UNIVERSITÄT BIELEFELD
Faculty of Business Administration and Economics

Facilitating probit likelihood optimization

Lennart Oelschläger   Dietmar Bauer

Bielefeld University, Empirical Methods Department, Econometrics Group

12 September 2023

Outline

Outline

Model motivation

One of the most important questions (for me) when attending a conference ...

Model motivation

One of the most important questions (for me) when attending a conference ... where to sleep?

	at home (commuting)		
cost per night	0 €	90 €	70 €
distance to venue	120 km	2 km	20 km

Model motivation

One of the most important questions (for me) when attending a conference ... where to sleep?

	at home (commuting)		
cost per night	0 €	90 €	70 €
distance to venue	120 km	2 km	20 km

This is a discrete choice setting:

- deciders choose among a discrete set of alternatives
- based on decider- and/or alternative-specific attributes.

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

and can answer questions like

- What is the probability of each choice being made?
- Which attributes greatly influence decisions?
- How are attribute trade-offs made?
- Do different population segments choose differently?
- What is the impact of new choice alternatives?

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

and can answer questions like

- What is the probability of each choice being made?
- Which attributes greatly influence decisions?
- How are attribute trade-offs made?
- Do different population segments choose differently?
- What is the impact of new choice alternatives?

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

and can answer questions like

- What is the probability of each choice being made?
- Which attributes greatly influence decisions?
- How are attribute trade-offs made?
- Do different population segments choose differently?
- What is the impact of new choice alternatives?

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

and can answer questions like

- What is the probability of each choice being made?
- Which attributes greatly influence decisions?
- How are attribute trade-offs made?
- Do different population segments choose differently?
- What is the impact of new choice alternatives?

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

and can answer questions like

- What is the probability of each choice being made?
- Which attributes greatly influence decisions?
- How are attribute trade-offs made?
- Do different population segments choose differently?
- What is the impact of new choice alternatives?

Model motivation

The multinomial probit model is widely used to analyze such discrete choices in fields like

- politics (Ampel vs. opposition vs. non-voter)
- marketing (Apple vs. Samsung vs. ...)
- transportation (private vehicle vs. public transport vs. bike vs. ...)

and can answer questions like

- What is the probability of each choice being made?
- Which attributes greatly influence decisions?
- How are attribute trade-offs made?
- Do different population segments choose differently?
- What is the impact of new choice alternatives?

Model definition

Formally, the model

- connects attributes X_n
- to the choice $y_n \in \{1, \dots, J\}$
- via latent utilities U_n that are defined as
  1. a linear function $V_n = X_n\beta$
  2. plus a Gaussian error ε_n
- assuming that deciders seek to maximize their utility.

Model definition

Formally, the model

- connects attributes X_n
- to the choice $y_n \in \{1, \dots, J\}$
- via latent utilities U_n that are defined as
  1. a linear function $V_n = X_n \beta$
  2. plus a Gaussian error ε_n
- assuming that deciders seek to maximize their utility.

In formulas:

$$U_n = V_n + \varepsilon_n$$
$$V_n = X_n \beta$$
$$\varepsilon_n \sim N(0, \Sigma)$$
$$y_n = \arg\max U_n$$

Model definition

Formally, the model

- connects attributes X_n
- to the choice $y_n \in \{1, \dots, J\}$
- via latent utilities U_n that are defined as
  1. a linear function $V_n = X_n\beta$
  2. plus a Gaussian error ε_n
- assuming that deciders seek to maximize their utility.

In formulas:

$$U_n = V_n + \varepsilon_n$$
$$V_n = X_n\beta$$
$$\varepsilon_n \sim N(0, \Sigma)$$
$$y_n = \arg\max U_n$$

Under this model, the choice probability for alternative $j \in \{1, \dots, J\}$ equals

$$P_{n,j} = \Phi(-\Delta_j X_n \beta \mid 0; \Delta_j \Sigma \Delta_j').$$

Technical detail: the operator Δ_j takes utility differences with respect to alternative j for identification.

Model definition

Formally, the model

- connects attributes X_n
- to the choice $y_n \in \{1, \dots, J\}$
- via latent utilities U_n that are defined as
  1. a linear function $V_n = X_n\beta$
  2. plus a Gaussian error ε_n
- assuming that deciders seek to maximize their utility.

In formulas:

$$U_n = V_n + \varepsilon_n$$
$$V_n = X_n\beta$$
$$\varepsilon_n \sim N(0, \Sigma)$$
$$y_n = \arg\max U_n$$

Under this model, the choice probability for alternative $j \in \{1, \dots, J\}$ equals

$$P_{n,j} = \Phi(-\Delta_j X_n \beta \mid 0; \Delta_j \Sigma \Delta_j').$$

This involves evaluating the Gaussian CDF, which has no closed form.

Technical detail: the operator Δ_j takes utility differences with respect to alternative j for identification.

Model estimation

How to fit this model to observed choice data $(X_n, y_n)_n$?

Model estimation

How to fit this model to observed choice data $(X_n, y_n)_n$? The widely used procedure is to

1. define the log-likelihood function

$$\ell(\beta, \Sigma \mid X, y) = \sum_{n,j} 1(y_n = j) \log P_{n,j}$$

2. and maximize it numerically over the parameters (β, Σ).

Technical detail: Σ is a restricted matrix but can be parametrized as a vector.

Model estimation

How to fit this model to observed choice data $(X_n, y_n)_n$? The widely used procedure is to

1. define the log-likelihood function

$$\ell(\beta, \Sigma \mid X, y) = \sum_{n,j} 1(y_n = j) \log P_{n,j}$$

2. and maximize it numerically over the parameters (β, Σ).

But this involves two challenges:

- curse of dimensionality $\Rightarrow$ optimization becomes slow
- approximation required $\Rightarrow$ optimization becomes unstable

Technical detail: Σ is a restricted matrix but can be parametrized as a vector.

Model estimation

How to fit this model to observed choice data $(X_n, y_n)_n$? The widely used procedure is to

1. define the log-likelihood function

$$\ell(\beta, \Sigma \mid X, y) = \sum_{n,j} 1(y_n = j) \log P_{n,j}$$

2. and maximize it numerically over the parameters (β, Σ).

But this involves two challenges:

- curse of dimensionality $\Rightarrow$ optimization becomes slow
- approximation required $\Rightarrow$ optimization becomes unstable

Research question: How to make probit estimation **faster** and **more reliable**?

Technical detail: Σ is a restricted matrix but can be parametrized as a vector.

Outline

1 The multinomial probit model: purpose and estimation

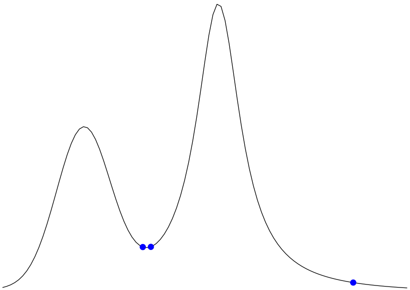
2 Numerical optimization and the initialization effect

3 Our initialization strategy for probit likelihood optimization

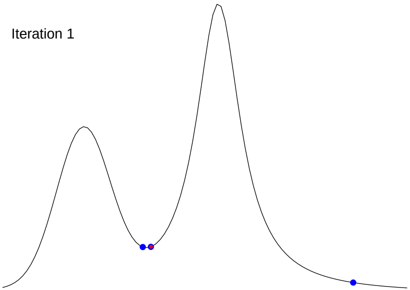
4 How does the strategy perform in comparison to random initialization?

5 Takeaways

Numerical optimization of $\ell(\theta)$ means iteratively adjusting θ to find the *globally* best $\ell(\theta)$. And most algorithms need to start at some user-defined initial value θ_0.



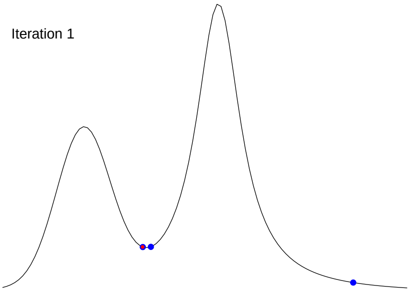
Numerical optimization of $\ell(\theta)$ means iteratively adjusting θ to find the *globally* best $\ell(\theta)$. And most algorithms need to start at some user-defined initial value θ_0.



Depending on θ_0, the optimization

- works fine 🙂

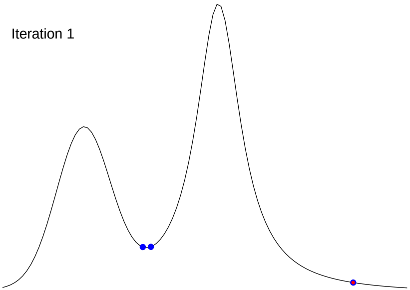
Numerical optimization of $\ell(\theta)$ means iteratively adjusting θ to find the *globally* best $\ell(\theta)$.
And most algorithms need to start at some user-defined initial value θ_0.



Depending on θ_0, the optimization

- works fine 🙂
- does not work ☹️

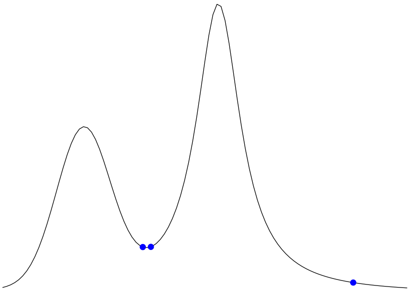
Numerical optimization of $\ell(\theta)$ means iteratively adjusting θ to find the *globally* best $\ell(\theta)$.
And most algorithms need to start at some user-defined initial value θ_0.



Depending on θ_0, the optimization

- works fine 🙂
- does not work ☹️
- takes an eternity 😴

Numerical optimization of $\ell(\theta)$ means iteratively adjusting θ to find the *globally* best $\ell(\theta)$. And most algorithms need to start at some user-defined initial value θ_0.



Depending on θ_0, the optimization

- works fine 🙄
- does not work ☹️
- takes an eternity 😴

Aim to initialize close to the global optimum

1. to reduce computation time and
2. to avoid local optima.

Outline

Initialization strategy

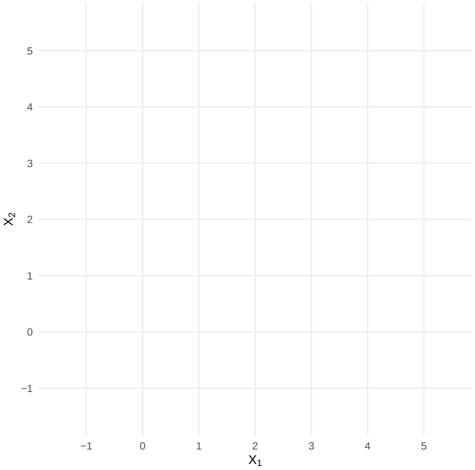
For the numerical optimization of the probit log-likelihood function

$$\ell(\beta, \Sigma \mid X, y) = \sum_{n,j} 1(y_n = j) \log P_{n,j}$$

we can **quickly** find **consistent** initial values

- β_0 via exploiting a constant utility direction,
- and Σ_0 conditional on β_0 via Gibbs sampling.

Initialize β

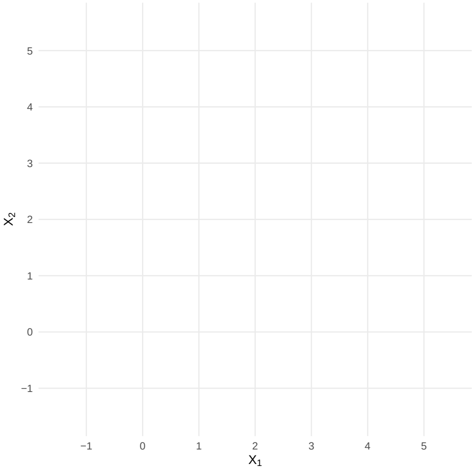


Assume

$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

(such a definition is typical for alternative-varying X with alternative-specific β, e.g., travel time)

Initialize β



Assume

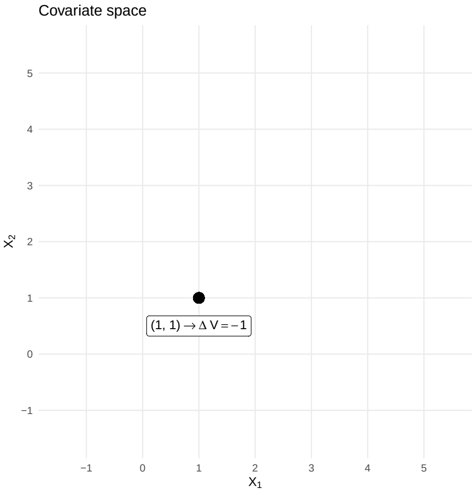
$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

(such a definition is typical for alternative-varying X with alternative-specific β, e.g., travel time)

Let

$$\begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

Initialize β



Assume

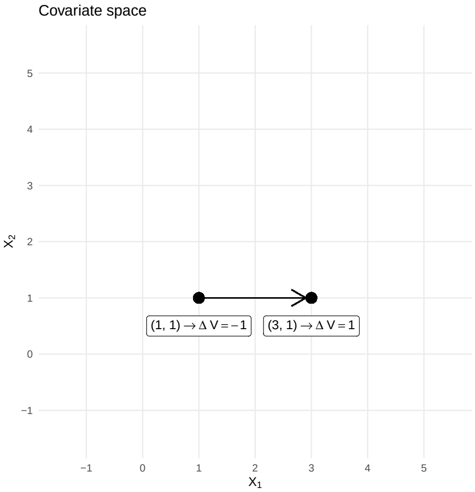
$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

(such a definition is typical for alternative-varying X with alternative-specific β, e.g., travel time)

Let

$$\begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

Initialize β



Assume

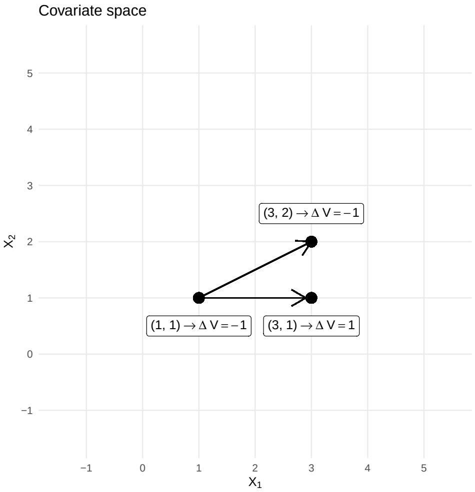
$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

(such a definition is typical for alternative-varying X with alternative-specific β, e.g., travel time)

Let

$$\begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

Initialize β



Assume

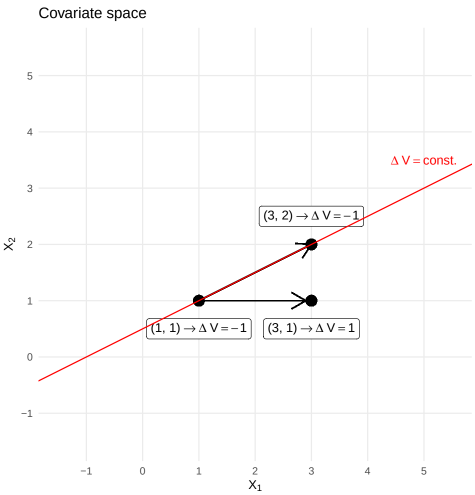
$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

(such a definition is typical for alternative-varying X with alternative-specific β, e.g., travel time)

Let

$$\begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

Initialize β



Assume

$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

(such a definition is typical for alternative-varying X with alternative-specific β, e.g., travel time)

Let

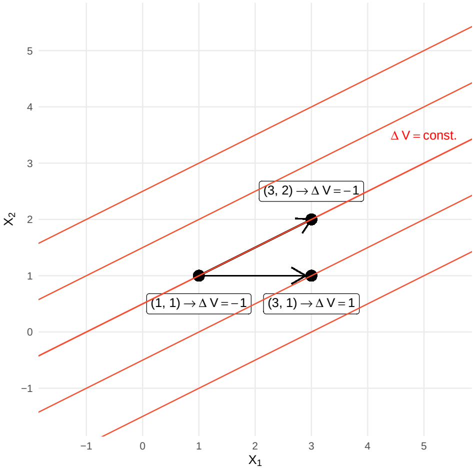
$$\begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

We find the direction

$$\overrightarrow{\begin{pmatrix} 1 \\ 0.5 \end{pmatrix}} = \overrightarrow{\begin{pmatrix} 1/\beta_1 \\ 1/\beta_2 \end{pmatrix}}$$

in which $\Delta V = V_1 - V_2 = \text{const.}$

Initialize β



Assume

$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

(such a definition is typical for alternative-varying X with alternative-specific β, e.g., travel time)

Let

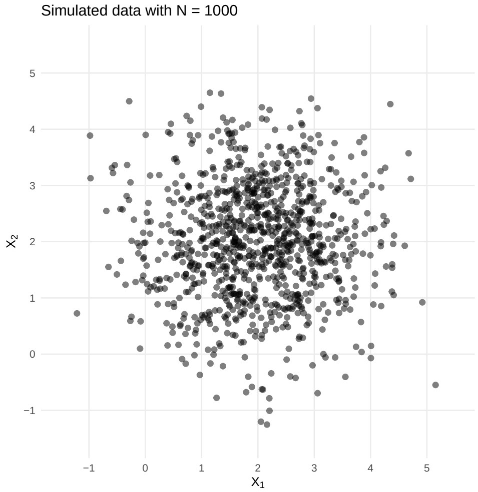
$$\begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

We find the direction

$$\overrightarrow{\begin{pmatrix} 1 \\ 0.5 \end{pmatrix}} = \overrightarrow{\begin{pmatrix} 1/\beta_1 \\ 1/\beta_2 \end{pmatrix}}$$

in which $\Delta V = V_1 - V_2 = \text{const.}$

Initialize β



Assume

$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

(such a definition is typical for alternative-varying X with alternative-specific β, e.g., travel time)

Let

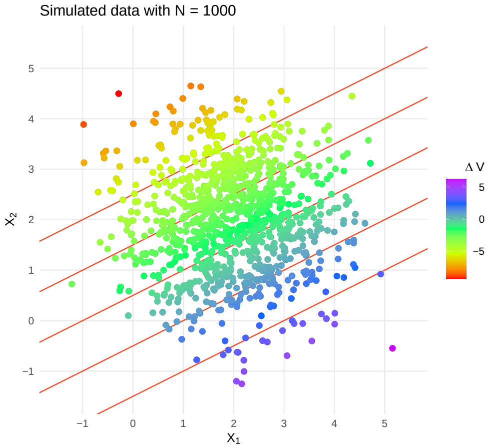
$$\begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

We find the direction

$$\overrightarrow{\begin{pmatrix} 1 \\ 0.5 \end{pmatrix}} = \overrightarrow{\begin{pmatrix} 1/\beta_1 \\ 1/\beta_2 \end{pmatrix}}$$

in which $\Delta V = V_1 - V_2 = \text{const.}$

Initialize β



Assume

$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} X_1 & 0 \\ 0 & X_2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

(such a definition is typical for alternative-varying X with alternative-specific β, e.g., travel time)

Let

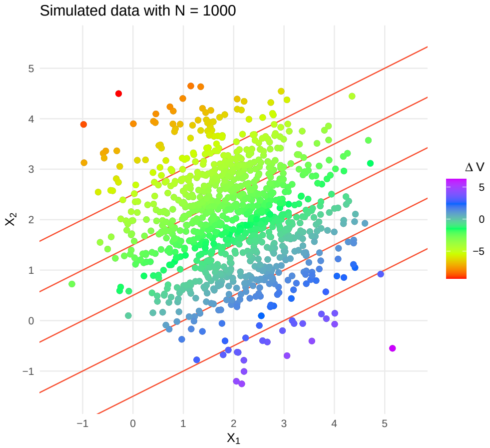
$$\begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

We find the direction

$$\overrightarrow{\begin{pmatrix} 1 \\ 0.5 \end{pmatrix}} = \overrightarrow{\begin{pmatrix} 1/\beta_1 \\ 1/\beta_2 \end{pmatrix}}$$

in which $\Delta V = V_1 - V_2 = \text{const.}$

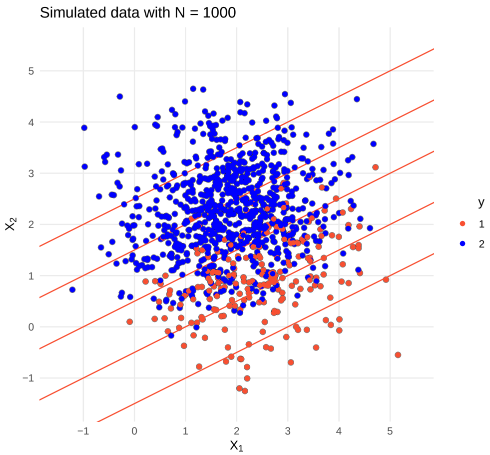
Initialize β



We do not observe ΔV

(depends on the unknown β)

Initialize β



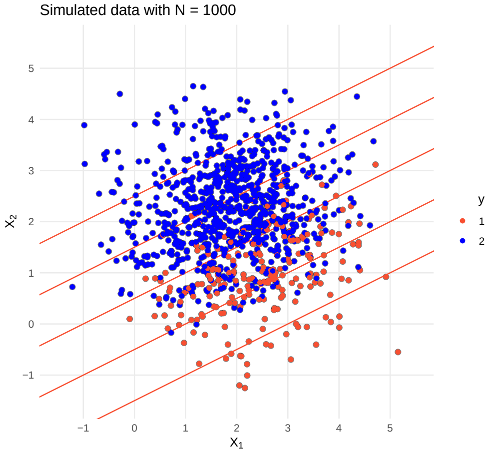
We do not observe ΔV

(depends on the unknown β)

but we observe the choices y.

UNIVERSITÄT BIELEFELD
Faculty of Business Administration and Economics

Initialize β



We do not observe ΔV

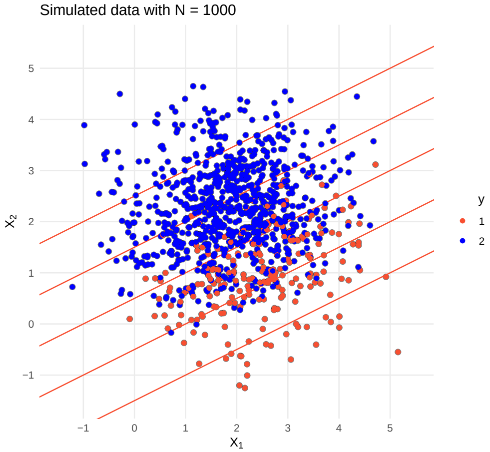
(depends on the unknown β)

but we observe the choices y.

They are disturbed by the error-term ε.

(here I used $\Sigma = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$)

Initialize β



We do not observe ΔV

(depends on the unknown β)

but we observe the choices y.

They are disturbed by the error-term ε.

(here I used $\Sigma = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$)

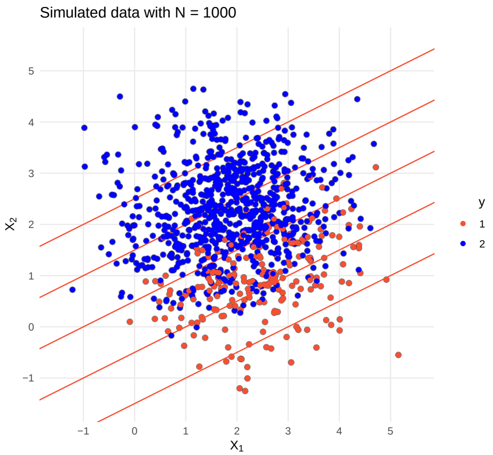
But we can still identify

$$\overrightarrow{\begin{pmatrix} 1/\beta_1 \\ 1/\beta_2 \end{pmatrix}}$$

as the kernel of $\text{Cor}(X, y)$.

(constant choice probability in this direction)

Initialize β



We do not observe ΔV

(depends on the unknown β)

but we observe the choices y.

They are disturbed by the error-term ε.

(here I used $\Sigma = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$)

But we can still identify

$$\overrightarrow{\begin{pmatrix} 1/\beta_1 \\ 1/\beta_2 \end{pmatrix}}$$

as the kernel of $\operatorname{Cor}(X, y)$.

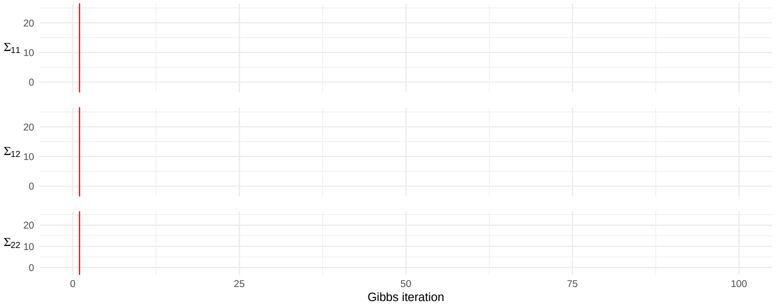
(constant choice probability in this direction)

This gives an initial estimator $\hat{\beta}_0$ that can be shown to consistent as $N \to \infty$.

Initialize Σ

Now that we have a guess β_0, we can draw Σ conditional on β_0:

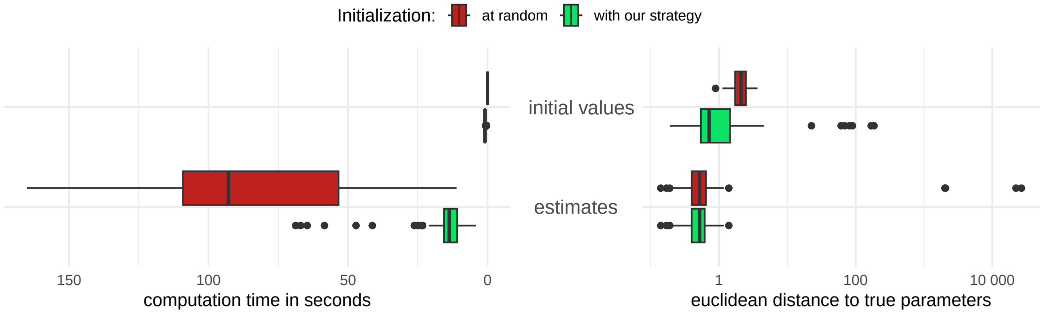
1. $(U_n)_n \mid \Sigma, \beta_0, (X_n, y_n)_n \sim$ truncated normal
2. $\Sigma \mid \beta_0, (U_n)_n \sim$ inverse Wishart



Outline

		N: 100	200	1000
J	2	☐	☐	☐
	3	☐	■	☐
	4	☐	☐	☐

100 simulated data sets with 200 deciders and 3 alternatives

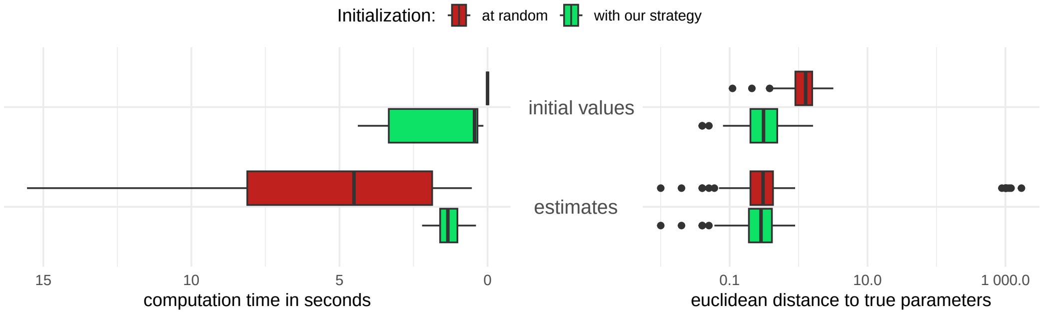


Simulation details: one alternative-varying regressor connected to alternative-varying coefficients; fully flexible Σ; true parameters and random initial values were drawn from a standard normal distribution.

UNIVERSITÄT BIELEFELD
Faculty of Business Administration and Economics

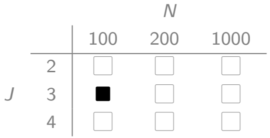
		N		
		100	200	1000
J	2	■	□	□
	3	□	□	□
	4	□	□	□

100 simulated data sets with 100 deciders and 2 alternatives



Simulation details: one alternative-varying regressor connected to alternative-varying coefficients; fully flexible Σ; true parameters and random initial values were drawn from a standard normal distribution.

UNIVERSITÄT BIELEFELD
Faculty of Business Administration and Economics



100 simulated data sets with 100 deciders and 3 alternatives

Simulation details: one alternative-varying regressor connected to alternative-varying coefficients; fully flexible Σ; true parameters and random initial values were drawn from a standard normal distribution.

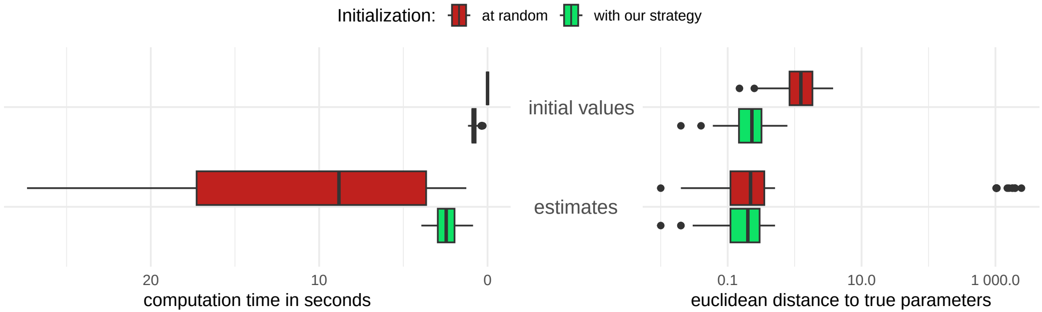
		N		
		100	200	1000
J	2	□	□	□
	3	□	□	□
	4	■	□	□

100 simulated data sets with 100 deciders and 4 alternatives

Simulation details: one alternative-varying regressor connected to alternative-varying coefficients; fully flexible Σ; true parameters and random initial values were drawn from a standard normal distribution.

		N		
		100	200	1000
J	2	□	■	□
	3	□	□	□
	4	□	□	□

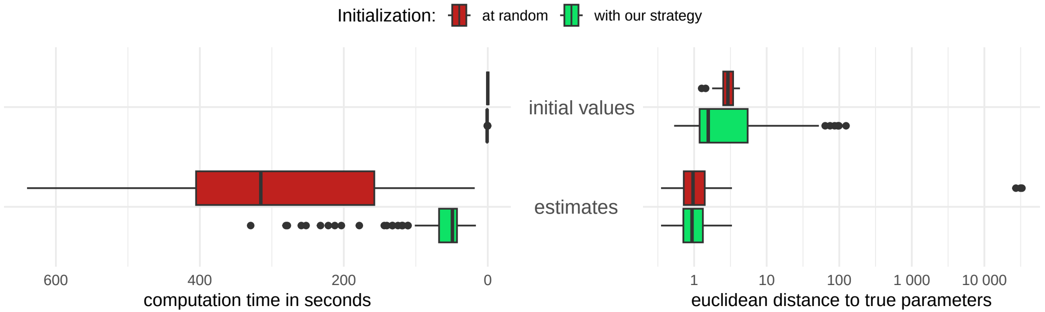
100 simulated data sets with 200 deciders and 2 alternatives



💡 Simulation details: one alternative-varying regressor connected to alternative-varying coefficients; fully flexible Σ; true parameters and random initial values were drawn from a standard normal distribution.

		N		
		100	200	1000
J	2	□	□	□
	3	□	□	□
	4	□	■	□

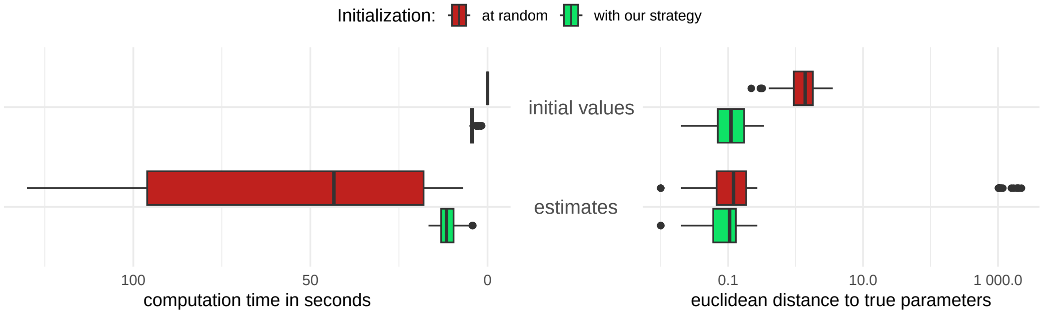
100 simulated data sets with 200 deciders and 4 alternatives



Simulation details: one alternative-varying regressor connected to alternative-varying coefficients; fully flexible Σ; true parameters and random initial values were drawn from a standard normal distribution.

		N		
		100	200	1000
J	2	□	□	■
	3	□	□	□
	4	□	□	□

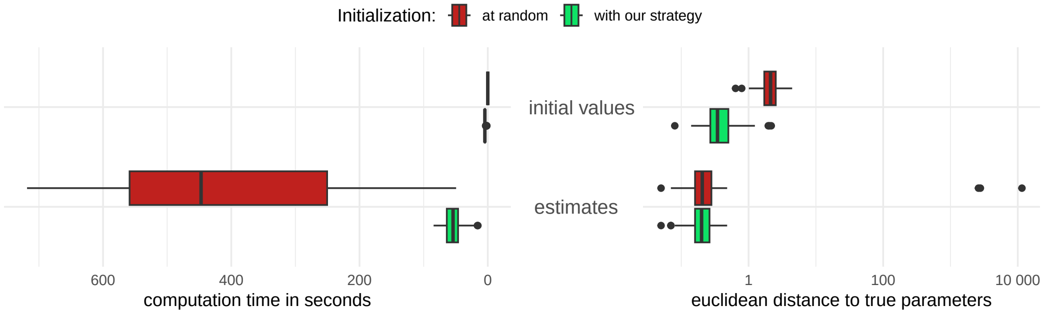
100 simulated data sets with 1000 deciders and 2 alternatives



💡 Simulation details: one alternative-varying regressor connected to alternative-varying coefficients; fully flexible Σ; true parameters and random initial values were drawn from a standard normal distribution.

		N		
		100	200	1000
J	2	□	□	□
	3	□	□	■
	4	□	□	□

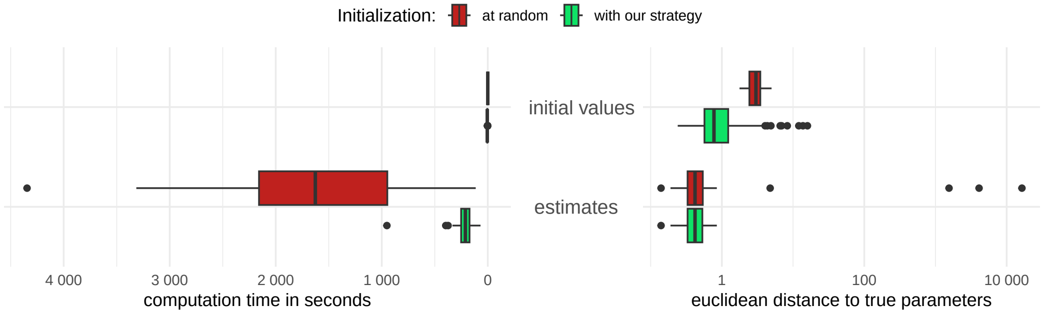
100 simulated data sets with 1000 deciders and 3 alternatives



💡 Simulation details: one alternative-varying regressor connected to alternative-varying coefficients; fully flexible Σ; true parameters and random initial values were drawn from a standard normal distribution.

		N		
		100	200	1000
J	2	☐	☐	☐
	3	☐	☐	☐
	4	☐	☐	■

100 simulated data sets with 1000 deciders and 4 alternatives



Simulation details: one alternative-varying regressor connected to alternative-varying coefficients; fully flexible Σ; true parameters and random initial values were drawn from a standard normal distribution.

Outline

Takeaways

- Probit models are widely used in discrete choice applications.
- But estimation quickly becomes computational challenging.
- Our initialization idea improves optimization time and convergence rate.

UNIVERSITÄT BIELEFELD
Faculty of Business Administration and Economics

Takeaways

- Probit models are widely used in discrete choice applications.
- But estimation quickly becomes computational challenging.
- Our initialization idea improves optimization time and convergence rate.

Thanks for your attention! Do you have any questions or comments?

lennart.oelschlaeger@uni-bielefeld.de
loelschlaeger.de/talks