xi}_{t,i}$ has dimension 2 and $\boldsymbol{\zeta}_{t,i,\ell}$ has dimension 3.

The vector of parameters is then

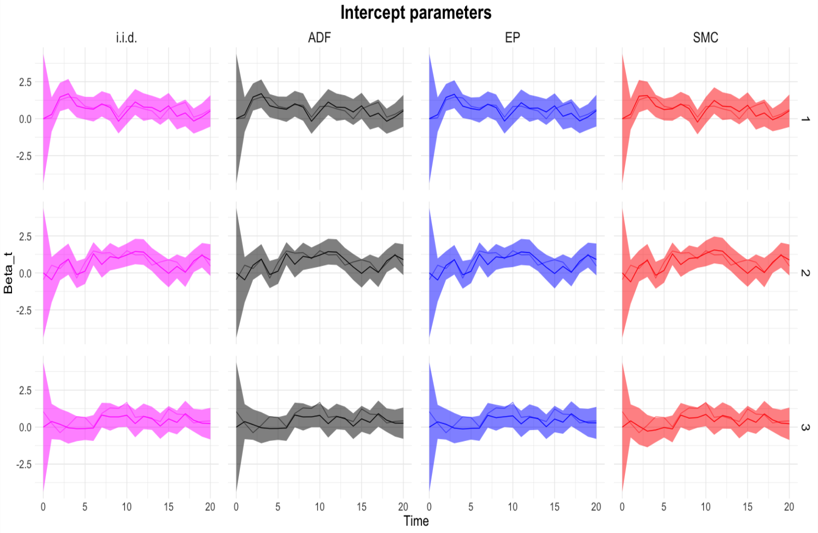
$$\boldsymbol{\beta}_t = (\alpha_{t,0,1}, \alpha_{t,0,2}, \alpha_{t,0,3}, \boldsymbol{\alpha}_{t,1}^{\intercal}, \boldsymbol{\alpha}_{t,2}^{\intercal}, \boldsymbol{\alpha}_{t,3}^{\intercal}, \gamma_t^{\intercal})^{\intercal} \in \mathbb{R}^{12}.$$

Filter experiments: running times

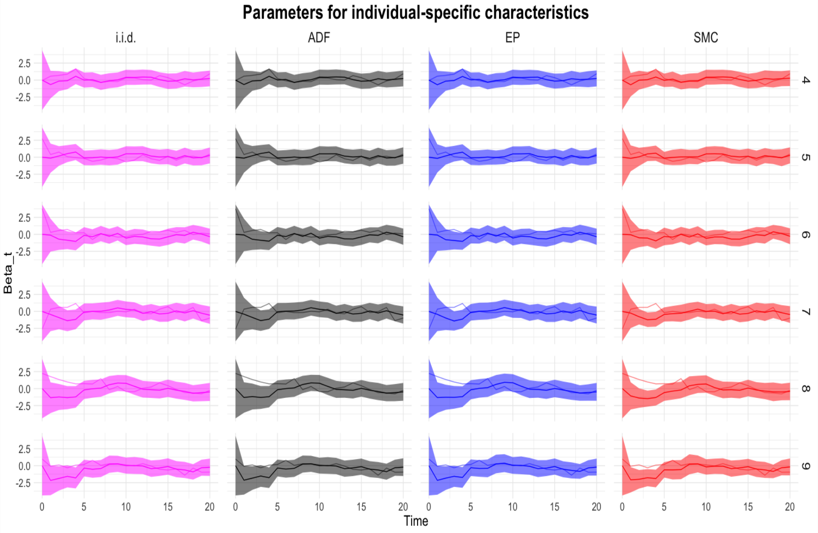
Method	i.i.d.	ADF	EP	SMC
Time (in seconds)	66,766.10	7.05	17.23	528.60

Table 1: Running times based on 5,000 samples.

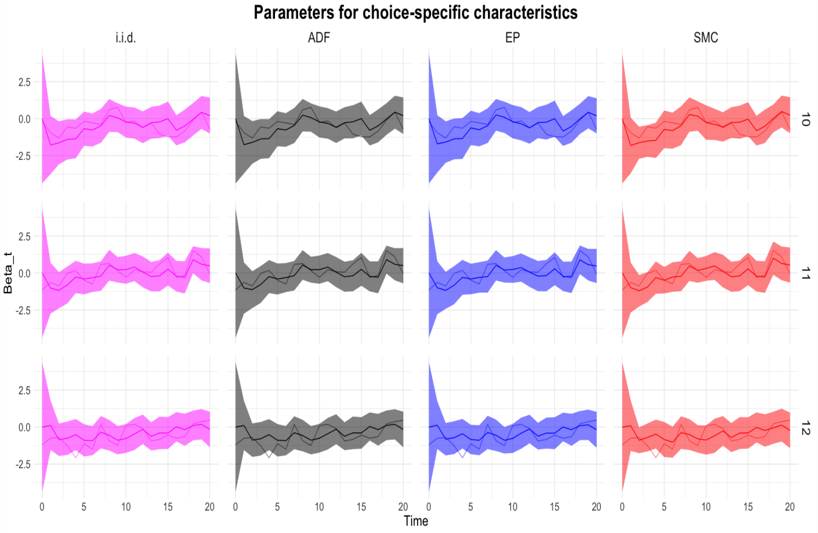
Filter experiments: intercepts parameters



Filter experiments: individual-specific characteristics parameters



Filter experiments: choice-specific characteristics parameters



Filter experiments: higher dimension

If we increase the dimension of the problem, taking

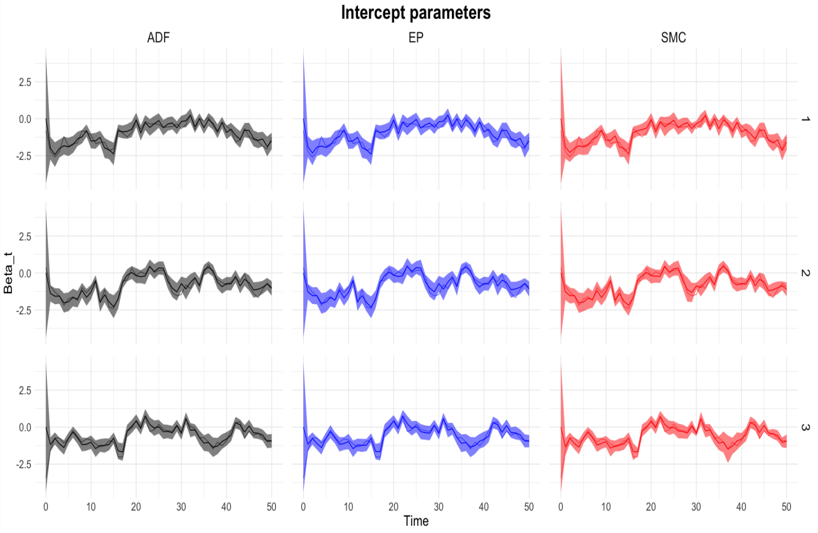
- $T = 50$,
- $n_t \sim \text{Pois}(50)$ for each t,
- the rest as before:
  - $L = 3$,
  - 2 individual-specific characteristics,
  - 3 choice-specific characteristics.

we obtain the following running times

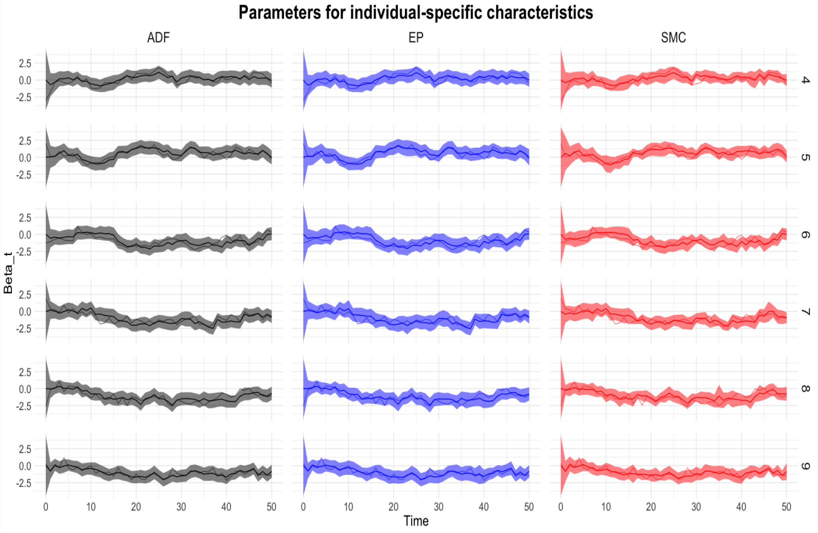
Method	i.i.d.	ADF	EP	SMC
Time (in seconds)	NA	599.98	199.69	6323.35

Table 2: Running times based on 5,000 samples.

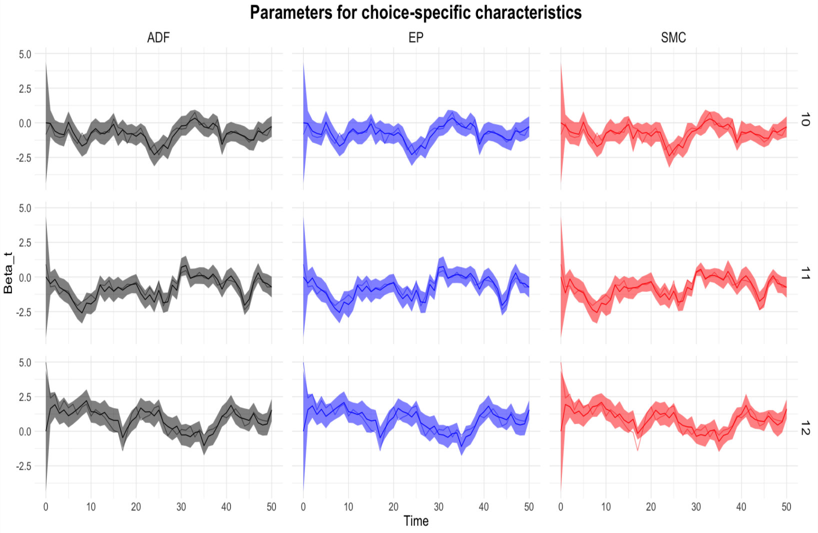
Filter experiments: intercepts parameters



Filter experiments: individual-specific characteristics parameters



Filter experiments: choice-specific characteristics parameters



Conclusions

- We have considered multiple filtering methods to make inference in dynamic discrete-choice models based on multinomial probit likelihoods.
- The **i.i.d. sampler** seems **impractical** already in moderate dimensions.
- **Sequential Monte Carlo** (SMC) procedures do **not** seem an **appropriate** solution due to the need to compute the MNP likelihood multiple times.
- The **assumed density filtering** (ADF) procedure is **efficient in scenarios where $n_t \times L$ never exceeds a few hundred**.
- In **higher dimensions**, **sequential expectation propagation** (EP) is the **most efficient** method, with accuracy comparable to the one of ADF.
- An **expectation-maximization** (EM) can be used to estimate the hyperparameters, like, e.g., the covariance matrix of the error terms $\boldsymbol{\Sigma}$.
- Ongoing work on an Expedia dataset about bookings of properties across time: arxiv preprint coming in the coming months!

Thank You!