Macroeconometrics

Lecture 4 Numerical optimization and integration

Tomasz Woźniak

Department of Economics
University of Melbourne

Numerical optimization

Numerical integration

Materials:

Woźniak (2021) Posterior derivations for a simple linear regression model, Lecture notes

An R file `L4 mcxs.R` for the reproduction the results

Numerical optimization

Maximizing the log-likelihood function

Numerical optimization

Motivation.
For many econometric models the MLE cannot be found analytically as the system of equations for the first order conditions cannot or is difficult to solve.

$$G\left(\hat{\theta}\right) = \mathbf{0}$$

In such cases, we rely on numerical optimization methods that potentially give an approximate solution to the problem above that is as close to the exact solution as possible or required.

Numerical optimization: the idea

Use an algorithm that requires:

starting values for the parameter vector, denoted by $\theta_{(0)}$, at which the algorithm begins the search of the solution

a dynamic rule that generates values of parameters in subsequent iterations of the algorithm, denoted by $\theta_{(k)}$, that are closer and closer to the solution

a stopping rule that stops the algorithm at a point that is close enough to the solution given the required precision

Numerical optimization: starting values

Are usually generated from:

preliminary data analysis some summary statistics can be informative about approximate values of the parameters

a simplified model that can be easily estimated using a simpler method

a grid of admissible values that is a robust way of searching for the global maximum (away from a boundary of the parameter space) in cases where the solutions provided by algorithm are sensitive to starting values

Numerical optimization: Newton-Raphson method

$$\theta_{(k)} = \theta_{(k-1)} - H^{-1}\left(\theta_{(k-1)}\right) G\left(\theta_{(k-1)}\right)$$

$\theta_{(k)}$ – value of parameters in the current step of the algorithm

$\theta_{(k-1)}$ – value of parameters in the previous step of the algorithm

$H^{-1}\left(\theta_{(k-1)}\right)$ – inverse of Hessian matrix evaluated at $\theta_{(k-1)}$

$G\left(\theta_{(k-1)}\right)$ – the gradient vector evaluated at $\theta_{(k-1)}$

Numerical optimization: Newton-Raphson method

Gradient vector and Hessian matrix are computed using:

analytical formulae – provided by the user

numerical approximation – which is the default option in existing software packages, it is also time-consuming

automatic differentiation – where the exact derivatives of the objective function are computed using designated software – no existing software packages offer this functionality

Numerical optimization: BHHH method

Based on Brendt, Hall, Hall, Hausman (1974)

$$\theta_{(k)} = \theta_{(k-1)} + J^{-1}\left(\theta_{(k-1)}\right) G\left(\theta_{(k-1)}\right)$$

Hessian matrix ordinate $-H^{-1}\left(\theta_{(k-1)}\right)$ is aproximated by the outer product of gradients $J^{-1}\left(\theta_{(k-1)}\right)$.

$$J\left(\theta_{(k-1)}\right) = \frac{1}{T}\sum_{t=1}^{T} g_t\left(\theta_{(k-1)}\right) g_t\left(\theta_{(k-1)}\right)'$$

$$g_t\left(\theta_{(k-1)}\right) = \frac{\partial l(y_t|x_t,\theta)}{\partial\theta}\Big|_{\theta=\theta_{(k-1)}}$$

Numerical optimization: stopping rules

Let ϵ be a small positive number.

$\epsilon = \texttt{1e} - \texttt{06}$ – for preliminary estimations

$\epsilon < \texttt{1e} - \texttt{08}$ – for reporting final results

Stopping rules.

Stop the algorithm when either of the criterion holds:

$\left|G\left(\theta_{(k)}\right)\right| < \epsilon$ – related directly to absolute value of gradient

$\left\|\theta_{(k)} - \theta_{(k-1)}\right\| < \epsilon$ – an increment in parameter values over iterations is negligible, where $||\cdot||$ is some norm, e.g., a maximum value of elements of vector

$k > k_{max}$ – a rule thanks to which we avoid running the computations for a long time without achieving convergence. This might happen when we set ϵ to too small a value. This rule should not be binding for the final results.

Always check the message regarding the convergence!

Numerical optimization: illustration

A simple linear regression model with known slope $\beta = 1$

$$Y = X + E$$
$$E|X \sim \mathcal{N}\left(\mathbf{0}_T, \sigma^2 I_T\right)$$
$$\downarrow$$
$$Y|X, \beta = 1 \sim \mathcal{N}\left(X, \sigma^2 I_T\right)$$

The log-likelihood function.

$$l\left(\sigma^2|Y,X\right) = -\frac{T}{2}\log(2\pi) - \frac{T}{2}\log\left(\sigma^2\right) - \frac{1}{2}\frac{1}{\sigma^2}(Y-X)'(Y-X)$$

MLE: $\hat{\sigma}^2 = \frac{1}{T}(Y-X)'(Y-X) = \frac{1}{T}E'E$

Numerical optimization: illustration

Derivatives.

$$G\left(\sigma^2\right) = -\frac{T}{2}\frac{1}{\sigma^2} + \frac{1}{2}\frac{E'E}{\left(\sigma^2\right)^2}$$

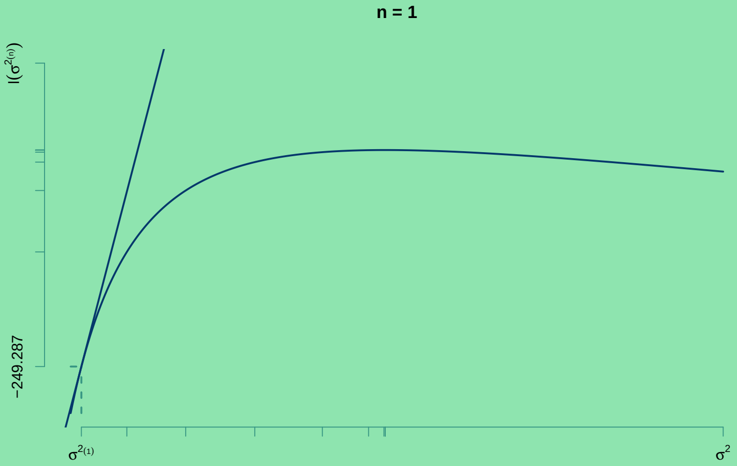
$$g_t\left(\sigma^2\right) = -\frac{T}{2}\frac{1}{\sigma^2} + \frac{1}{2}\frac{\left(y_t - x_t\right)^2}{\left(\sigma^2\right)^2}$$

$$H\left(\sigma^2\right) = \frac{T}{2}\frac{1}{\left(\sigma^2\right)^2} - \frac{E'E}{\left(\sigma^2\right)^3}$$

Newton-Raphson rule.

$$\begin{aligned}\sigma^2_{(k)} &= \sigma^2_{(k-1)} + \left[\frac{E'E}{\left(\sigma^2_{(k-1)}\right)^3} - \frac{T}{2}\frac{1}{\left(\sigma^2_{(k-1)}\right)^2}\right]^{-1}\left[\frac{1}{2}\frac{E'E}{\left(\sigma^2_{(k-1)}\right)^2} - \frac{T}{2}\frac{1}{\sigma^2_{(k-1)}}\right] \\ &= \sigma^2_{(k-1)} + \sigma^2_{(k-1)}\left[2E'E - T\sigma^2_{(k-1)}\right]^{-1}\left[E'E - T\sigma^2_{(k-1)}\right]\end{aligned}$$

Numerical optimization: illustration



Numerical optimization: illustration

Data for the animation.

k	$\sigma^2_{(k)}$	$l(\sigma^2_{(k)}\|Y,X)$	$\|G(\sigma^2_{(k)})\|$	$\|G(\sigma^2_{(k)})\| < \epsilon$	$\|\sigma^2_{(k)} - \sigma^2_{(k-1)}\|$	$\|\sigma^2_{(k)} - \sigma^2_{(k-1)}\| < \epsilon$
1	0.500	-249.287	284.1	FALSE	0.213	FALSE
2	0.713	-209.707	118.9	FALSE	0.275	FALSE
3	0.988	-188.495	47.8	FALSE	0.323	FALSE
4	1.311	-178.686	17.6	FALSE	0.316	FALSE
5	1.627	-175.253	5.6	FALSE	0.216	FALSE
6	1.842	-174.567	1.1	FALSE	0.072	FALSE
7	1.9144	-174.523	0.08	FALSE	0.006	FALSE
8	1.9205	-174.523	0.001	FALSE	0.000	FALSE
9	1.9205	-174.523	0.000	TRUE	0.000	TRUE

$\hat{\sigma}^2 = 1.9205, \qquad \epsilon = 10^{-6}$

Numerical integration Gibbs sampler

Numerical integration

Motivation.

For most of econometric models analytical derivation of the joint posterior distribution of the parameters is impossible. The distribution is known only up to its kernel:

$$p(\theta|Y) \propto L(\theta|Y)p(\theta)$$

In such cases, we use numerical integration algorithms from the family of Monte Carlo Markov Chain (MCMC) methods to generate random draws from the joint posterior distribution and we use them to compute all of the required characteristics of this distribution.

Gibbs sampler is an MCMC algorithm that is feasible for most of the models discussed in this subject.

Gibbs sampler

Idea for the algorithm.

Suppose that the parameters can be conveniently divided in two groups: $\theta = (\theta_1, \theta_2)$.

For many models, when the joint posterior distribution $p(\theta_1, \theta_2 | Y)$ cannot be derived, the full conditional posterior distributions, $p(\theta_1 | Y, \theta_2)$ and $p(\theta_2 | Y, \theta_1)$, are available.

Gibbs sampler

Construction of the algorithm.

Initialize θ_2 at $\theta_2^{(0)}$

At each iteration s:

1. **Draw** a random number/vector from $\theta_1^{(s)} \sim p\left(\theta_1 | Y, \theta_2^{(s-1)}\right)$
2. **Draw** a random number/vector from $\theta_2^{(s)} \sim p\left(\theta_2 | Y, \theta_1^{(s)}\right)$

Repeat steps 1. and 2. $S_1 + S_2$ times

Discard the first S_1 draws that allowed the algorithm to converge to the stationary posterior distribution

Output is a sample of draws from the joint posterior distribution $\left\{\theta_1^{(s)}, \theta_2^{(s)}\right\}_{s=S_1+1}^{S_2}$

A simple linear regression model

$$
\begin{aligned}
Y &= \beta X + E \\
E|X &\sim \mathcal{N}\left(\mathbf{0}_T, \sigma^2 I_T\right) \\
&\downarrow \\
Y|X &\sim \mathcal{N}\left(\beta X, \sigma^2 I_T\right)
\end{aligned}
$$

The likelihood function.

$$L\left(\theta|Y,X\right) = (2\pi)^{-\frac{T}{2}}\left(\sigma^2\right)^{-\frac{T}{2}}\exp\left\{-\frac{1}{2}\frac{1}{\sigma^2}(Y-\beta X)'(Y-\beta X)\right\}$$

Conditionally-conjugate prior distribution

A conditionally-conjugate prior distribution leads to the full conditional posterior distribution of the same form.

Let (β, σ^2) follow an independent normal and inverse gamma 2 distribution:

$$p\left(\beta, \sigma^2\right) = p\left(\beta\right) p\left(\sigma^2\right)$$

$$p\left(\beta\right) = \mathcal{N}\left(\underline{\beta}, \underline{\sigma}^2_{\beta}\right)$$

$$p\left(\sigma^2\right) = \mathcal{IG}2(\underline{s}, \underline{\nu})$$

pdf.

$$p\left(\beta, \sigma^2\right) \propto \exp\left\{-\frac{1}{2}\frac{1}{\underline{\sigma}^2_{\beta}}(\beta - \underline{\beta})'(\beta - \underline{\beta})\right\} \times \left(\sigma^2\right)^{-\frac{\underline{\nu}+2}{2}} \exp\left\{-\frac{1}{2}\frac{\underline{s}}{\sigma^2}\right\}$$

Derive Gibbs sampler

Full conditional posterior distribution of β: $p\left(\beta|Y,X,\sigma^2\right)$

Conditioning on σ^2 implies that for the sake of deriving the full conditional posterior distribution of β we treat it as non-random.

Bayes' rule.

$$p\left(\beta|Y,X,\sigma^2\right) \propto L\left(\beta,\sigma^2|Y,X\right)p\left(\beta\right)$$

kernel of the full conditional distribution.

$$p\left(\beta|Y,X,\sigma^2\right) \propto \exp\left\{-\frac{1}{2}\frac{1}{\sigma^2}(Y-\beta X)'(Y-\beta X)\right\}\times\exp\left\{-\frac{1}{2}\frac{1}{\underline{\sigma}_\beta^2}(\beta-\underline{\beta})'(\beta-\underline{\beta})\right\}$$

Derive Gibbs sampler

Full conditional posterior distribution of β: $p\left(\beta|Y,X,\sigma^2\right)$

$$
\begin{aligned}
p\left(\beta|Y,X,\sigma^2\right) &\propto \exp\left\{-\frac{1}{2}\frac{1}{\sigma^2}(Y-\beta X)'(Y-\beta X)\right\}\exp\left\{-\frac{1}{2}\frac{1}{\underline{\sigma}_\beta^2}(\beta-\underline{\beta})'(\beta-\underline{\beta})\right\} \\
&= \exp\left\{-\frac{1}{2}\left[\frac{1}{\sigma^2}(Y-\beta X)'(Y-\beta X)+\frac{1}{\underline{\sigma}_\beta^2}(\beta-\underline{\beta})'(\beta-\underline{\beta})\right]\right\} \\
&= \exp\left\{-\frac{1}{2}\left[\beta^2\left(\underline{\sigma}_\beta^{-2}+\sigma^{-2}X'X\right)-\beta 2\left(\underline{\sigma}_\beta^{-2}\underline{\beta}+\sigma^{-2}X'Y\right)+\dots\right]\right\}
\end{aligned}
$$

Which is in a form of a normal distribution.

Derive Gibbs sampler

Full conditional posterior distribution of β: $p\left(\beta | Y, X, \sigma^2\right)$

$$p\left(\beta | Y, X, \sigma^2\right) = \mathcal{N}\left(\overline{\beta}, \overline{\sigma}_\beta^2\right)$$

$$\begin{aligned} \overline{\sigma}_\beta^2 &= \left(\underline{\sigma}_\beta^{-2} + \sigma^{-2} X'X\right)^{-1} \\ \overline{\beta} &= \overline{\sigma}_\beta^2 \left(\underline{\sigma}_\beta^{-2} \underline{\beta} + \sigma^{-2} X'Y\right) \end{aligned}$$

Derive Gibbs sampler

Full conditional posterior distribution of σ^2: $p\left(\sigma^2|Y,X,\beta\right)$
Conditioning on β implies that for the sake of deriving the full conditional posterior distribution of σ^2 we treat it as non-random.

Bayes' rule.

$$p\left(\sigma^2|Y,X,\beta\right) \propto L\left(\beta,\sigma^2|Y,X\right)p\left(\sigma^2\right)$$

kernel of the full conditional distribution.

$$p\left(\sigma^2|Y,X,\beta\right) \propto \left(\sigma^2\right)^{-\frac{T}{2}} \exp\left\{-\frac{1}{2}\frac{1}{\sigma^2}(Y-\beta X)'(Y-\beta X)\right\}\times\left(\sigma^2\right)^{-\frac{\underline{\nu}+2}{2}} \exp\left\{-\frac{1}{2}\frac{\underline{s}}{\sigma^2}\right\}$$

Derive Gibbs sampler

Full conditional posterior distribution of σ^2: $p\left(\sigma^2|Y,X,\beta\right)$

$$
\begin{aligned}
p\left(\sigma^2|Y,X,\beta\right) &\propto \left(\sigma^2\right)^{-\frac{T}{2}} \exp\left\{-\frac{1}{2}\frac{1}{\sigma^2}(Y-\beta X)'(Y-\beta X)\right\} \times \left(\sigma^2\right)^{-\frac{\underline{\nu}+2}{2}} \exp\left\{-\frac{1}{2}\frac{\underline{s}}{\sigma^2}\right\} \\
&= \left(\sigma^2\right)^{-\frac{T+\underline{\nu}+2}{2}} \exp\left\{-\frac{1}{2}\frac{1}{\sigma^2}\left[(Y-\beta X)'(Y-\beta X)+\underline{s}\right]\right\} \\
&= \left(\sigma^2\right)^{-\frac{\overline{\nu}+2}{2}} \exp\left\{-\frac{1}{2}\frac{\overline{s}}{\sigma^2}\right\}
\end{aligned}
$$

The final line is the kernel of the inverse gamma 2 distribution.

Derive Gibbs sampler

Full conditional posterior distribution of σ^2: $p\left(\sigma^2|Y, X, \beta\right)$

$$p\left(\sigma^2|Y, X, \beta\right) = \mathcal{IG}2\left(\overline{s}, \overline{\nu}\right)$$

$$\overline{s} = \underline{s} + (Y - \beta X)'(Y - \beta X)$$

$$\overline{\nu} = \underline{\nu} + T$$

Gibbs sampler

Sampling random numbers from $\mathcal{IG}2\,(s,\nu)$

Step 1: Draw a random number from $\tilde{s} \sim \chi^2(\nu)$ using R function `rchisq()`

Step 2: Return $s/\tilde{s}$ as a draw from $\mathcal{IG}2\,(s,\nu)$

Sampling random numbers from $\mathcal{N}_1\left(\mu,\sigma^2\right)$

Use R function `rnorm()`

Sampling random numbers from $\mathcal{N}_N\,(\mu,\Sigma)$

Use R function `rmvnorm()` from package `mvtnorm`

Gibbs sampler

Initialize σ^2 at $\sigma^{2(0)}$

At each iteration s:

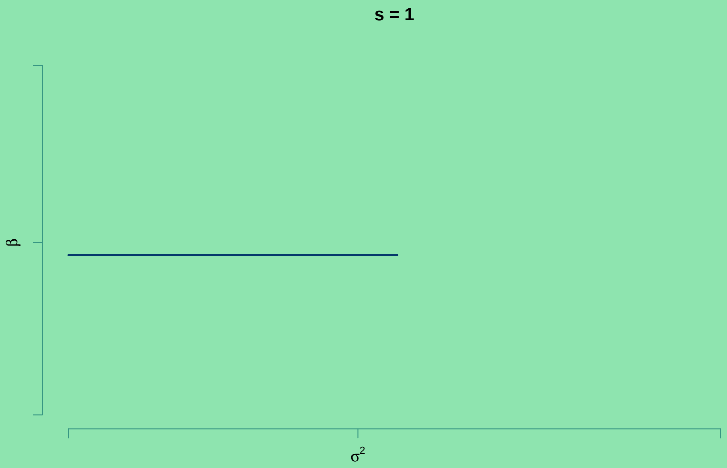
1. **Draw** $\beta^{(s)} \sim p\left(\beta|Y, X, \sigma^{2(s-1)}\right) = \mathcal{N}\left(\overline{\beta}, \overline{\sigma}_{\beta}^{2}\right)$
2. **Draw** $\sigma^{2(s)} \sim p\left(\sigma^2|Y, X, \beta^{(s)}\right) = \mathcal{IG}2\left(\overline{s}, \overline{\nu}\right)$

Repeat steps 1. and 2. $S_1 + S_2$ times

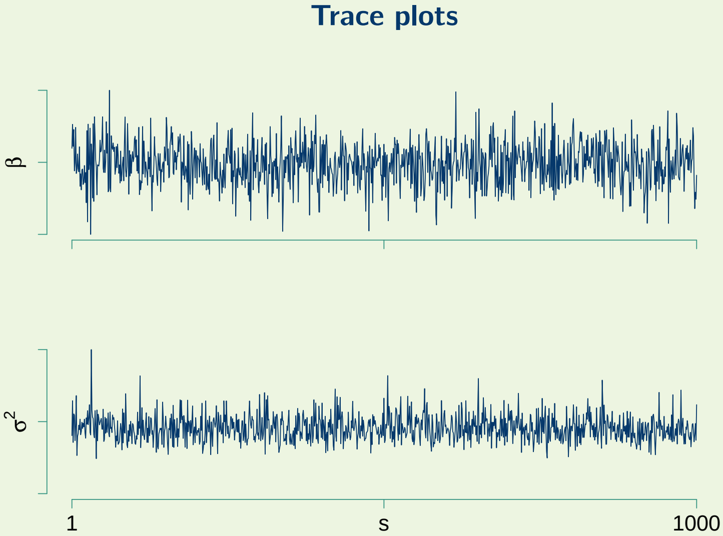
Discard the first S_1 draws that allowed the algorithm to converge to the stationary posterior distribution

Output is a sample of draws from the joint posterior distribution $\left\{\beta^{(s)}, \sigma^{2(s)}\right\}_{s=S_1+1}^{S_2}$

Gibbs sampler: illustration

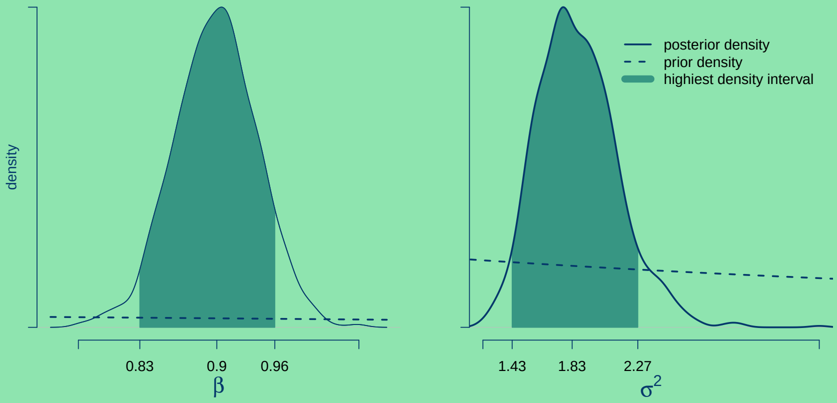


Gibbs sampler: illustration



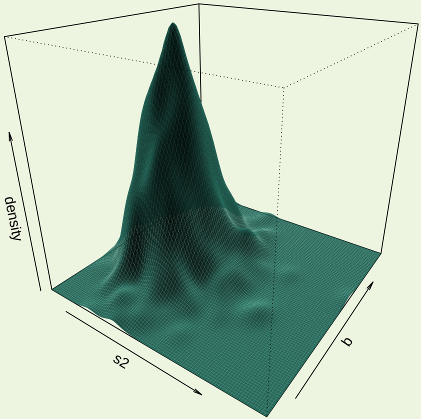
Gibbs sampler: illustration

Marginal posterior densities: $p(\beta|Y,X)$ and $p\left(\sigma^2|Y,X\right)$



Gibbs sampler: illustration

Joint posterior density: $p\left(\beta, \sigma^2 | Y, X\right)$



Numerical optimization

Newton-Raphson method

BHHH method

Numerical integration

Gibbs sampler

full conditional posterior distributions

joint and marginal posterior distributions