

UNIVERSIDAD CENTROCCIDENTAL
“LISANDRO ALVARADO”

Decanato de Ciencias y Tecnología
Licenciatura en Ciencias Matemáticas



“ALGORITMOS MCMC PARA ESTIMAR UN MODELO
MACROECONÓMICO DINÁMICO”

TRABAJO ESPECIAL DE GRADO PRESENTADO POR

BR. ANAIS FRANGELINE ACUÑA SOSA

COMO REQUISITO FINAL
PARA OBTENER EL TÍTULO DE LICENCIADA
EN CIENCIAS MATEMÁTICAS

ÁREA DE CONOCIMIENTO: PROBABILIDAD Y ESTADÍSTICA.

TUTOR: LIC. MSC. JHONNY ESCALONA

Barquisimeto, Venezuela. Julio de 2012



Universidad Centroccidental
 “Lisandro Alvarado”
 Decanato de Ciencias y Tecnología
 Licenciatura en Ciencias Matemáticas



ACTA
 TRABAJO ESPECIAL DE GRADO

Los suscritos miembros del Jurado designado por el Jefe del Departamento de Matemáticas del Decanato de Ciencias y Tecnología de la Universidad Centroccidental “Lisandro Alvarado”, para examinar y dictar el veredicto sobre el Trabajo Especial de Grado titulado:

“ALGORITMOS MCMC PARA ESTIMAR UN MODELO
 MACROECONÓMICO DINÁMICO”

Presentado por la ciudadana BR. ANAIS FRANGELINE ACUÑA SOSA titular de la Cédula de Identidad No. 19.884.298, con el propósito de cumplir con el requisito académico final para el otorgamiento del título de Licenciada en Ciencias Matemáticas.

Luego de realizada la Defensa y en los términos que imponen los Lineamientos para el Trabajo Especial de Grado de la Licenciatura en Ciencias Matemáticas, se procedió a discutirlo con el interesado habiéndose emitido el veredicto que a continuación se expresa:

¹ _____

Con una calificación de _____ puntos.

En fe de lo expuesto firmamos la presente Acta en la Ciudad de Barquisimeto a los ____ días del mes de _____ de _____.

 TUTOR

 FIRMA

 PRINCIPAL

 FIRMA

 PRINCIPAL

 FIRMA

OBSERVACIONES:

¹ Aprobado ó Reprobado

Algoritmos MCMC para estimar un Modelo Macroeconómico Dinámico

Br. Anais Frangeline Acuña Sosa.

RESUMEN:

En este trabajo estudiamos los algoritmos muestreo de importancia y Metropolis-Hastings para la estimación y evaluación de una clase de modelos macroeconómicos dinámicos, denominados modelos estocásticos de equilibrio general dinámicos (DSGE, siglas en inglés). Desarrollamos para esto el algoritmo de filtro de Kalman, el cual nos permitió encontrar estimaciones de la función de verosimilitud y se halló la matriz de información de un modelo DSGE, posteriormente fueron empleadas las diferentes técnicas de estos algoritmos para estimar los parámetros de un modelo DSGE estándar, utilizando como datos el consumo total y el producto interno bruto en Venezuela entre los años 1984-2010, cuyas observaciones están dadas por trimestres. Comparamos los resultados de estos algoritmos.

A Dios, mis padres, hermanos y a Jacobo.

AGRADECIMIENTOS

Quiero comenzar estas líneas agradeciendo a Dios por concederme la vida y llenarla de grandes bendiciones.

A mis padres, por brindarme su amor incondicional, por estar a mi lado, por apoyarme y guiarme, este logro es de ustedes.

A mis hermanos, Ana G. por ser un modelo a seguir y ser más que mi hermana, mi mejor amiga; a Marcos, por los momentos compartidos y tantas sonrisas.

A mi abuela Ana, por ser la abuelita más especial y linda del mundo, por su grandioso amor, por su preocupación y bendiciones, por tantos momentos a mi lado reconfortandome.

A Jacobo, por estar a mi lado, comprenderme, ayudarme, ofrecerme su inmenso cariño y fortaleza. Por llenar mi vida de luz.

A mis amigos, Crisor Pinto y Carlos Castillo, por su valiosa y sincera amistad, por darme momentos dulces y acompañarme en diferentes etapas de la vida.

A mis tíos y primos que de alguna forma estuvieron allí y me brindaron su mano amiga.

A mis profesores, quienes me formaron y guiaron, ofrecieron sus consejos y apoyo, en especial a los profesores Nicolás Arias, Freddy Jiménez, Victor Bernal, Miguel Vivas, Mario Rodríguez, Edner Pineda, Eibar Hernández, Ismael Huerta, Sergio Muñoz, Yenny Rangel, Abelardo Monsalve y Javier Hernández.

A mi tutor y profesor, Jhonny Escalona; por sus enseñanzas, dedicación, confianza

y guía. Por su disposición en la realización de este trabajo y por su apreciada amistad.

A Jesús Morales, Giovanni Guedez y Mariela Perdomo; por las lecciones enseñadas y colaboración durante y después de mis pasantías en el Banco Central de Venezuela.

A mis compañeros, en especial a mis amigas Mariana Álvarez y Massiel Gatica, por horas de estudio compartidas, sus ayudas y consejos.

Y en general, a todos que de alguna u otra forma me ayudaron en la realización de este trabajo.

ÍNDICE

Agradecimientos	i
Introducción	1
1. Preliminares	3
1.1. Estadística vectorial	3
1.1.1. Distribución normal multivariada	4
1.1.2. Distribución t multivariada	5
1.1.3. Versión vectorial de la ley fuerte de los grandes números	6
1.2. Función de verosimilitud	6
1.3. Modelos DSGE	7
1.3.1. Log-linealización	8
1.3.2. Método de coeficientes indeterminados	10
2. Filtro de Kalman	12
2.1. Representación espacio-estado	12
2.1.1. Formulación	13
2.1.2. Ejemplos	14
2.2. Filtro de Kalman	15
2.2.1. Proceso de estimación	16
2.2.2. Consistencia del filtro de Kalman	20
2.2.3. Algoritmo	21
2.2.4. Estimación de la función de verosimilitud	23
3. Matriz de Información	26
3.1. Introducción	26

3.2. Estimación de la matriz de información	27
3.2.1. Cálculo de la matriz H	29
3.2.2. Cálculo de la matriz \mathfrak{J}_ϕ	36
4. Algoritmos MCMC	39
4.1. Introducción	39
4.2. Algoritmo Muestreo de Importancia	41
4.3. Cadenas de Markov	43
4.4. Algoritmo Metropolis-Hastings	47
5. Resultados	50
5.1. Un modelo DSGE estándar	50
5.1.1. Solución del modelo DSGE	51
5.2. Resultados	59
5.2.1. Resultados: Muestreo de Importancia	61
5.2.2. Resultados: Metropolis-Hastings	64
5.2.3. Resultados adicionales	67
Conclusión	69
A.	70
A.1. Teorema de la función implícita	70
A.2. Producto Kronecker	70
A.3. Operador vec	71
A.4. Derivadas de las ecuaciones del filtro de Kalman	72
B. Códigos desarrollados en Matlab	75
B.1. Cálculo de la matriz de información y filtro de Kalman	75
B.1.1. gradiente.m	75
B.1.2. Filtro_Kalman.m	77
B.2. Algoritmos muestreo de importancia y Metropolis-Hastings	83
B.2.1. MetropolisHastings.m	83

B.2.2. MuestreoImportancia.m	85
Referencias	89

ÍNDICE DE TABLAS

5.1. Prior uniformes para θ	60
5.2. Prior para θ	61
5.3. Resultados muestreo de importancia con prior de la tabla 5.1.	61
5.4. Resultados muestreo de importancia con prior de la tabla 5.2.	62
5.5. Resultados muestreo de importancia con parámetros calibrados y prior de la tabla 5.2.	63
5.6. Resultados Metropolis-Hastings con prior de la tabla 5.1.	65
5.7. Resultados Metropolis-Hastings con prior de la tabla 5.2.	66
5.8. Resultados Metropolis-Hastings con parámetros calibrados y prior de la tabla 5.2.	67
5.9. Desviaciones estándar de los parámetros a través de la inversa de la matriz de información.	68

Índice de figuras

5.1. Datos: logaritmo del cociente entre el PIB y el consumo.	60
5.2. Histogramas de frecuencias del algoritmo muestreo de importancia con los prior de la tabla 5.1.	62
5.3. Histogramas de frecuencias del algoritmo muestreo de importancia con los prior de la tabla 5.2.	63
5.4. Histogramas de frecuencias del algoritmo muestreo de importancia con parámetros calibrados y los prior de la tabla 5.2.	64
5.5. Histogramas de frecuencias del algoritmo Metropolis-Hastings con los prior de la tabla 5.1.	65
5.6. Histogramas de frecuencias del algoritmo Metropolis-Hastings con los prior de la tabla 5.2.	66
5.7. Histogramas de frecuencias del algoritmo Metropolis-Hastings con parámetros calibrados y los prior de la tabla 5.2.	67

Introducción

Los modelos DSGE (Dinamic Stochastic General Equilibrium) se han convertido en forma creciente en los últimos años, en una herramienta utilizada por los bancos centrales y otras instituciones financieras para sistematizar, analizar y proyectar complejos procesos económicos susceptibles del ser analizados con métodos cuantitativos modernos, incluyendo el análisis de las políticas macroeconómicas.

Estos modelos parten de la tradición del equilibrio general, comenzado con el trabajo pionero de León Walras: "Elementos de economía política pura, o teoría de la riqueza social"(1874) y modernizado con métodos matemáticos y conceptos económicos desarrollados muy posteriormente. En particular, se suele especificar el comportamiento de agentes económicos: familias, empresas, gobierno, etc; así como los mercados en que interactúan, precisando la estructura de cada uno de ellos.

Un modelos DSGE puede ser representado de la siguiente forma:

$$\mathbb{E}_t[f(x_{t-1}, x_t, x_{t+1}, U_t, \theta)] = 0, \quad (1)$$

donde \mathbb{E}_t es la esperanza condicionada a la información disponible a tiempo t , x_t es un vector m -dimensional de variables endógenas, U_t es un vector n -dimensional gaussiano de variables aleatorias independientes e idénticamente distribuidas (i.i.d) con $\mathbb{E}(U_t) = 0$ y $\mathbb{E}(U_t U_t') = I_n$ (I_n , identidad de orden $n \times n$); y θ es el vector k -dimensional de parámetros del modelo.

Se han desarrollado técnicas para aproximar la solución del modelo (1) (ver [12]) y técnicas con el objeto de estimar la distribución a posteriori de θ , $\pi(\theta|Z)$, donde Z es la serie de observaciones macroeconómicas (ver [2]). Estas técnicas de estimación parten del supuesto que los parámetros del modelo son variables aleatorias en lu-

gar de constantes desconocidas, y hacen uso de poderosos algoritmos de simulación bayesiana, como el algoritmo muestreo de importancia y el algoritmo Metropolis-Hastings, que permiten estimar con gran precisión la densidad a posteriori de los parámetros del modelo.

En este trabajo se propuso resolver un modelo DSGE estándar, se linealizó y se construyó una representación de la forma:

$$x_t = Ax_{t-1} + BU_t, \quad \text{donde } U_t \sim N(0, Q) \quad (2)$$

$$z_t = Cx_t + V_t, \quad \text{donde } V_t \sim N(0, R). \quad (3)$$

Luego, a partir de las ecuaciones (2) y (3), a través del filtro de Kalman fue posible la obtención de estimaciones de la función de log-verosimilitud y dados unos prior para los elementos del vector de parámetros profundos del modelo, se construyó la densidad a posteriori con la utilización de los algoritmos muestreo de importancia y Metropolis-Hastings, lo que en la práctica requiere la evaluación de la segunda derivada (el Hessiano) de la función log-verosimilitud, por lo cual es calculada con anterioridad la matriz de información del modelo.

El trabajo se encuentra estructurado de la siguiente manera: en el capítulo 1, son presentadas definiciones y resultados útiles en su desarrollo; en el capítulo 2, se muestra la teoría de filtro de Kalman, proceso de estimación, consistencia y algoritmo, además se explica como realizar la estimación de la función de verosimilitud a través de él; en el capítulo 3, se desarrolla el procedimiento descrito en [6] para encontrar una expresión analítica de la matriz de información de un modelo DSGE; en el capítulo 4, son descritas las técnicas empleadas por los algoritmos muestreo de importancia y Metropolis-Hastings; en el capítulo 5, son mostrados los resultados de la implementación de los algoritmos expuestos para un modelo DSGE estándar.

CAPÍTULO 1

PRELIMINARES

En este capítulo enunciaremos algunas definiciones y resultados que serán de utilidad en el desarrollo de este trabajo.

1.1. Estadística vectorial

Definición 1.1.1. Se dice que $X = (X_1, X_2, \dots, X_p)$ es un p -vector aleatorio, si cada X_i con $i \in \{1, 2, \dots, p\}$ es una variable aleatoria unidimensional o escalar.

Definición 1.1.2. Sea X un p -vector aleatorio, $X = (X_1, X_2, \dots, X_p)$, se define el vector de esperanza $\mathbb{E}(X)$ como:

$$\mathbb{E}(X) := (\mathbb{E}(X_1), \mathbb{E}(X_2), \dots, \mathbb{E}(X_p)).$$

Propiedad 1.1.1. Sean X y Y p -vectores aleatorios cualesquiera. Si $\mathbb{E}(X)$ y $\mathbb{E}(Y)$ existen, entonces:

- Para todo $\alpha \in \mathbb{R}$, $\mathbb{E}(\alpha X)$ existe y además:

$$\mathbb{E}(\alpha X) = \alpha \mathbb{E}(X).$$

- $\mathbb{E}(X + Y)$ existe, y se cumple:

$$\mathbb{E}(X + Y) = \mathbb{E}(X) + \mathbb{E}(Y).$$

- Para todo $\mathbf{a} \in \mathbb{R}^p$, $\mathbb{E}(X + \mathbf{a})$ existe y se tiene que:

$$\mathbb{E}(X + \mathbf{a}) = \mathbb{E}(X) + \mathbf{a}.$$

Definición 1.1.3. Sea X un p -vector aleatorio, $X = (X_1, X_2, \dots, X_p)$, se define la matriz de covarianza de X , como:

$$\text{cov}(X) = \begin{pmatrix} \text{cov}(X_1, X_1) & \cdots & \text{cov}(X_1, X_p) \\ \text{cov}(X_2, X_1) & \cdots & \text{cov}(X_2, X_p) \\ \vdots & & \vdots \\ \text{cov}(X_p, X_1) & \cdots & \text{cov}(X_p, X_p) \end{pmatrix},$$

donde $\text{cov}(X_i, X_j) = \text{cov}(X_j, X_i)$, para todo $i, j \in \{1, 2, \dots, p\}$.

Observación 1.1.1. Dado que $\text{cov}(X_i, X_j) = \text{cov}(X_j, X_i)$, para todo $i, j \in \{1, 2, \dots, p\}$, entonces la matriz de covarianza para cualquier p -vector aleatorio X es simétrica.

Observación 1.1.2. Por definición, $\text{cov}(X_i, X_j) = \mathbb{E}[(X_i - \mathbb{E}(X_i))(X_j - \mathbb{E}(X_j))]$. En la práctica cuando $\sigma_{X_i}^2 < \infty$ y $\sigma_{X_j}^2 < \infty$, es más usada la propiedad:

$$\text{cov}(X_i, X_j) = \mathbb{E}(X_i X_j) - \mathbb{E}(X_i) \mathbb{E}(X_j).$$

1.1.1. Distribución normal multivariada

La distribución normal posee una singular importancia dentro de la estadística, una de las principales razones esta relacionada con sus propiedades matemáticas. Su definición puede ser extendida al caso multivariado o vectorial, como veremos a continuación.

Definición 1.1.4. Se dice que un p -vector aleatorio $X = (X_1, X_2, \dots, X_p)$ posee distribución normal (multivariada), si las variables escalares X_1, X_2, \dots, X_p constituyen un sistema estadísticamente independiente y cada una de ellas posee distribución normal (escalar o univariada).

Definición 1.1.5. La **función de densidad** f de un p -vector aleatorio X , el cual posee distribución normal (multivariada) con media $\mu = (\mu_1, \mu_2, \dots, \mu_p)$ y matriz de covarianza Σ , donde $\Sigma \in M_{p \times p}$ es una matriz definida positiva. Viene dada por:

$$f(X) = \frac{1}{(2\pi)^{\frac{p}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} (X - \mu)' \Sigma^{-1} (X - \mu)\right).$$

Nota 1.1.1. Denotaremos por $X \sim N(\mu, \Sigma)$, cuando el p -vector aleatorio X posee distribución normal, media μ y matriz de covarianza Σ .

Propiedad 1.1.2. Si $X \sim N(\mu, \Sigma)$ y \mathbf{a} es un p -vector escalar, entonces $X + \mathbf{a}$ posee distribución normal, con media $\mu + \mathbf{a}$ y matriz de covarianza Σ , esto es, $X + \mathbf{a} \sim N(\mu + \mathbf{a}, \Sigma)$.

1.1.2. Distribución t multivariada

La distribución t también conocida como distribución de t Student en honor a W.S. Gosset, quién publicó sus estudios de esta distribución en 1908 con el seudónimo “Student”, esta estrechamente relacionada con la distribución normal y ha sido ampliamente empleada en problemas importantes de inferencia estadística, en los cuales podemos mencionar el análisis Bayesiano multivariado. La distribución t multivariada se define de la siguiente manera:

Definición 1.1.6. Consideremos Z un p -vector aleatorio donde $Z \sim N(0, \Sigma)$ y u una variable aleatoria proveniente de la distribución ji-cuadrado con v grados de libertad, lo cual se denota como $u \sim \chi_v^2$. Si Z y u son estadísticamente independientes, el p -vector aleatorio X definido por la siguiente ecuación:

$$X = \frac{Z}{\sqrt{\frac{u}{v}}},$$

posee distribución t con media $\mu = 0$, matriz de covarianza Σ y v grados de libertad, lo cual denotamos de la siguiente manera: $X \sim T_v(0, \Sigma)$.

Definición 1.1.7. La función de densidad f de un p -vector aleatorio $X \sim T_v(\mu, \Sigma)$, esta dada por:

$$f(X) = \frac{\Gamma[(v+p)/2]}{\Gamma(v/2) v^{p/2} \pi^{p/2} |\Sigma|^{1/2} \left[1 + \frac{1}{v} (X - \mu)' \Sigma^{-1} (X - \mu)\right]^{(v+p)/2}},$$

donde $\Gamma(\cdot)$ denota la función gamma.

Propiedad 1.1.3. Sean X un p -vector aleatorio tal que $X \sim T_v(\mu, \Sigma)$ entonces, para cualesquiera $C \in M_{p \times p}$ una matriz escalar no singular y \mathbf{a} un p -vector escalar, $CX + \mathbf{a} \sim T_v(C\mu + \mathbf{a}, C\Sigma C')$.

1.1.3. Versión vectorial de la ley fuerte de los grandes números

La ley fuerte de los grandes números establece que el promedio de una muestra de gran tamaño, tomada al azar de una población tenderá a estar cerca de la media de la población completa. Este resultado fue enunciado por primera vez sin prueba por el matemático italiano Gerolamo Cardano en 1565. Dada su utilidad, enunciaremos la versión vectorial de este teorema.

Sean X un p -vector aleatorio, X_1, X_2, \dots una secuencia de p -vectores aleatorios y $(\Omega, \mathfrak{U}, \mathbb{P})$ el espacio de probabilidad asociado a estos vectores.

Definición 1.1.8. Se dice que la secuencia X_1, X_2, \dots converge fuertemente (o casi segura) a X , si existe un conjunto $A \in \mathfrak{U}$ tal que $\mathbb{P}(A) = 1$ y se satisfaga que:

$$\lim_{n \rightarrow \infty} X_n(w) = X(w), \quad \forall w \in A.$$

Teorema 1.1.1. Sea X_1, X_2, \dots una muestra infinita de una población con vector esperanza μ . Entonces,

$$\frac{X_1 + X_2 + \dots + X_n}{n} \rightarrow \mu, \quad \text{fuertemente.}$$

1.2. Función de verosimilitud

La función de verosimilitud es una función de los parámetros de un modelo estadístico que permite realizar inferencias acerca de su valor a partir de un conjunto de observaciones. Esta función abre las vías a una de las técnicas más habituales en inferencia estadística, la de máxima verosimilitud, la cual fue introducida por R. A. Fisher, genetista y experto en estadística en la década de 1920; es de gran importancia dado que la evaluación por este método procura encontrar los valores más probables de los parámetros de la distribución para un conjunto de datos. Esta función se define como sigue:

Definición 1.2.1. Sea X_1, \dots, X_n una muestra aleatoria de una población con función de densidad p_λ . La función de verosimilitud asociada a la muestra aleatoria es la función de densidad del vector (X_1, \dots, X_n) . Es decir, la función de verosimilitud es la función $\mathcal{L} : \mathbb{R}^n \rightarrow \mathbb{R}$ definida por:

$$\mathcal{L}(\lambda|x_1, x_2, \dots, x_n) = \prod_{t=1}^n p_\lambda(x_t),$$

donde x_1, x_2, \dots, x_n son resultados de la muestra aleatoria.

1.3. Modelos DSGE

Los modelos estocásticos de equilibrio general (DSGE, por sus siglas en inglés) son una herramienta ampliamente utilizada en el modelaje macroeconómico moderno por las instituciones diseñadoras y ejecutoras de políticas macroeconómicas, como por ejemplo, los bancos centrales; ya que permiten analizar y cuantificar los efectos que pueden tener distintas variables que intervienen en ella, y caracterizan su evolución a través del tiempo.

Un modelo DSGE puede ser representado de la siguiente forma:

$$\mathbb{E}_t[f(x_{t-1}, x_t, x_{t+1}, U_t, \theta)] = 0,$$

donde \mathbb{E}_t es la esperanza condicionada a la información disponible a tiempo t , x_t es un vector m -dimensional de variables endógenas, U_t es un vector n -dimensional gaussiano de variables aleatorias independientes e idénticamente distribuidas (i.i.d) con $\mathbb{E}(U_t) = 0$ y $\mathbb{E}(U_t U_t') = I_n$ (I_n , identidad de orden $n \times n$); y θ es el vector k -dimensional de parámetros del modelo.

En general, la función f es no lineal más sin embargo existen métodos que permiten linealizar el modelo para su posterior solución.

1.3.1. Log-linealización

Un método para resolver y analizar modelos estocásticos dinámicos no lineales, es el de aproximar las ecuaciones no lineales que caracterizan el equilibrio del modelo con las log-linealizaciones. Esta estrategia consiste en utilizar una aproximación de Taylor de primer orden en torno al logaritmo del estado estacionario (donde las variables no dependen del tiempo), para reemplazar las ecuaciones con las aproximaciones, las cuales son lineales en las log-desviaciones de las variables.

La metodología es la siguiente:

Sea X_t una variable estrictamente positiva y X su estado estacionario, definimos la desviación logarítmica como:

$$\widehat{X}_t = \log X_t - \log X.$$

Supongamos que tenemos una ecuación de la forma:

$$f(X_t, Y_t) = g(Z_t), \quad (1.1)$$

donde X_t , Y_t y Z_t son variables estrictamente positivas.

Esta ecuación claramente es válida en el estado estacionario:

$$f(X, Y) = g(Z). \quad (1.2)$$

Para encontrar la versión log-linealizada de (1.1), se reescriben las variables usando la identidad $X_t = e^{\log(X_t)}$ y tomando logaritmo en ambos lados:

$$\log(f(e^{\log(X_t)}, e^{\log(Y_t)})) = \log(f(e^{\log(Z_t)})). \quad (1.3)$$

Ahora, tomando una aproximación de Taylor de primer orden alrededor del estado estacionario $(\log(X), \log(Y), \log(Z))$. Luego de algunos cálculos, podemos escribir el lado izquierdo de la ecuación (1.2) como:

$$\log(f(X, Y)) + \frac{1}{f(X, Y)} [f_1(X, Y) X (\log(X_t) - \log(X)) + f_2(X, Y) Y (\log(Y_t) - \log(Y))], \quad (1.4)$$

donde f_i con $i = 1, 2$ denota la derivada parcial de f con respecto a la componente i .

De forma similar, podemos escribir el lado derecho de (1.2) como:

$$\log(g(Z)) + \frac{1}{g(Z)} [g'(Z) Z (\log(Z_t - \log(Z)))] . \quad (1.5)$$

Igualando las ecuaciones (1.4) y (1.5), y usando las ecuaciones (1.1) y (1.3), se obtiene la siguiente ecuación log-linealizada:

$$\left[f_1(X, Y) X \widehat{X}_t + f_2(X, Y) Y \widehat{Y}_t \right] \simeq \left[g'(Z) Z \widehat{Z}_t \right] .$$

Esta metodología puede ser generalizada para una ecuación de la forma:

$$f(X_t^1, X_t^2, \dots, X_t^n) = g(Y_t^1, Y_t^2, \dots, Y_t^m) ,$$

donde f y g son funciones no lineales, y $X_t^1, X_t^2, \dots, X_t^n, Y_t^1, Y_t^2, \dots, Y_t^m$ son variables estrictamente positivas. Su log-linealización viene dada por:

$$\sum_{i=1}^n f_i(X^1, X^2, \dots, X^n) X^i \widehat{X}_t^i \simeq \sum_{j=1}^m g_j(Y^1, Y^2, \dots, Y^m) Y^j \widehat{Y}_t^j . \quad (1.6)$$

Ejemplo 1. Consideremos una restricción típica en economía, la cual es la condición de equilibrio para el mercado de bienes:

$$C_t + I_t = Y_t, \quad (1.7)$$

donde C_t es el consumo, I_t la inversión y Y_t la producción

Si escribimos:

$$f(C_t, I_t) = C_t + I_t,$$

y

$$g(Y_t) = Y_t.$$

Siguiendo la ecuación (1.6), obtenemos:

$$f_1(C, I) C \widehat{C}_t + f_2(C, I) I \widehat{I}_t = g'(Y) Y \widehat{Y}_t.$$

Así, la ecuación log-linealizada de (1.7) es:

$$C \widehat{C}_t + I \widehat{I}_t = Y \widehat{Y}_t.$$

1.3.2. Método de coeficientes indeterminados

Este método fue propuesto por H. Uhlig¹ en 1995 (ver [12]), provee una herramienta para resolver y analizar modelos estocásticos dinámicos, no lineales de tiempo discreto; se basa en resolver la ley recursiva de movimiento de equilibrio con el método de coeficientes indeterminados a partir de log-linealizar las ecuaciones necesarias que caracterizan el equilibrio del modelo.

El primer paso es obtener las ecuaciones de primer orden del modelo log-linealizadas y escribirlas como en el siguiente sistema:

$$\left. \begin{aligned} 0 &= AAx_t^* + BBx_{t-1}^* + CCy_t^* + DDz_t^* \\ 0 &= \mathbb{E}_t[FFx_{t+1}^* + GGx_t^* + HHx_{t-1}^* + JJy_{t+1}^* + KKy_t^* + LLz_{t+1}^* + MMz_t^*] \\ z_{t+1}^* &= NNz_t^* + \varepsilon_{t+1}, \quad E_t[\varepsilon_{t+1}] \end{aligned} \right\}, \quad (1.8)$$

donde x_t^* es el vector de estado de dimensión $m \times 1$, y_t^* el vector de variables endógenas de tamaño $n \times 1$ y z_t^* el vector de procesos estocásticos exógenos de tamaño $k \times 1$.

Uhlig demostró que este sistema se puede reescribir como una ley de movimiento

¹ Su artículo, además de los programas desarrollados en MATLAB se encuentran disponibles en: <http://www2.wiwi.hu-berlin.de/institute/wpol/html/toolkit.htm>

de equilibrio recursivo, como sigue:

$$\begin{aligned}x_t^* &= PPx_{t-1}^* + QQz_t^* \\y_t^* &= RRx_{t-1}^* + SSz_t^*\end{aligned}$$

De esta forma, el procedimiento general para resolver y analizar estos modelos a través de este método, es el siguiente:

1. Hallar las ecuaciones necesarias que caracterizan el equilibrio, esto es, las limitaciones y condiciones de primer orden.
2. Establecer los parámetros y encontrar el estado estacionario.
3. Log-linealizar las ecuaciones necesarias que caracterizan el equilibrio del sistema para hacer las ecuaciones aproximadamente lineales en log-desviaciones alrededor del estado estacionario.
4. Resolver la ley de movimiento de equilibrio vía el método de coeficientes indeterminados.

CAPÍTULO 2

FILTRO DE KALMAN

2.1. Representación espacio-estado

Los modelos espacio-estados surgieron en el campo de la ingeniería, en el contexto de los sistemas dinámicos, sin embargo es a comienzo de los años sesenta con el artículo de Kalman donde se produce un importante salto cualitativo.

La representación espacio-estado es esencialmente una notación conveniente para la estimación de modelos estocásticos donde se suponen errores en la medición del sistema, cuya naturaleza markoviana permite que los cálculos necesarios para su aplicación práctica se efectúen de forma recursiva, lo que permite abordar el manejo de un amplio rango de modelos de series de tiempo. Entre sus usos particulares se encuentra la modelación de componentes no observables y parámetros que cambien en el tiempo.

El modelo espacio-estado dentro del cual se enmarca la formulación del filtro de Kalman, supone la existencia de una o más variables (variables de estado), con las cuales podemos describir completamente el estado en que se encuentra algún sistema en un momento dado. El conjunto de variables de estado se conoce como el vector de estado y lo denotamos x_t .

En el modelo, x_t no es observable directamente, en su lugar observamos z_t (vector de observaciones), el cual es una combinación lineal de las variables de estado x_t más un error de medición.

2.1.1. Formulación

Un modelo de espacio-estado consiste en dos ecuaciones:

Ecuación de estado:

Determina el vector m -dimensional de estado x_t , en términos del estado anterior x_{t-1} , más un error. Es decir,

$$x_t = Ax_{t-1} + B U_t, \quad (2.1)$$

donde $U_t \sim N(0, Q)$ de dimensión $n \times 1$, $A \in M_{m \times m}$ y $B \in M_{m \times n}$.

Ecuación de observación:

Expresa la observación z_t de dimensión $p \times 1$, como una función lineal de una variable de estado x_t de dimensión $m \times 1$, más un error. Esto es:

$$z_t = Cx_t + V_t, \quad (2.2)$$

donde $V_t \sim N(0, R)$ de dimensión $p \times 1$ y $C \in M_{p \times m}$.

Observación 2.1.1. En formulaciones más generales, las matrices A , B y C dependen del tiempo t . Para nuestros propósitos, las consideraremos independientes del tiempo.

Observación 2.1.2. Bajo las suposiciones en las cuales enmarcamos la representación espacio-estado, resulta claro que: $\mathbb{E}(U_t x_r') = 0$, $\mathbb{E}(U_t z_r') = 0$, $\mathbb{E}(V_t x_r') = 0$ y $\mathbb{E}(V_t z_r') = 0$, para todo $1 \leq r \leq t$. Además:

$$\mathbb{E}(V_r V_t') = \begin{cases} R, & \text{si } r = t \\ 0, & \text{en otro caso} \end{cases}$$

y

$$\mathbb{E}(U_r U_t') = \begin{cases} Q, & \text{si } r = t \\ 0, & \text{en otro caso} \end{cases}$$

2.1.2. Ejemplos

Ejemplo 2. Si tenemos el siguiente proceso AR(2), descrito por:

$$z_t = \rho_1 z_{t-1} + \rho_2 z_{t-2} + V_t, \quad V_t \sim N(0, \sigma_V^2).$$

Podemos describirlo mediante una representación espacio-estado, con:

Ecuación de estado:

$$x_t = \begin{bmatrix} \rho_1 & 1 \\ \rho_2 & 0 \end{bmatrix} x_{t-1} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} V_t,$$

donde $x_t = \begin{bmatrix} y_t & \rho_2 y_{t-1} \end{bmatrix}'$.

Ecuación de observación:

$$z_t = \begin{bmatrix} 1 & 0 \end{bmatrix} x_t.$$

Ejemplo 3. Para el siguiente proceso MA(1):

$$z_t = V_t + \theta V_{t-1}, \quad V_t \sim N(0, \sigma_V^2), \quad y \quad \mathbb{E}(V_t V_s') = 0, \quad \forall s \neq t.$$

Se tiene una representación espacio-estado, dada por:

Ecuación de estado:

$$x_t = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_{t-1} + \begin{bmatrix} 1 \\ \theta \end{bmatrix} V_t,$$

donde $x_t = \begin{bmatrix} y_t & \theta V_t \end{bmatrix}'$.

y **Ecuación de observación:**

$$z_t = \begin{bmatrix} 1 & 0 \end{bmatrix} x_t.$$

Otra posible representación espacio-estado que permite describir el mismo proceso (quizás mas sencilla), es la siguiente:

Ecuación de estado:

$$x_t = V_{t-1},$$

donde $x_t = [V_{t-1}]'$.

y **Ecuación de observación:**

$$z_t = \theta x_t + V_t.$$

2.2. Filtro de Kalman

El filtro de Kalman, es un algoritmo desarrollado por Rudolf E. Kalman en 1960 (ver [7]) para estimar sistemas dinámicos representados en la forma de espacio-estado, proporciona el medio de estimar el estado inobservable a partir de alguna magnitud observable relacionada con éste, de forma que la estimación se actualice cada vez que se dispone de nueva información.

Dentro de la notación espacio-estado, la derivación del filtro de Kalman se basa en el supuesto de normalidad del vector de estado inicial y de las perturbaciones del sistema. De tal forma que es posible calcular la función de verosimilitud sobre el error de predicción con lo cual se lleva a cabo la estimación de los parámetros no

conocidos de este sistema.

El procedimiento de estimación completo es el siguiente: luego de que el modelo es formulado en su representación espacio-estado y para un conjunto inicial de parámetros dados, los errores de predicción del modelo son generados por el filtro y estos son utilizados para evaluar recursivamente la función de verosimilitud.

2.2.1. Proceso de estimación

Supongamos que se tiene un modelo de espacio-estado, mediante las ecuaciones (2.1) y (2.2). Denotamos por z^r al vector que contiene las variables observables hasta el tiempo r .

Definamos al mejor predictor lineal dada la información de las variables observables hasta el tiempo r , de:

- x_t , por:

$$x_{t|r} = \mathbb{E}(x_t | z^r), \quad (2.3)$$

- z_t , por:

$$z_{t|r} = \mathbb{E}(z_t | z^r), \quad (2.4)$$

Ahora, definamos al error de predicción de la matriz de covarianza dada la información de las variables observables hasta el tiempo r , de:

- x_t , por:

$$\Sigma_{t|r} = \mathbb{E}\left((x_t - x_{t|r})(x_t - x_{t|r})' | z^r\right), \quad (2.5)$$

- z_t , por:

$$\Omega_{t|r} = \mathbb{E}\left((z_t - z_{t|r})(z_t - z_{t|r})' | z^r\right). \quad (2.6)$$

De acuerdo a las definiciones (2.3), (2.4) y usando propiedades de la esperanza condicional:

- $z_{t|t-1} = \mathbb{E}(z_t|z^{t-1}) = \mathbb{E}(Cx_t + V_t|z^{t-1}) = C\mathbb{E}(x_t|z^{t-1}) + \mathbb{E}(V_t|z^{t-1}) = Cx_{t|t-1}$;
pues V_t es independiente de z^t y $\mathbb{E}(V_t) = 0$.
- $x_{t+1|t} = \mathbb{E}(x_{t+1}|z^t) = \mathbb{E}(Ax_t + BU_{t+1}|z^t) = A\mathbb{E}(x_t|z^t) + B\mathbb{E}(U_{t+1}|z^t) = Ax_{t|t}$;
dado que U_{t+1} es independiente de z^t y $\mathbb{E}(U_t) = 0$.

Análogamente, $z_{t+1|t} = Cx_{t+1|t}$.

Consideremos la siguiente ecuación, la cual nos permite obtener $x_{t|t}$ de z_t y $x_{t|t-1}$:

$$x_{t|t} = x_{t|t-1} + K_t (z_t - z_{t|t-1}) = x_{t|t-1} + K_t (z_t - Cx_{t|t-1}), \quad (2.7)$$

donde K_t es llamada **ganancia de Kalman** y mide como actualizamos $x_{t|t-1}$ como una función de nuestro error de predicción para z_t .

El filtro de Kalman trata de como construir K_t de manera que actualicemos de forma óptima $x_{t|t}$ de $x_{t|t-1}$ y z_t . La llamada ganancia de Kalman K_t debe ser tal que minimice el error de predicción de la matriz de covarianza de x_t , dada la información de las variables observables hasta el tiempo t , $\Sigma_{t|t}$. Así:

$$K_t = \Sigma_{t|t-1} C' (C \Sigma_{t|t-1} C' + R)^{-1}. \quad (2.8)$$

Luego, dados $\Sigma_{t|t-1}$, z_t y $x_{t|t-1}$, de (2.6) y (2.7):

$$\begin{aligned} \Omega_{t|t-1} &= \mathbb{E} \left\{ (z_t - z_{t|t-1}) (z_t - z_{t|t-1})' | z^{t-1} \right\} \\ &= \mathbb{E} \left\{ (Cx_t + V_t - Cx_{t|t-1}) (Cx_t + V_t - Cx_{t|t-1})' | z^{t-1} \right\} \end{aligned}$$

$$\begin{aligned}
&= \mathbb{E} \left\{ (C(x_t - x_{t|t-1}) + V_t) (C(x_t - x_{t|t-1}) + V_t)' | z^{t-1} \right\} \\
&= \mathbb{E} \left\{ (C(x_t - x_{t|t-1}) + V_t) \left((x_t - x_{t|t-1})' C' + V_t' \right) | z^{t-1} \right\} \\
&= \mathbb{E} \left\{ \begin{array}{c} C(x_t - x_{t|t-1}) (x_t - x_{t|t-1})' C' + \\ C(x_t - x_{t|t-1}) V_t' + V_t (x_t - x_{t|t-1})' C' + \\ V_t V_t' | z^{t-1} \end{array} \right\} \\
&= \left[\begin{array}{c} C \mathbb{E} \left\{ (x_t - x_{t|t-1}) (x_t - x_{t|t-1})' | z^{t-1} \right\} C' + C \mathbb{E} \left\{ (x_t - x_{t|t-1}) V_t' | z^{t-1} \right\} + \\ \mathbb{E} \left\{ V_t (x_t - x_{t|t-1}) | z^{t-1} \right\} C' + \mathbb{E} \left\{ V_t V_t' | z^{t-1} \right\} \end{array} \right] \\
&= C \mathbb{E} \left\{ (x_t - x_{t|t-1}) (x_t - x_{t|t-1})' | z^{t-1} \right\} C' + \mathbb{E} \left\{ V_t V_t' \right\} \quad (2.9) \\
&= C \Sigma_{t|t-1} C' + R. \quad (2.10)
\end{aligned}$$

De aquí tenemos que la ganancia de Kalman que minimiza el error de predicción de la matriz de covarianza de x_t , depende del error de predicción de z_t , esto es,

$$K_t = \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1}. \quad (2.11)$$

A partir de estos resultados es posible calcular $\Sigma_{t|t}$:

$$\begin{aligned}
\Sigma_{t|t} &= \mathbb{E} \left\{ (x_t - x_{t|t}) (x_t - x_{t|t})' | z^t \right\} \\
&= \mathbb{E} \left\{ (x_t - (x_{t|t-1} + K_t (z_t - Cx_{t|t-1}))) (x_t - (x_{t|t-1} + K_t (z_t - Cx_{t|t-1})))' | z^t \right\} \\
&= \mathbb{E} \left\{ ((x_t - x_{t|t-1}) - K_t (z_t - Cx_{t|t-1})) \left((x_t - x_{t|t-1})' - (z_t - Cx_{t|t-1})' K_t' \right) | z^t \right\} \\
&= \mathbb{E} \left\{ \begin{array}{c} (x_t - x_{t|t-1}) (x_t - x_{t|t-1})' - (x_t - x_{t|t-1}) (z_t - Cx_{t|t-1})' K_t' - \\ K_t (z_t - Cx_{t|t-1}) (x_t - x_{t|t-1})' + K_t (z_t - Cx_{t|t-1}) (z_t - Cx_{t|t-1})' K_t' | z^t \end{array} \right\} \\
&= \left[\begin{array}{c} \mathbb{E} \left\{ (x_t - x_{t|t-1}) (x_t - x_{t|t-1})' | z^{t-1} \right\} - \\ \mathbb{E} \left\{ (x_t - x_{t|t-1}) (z_t - Cx_{t|t-1})' | z^{t-1} \right\} K_t' - \\ K_t \mathbb{E} \left\{ (z_t - Cx_{t|t-1}) (x_t - x_{t|t-1})' | z^{t-1} \right\} + \\ K_t \mathbb{E} \left\{ (z_t - Cx_{t|t-1}) (z_t - Cx_{t|t-1})' | z^{t-1} \right\} K_t' \end{array} \right], \quad (2.12)
\end{aligned}$$

Reescribiendo los términos del lado derecho en la igualdad anterior, considerando las propiedades de la esperanza condicional y la independencia de las variables aleatorias, tenemos que:

$$\mathbb{E} \left\{ (x_t - x_{t|t-1}) (x_t - x_{t|t-1})' | z^{t-1} \right\} = \Sigma_{t|t-1},$$

$$\begin{aligned} \mathbb{E} \left\{ (x_t - x_{t|t-1}) (z_t - Cx_{t|t-1})' | z^{t-1} \right\} K_t' &= \mathbb{E} \left\{ (x_t - x_{t|t-1}) (Cx_t + V_t - Cx_{t|t-1})' | z^{t-1} \right\} K_t' \\ &= \mathbb{E} \left\{ (x_t - x_{t|t-1}) (C(x_t - x_{t|t-1}) + V_t)' | z^{t-1} \right\} K_t' \\ &= \mathbb{E} \left\{ (x_t - x_{t|t-1}) \left((x_t - x_{t|t-1})' C' + V_t' \right) | z^{t-1} \right\} K_t' \\ &= \left[\begin{array}{c} \mathbb{E} \left\{ (x_t - x_{t|t-1}) (x_t - x_{t|t-1})' | z^{t-1} \right\} C' K_t' + \\ \mathbb{E} \left\{ (x_t - x_{t|t-1}) V_t' | z^{t-1} \right\} K_t' \end{array} \right] \\ &= \mathbb{E} \left\{ (x_t - x_{t|t-1}) (x_t - x_{t|t-1})' | z^{t-1} \right\} C' K_t' \\ &= \Sigma_{t|t-1} C' K_t', \end{aligned}$$

$$\begin{aligned} K_t \mathbb{E} \left\{ (z_t - Cx_{t|t-1}) (z_t - Cx_{t|t-1})' | z^{t-1} \right\} K_t' &= K_t \mathbb{E} \left\{ (z_t - z_{t|t-1}) (z_t - z_{t|t-1})' | z^{t-1} \right\} K_t' \\ &= K_t \Omega_{t|t-1} K_t' \\ &= \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} \Omega_{t|t-1} K_t', \quad \text{por (2.11)} \\ &= \Sigma_{t|t-1} C' K_t'. \end{aligned}$$

De manera similar:

$$K_t \mathbb{E} \left\{ (z_t - Cx_{t|t-1}) (x_t - x_{t|t-1})' | z^{t-1} \right\} = K_t C \Sigma_{t|t-1}.$$

Por lo tanto, (2.12) queda determinada por:

$$\Sigma_{t|t} = \Sigma_{t|t-1} - \Sigma_{t|t-1} C' K_t' - K_t C \Sigma_{t|t-1} + \Sigma_{t|t-1} C' K_t' = \Sigma_{t|t-1} - K_t C \Sigma_{t|t-1}. \quad (2.13)$$

Por otro lado:

$$\begin{aligned}
\Sigma_{t+1|t} &= \mathbb{E} \left\{ (x_{t+1} - x_{t+1|t}) (x_{t+1} - x_{t+1|t})' | z^t \right\} \\
&= \mathbb{E} \left\{ (Ax_t + B U_{t+1} - Ax_{t|t}) (Ax_t + B U_{t+1} - Ax_{t|t})' | z^t \right\} \\
&= \mathbb{E} \left\{ (A(x_t - x_{t|t}) + B U_{t+1}) (A(x_t - x_{t|t}) + B U_{t+1})' | z^t \right\} \\
&= \mathbb{E} \left\{ (A(x_t - x_{t|t}) + B U_{t+1}) \left((x_t - x_{t|t})' A' + U_{t+1}' B' \right) | z^t \right\} \\
&= \left[\begin{array}{l} \mathbb{E} \left\{ A(x_t - x_{t|t}) (x_t - x_{t|t})' A' | z^t \right\} + \mathbb{E} \left\{ A(x_t - x_{t|t}) U_{t+1}' B' | z^t \right\} + \\ \mathbb{E} \left\{ B U_{t+1} (x_t - x_{t|t})' A' | z^t \right\} + \mathbb{E} \left\{ B U_{t+1} U_{t+1}' B' | z^t \right\} \end{array} \right] \\
&= A \mathbb{E} \left\{ (x_t - x_{t|t}) (x_t - x_{t|t})' | z^t \right\} A' + B \mathbb{E} \left\{ U_{t+1} U_{t+1}' \right\} B' \\
&= A \Sigma_{t|t} A' + B Q B'.
\end{aligned}$$

2.2.2. Consistencia del filtro de Kalman

Teniendo en cuenta la independencia y la normalidad del estado inicial, $x_{1|0} \sim N(\mu_0, \sigma_0)$ y el error $U_t \sim N(0, Q)$; obtenemos que $x_t \sim N(x_{t|t-1}, \Sigma_{t|t-1})$ y $z_t \sim N(z_{t|t-1}, \Omega_{t|t-1})$, para todo t .

Entonces,

$$\begin{pmatrix} x_t \\ z_t \end{pmatrix} | z^{t-1} \sim N \left(\begin{bmatrix} x_{t|t-1} \\ z_{t|t-1} \end{bmatrix}, \begin{bmatrix} \Sigma_{t|t-1} & \Sigma_{t|t-1} C' \\ C \Sigma_{t|t-1} & \Omega_{t|t-1} \end{bmatrix} \right),^1$$

y

$$x_t | z_t, z^{t-1} = x_t | z^t \sim N \left(x_{t|t-1} + \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} (z_t - z_{t|t-1}), \Sigma_{t|t-1} - \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} C \Sigma_{t|t-1} \right).^2$$

¹ Si $\mathbf{X} = (x_1', x_2')'$ donde $x_1 = (x_{11}, x_{12}, \dots, x_{1p_1})'$ y $x_2 = (x_{21}, x_{22}, \dots, x_{2p_2})'$ son vectores con distribución normal de medias μ_1 y μ_2 respectivamente, entonces \mathbf{X} tiene media $\mu = (\mu_1', \mu_2')'$ de dimensión $p \times 1$ y $\text{cov}(\mathbf{X}) = \begin{pmatrix} \text{cov}(x_1) & \text{cov}(x_1, x_2) \\ \text{cov}(x_2, x_1) & \text{cov}(x_2) \end{pmatrix}$ de dimensión $p \times p$, donde $p = p_1 + p_2$.

² Bajo las mismas hipótesis de ¹, se tiene que: $\mathbb{E}(x_1 | x_2) = \mu_1 + \text{cov}(x_1, x_2) \text{cov}(x_2)^{-1} (z_t - \mu_2)$ y $\text{cov}(x_1 | x_2) = \text{cov}(x_1) - \text{cov}(x_1, x_2) \text{cov}(x_2)^{-1} \text{cov}(x_2, x_1)$.

Luego, de las ecuaciones (2.7), (2.8), (2.10), y (2.13) se sigue que:

$$\begin{aligned} x_{t|t-1} + \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} (z_t - z_{t|t-1}) &= x_{t|t-1} + \Sigma_{t|t-1} C' (C \Sigma_{t|t-1} C' + R)^{-1} (z_t - C x_{t|t-1}) \\ &= x_{t|t-1} + K_t (z_t - z_{t|t-1}) \\ &= x_{t|t}, \end{aligned}$$

y

$$\begin{aligned} \Sigma_{t|t-1} - \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} C \Sigma_{t|t-1} &= \Sigma_{t|t-1} - \Sigma_{t|t-1} C' (C \Sigma_{t|t-1} C' + R)^{-1} C \Sigma_{t|t-1} \\ &= \Sigma_{t|t-1} - K_t C \Sigma_{t|t-1} \\ &= \Sigma_{t|t}. \end{aligned}$$

Así,

$$x_t | z_t, z^{t-1} = x_t | z^t \sim N(x_{t|t}, \Sigma_{t|t}).$$

Esto demuestra la consistencia del algoritmo del Filtro de Kalman; dado que en efecto $x_{t|t}$ es el mejor predictor de x_t y $\Sigma_{t|t}$ el mejor predictor de la matriz de covarianza de x_t dada la información de las variables observables hasta el tiempo t .

2.2.3. Algoritmo

Para un modelo espacio-estado especificado en las ecuaciones (2.1) y (2.2), dados el vector de estado inicial $x_{1|0}$ y el vector de observaciones z_1 , necesitamos encontrar $\Sigma_{1|0}$.

Definición 2.2.1. Se dice que un sistema es estable si, para cualquier estado inicial $x_{1|0}$, el vector de estados x_t , converge a un único \bar{x} .

Cuando un sistema es estable, la ecuación:

$$\Sigma_{t+1|t} = A \Sigma_{t|t} A' + B Q B',$$

se convierte en:

$$\bar{\Sigma} = A \bar{\Sigma} A' + B Q B',$$

la cual es una ecuación de Lyapunov.

Partiendo de que nuestro sistema es estable, para obtener el valor inicial $\Sigma_{1|0} = \bar{\Sigma}$, resolvemos la ecuación:

$$\text{vec}(\bar{\Sigma}) = [I - A \otimes A]^{-1} \text{vec}(BQB').$$

Algoritmo de filtro de Kalman

Dados $\Sigma_{t|t-1}$, $x_{t|t-1}$ y el vector de observaciones z_t .

Para $t = 1, 2, \dots$

$$\Omega_{t|t-1} = C\Sigma_{t|t-1}C' + R,$$

$$z_{t|t-1} = Cx_{t|t-1},$$

$$\Sigma_{t|t} = \Sigma_{t|t-1} - K_t C\Sigma_{t|t-1},$$

$$x_{t|t} = x_{t|t-1} + K_t (z_t - z_{t|t-1}),$$

$$\Sigma_{t+1|t} = A\Sigma_{t|t}A' + BQB',$$

$$x_{t+1|t} = Ax_{t|t}.$$

donde K_t viene dado por (2.11).

2.2.4. Estimación de la función de verosimilitud

Sea $\Theta = (A, B, Q)$, el vector de parámetros. Estamos interesados en hallar la función de verosimilitud de $z^T = \{z_t\}_{t=1}^T$. Es decir:

$$\mathcal{L}(z^T | \Theta),$$

Definamos el vector de innovaciones de dimensión $p \times 1$, como:

$$\epsilon_t = z_t - z_{t|t-1}. \quad (2.14)$$

y estudiemos sus características:

- $\mathbb{E}(\epsilon_t) = \mathbb{E}(z_t - z_{t|t-1}) = \mathbb{E}(z_t) - \mathbb{E}(\mathbb{E}(z_t | z^{t-1})) = \mathbb{E}(z_t) - \mathbb{E}(z_t) = 0.$

- $\Sigma_{\epsilon_t} = \text{var}(\epsilon_t) = \text{var}(z_t - z_{t|t-1})$

$$= \text{var}(z_t - Cx_{t|t-1})$$

$$= \text{var}(Cx_t + V_t - Cx_{t|t-1})$$

$$= \text{var}(C(x_t - x_{t|t-1}) + V_t)$$

$$= \text{cov}\left(C(x_t - x_{t|t-1}) + V_t, (C(x_t - x_{t|t-1}) + V_t)'\right)$$

$$= \begin{bmatrix} \text{cov}\left(C(x_t - x_{t|t-1}), (x_t - x_{t|t-1})' C'\right) + \text{cov}\left(C(x_t - x_{t|t-1}), V_t'\right) + \\ \text{cov}\left(V_t, (x_t - x_{t|t-1})' C'\right) + \text{cov}(V_t, V_t') \end{bmatrix}$$

$$= C\Sigma_{t|t-1}C' + R$$

$$= \Omega_{t|t-1}.$$

Esto dado que $V_t \sim N(0, R)$ y su independencia con respecto a las demás variables aleatorias.

Así, hemos demostrado que $\epsilon_t \sim N(0, \Omega_{t|t-1})$. Además estas innovaciones dependen del vector de parámetros Θ .

La función de verosimilitud es calculada utilizando las innovaciones definidas en (2.14), el supuesto de normalidad de $x_{1|0}$ ($x_{1|0} \sim N(\mu_0, \sigma_0)$), que los errores $U_t \sim N(0, R)$, $V_t \sim N(0, Q)$ y su independencia. De la siguiente manera:

Primero calculamos la función de verosimilitud de las innovaciones hasta el tiempo T , $\epsilon = (\epsilon_1, \epsilon_2, \dots, \epsilon_T)$. Veamos:

$$\mathcal{L}(\epsilon|\Theta) = f(\epsilon_1, \epsilon_2, \dots, \epsilon_T|\Theta),$$

donde f es la función de densidad de ϵ , la cual corresponde a la distribución normal con media 0 y matriz de covarianza $\Omega_{t|t-1}$ (como lo hemos probado). Así:

$$\begin{aligned} \mathcal{L}(\epsilon|\Theta) &= f(\epsilon_1, \epsilon_2, \dots, \epsilon_T|\Theta) \\ &= \prod_{t=1}^T f(\epsilon_t|\Theta), \quad \text{por independencia} \\ &= \prod_{t=1}^T \frac{|\Omega_{t|t-1}|^{-\frac{1}{2}}}{(2\pi)^{\frac{p}{2}}} \exp\left(-\frac{1}{2}\epsilon_t'\Omega_{t|t-1}^{-1}\epsilon_t\right), \end{aligned}$$

Tomando logaritmo en ambos lados de la igualdad:

$$\begin{aligned} \log \mathcal{L}(\epsilon|\Theta) &= \log \left\{ \prod_{t=1}^T \frac{|\Omega_{t|t-1}|^{-\frac{1}{2}}}{(2\pi)^{\frac{p}{2}}} \exp\left(-\frac{1}{2}\epsilon_t'\Omega_{t|t-1}^{-1}\epsilon_t\right) \right\} \\ &= \sum_{t=1}^T \left(-\frac{p}{2} \log(2\pi) - \frac{1}{2} \log |\Omega_{t|t-1}| - \frac{1}{2} \epsilon_t'\Omega_{t|t-1}^{-1}\epsilon_t \right) \\ &= -\sum_{t=1}^T \left(\frac{p}{2} \log(2\pi) + \frac{1}{2} \log |\Omega_{t|t-1}| + \frac{1}{2} \epsilon_t'\Omega_{t|t-1}^{-1}\epsilon_t \right). \end{aligned}$$

Por otra parte,

$$\mathcal{L}(z^T|\Theta) = \prod_{t=1}^T f(z_t|z^{t-1}, \Theta)$$

donde f es la función de densidad de z^T .

Luego, como $z_t = \epsilon_t + z_{t|t-1}$, entonces $f(z_t|z^{t-1}, \Theta) = f(\epsilon_t|\Theta)$. Así,

$$\log \mathcal{L}(\epsilon|\Theta) = \log \mathcal{L}(z^T|\Theta).$$

Por tanto,

$$\log \mathcal{L}(z^T|\Theta) = - \sum_{t=1}^T \left(\frac{p}{2} \log(2\pi) + \frac{1}{2} \log |\Omega_{t|t-1}| + \frac{1}{2} \epsilon_t' \Omega_{t|t-1}^{-1} \epsilon_t \right). \quad (2.15)$$

Observación 2.2.1. En cada corrida del filtro de Kalman obtenemos un ϵ_t ; definido como $\epsilon_t = z_t - z_{t|t-1}$, por tanto haciendo T corridas del filtro de Kalman obtenemos $\epsilon = (\epsilon_1, \epsilon_2, \dots, \epsilon_T)$ y de esta manera obtenemos la estimación de la función de verosimilitud.

CAPÍTULO 3

MATRIZ DE INFORMACIÓN

3.1. Introducción

Gran parte de la creciente literatura sobre la evaluación empírica de los modelos DSGE utilizan métodos de estimación basados en la función de verosimilitud, luego de disponer de la representación espacio-estado del modelo es posible obtener la evaluación de la función de verosimilitud vía filtro de Kalman; pero en la práctica estos métodos requieren de la evaluación de la segunda derivada (el Hessiano) del logaritmo de esta función (la función log-verosimilitud). En la estimación por máxima verosimilitud la matriz de covarianza asintótica resulta igual a la matriz de información y es por esto que estaremos interesados en calcularla, dado que algoritmos como los que implementaremos (Metropolis-Hasting y muestreo de importancia) necesitan de la inversa de la estimación del Hessiano para encontrar la distribución a posteriori de los parámetros.

A pesar de su importancia su cálculo no es sencillo, en la mayoría de los modelos DSGE el mapeo entre los parámetros profundos y los parámetros de la forma reducida es altamente no lineal y puede ser encontrada solo numéricamente. Derivar una expresión analítica para el Hessiano es imposible por diferenciación directa de la función de verosimilitud, por lo que los investigadores han estado usando derivadas numéricas, lo cual se sabe que tiene deficiencias importantes, y produce un gran costo computacional. Más aun, estos resultados pueden ser imprecisos cuando la función de verosimilitud no tiene un buen comportamiento, lo que conlleva a resultados erróneos; por consiguiente, es deseable utilizar expresiones analíticas para estas derivadas siempre que esto sea factible.

En este capítulo se desarrollarán las técnicas propuestas en [6], donde se propone estimar la matriz de información para modelos DSGE, sin necesidad de implementar derivadas numéricas. Como hemos mencionado, esto es de gran utilidad para métodos iterativos como el Metropolis-Hasting y muestreo de importancia; nuestra finalidad será implementar dicho cálculo de la matriz de información en estos algoritmos para un modelo DSGE estándar.

3.2. Estimación de la matriz de información

Un modelo DSGE luego de ser linealizado puede ser expresado de la siguiente forma estándar:

$$\Gamma_0 x_t = \Gamma_1 \mathbb{E}_t x_{t+1} + \Gamma_2 x_{t-1} + \Gamma_3 U_t, \quad (3.1)$$

donde x_t es un vector m -dimensional de variables endógenas, los errores estructurales U_t son un vector n -dimensional gaussiano de variables aleatorias independientes e idénticamente distribuidas (i.i.d) con $\mathbb{E}(U_t) = 0$ y $\mathbb{E}(U_t U_t') = I_n$. Las matrices de coeficientes Γ_0 , Γ_1 , Γ_2 y Γ_3 son funciones del vector k -dimensional de parámetros profundos θ ; donde Γ_0 , Γ_1 , $\Gamma_2 \in M_{n \times n}$ y $\Gamma_3 \in M_{m \times n}$.

Bajo ciertas condiciones, según [11], el modelo (3.1) posee única solución dada por la siguiente forma reducida:

$$x_t = Ax_{t-1} + BU_t, \quad (3.2)$$

$$z_t = Cx_t, \quad (3.3)$$

donde z_t es un vector p -dimensional de variables observables y C es una matriz de orden $p \times m$ de constantes conocidas.

La ecuación (3.2) representa la ecuación de estado y la ecuación (3.3) la ecuación de observación del modelo DSGE.

Sea ϕ el vector de parámetros de la forma reducida, de dimensión $m^2 + mn$, definida por:

$$\phi = [\text{vec}(A)', \text{vec}(B)']'.$$

Resolver el modelo lineal DSGE (3.1), significa hallar para un valor dado de parámetros profundos θ un único valor de parámetros de la forma reducida, ϕ . Denotamos la función que mapea (envía) θ en ϕ por h , esto es,

$$\phi = h(\theta).$$

Sea $H = h_\theta(\theta)$, el gradiente de h . Con excepción de algunos modelos muy sencillos es imposible encontrar h analíticamente; y es por esto que también es imposible encontrar H derivando directamente.

Estamos interesados en encontrar la matriz de información \mathfrak{J}_θ , definida como:

$$\mathfrak{J}_\theta = -\mathbb{E}[\mathcal{L}_{\theta\theta}(z, \theta)],$$

donde $\mathcal{L}_{\theta\theta}$ es la segunda derivada de la función log-verosimilitud con respecto al vector de parámetros profundos θ .

Dado que la función log-verosimilitud para un modelo espacio-estado viene dada por (2.15), el elemento (i, j) de la matriz de información \mathfrak{J}_ϕ esta definido de la siguiente manera:

$$-\mathbb{E}\left[\frac{\partial^2 \mathcal{L}(z^t|\phi)}{\partial \phi_i \partial \phi_j}\right] = \mathbb{E}\left[\sum_{t=1}^T \frac{\partial^2 l_t}{\partial \phi_i \partial \phi_j}\right],$$

donde $l_t = \left(\frac{p}{2} \log(2\pi) + \frac{1}{2} \log |\Omega_{t|t-1}| + \frac{1}{2} \epsilon_t' \Omega_{t|t-1}^{-1} \epsilon_t\right)$.

Según [9], \mathfrak{J}_θ puede expresarse de la siguiente forma:

$$\mathfrak{J}_\theta = H' \mathfrak{J}_\phi H, \quad (3.4)$$

donde \mathfrak{J}_ϕ es la matriz de información del modelo en la forma reducida.

Así, podríamos obtener las matrices H y \mathfrak{J}_ϕ , y a través de la ecuación (3.4) encontrar una expresión analítica para \mathfrak{J}_θ , en lugar de obtenerla por medio de métodos numéricos.

En varios trabajos se han desarrollado expresiones analíticas para la matriz de información para modelos espacio-estado lineales. Más recientemente, Klein y Neudecker (2000) usan la diferenciación de matrices para obtener la matriz explícitamente. Segal y Weinstein (1989) proveen un método para la evaluación de la matriz de información tanto como el score y el Hessiano del logaritmo de la función de verosimilitud para modelos espacio-estado. Tales algoritmos también han sido desarrollados en Zdrozny (1989) y Terceiro (1990).

3.2.1. Cálculo de la matriz H

Como la función h no está disponible analíticamente, como mencionamos, no es posible calcular su gradiente H por diferenciación directa. Sin embargo, es posible encontrar la matriz H con la ayuda de la función implícita, como mostraremos.

De la ecuación (3.2), tenemos:

$$\begin{aligned} \mathbb{E}_t(x_{t+1}) &= \mathbb{E}_t(Ax_t + BU_t) \\ &= A\mathbb{E}_t(x_t) + B\mathbb{E}_t(U_t) \\ &= A\mathbb{E}_t(x_t), \quad \text{dado que } \mathbb{E}_t(U_t) = 0 \\ &= Ax_t. \end{aligned}$$

Sustituyendo esta última ecuación en (3.1):

$$\Gamma_0 x_t = \Gamma_1 A x_t + \Gamma_2 x_{t-1} + \Gamma_3 U_t,$$

o equivalentemente:

$$(\Gamma_0 - \Gamma_1 A) x_t = \Gamma_2 x_{t-1} + \Gamma_3 U_t. \quad (3.5)$$

Sustituyendo (3.2) en (3.5):

$$(\Gamma_0 - \Gamma_1 A) A x_{t-1} + (\Gamma_0 - \Gamma_1 A) B U_t = \Gamma_2 x_{t-1} + \Gamma_3 U_t.$$

Luego, igualando término a término, obtenemos que:

$$\begin{aligned} (\Gamma_0 - \Gamma_1 A) A &= \Gamma_2 & \text{y} \\ (\Gamma_0 - \Gamma_1 A) B &= \Gamma_3. \end{aligned}$$

Así, resultan los siguientes dos conjuntos de ecuaciones:

$$(\Gamma_0 - \Gamma_1 A) A - \Gamma_2 = 0_{m \times m}, \quad (3.6)$$

$$(\Gamma_0 - \Gamma_1 A) B - \Gamma_3 = 0_{m \times n}. \quad (3.7)$$

Las expresiones (3.6) y (3.7) definen una función implícita de \mathbb{R}^{k+m^2+mn} a \mathbb{R}^{m^2+mn} , por:

$$F(\theta, \phi) = F(\theta, h(\theta)) = \begin{bmatrix} \text{vec}(B_1) \\ \text{vec}(B_2) \end{bmatrix} = 0_{m^2+mn}, \quad (3.8)$$

donde $B_1 = (\Gamma_0 - \Gamma_1 A) A - \Gamma_2$ y $B_2 = (\Gamma_0 - \Gamma_1 A) B - \Gamma_3$.

Por tanto, tenemos una función de \mathbb{R}^{k+m^2+mn} a \mathbb{R}^{m^2+mn} , tal que $F(\theta, h(\theta)) = 0_{m^2+mn}$, luego por el teorema de la función implícita (ver sección apéndice A), obtenemos que:

$$H = \frac{\partial h(\theta)}{\partial \theta'} = - (F_\phi(\theta, h(\theta)))^{-1} F_\theta(\theta, h(\theta)). \quad (3.9)$$

Ejemplo 4. Supongamos que el modelo DSGE (3.1) viene dado por:

$$x_t = \alpha \mathbb{E}_t(x_{t+1}) + (1 - \alpha)x_{t-1} + \sigma U_t,$$

donde x_t es univariada, $\alpha > .5$ y $U_t \sim N(0, 1)$.

Para este ejemplo, el vector de parámetros profundos es $\theta = [\alpha, \sigma]'$, $\Gamma_0 = 1$, $\Gamma_1 = \alpha$, $\Gamma_2 = 1 - \alpha$, y $\Gamma_3 = \sigma$.

La representación espacio-estado viene dada por:

$$x_t = Ax_{t-1} + B U_t,$$

donde la variable x_t es observable, es decir, $z_t = x_t$ y el vector de parámetros de la forma reducida es $\phi = [A, B]'$.

De las ecuaciones (3.6) y (3.7) es posible encontrar fácilmente los valores de A y B :

$$A = \frac{1 - \alpha}{\alpha}, \quad B = \frac{\alpha}{\sigma}.$$

Por tanto, H puede ser calculada por derivación directa,

$$H = \frac{\partial h(\theta)}{\partial \theta'} = \begin{bmatrix} -\frac{1}{\alpha^2} & 0 \\ -\frac{\sigma}{\alpha^2} & \frac{1}{\alpha} \end{bmatrix}. \quad (3.10)$$

Este cálculo directo resulta imposible en la mayoría de los casos. Sin embargo, como hemos probado H puede ser encontrada analíticamente usando la función implícita definida por (3.8) y aplicando el teorema de la función implícita. Para nuestro caso, dicha función implícita F , viene dada por:

$$F(\theta, h(\theta)) = \begin{bmatrix} A - \alpha A^2 - 1 + \alpha \\ B - \alpha AB - \sigma \end{bmatrix} = 0_{\mathbb{R}^2}.$$

Luego,

$$F_\theta(\theta, h(\theta)) = \begin{bmatrix} -A^2 + 1 & 0 \\ -AB & -1 \end{bmatrix},$$

$$F_\phi(\theta, h(\theta)) = \begin{bmatrix} 1 - 2\alpha A & 0 \\ -\alpha B & 1 - \alpha A \end{bmatrix},$$

y

$$[F_\phi(\theta, h(\theta))]^{-1} = \begin{bmatrix} \frac{1}{1-2\alpha A} & 0 \\ -\frac{\alpha B}{(1-2\alpha A)(1-\alpha A)} & \frac{1}{1-\alpha A} \end{bmatrix}.$$

Finalmente, por la ecuación (3.9), se tiene:

$$\begin{aligned} H &= -(F_\phi(\theta, h(\theta)))^{-1} F_\theta(\theta, h(\theta)) \\ &= - \begin{bmatrix} \frac{1}{1-2\alpha A} & 0 \\ -\frac{\alpha B}{(1-2\alpha A)(1-\alpha A)} & \frac{1}{1-\alpha A} \end{bmatrix} \begin{bmatrix} -A^2 + 1 & 0 \\ -AB & -1 \end{bmatrix} \\ &= - \begin{bmatrix} \frac{-A^2+1}{1-2\alpha A} & 0 \\ \frac{\alpha B(-A^2+1)-(1-2\alpha A)AB}{(1-2\alpha A)(1-\alpha A)} & \frac{-1}{1-\alpha A} \end{bmatrix}. \end{aligned}$$

Sustituyendo en la última expresión los valores de A y B :

$$H = \frac{\partial h(\theta)}{\partial \theta'} = \begin{bmatrix} -\frac{1}{\alpha^2} & 0 \\ -\frac{\sigma}{\alpha^2} & \frac{1}{\alpha} \end{bmatrix}. \quad (3.11)$$

Por tanto, la expresión (3.11) resulta igual a la expresión (3.10), así de manera analítica a través de la función implícita (3.8) se encuentra el mismo resultado para la matriz H , sin necesidad de cálculos numéricos.

Cálculo de la matriz H por factorización de la función F :

Los cálculos de los gradientes requeridos en (3.9) pueden simplificarse por el hecho de que la función F definida por (3.8) puede ser factorizada, como:

$$F(\theta, h(\theta)) = F_1(h(\theta)) F_2(\theta).$$

Veamos:

Sea $\gamma = [vec(\gamma_0)', vec(\gamma_1)', vec(\gamma_2)', vec(\gamma_3)']'$ el vector de parámetros estructurales. En otras palabras, γ contiene los parámetros en los cuales el modelo (3.1) es lineal.

Definimos al vector s -dimensional $\tilde{\gamma}$, como el vector γ sin repeticiones, es decir, si en el vector γ hay entradas que se repiten estas se eliminan y el resultado es $\tilde{\gamma}$.

Las matrices $\Gamma_0, \Gamma_1, \Gamma_2, \Gamma_3$ contienen ceros, otras constantes conocidas y elementos de $\tilde{\gamma}$. Entonces, para $i = 0, 1, 2$, podemos escribir:

$$vec(\Gamma_i) = G_i \tilde{\gamma}, \quad (3.12)$$

donde cada G_i es una matriz de dimensión $m^2 \times s$ que contiene solo ceros y unos.

Análogamente,

$$vec(\Gamma_3) = G_3 \tilde{\gamma}, \quad (3.13)$$

donde G_3 es una matriz de dimensión $mn \times s$, que contiene solo ceros y unos.

Ahora, vectorizando a ambos lados la ecuación (3.6):

$$vec(\Gamma_0 A) - vec(\Gamma_1 A^2) - vec(\Gamma_2) = 0_{m^2}.$$

Aplicando propiedades del operador vec , esta última expresión se transforma en:

$$(A' \otimes I_m) vec(\Gamma_0) - ((A^2)' \otimes I_m) vec(\Gamma_1) - vec(\Gamma_2) = 0_{m^2},$$

donde I_m es la matriz identidad de orden m y 0_{m^2} es el vector nulo de dimensión m^2 .

De la ecuación anterior y la ecuación (3.12), se tiene:

$$[(A' \otimes I_m) G_0 - ((A^2)' \otimes I_m) G_1 - G_2] \tilde{\gamma} = 0_{m^2},$$

donde el lado derecho del producto no depende directamente de θ , sino solo de ϕ , y donde $\tilde{\gamma}$ depende de θ .

De la misma manera; vectorizando la ecuación (3.7), aplicando propiedades del operador vec , y de las ecuaciones (3.12) y (3.13), obtenemos:

$$[(B' \otimes I_m) G_0 - (B' A' \otimes I_m) G_1 - G_3] \tilde{\gamma} = 0_{mn}.$$

Así, el sistema conformado por (3.6) y (3.7) se transforma en:

$$\begin{bmatrix} (A' \otimes I_m) G_0 - ((A^2)' \otimes I_m) G_1 - G_2 \\ (B' \otimes I_m) G_0 - (B' A' \otimes I_m) G_1 - G_3 \end{bmatrix} \tilde{\gamma} = \begin{bmatrix} 0_{m^2} \\ 0_{mn} \end{bmatrix}. \quad (3.14)$$

Hagamos $F_1(h(\theta))$ y $F_2(\theta)$ el primer y segundo término del producto del lado izquierdo de la ecuación (3.14), respectivamente, así:

$$F(\theta, h(\theta)) = F_1(h(\theta)) F_2(\theta). \quad (3.15)$$

Ahora, encontremos las expresiones para $F_\theta(\theta, h(\theta))$ y $F_\phi(\theta, h(\theta))$. Haciendo $d = m^2 + mn$, tenemos:

$$\bullet F_\theta(\theta, h(\theta)) = \frac{\partial (F_1(h(\theta)) F_2(\theta))}{\partial \theta'}, \quad \text{por (3.15)}$$

$$= ((F_2(\theta))' \otimes I_d) \frac{\partial \text{vec}(F_1(h(\theta)))}{\partial \theta'} + F_1(h(\theta)) \frac{\partial F_2(\theta)}{\partial \theta'}, \quad (3.16)$$

pero como:

$$\frac{\partial \text{vec}(F_1(h(\theta)))}{\partial \theta'} = 0,$$

entonces la ecuación (3.16), queda de la siguiente forma:

$$F_\theta(\theta, h(\theta)) = F_1(h(\theta)) \frac{\partial F_2(\theta)}{\partial \theta'}. \quad (3.17)$$

De igual forma:

$$\bullet F_\phi(\theta, h(\theta)) = \frac{\partial (F_1(h(\theta)) F_2(\theta))}{\partial \phi'}, \quad \text{por (3.15)}$$

$$= ((F_2(\theta))' \otimes I_d) \frac{\partial \text{vec}(F_1(h(\theta)))}{\partial \phi'} + F_1(h(\theta)) \frac{\partial F_2(\theta)}{\partial \phi'}, \quad (3.18)$$

pero dado que:

$$\frac{\partial F_2(\theta)}{\partial \phi'} = 0,$$

entonces la ecuación (3.18), queda:

$$F_\phi(\theta, h(\theta)) = ((F_2(\theta))' \otimes I_d) \frac{\partial \text{vec}(F_1(h(\theta)))}{\partial \phi'}. \quad (3.19)$$

Empleando las ecuaciones (3.17) y (3.19) en (3.9), obtenemos el resultado deseado.

3.2.2. Cálculo de la matriz \mathfrak{J}_ϕ

Hasta ahora disponemos de una expresión analítica para obtener la matriz H , falta realizar el cálculo de la matriz \mathfrak{J}_ϕ y de esa forma obtener la matriz de información \mathfrak{J}_θ del modelo DSGE a través de la ecuación (3.4).

La expresión general para la matriz de información \mathfrak{J}_ϕ según [8] puede ser escrita como:

$$\mathfrak{J}_\phi = \frac{1}{2} \sum_{t=1}^T \left(\frac{\partial \text{vec}(\Omega_{t|t-1})}{\partial \phi'} \right)' (\Omega_{t|t-1}^{-1} \otimes \Omega_{t|t-1}^{-1}) \left(\frac{\partial \text{vec}(\Omega_{t|t-1})}{\partial \phi'} \right) + \mathbb{E} \left[\sum_{t=1}^T \left(\frac{\partial \epsilon_t}{\partial \phi'} \right)' \Omega_{t|t-1} \left(\frac{\partial \epsilon_t}{\partial \phi'} \right) \right] \quad (3.20)$$

Dado que por medio del filtro de Kalman disponemos de expresiones analíticas para la matriz $\Omega_{t|t-1}$ y el vector de innovaciones ϵ_t , entonces es posible evaluar sus derivadas analíticamente, así para obtener la matriz de información \mathfrak{J}_ϕ del modelo en la forma reducida (en su representación espacio-estado) debemos implementar un algoritmo recursivo que se ejecute en paralelo con el algoritmo de filtro de Kalman y obtener en cada iteración las derivadas necesarias; además dado que el filtro de Kalman proporciona buenas estimaciones para ϵ_t y $\Omega_{t|t-1}$ es posible eliminar el operador esperanza de la ecuación (3.20) y la expresión obtenida es asintóticamente equivalente, la cual en la mayoría de los casos puede ser evaluada mas fácilmente.

En resumen, el procedimiento será el siguiente: derivaremos las expresiones necesarias del filtro de Kalman (ver apéndice A) y obtendremos su evaluación en cada iteración del algoritmo y a partir de esto, a través de la expresión (3.20) obtendremos la matriz \mathfrak{J}_ϕ .

Las ecuaciones derivadas necesarias del filtro de Kalman son las siguientes:

- $\frac{\partial}{\partial \phi'} (x_{t|t-1}) = \left(x'_{t-1|t-1} \otimes I_m \right) \frac{\partial}{\partial \phi'} \text{vec} (A) + A \frac{\partial}{\partial \phi'} (x_{t-1|t-1}) .$
- $\frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) = (A \otimes I_m) \left[\left((\Sigma_{t-1|t-1})' \otimes I_m \right) \frac{\partial}{\partial \phi'} \text{vec} (A) + (I_m \otimes A) \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t-1|t-1}) \right] + (I_m \otimes (A \Sigma_{t-1|t-1})) \frac{\partial}{\partial \phi'} (A') + (BQ' \otimes I_m) \frac{\partial}{\partial \phi'} \text{vec} (B) + (I_m \otimes (BQ)) \frac{\partial}{\partial \phi'} (B') .$
- $\frac{\partial}{\partial \phi'} (\epsilon_t) = -C \frac{\partial}{\partial \phi'} (x_{t|t-1}) .$
- $\frac{\partial}{\partial \phi'} \text{vec} (\Omega_{t|t-1}) = (C \otimes C) \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) .$
- $\frac{\partial}{\partial \phi'} \text{vec} (\Omega_{t|t-1}^{-1}) = - \left((\Omega_{t|t-1}^{-1})' \otimes \Omega_{t|t-1}^{-1} \right) \frac{\partial}{\partial \phi'} \text{vec} (\Omega_t) .$
- $\frac{\partial}{\partial \phi'} (x_{t|t}) = \frac{\partial}{\partial \phi'} (x_{t|t-1}) + \left((\Omega_{t|t-1}^{-1} \epsilon_t)' \otimes I_m \right) (C \otimes I_m) \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) + (\Sigma_{t|t-1} C') \left[(\epsilon_t' \otimes I_p) \frac{\partial}{\partial \phi'} \text{vec} (\Omega_{t|t-1}^{-1}) + \Omega_{t|t-1}^{-1} \frac{\partial}{\partial \phi'} \text{vec} (\epsilon_t) \right] .$
- $\frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t}) = \frac{\partial}{\partial \phi'} (\Sigma_{t|t-1}) - \left((C \Sigma_{t|t-1})' \otimes I_m \right) \left[\left((\Omega_{t|t-1}^{-1})' C \otimes I_m \right) \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) + (I_p \otimes \Sigma_{t|t-1} C') \frac{\partial}{\partial \phi'} \text{vec} (\Omega_{t|t-1}^{-1}) \right] - \left(I_m \otimes \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} \right) (I_m \otimes C) \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) .$

Observación 3.2.1. Los valores iniciales para $\frac{\partial}{\partial \phi'}(x_{0|0})$ y $\frac{\partial}{\partial \phi'} \text{vec}(\Sigma_{0|0})$ dependen de los valores iniciales del filtro de Kalman, si $x_{0|0}$ y $\Sigma_{0|0}$ son independientes de ϕ , se puede tomar $\frac{\partial}{\partial \phi'}(x_{0|0}) = 0$ y $\frac{\partial}{\partial \phi'} \text{vec}(\Sigma_{0|0}) = 0$.

CAPÍTULO 4

ALGORITMOS MCMC

4.1. Introducción

Los métodos Bayesianos son reconocidos como una manera coherente de hacer inferencia, en contraste con los métodos clásicos, donde los datos obtenidos de estudios observacionales y/o experimentales son analizados con modelos que dependen del tipo de datos con procedimientos de inferencia y decisión particulares para cada caso. El análisis Bayesiano trata de una manera unificada la inferencia y la decisión, tomando en consideración la incertidumbre asociada al modelo y a los parámetros, proporcionando de una vez las herramientas para cuantificar esta incertidumbre. Por otra parte, el tratamiento de las cantidades no observadas como variables aleatorias y el análisis condicional, permiten naturalmente considerar modelos jerárquicos o de variables latentes que son difíciles o imposibles de manejar con la estadística clásica. Su nombre de “Bayesiana” proviene del uso frecuente que hace del teorema de Bayes, el cual establece lo siguiente:

Suponga que se observa una variable aleatoria Z y se desea hacer inferencias acerca de otra variable aleatoria θ , donde θ se extrae de alguna densidad $\pi(\theta)$, a esta distribución se le conoce como distribución previa o a priori de θ . Entonces:

$$\pi(\theta|Z) = \frac{\pi(Z|\theta)\pi(\theta)}{\pi(Z)}, \quad (4.1)$$

donde:

- En el caso discreto: $\pi(Z) = \sum_{\theta} \pi(\theta)\pi(Z|\theta)$ (suma sobre los posibles valores de θ).

- En el caso continuo: $\pi(Z) = \int \pi(\theta) \pi(Z|\theta) d\theta$.

Este resultado fue enunciado por Thomas Bayes en 1763, pero no es hasta finales de la década de 1980 y principios de la década de 1990 con la aparición de métodos computacionales basados en simulación, los que permiten implementar el paradigma Bayesiano en la práctica. Una de las desventajas que posee la inferencia Bayesiana, es que en ella se requiere encontrar la densidad a posteriori de los parámetros o cantidades desconocidas de los modelos. La cual se define como:

Definición 4.1.1. Sea $Z = (z_1, z_2, \dots, z_n)$ una muestra aleatoria de una población con función de densidad f_θ , donde θ es un parámetro desconocido y función de verosimilitud $\mathcal{L}(\theta, Z) = \prod_{i=1}^n f_\theta(z_i)$. Se define la densidad a posteriori para el caso continuo como:

$$\begin{aligned} \pi(\theta|Z) &= \frac{\mathcal{L}(\theta|Z) \pi(\theta)}{\int \mathcal{L}(\theta|Z) \pi(\theta) d\theta} \\ &\propto \mathcal{L}(\theta|Z) \pi(\theta), \end{aligned}$$

donde \propto significa “proporcional a” y la distribución $\pi(\theta)$ es la distribución previa o a priori de θ .

Obtener expresiones analíticas de la densidad a posteriori sólo es posible en casos particulares que usualmente representan modelos muy sencillos; pero dado que través del filtro de Kalman es posible obtener estimaciones de la función de verosimilitud entonces, resulta también posible la obtención de la estimación de la densidad a posteriori, través de métodos de simulación. En este capítulo presentaremos dos de estos métodos, el algoritmo muestreo de importancia y el algoritmo Metropolis-Hastings.

Estos algoritmos de simulación se refieren a la obtención de pseudo-muestras provenientes de una distribución de probabilidad, donde estas muestras pueden cumplir con ciertas condiciones que permiten estimar propiedades probabilísticas que no pueden ser obtenidas por métodos analíticos. Como este tipo de métodos introducen un nivel de aleatoriedad en el análisis, se les considera como métodos

de Monte Carlo, en honor al famoso casino del principado de Mónaco. El algoritmo Metropolis-Hastings pertenece a la clase de los algoritmos MCMC (Monte Carlo Markov Chain) y si bien, el algoritmo muestreo de importancia no pertenece a la familia de los algoritmos MCMC, como hemos comentado, es un algoritmo Monte Carlo y es considerado dentro de este estudio dado que ambos nos permite encontrar la esperanza posterior de cualquier función a través del cálculo adecuado de un promedio, y esta eficacia esta garantizada gracias a ley fuerte de los grandes números (ver sección 1.1.3).

4.2. Algoritmo Muestreo de Importancia

El algoritmo muestreo de importancia es utilizado para calcular la densidad a posteriori de los parámetros de un modelo estocástico de crecimiento linealizado.

La idea del algoritmo esta basada en la siguiente identidad:

$$\mathbb{E}[h(\theta)|Z] = \int h(\theta) \pi(\theta|Z) d\theta = \int \frac{h(\theta) \pi(\theta|Z)}{q(\theta)} q(\theta) d\theta, \quad (4.2)$$

donde q es una función densidad de probabilidad conocida como función de importancia.

El muestreo desde la densidad a posteriori $\pi(\theta|Z)$ es reemplazado por el muestreo desde la densidad $q(\theta)$ y ponderado por el radio de importancia $\pi(\theta|Z)/q(\theta)$ para obtener una aproximación numérica del valor esperado del posterior de interés. Este algoritmo se basa en la buena escogencia de la función q , tal que sea la función de densidad de una distribución de la que se puede simular fácilmente.

Hammersley y Handscomb (1964) fueron los primeros en proponer este método y Geweke (1989) provee resultados importantes de convergencia. Una versión particular de este algoritmo es utilizado por An y Shorfheide (2007) (ver [2]), donde como función de importancia es utilizada la función de densidad de una distribu-

ción t multivariada. Esta versión del algoritmo fue utilizada en este trabajo, la cual presentamos a continuación:

Algoritmo Muestreo de Importancia
<p>Inicialización:</p> <ul style="list-style-type: none"> • Denote la moda del posterior por $\tilde{\theta}$. • Sea $\tilde{\Sigma}$ la inversa de la matriz hessiana calculada en $\tilde{\theta}$. • Sea $q(\theta)$ la densidad de una t-distribución multivariada con media $\tilde{\theta}$, matriz de covarianza $c^2\tilde{\Sigma}$, y v grados de libertad.
<p>Para $s = 1, \dots, n_{sim}$</p> <ol style="list-style-type: none"> 1. Generar $\theta^{(s)}$ desde $q(\theta)$. 2. Calcular $\tilde{w}_s(\theta^{(s)}) = \frac{\mathcal{L}(\theta^{(s)} Z)\pi(\theta^{(s)})}{q(\theta^{(s)})}$. 3. Aproximar el valor esperado del posterior de la función $h(\theta)$ por: $\sum_{s=1}^{n_{sim}} \frac{\tilde{w}_s(\theta^{(s)}) h(\theta^{(s)})}{\hat{w}},$ <p>donde $\hat{w} = \sum_{s=1}^{n_{sim}} \tilde{w}_s(\theta^{(s)})$.</p>

Observación 4.2.1. El escalar c empleado en este algoritmo es de vital importancia, dado que su utilización debe permitir que la evaluación del prior (función π) en los vectores generados desde la función de importancia, no estén frecuentemente fuera del soporte.

4.3. Cadenas de Markov

Antes de presentar el algoritmo Metropolis-Hastings, exponemos algunos resultados necesarios relativos a las cadenas de Markov.

Definición 4.3.1. Un proceso estocástico es una familia de variables aleatorias, X_t , donde $t \in T$, y T es un conjunto adecuado de índices.

Definición 4.3.2. Un proceso de Markov $\{X_t\}$ es un proceso estocástico con la siguiente propiedad: dado el valor de X_t , los valores de X_s con $s > t$ no están influenciados por los valores de X_u con $u < t$. Es decir, la probabilidad de cualquier comportamiento futuro del proceso, cuando el estado actual se conoce con exactitud, no se ve alterada por el conocimiento de su comportamiento en el pasado. **Una cadena de Markov de tiempo discreto**, es un proceso de Markov cuyo espacio-estado (rango de posibles valores de las variables aleatorias) es finito o un conjunto numerable, y donde el conjunto de índices es $T = (0, 1, 2, 3, \dots)$.

En términos formales, la propiedad de Markov es la siguiente:

$$Pr \{X_{n+1} = j | X_0 = i_0, \dots, X_{n-1} = i_{n-1}, X_n = i\} = Pr \{X_{n+1} = j | X_n = i\}, \quad (4.3)$$

para todo tiempo n y todos los estados $i_0, i_1, \dots, i_{n-1}, i, j$.

Observación 4.3.1. En una cadena de Markov, interpretamos a X_n como el “estado del sistema en el instante n ”.

La probabilidad de que X_{n+1} este en el estado j dado que X_n esta en el estado i es llamada **la probabilidad de transición de un paso** y es denotado por $P_{ij}^{n,n+1}$. Esto es,

$$P_{ij}^{n,n+1} = Pr \{X_{n+1} = j | X_n = i\}.$$

Esta notación hace hincapié en que en general la probabilidad de transición no solo es función del estado inicial y final, si no también del tiempo de transición. Cuando la probabilidad de transición de los pasos son independientes de la variable de tiempo n , se dice que la cadena de Markov tiene probabilidad de transición estacionaria. Como la mayoría de las cadenas de Markov que se encuentran poseen esta

propiedad, pondremos nuestro interés en este caso. Así, tenemos que $P_{ij}^{n,n+1} = P_{ij}$ y P_{ij} es la probabilidad condicional de que el valor del estado i se mueva al estado j en un paso.

Observación 4.3.2. Muchas veces es conveniente etiquetar el espacio-estado de una cadena de Markov con los enteros no-negativos $\{0, 1, 2, \dots\}$.

Definición 4.3.3. Se define la matriz de probabilidades transición o matriz de Markov, como la matriz \mathbf{P} cuyo elemento (i, j) es P_{ij} , es decir, la matriz en cuyo elemento (i, j) se encuentra la probabilidad de pasar del estado i al estado j .

Así, la matriz \mathbf{P} queda como sigue:

$$\mathbf{P} = \begin{bmatrix} P_{00} & P_{01} & P_{02} & P_{03} & \cdots \\ P_{10} & P_{11} & P_{12} & P_{13} & \cdots \\ P_{20} & P_{21} & P_{22} & P_{23} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \\ P_{i0} & P_{i1} & P_{i2} & P_{i3} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \end{bmatrix}$$

Si el número de estados es finito, entonces \mathbf{P} es una matriz cuadrada finita, cuyo orden (el número de filas) es igual al número de estados. Claramente, las cantidades P_{ij} satisfacen las condiciones:

$$P_{ij} \geq 0, \quad \forall i, j = 0, 1, 2, \dots \quad (4.4)$$

$$\sum_{j=0}^{\infty} P_{ij} = 1, \quad \forall i, j = 0, 1, 2, \dots \quad (4.5)$$

La condición (4.5) simplemente indica que alguna transición se produce en cada ensayo (por conveniencia se dice que la transición se ha producido incluso si el estado

se mantiene sin cambios).

Un proceso de Markov es completamente definido una vez que la matriz de probabilidades de transición y el estado inicial X_0 (de forma más general, la distribución de probabilidad de X_0) están especificados. Probemos este hecho:

Sea $Pr \{X_0 = i_0\} = p_{i_0}$. Es suficiente mostrar como calcular las cantidades:

$$Pr \{X_0 = i_0, X_1 = i_1, X_2 = i_2, \dots, X_n = i_n\}, \quad (4.6)$$

ya que cualquier probabilidad que involucra X_{j_1}, \dots, X_{j_k} , para $j_1 < \dots < j_k$, pueden ser obtenida, acorde al axioma de probabilidad total, por la suma de términos de la forma (4.6).

Por definición de la probabilidad condicional:

$$\begin{aligned} & Pr \{X_0 = i_0, X_1 = i_1, X_2 = i_2, \dots, X_n = i_n\} \\ &= Pr \{X_0 = i_0, X_1 = i_1, X_2 = i_2, \dots, X_{n-1} = i_{n-1}\} \\ &\times Pr \{X_n = i_n | X_0 = i_0, X_1 = i_1, \dots, X_{n-1} = i_{n-1}\}. \end{aligned} \quad (4.7)$$

Ahora, por definición de cadena de Markov,

$$\begin{aligned} Pr \{X_n = i_n | X_0 = i_0, X_1 = i_1, X_2 = i_2, \dots, X_{n-1} = i_{n-1}\} &= Pr \{X_n = i_n | X_{n-1} = i_{n-1}\} \\ &= P_{i_{n-1}i_n}. \end{aligned} \quad (4.8)$$

Sustituyendo (4.8) en (4.7):

$$\begin{aligned} & Pr \{X_0 = i_0, X_1 = i_1, X_2 = i_2, \dots, X_n = i_n\} \\ &= Pr \{X_0 = i_0, X_1 = i_1, X_2 = i_2, \dots, X_{n-1} = i_{n-1}\} P_{i_{n-1}i_n}. \end{aligned} \quad (4.9)$$

Al repetir el proceso $n - 1$ veces, (4.6) se convierte en:

$$Pr \{X_0 = i_0, X_1 = i_1, \dots, X_n = i_n\} = p_{i_0} P_{i_0 i_1} \cdots P_{i_{n-2} i_{n-1}} P_{i_{n-1} i_n}.$$

Esto demuestra que toda probabilidad finita dimensional es especificada una vez que la matriz de probabilidades de transición y la distribución del estado inicial son dados, y el proceso es definido por estas cantidades.

Ejemplo 5. Después de algunos estudios sobre el clima en determinada ciudad, se ha observado que si un día está soleado en el 70 % de los casos el día siguiente continúa soleado y en el 30 % nublado; y si un día está nublado, la probabilidad de que esté soleado el día siguiente es de 60 % y la probabilidad de que se siga nublado es de 40 %. Además, según estos estudios el clima de un determinado día solo depende del clima del día anterior.

Para estudiar este problema primeramente encontramos las probabilidades de transición, es decir las probabilidades de que teniendo cierto clima un día, al día siguiente se tenga otro o el mismo clima. Así, si indicamos con s a un día soleado y con n a un día nublado, tendremos:

$$P_{ss} = 0.7,$$

$$P_{sn} = 0.3,$$

$$P_{ns} = 0.6,$$

$$P_{nn} = 0.4.$$

Si acomodamos estos datos en una matriz, obtenemos la matriz de probabilidades de transición:

$$\mathbf{P} = \begin{bmatrix} 0.7 & 0.3 \\ 0.6 & 0.4 \end{bmatrix},$$

donde las filas indican el clima en el día, fila 1 soleado y fila 2 nublado, y las columnas el clima en el día siguiente, columna 1 soleado y columna 2 nublado.

Observamos que en esta matriz no sólo los coeficientes son no-negativos, sino que al sumarlos por filas obtenemos 1. Este es un ejemplo de cadena de Markov ya que:

1. Tenemos ciertos estados, en este caso s y n .
2. En cada momento estamos en uno de estos estados.

3. En el próximo momento volveremos a estar en ese u otro estado.
4. Pasamos de un estado a otro con cierta probabilidad, que sólo puede depender del estado inmediatamente anterior y esa probabilidad no cambia con el transcurso del tiempo.

4.4. Algoritmo Metropolis-Hastings

La referencia al algoritmo de Metropolis-Hastings corresponde a un término general que se utiliza para una familia de métodos de simulación, compuesta de algoritmos universales que generan cadenas de Markov con distribución estacionaria que corresponde al momento del posterior de interés, estos se derivan del algoritmo de Metrópolis.

El algoritmo de Metropolis, es una modificación de un paseo al azar que utiliza una regla de aceptación-rechazo para obtener convergencia de la cadena a una distribución específica. Este algoritmo consiste de los siguientes pasos:

1. Simular un punto inicial para el cual $\pi(\theta^0|Z) > 0$, a partir de de una distribución inicial $p_0(\theta)$.

Para $s = 1, \dots, n_{sim}$

2. Obtener una realización candidata ϑ a partir de una distribución de salto en el tiempo s , $J_s(\vartheta|\theta^{(s-1)})$. Esta distribución debe ser simétrica en el sentido de que $J_s(\theta^{(a)}|\theta^{(b)}) = J_s(\theta^{(b)}|\theta^{(a)})$, para todo $\theta^{(a)}, \theta^{(b)}, s$.

3. Calcular:

$$r = \frac{\pi(\vartheta|Z)}{\pi(\theta^{(s-1)}|Z)}.$$

4. Hacer:

$$\theta^{(s)} = \begin{cases} \vartheta, & \text{con probabilidad mín } \{1, r\} \\ \theta^{(s-1)}, & \text{en otro caso} \end{cases}$$

La regla de aceptación y rechazo del algoritmo anterior se puede interpretar como sigue: si el “salto” produce un valor para el que se aumenta la densidad posterior,

hacer $\theta^{(s)} = \vartheta$; si el “salto” no aumenta la densidad a posteriori, hacer $\theta^{(s)} = \vartheta$ con probabilidad r y $\theta^{(s)} = \theta^{(s-1)}$, si no. Esto puede ser visto como una versión estocástica de un algoritmo de búsqueda de moda por pasos.

Esta primera versión del algoritmo fue construida por Metropolis (1949) y más tarde, es generalizada por Hastings (1970), cuya generalización consiste esencialmente en que las reglas de salto, dadas por J_s no necesitan ser simétricas y el radio r es reemplazado por,

$$r = \frac{\pi(\vartheta|Z) / J_s(\vartheta|\theta^{(s-1)})}{\pi(\theta^{(s-1)}|Z) / J_s(\theta^{(s-1)}|\vartheta)}.$$

Al ser este algoritmo de aceptación-rechazo, la eficiencia en la generación de la cadena dependerá de las propiedades de la distribución de salto; una buena distribución de salto debería cumplir con las siguientes propiedades:

- Para cualquier θ , es fácil muestrear de $J(\vartheta|\theta)$.
- Es fácil calcular los cocientes de importancia r .
- Cada salto produce resultados a una distancia razonable en el espacio de parámetros.
- Los saltos no son rechazados muy frecuentemente.

En esta sección describiremos el algoritmo RWM (Random Walk Metrópolis) el cual pertenece a la clase más general de los algoritmos Metropolis-Hastings. El algoritmo RWM fue usado por primera vez para generar muestras de la distribución del posterior de los parámetros de un modelo DSGE, por Shorfheide (2000) y Otrok (2001). En este trabajo presentamos la versión del algoritmo expuesto en por An y Shorfheide (2007) (ver [2]), en la cual es utilizada como distribución de salto, la distribución normal y donde h es la función que se desea estimar.

Algoritmo Metropolis-Hastings

Inicialización:

- Denote la moda del posterior por $\tilde{\theta}$.
- Sea $\tilde{\Sigma}$ la inversa de la matriz hessiana calculada en $\tilde{\theta}$.

1. Generar $\theta^{(0)}$ desde $N(\tilde{\theta}, c^2\tilde{\Sigma})$ o dar un valor inicial específico.

Para $s = 1, \dots, n_{sim}$.

2. Generar ϑ desde $N(\theta^{(s-1)}, c^2\tilde{\Sigma})$.

3. Calcular:

$$r(\theta^{(s-1)}, \vartheta|Z) = \frac{\mathcal{L}(\vartheta|Z) \pi(\vartheta)}{\mathcal{L}(\theta^{(s-1)}|Z) \pi(\theta^{(s-1)})}.$$

4. Con probabilidad $\min\{1, r(\theta^{(s-1)}, \vartheta|Z)\}$ aceptar ϑ y hacer $\theta^{(s)} = \vartheta$, en otro caso, rechazar ϑ y hacer $\theta^{(s)} = \theta^{(s-1)}$.
5. Aproximar el valor esperado del posterior de la función $h(\theta)$ por:

$$\frac{1}{n_{sim}} \sum_{s=1}^{n_{sim}} h(\theta^{(s)}).$$

Observación 4.4.1. El escalar c utilizado en el algoritmo, permite re-escalar la inversa de la matriz hessiana $\tilde{\Sigma}$ de tal modo que los vectores generados ϑ sean aceptados en la mayoría de los casos, lo cual es esperado en este algoritmo.

CAPÍTULO 5

RESULTADOS

En este capítulo mostraremos los resultados dados por los algoritmos muestreo de importancia y Metropolis-Hastings para un modelo DSGE estándar. En principio presentamos este modelo y su solución.

5.1. Un modelo DSGE estándar

En esta sección presentamos el modelo esbozado por Uhlig (1995) (ver [12]), el cual fue estudiado por Hansens (1985) y es uno de los modelos de ciclo económico más representativo. El modelo consta de un número finito de agentes representativos caracterizados por una vida infinita, cada agente maximiza su utilidad esperada de por vida. En cada período el agente elige de forma óptima el consumo, las inversiones y el esfuerzo de la mano de obra, bajo la restricciones dada por sus ingresos.

En términos del planificador social, el problema es el de maximizar la utilidad total del agente dada por:

$$\text{máx } \mathbb{E}_t \sum_{t=0}^{\infty} \beta^t \left[\frac{C_t^{1-\eta} - 1}{1-\eta} - \mathcal{A}N_t \right],$$

donde β es el factor de descuento, C_t el consumo, η es el coeficiente de aversión al riesgo relativo, N_t es el número de horas trabajadas y \mathcal{A} es el parámetro que caracteriza la función de utilidad.

El problema de maximización se realiza bajo una serie de limitaciones. La primera

restricción es la condición de equilibrio para el mercado de bienes, a saber:

$$C_t + I_t = Y_t, \quad (5.1)$$

además en cada período el agente se enfrenta a otra restricción presupuestaria:

$$K_t = I_t + (1 - \delta)K_{t-1}, \quad (5.2)$$

y la siguiente restricción está dada a la función de producción. La cual se asume es la función de producción de Cobb-Douglas:

$$Y_t = Z_t K_{t-1}^\rho N_t^{1-\rho}. \quad (5.3)$$

donde I_t es la inversión, Y_t la producción, K_t el capital, Z_t el factor de productividad total, ρ la elasticidad de la producción con respecto al capital y δ es la tasa de depreciación del capital.

La última de estas restricciones está dada por la especificación de la dinámica del factor de productividad total, a través de la ecuación:

$$\log Z_t = (1 - \psi) \log \bar{Z} + \psi \log Z_{t-1} + \epsilon_t, \quad \epsilon_t \sim i.i.dN(0, \sigma^2). \quad (5.4)$$

5.1.1. Solución del modelo DSGE

El **lagrangiano** para el problema de optimización viene dado por:

$$\Lambda = \mathbb{E}_t \left\{ \sum_{t=0}^{\infty} \beta^t \left(\frac{C_t^{1-\eta} - 1}{1-\eta} - \mathcal{A}N_t \right) - \sum_{t=0}^{\infty} \beta^t \lambda_t (K_t - K_{t-1} + \delta K_{t-1} - Z_t K_{t-1}^\rho N_t^{1-\rho} + C_t) \right\}.$$

donde λ_t representa el multiplicador de Lagrange.

Realicemos el cálculo de las **condiciones de primer orden**:

$$\begin{aligned}
\frac{\partial \Lambda}{\partial C_t} = 0 &\implies \mathbb{E}_t \{ \beta^t C_t^{-\eta} - \beta^t \lambda_t \} = 0 \\
&\implies \mathbb{E}_t \{ \beta^t \lambda_t \} = \mathbb{E}_t \{ \beta^t C_t^{-\eta} \} \\
&\implies \lambda_t = C_t^{-\eta}.
\end{aligned} \tag{5.5}$$

$$\begin{aligned}
\frac{\partial \Lambda}{\partial N_t} = 0 &\implies \mathbb{E}_t \{ -\beta_t \mathcal{A} + \beta_t \lambda_t (1 - \rho) Z_t K_{t-1}^\rho N_t^{-\rho} \} = 0 \\
&\implies (1 - \rho) \lambda_t Z_t K_{t-1}^\rho N_t^{-\rho} = \mathcal{A}.
\end{aligned} \tag{5.6}$$

$$\begin{aligned}
\frac{\partial \Lambda}{\partial K_t} = 0 &\implies \mathbb{E}_t \left\{ -\beta^t \lambda_t + \beta^{t+1} \lambda_{t+1} (1 - \delta + \rho Z_{t+1} K_t^{\rho-1} N_{t+1}^{1-\rho}) \right\} = 0 \\
&\implies \lambda_t = \beta \mathbb{E}_t \left\{ \lambda_{t+1} (1 - \delta + \rho Z_{t+1} K_t^{\rho-1} N_{t+1}^{1-\rho}) \right\}.
\end{aligned} \tag{5.7}$$

$$\begin{aligned}
\frac{\partial \Lambda}{\partial \lambda_t} = 0 &\implies \mathbb{E}_t \left\{ -\beta^t \left(K_t - K_{t-1} + \delta K_{t-1} - Z_t K_{t-1}^\rho N_t^{1-\rho} + C_t \right) \right\} = 0 \\
&\implies K_t - K_{t-1} + \delta K_{t-1} - Z_t K_{t-1}^\rho N_t^{1-\rho} + C_t = 0
\end{aligned} \tag{5.8}$$

Sustituyendo la ecuación (5.5) en la ecuación (5.6), tenemos:

$$\begin{aligned}
\mathcal{A} &= C_t^{-\eta} (1 - \rho) Z_t K_{t-1}^\rho N_t^{-\rho} \\
\implies \mathcal{A} &= C_t^{-\eta} (1 - \rho) \frac{Y_t}{K_{t-1}^\rho N_t^{1-\rho}} K_{t-1}^\rho N_t^{-\rho}, && \text{por (5.3)} \\
\implies \mathcal{A} &= C_t^{-\eta} (1 - \rho) \frac{Y_t}{N_t}.
\end{aligned}$$

Igualando la ecuación (5.5) y (5.7), obtenemos:

$$\begin{aligned}
C_t^{-\eta} &= \beta \mathbb{E}_t \left\{ \lambda_{t+1} (1 - \delta + \rho Z_{t+1} K_t^{\rho-1} N_{t+1}^{1-\rho}) \right\} \\
\implies 1 &= \beta C_t^\eta \mathbb{E}_t \left\{ \lambda_{t+1} \left(1 - \delta + \rho \frac{Y_{t+1}}{K_t^\rho N_{t+1}^{1-\rho}} K_t^{\rho-1} N_{t+1}^{1-\rho} \right) \right\}, & \text{por (5.3)} \\
\implies 1 &= \beta C_t^\eta \mathbb{E}_t \left\{ \lambda_{t+1} \left(1 - \delta + \rho \frac{Y_{t+1}}{K_t} \right) \right\} \\
\implies 1 &= \beta C_t^\eta \mathbb{E}_t \left\{ C_{t+1}^{-\eta} \left(1 - \delta + \rho \frac{Y_{t+1}}{K_t} \right) \right\}, & \text{por (5.5)} \\
\implies 1 &= \beta \mathbb{E}_t \left[\left(\frac{C_t}{C_{t+1}} \right)^\eta R_{t+1} \right].
\end{aligned}$$

donde, $R_t = \rho \frac{Y_t}{K_{t-1}} + 1 - \delta$.

Finalmente, las condiciones de primer orden para el modelo son:

$$\mathcal{A} = C_t^{-\eta} (1 - \rho) \frac{Y_t}{N_t}, \quad (5.9)$$

$$1 = \beta \mathbb{E}_t \left[\left(\frac{C_t}{C_{t+1}} \right)^\eta R_{t+1} \right], \quad (5.10)$$

$$R_t = \rho \frac{Y_t}{K_{t-1}} + 1 - \delta, \quad (5.11)$$

$$K_t - K_{t-1} + \delta K_{t-1} - Z_t K_{t-1}^\rho N_t^{1-\rho} + C_t = 0, \quad (5.12)$$

Ahora, encontremos el **estado estacionario del sistema**; como en este estado las variables no dependen del tiempo, las ecuaciones de primer orden quedan:

$$\begin{aligned}
\mathcal{A} &= C^{-\eta} (1 - \rho) \frac{Y}{N}, \\
1 &= \beta R, \\
R &= \rho \frac{Y}{K} + 1 - \delta.
\end{aligned}$$

Como siguiente paso, procedemos a **log-linealizar** alrededor del estado estacionario las ecuaciones que caracterizan el equilibrio del sistema (ver [1]):

1. Para la ecuación (5.1), tenemos:

$$C \widehat{C}_t + I \widehat{I}_t = Y \widehat{Y}_t.$$

2. Para la ecuación (5.2):

$$K \widehat{K}_t = I \widehat{I}_t + (1 - \delta) K \widehat{K}_{t-1}.$$

3. Para la ecuación (5.3):

$$\widehat{Y}_t = \widehat{Z}_t + \rho \widehat{K}_{t-1} + (1 - \rho) \widehat{N}_t.$$

4. Para la ecuación (5.4):

$$\widehat{Z}_t = \psi \widehat{Z}_{t-1} + \epsilon_t.$$

5. Para la ecuación (5.9):

$$-\eta \widehat{C}_t + \widehat{Y}_t - \widehat{N}_t = 0.$$

6. Para la ecuación (5.10):

$$\eta \left(\widehat{C}_t - \widehat{C}_{t+1} \right) + \widehat{R}_{t+1} = 0,$$

así:

$$\mathbb{E}_t \left[\eta \left(\widehat{C}_t - \widehat{C}_{t+1} \right) + \widehat{R}_{t+1} \right] = 0.$$

7. Para la ecuación (5.11):

$$\rho \frac{Y}{K} \left(\widehat{Y}_t - \widehat{K}_{t-1} \right) = R \widehat{R}_t.$$

donde $\widehat{C}_t, \widehat{I}_t, \widehat{Y}_t, \widehat{K}_t, \widehat{Z}_t$, y \widehat{R}_t denotan la desviación logarítmica de C_t, I_t, Y_t, K_t, Z_t y R_t para el tiempo t , respectivamente.

Por simplicidad, escribimos estas ecuaciones como en el siguiente sistema:

$$\left. \begin{aligned}
 0 &= -I \widehat{I}_t - C \widehat{C}_t + Y \widehat{Y}_t \\
 0 &= I \widehat{I}_t - K \widehat{K}_t + (1 - \delta) K \widehat{K}_{t-1} \\
 0 &= \rho \widehat{K}_{t-1} - \widehat{Y}_t + (1 - \rho) \widehat{N}_t + \widehat{Z}_t \\
 0 &= -\eta \widehat{C}_t + \widehat{Y}_t - \widehat{N}_t \\
 0 &= -\rho \frac{Y}{K} \widehat{K}_{t-1} + \rho \frac{Y}{K} \widehat{Y}_t - R \widehat{R}_t \\
 0 &= \mathbb{E} \left[-\eta \widehat{C}_{t+1} + \widehat{R}_{t+1} + \eta \widehat{C}_t \right] \\
 \widehat{Z}_t &= \psi \widehat{Z}_{t-1} + \epsilon_t.
 \end{aligned} \right\} . \quad (5.13)$$

El sistema (5.13) se puede escribir en forma compacta como:

$$\left. \begin{aligned}
 0 &= AAx_t^* + BBx_{t-1}^* + CCy_t^* + DDz_t^* \\
 0 &= \mathbb{E}_t[FFx_{t+1}^* + GGx_t^* + HHx_{t-1}^* + JJy_{t+1}^* + KKy_t^* + LLz_{t+1}^* + MMz_t^*] \\
 z_{t+1}^* &= NNz_t^* + \epsilon_{t+1}, \quad E_t[\epsilon_{t+1}]
 \end{aligned} \right\} . \quad (5.14)$$

donde $x_t^* = \widehat{K}_t$, $y_t^* = \left[\widehat{C}_t \quad \widehat{Y}_t \quad \widehat{N}_t \quad \widehat{R}_t \quad \widehat{I}_t \right]'$, $z_t^* = \widehat{Z}_t$ y

$$AA = \begin{bmatrix} 0 & -K & 0 & 0 & 0 \end{bmatrix}', \quad BB = \begin{bmatrix} 0 & (1 - \delta)K & \rho & 0 & -\rho \frac{Y}{K} \end{bmatrix}',$$

$$CC = \begin{bmatrix} -C & Y & 0 & 0 & -I \\ 0 & 0 & 0 & 0 & I \\ 0 & -1 & 1 - \rho & 0 & 0 \\ -\eta & 1 & -1 & 0 & 0 \\ 0 & \rho \frac{Y}{K} & 0 & -R & 0 \end{bmatrix},$$

$$DD = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \end{bmatrix}', \quad FF = [0], \quad GG = [0], \quad HH = [0],$$

$$JJ = \begin{bmatrix} -\eta & 0 & 0 & 1 & 0 \end{bmatrix}, \quad KK = \begin{bmatrix} \eta & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$LL = [0], \quad MM = [0], \quad \text{y} \quad NN = [\psi].$$

Para llevar a cabo el análisis y solución del modelo, nos valdremos del método de coeficientes indeterminados (ver [12]). Así, empleando el código desarrollado en MATLAB por Uhlig, obtenemos el siguiente sistema:

$$\left. \begin{aligned} x_t^* &= PP x_{t-1}^* + QQ z_t^* \\ y_t^* &= RR x_{t-1}^* + SS z_t^* \end{aligned} \right\}, \quad (5.15)$$

Luego, a partir del sistema (5.15) obtenemos una representación de espacio-estado de la forma de las ecuaciones (2.1) y (2.2). Donde:

$$x_t = \begin{bmatrix} x_t^* & y_t^{*'} & z_t^* \end{bmatrix}' = \begin{bmatrix} \hat{K}_t & \hat{C}_t & \hat{Y}_t & \hat{N}_t & \hat{R}_t & \hat{I}_t & \hat{Z}_t \end{bmatrix}',$$

y

$$z_t = \log \left(\frac{Y_t}{C_t} \right) - \log \left(\frac{Y}{C} \right).$$

de la siguiente manera:

Dado que: $x_t^* = \hat{K}_t$, $y_t^* = \left[\hat{C}_t \ \hat{Y}_t \ \hat{N}_t \ \hat{R}_t \ \hat{I}_t \right]'$ y $z_t^* = \hat{Z}_t$. Entonces, el sistema (5.15), queda:

$$\left. \begin{aligned} \hat{K}_t &= PP \hat{K}_{t-1} + QQ \hat{Z}_t \\ \left[\hat{C}_t \ \hat{Y}_t \ \hat{N}_t \ \hat{R}_t \ \hat{I}_t \right]' &= RR \hat{K}_{t-1} + SS \hat{Z}_t \end{aligned} \right\}, \quad (5.16)$$

donde $\hat{Z}_t = NN\hat{Z}_{t-1} + \varepsilon_t$.

Luego, el sistema (5.16) lo podemos representar por:

$$\begin{bmatrix} \hat{K}_t \\ \hat{C}_t \\ \hat{Y}_t \\ \hat{N}_t \\ \hat{R}_t \\ \hat{I}_t \\ \hat{Z}_t \end{bmatrix} = \begin{bmatrix} PP & 0 & 0 & 0 & 0 & 0 & QQNN \\ RR_{11} & 0 & 0 & 0 & 0 & 0 & SS_{11}NN \\ RR_{21} & 0 & 0 & 0 & 0 & 0 & SS_{21}NN \\ RR_{31} & 0 & 0 & 0 & 0 & 0 & SS_{31}NN \\ RR_{41} & 0 & 0 & 0 & 0 & 0 & SS_{41}NN \\ RR_{51} & 0 & 0 & 0 & 0 & 0 & SS_{51}NN \\ 0 & 0 & 0 & 0 & 0 & 0 & NN \end{bmatrix} \begin{bmatrix} \hat{K}_{t-1} \\ \hat{C}_{t-1} \\ \hat{Y}_{t-1} \\ \hat{N}_{t-1} \\ \hat{R}_{t-1} \\ \hat{I}_{t-1} \\ \hat{Z}_{t-1} \end{bmatrix} + \begin{bmatrix} QQ \\ SS_{11} \\ SS_{21} \\ SS_{31} \\ SS_{41} \\ SS_{51} \\ 1 \end{bmatrix} \varepsilon_t.$$

Así:

$$A = \begin{bmatrix} PP & 0 & 0 & 0 & 0 & 0 & QQNN \\ RR_{11} & 0 & 0 & 0 & 0 & 0 & SS_{11}NN \\ RR_{21} & 0 & 0 & 0 & 0 & 0 & SS_{21}NN \\ RR_{31} & 0 & 0 & 0 & 0 & 0 & SS_{31}NN \\ RR_{41} & 0 & 0 & 0 & 0 & 0 & SS_{41}NN \\ RR_{51} & 0 & 0 & 0 & 0 & 0 & SS_{51}NN \\ 0 & 0 & 0 & 0 & 0 & 0 & NN \end{bmatrix} \quad \text{y} \quad B = \begin{bmatrix} QQ \\ SS_{11} \\ SS_{21} \\ SS_{31} \\ SS_{41} \\ SS_{51} \\ 1 \end{bmatrix}.$$

Por otra parte:

$$z_t = \log\left(\frac{Y_t}{C_t}\right) - \log\left(\frac{Y}{C}\right) = \log(Y_t) - \log(C_t) - \log(Y) + \log(C) = \widehat{Y}_t - \widehat{C}_t,$$

pero,

$$\widehat{Y}_t - \widehat{C}_t = \begin{bmatrix} 0 & -1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \widehat{K}_t \\ \widehat{C}_t \\ \widehat{Y}_t \\ \widehat{N}_t \\ \widehat{R}_t \\ \widehat{I}_t \\ \widehat{Z}_t \end{bmatrix},$$

Por tanto, resulta que:

$$C = \begin{bmatrix} 0 & -1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

De esta forma, hemos encontrado las matrices A , B y C de la representación espacio-estado del modelo. Pero ya que:

$$\begin{aligned}\widehat{K}_t &= PP \widehat{K}_{t-1} + QQNN \widehat{Z}_{t-1} + QQ U_t \\ \implies \widehat{K}_t &= PP \widehat{K}_{t-1} + QQ [NN \widehat{Z}_{t-1} + U_t].\end{aligned}\quad (5.17)$$

Pero, por otro lado:

$$\begin{aligned}\widehat{K}_t &= PP \widehat{K}_{t-1} + QQ \widehat{Z}_t \\ \implies \widehat{K}_t &= PP \widehat{K}_{t-1} + QQ [NN Z_{t-1} + \varepsilon_t].\end{aligned}\quad (5.18)$$

De (5.17) y (5.18), tenemos que: $U_t = \varepsilon_t$.

Por tanto, si definimos:

$$B^* = \sigma B,$$

donde σ es la desviación estándar de ε_t (y así de U_t) la cual es conocida. Podemos facilitar los cálculos tomando como nuestra representación espacio-estado:

$$x_t = Ax_{t-1} + B^* U_t^*, \quad \text{donde } U_t^* \sim N(0, 1) \quad (5.19)$$

$$z_t = Cx_t. \quad (5.20)$$

5.2. Resultados

Dada la representación espacio-estado del modelo a través de las ecuaciones (5.19) y (5.20), empleamos el algoritmo de filtro de Kalman para encontrar las estimaciones de la función de verosimilitud. Además de esto, empleamos la metodología expuesta en el capítulo 3 para hallar la matriz de información en un θ dado. Así, luego de disponer de la moda del posterior, a través de los algoritmos de muestreo de importancia y Metropolis-Hastings se obtuvieron muestras de la densidad a posteriori $\pi(\theta|Z)$ y con estos, dadas sus técnicas expuestas en el capítulo 4, fue posible calcular cantidades de interés como la media y la desviación estándar. En la implementación de estos algoritmos fueron empleados datos extraídos por trimestres del producto

interno bruto (PIB) y el consumo total en Venezuela entre los años 1984-2010, desde el apartado de agregados macroeconómicos del Banco Central de Venezuela, donde se considera como vector de observaciones el logaritmo del cociente entre el PIB y el consumo; los cuales se muestran en la siguiente gráfica:

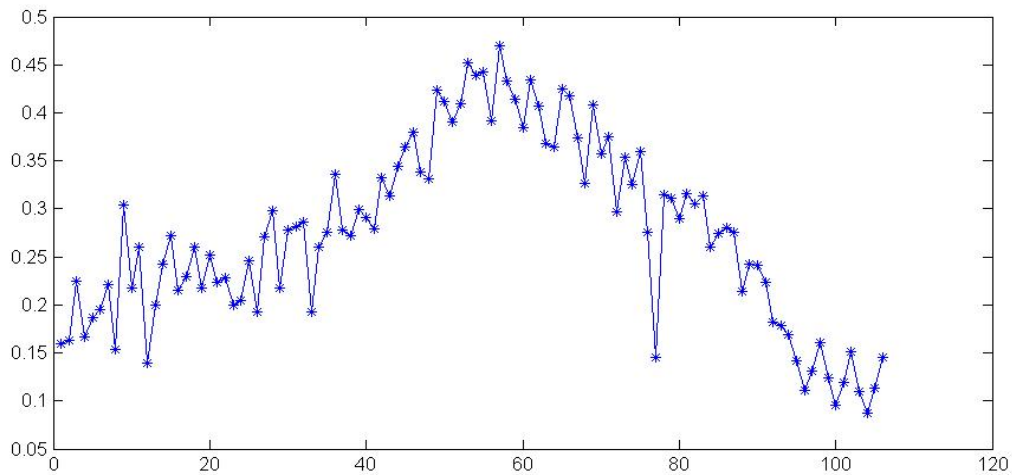


Figura 5.1: Datos: logaritmo del cociente entre el PIB y el consumo.

En las siguientes tablas se muestran los distintos prior utilizados para el modelo de ciclo económico real de Hansens, donde se considera como vector de parámetros profundos $\theta = \left[\beta \quad \eta \quad \delta \quad \rho \quad \psi \quad \sigma^2 \right]$.

Parámetro	Prior	Soporte
β	uniforme	$(0.75, 1)$
η	uniforme	$(0, 2)$
δ	uniforme	$(0, 0.1)$
ρ	uniforme	$(0, 1)$
ψ	uniforme	$(0, 1)$
σ^2	uniforme	$(0, 1)$

TABLA 5.1: PRIOR UNIFORMES PARA θ .

Parámetro	Prior	Media	Desviación estándar	Soporte
β	gamma inversa	0.99	0.1	(0.75, 1)
η	normal	1	0.25	(0, 2)
δ	beta	0.024	0.1	(0, 0.1)
ρ	normal	0.4	0.2	(0, 1)
ψ	beta	0.7	0.2	(0, 1)
σ^2	gamma inversa	0.02	∞	(0, 1)

TABLA 5.2: PRIOR PARA θ .

5.2.1. Resultados: Muestreo de Importancia

En esta sección presentamos los resultados arrojados por la implementación del algoritmo muestreo de importancia, donde fue empleado como el escalar que permite re-escalar la inversa de la matriz de información, $c = 0.002$ y fueron producidas 10.000 iteraciones. En la tabla 5.3 se muestran estos resultados con la utilización de los prior presentados en la tabla 5.1, donde la primera columna representa los parámetros del modelo; la segunda, la media; la tercera, la desviación estándar (Std) y la cuarta columna, la moda.

Parámetro	Media	Std	Moda
β	0.9987	0.0012	0.9999
η	1.9547	0.1260	1.9989
δ	0.0924	0.0090	0.0875
ρ	0.4165	0.0261	0.4010
ψ	0.9765	0.0004	0.9769
σ^2	0.0088	0.0007	0.0083

TABLA 5.3: RESULTADOS MUESTREO DE IMPORTANCIA CON PRIOR DE LA TABLA 5.1.

En la figura 5.2 se muestran los histogramas de frecuencia para β , η , δ , ρ , ψ y σ^2 .

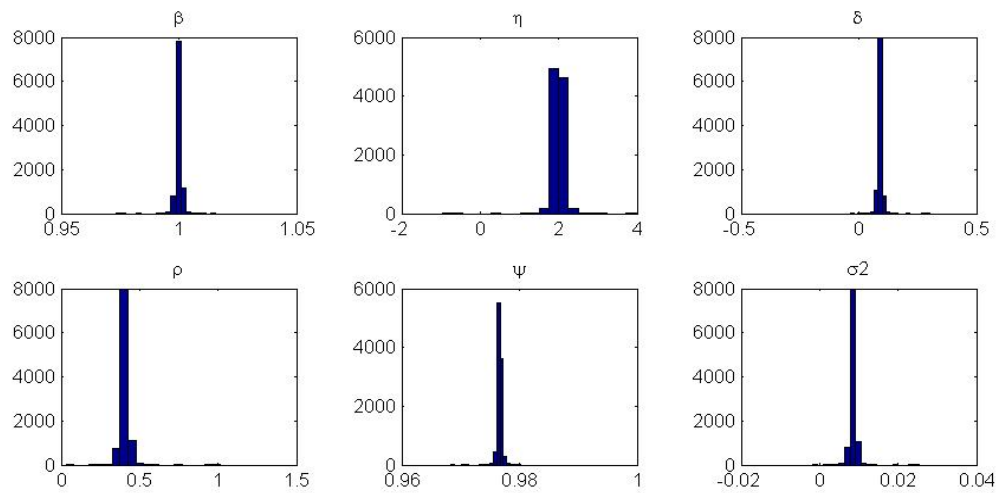


Figura 5.2: Histogramas de frecuencias del algoritmo muestreo de importancia con los prior de la tabla 5.1.

De igual forma, en la tabla 5.4 se muestran los mismos resultados, haciendo uso de los prior presentados en la tabla 5.2 y en la figura 5.3 se pueden observar los histograma de frecuencia para los parámetros.

Parámetro	Media	Std	Moda
β	0.9988	0.0008	0.9997
η	1.4240	0.1218	1.7328
δ	0.0965	0.0023	0.0923
ρ	0.4226	0.0145	0.4090
ψ	0.9871	0.0028	0.9819
σ^2	0.0210	0.0104	0.0121

TABLA 5.4: RESULTADOS MUESTREO DE IMPORTANCIA CON PRIOR DE LA TABLA 5.2.

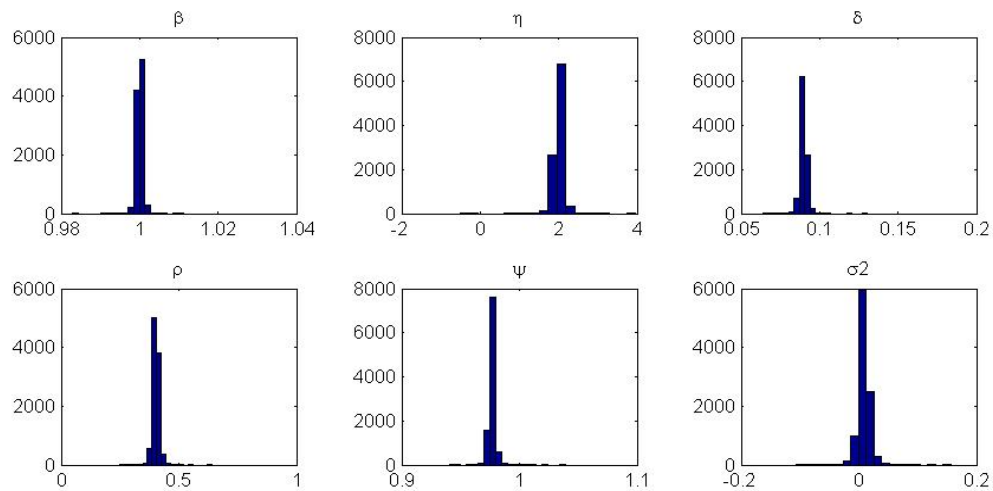


Figura 5.3: Histogramas de frecuencias del algoritmo muestreo de importancia con los prior de la tabla 5.2.

Luego de obtener estos resultados, los parámetros β , δ , ρ son calibrados usando los resultados en [5], y el resto de los parámetros siguen los prior presentados en la tabla 5.2. Las calibraciones son: $\beta = 0.99$, $\delta = 0.024$ y $\rho = 0.4$.

Los resultados a través de esta implementación se muestran en la tabla 5.5:

Parámetro	Media	Std	Moda
η	1.9557	0.1218	1.9999
ψ	0.9766	0.0004	0.9766
σ^2	0.0087	0.0007	0.0084

TABLA 5.5: RESULTADOS MUESTREO DE IMPORTANCIA CON PARÁMETROS CALIBRADOS Y PRIOR DE LA TABLA 5.2.

En la figura 5.4, son mostrados los histogramas para los parámetros η , ψ y σ^2 , en este caso.

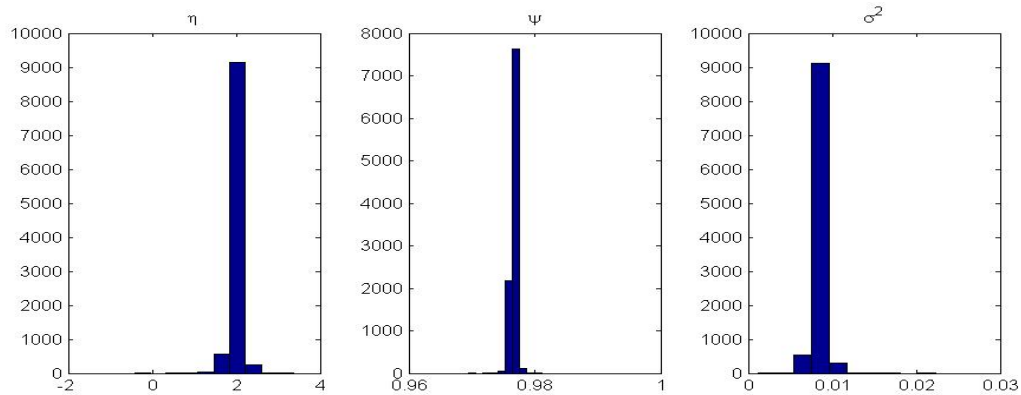


Figura 5.4: Histogramas de frecuencias del algoritmo muestreo de importancia con parámetros calibrados y los prior de la tabla 5.2.

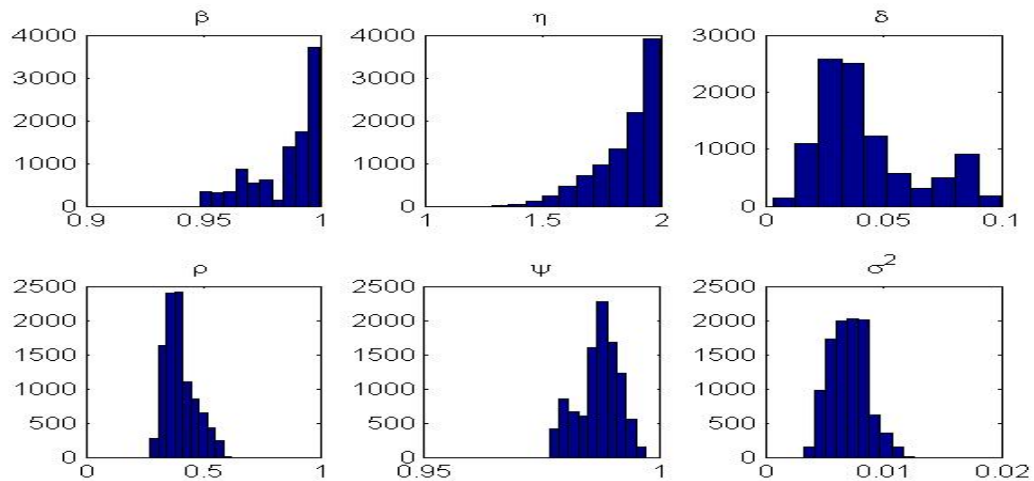
5.2.2. Resultados: Metropolis-Hastings

Mostraremos en esta sección los resultados del algoritmo Metropolis-Hastings, donde fue utilizado como el escalar que permite re-escalar la inversa de la matriz de información evaluada, $c = 0.004$ y como número de simulaciones $n_{sim} = 10.000$.

En la tabla 5.6 se muestran estos resultados con la utilización de los prior a los que se refieren la tabla 5.1 y en la figura 5.5 son mostrados los histogramas de frecuencia para los parámetros β , η , δ , ρ , ψ y σ^2 .

Parámetro	Media	Std	Moda
β	0.9857	0.0140	0.9540
η	1.8562	0.1318	1.9977
δ	0.0425	0.0210	0.0364
ρ	0.4002	0.0639	0.5247
ψ	0.9868	0.0044	0.9836
σ^2	0.0071	0.0015	0.0097

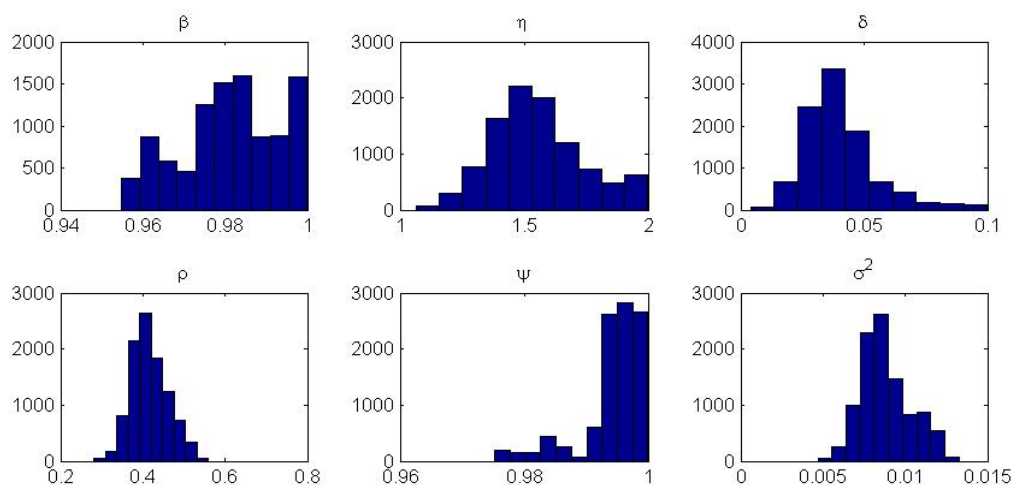
TABLA 5.6: RESULTADOS METROPOLIS-HASTINGS CON PRIOR DE LA TABLA 5.1.

**Figura 5.5:** Histogramas de frecuencias del algoritmo Metropolis-Hastings con los prior de la tabla 5.1.

De igual forma, en la tabla 5.7 se muestran los resultados con la utilización de los prior presentados en la tabla 5.2 y en la figura 5.6 se pueden observar los histogramas de frecuencia para β , η , δ , ρ , ψ y σ^2 .

Parámetro	Media	Std	Moda
β	0.9812	0.0120	0.9798
η	1.5531	0.1859	1.3346
δ	0.0401	0.0148	0.0250
ρ	0.4161	0.0442	0.4083
ψ	0.9943	0.0050	0.9988
σ^2	0.0088	0.0015	0.0078

TABLA 5.7: RESULTADOS METROPOLIS-HASTINGS CON PRIOR DE LA TABLA 5.2.

**Figura 5.6:** Histogramas de frecuencias del algoritmo Metropolis-Hastings con los prior de la tabla 5.2.

Siguiendo con las mismas ideas expuestas, fueron utilizadas de nuevo las calibraciones para los parámetros β , δ , ρ , con $\beta = 0.99$, $\delta = 0.024$ y $\rho = 0.4$, usando los resultados en [5] y el resto de los parámetros conservando los prior presentados en la tabla 5.2. Los resultados se resumen en la tabla 5.8.

Parámetro	Media	Std	Moda
η	1.4064	0.2926	1.1349
ψ	0.9905	0.0064	0.9967
σ^2	0.0053	0.0014	0.0046

TABLA 5.8: RESULTADOS METROPOLIS-HASTINGS CON PARÁMETROS CALIBRADOS Y PRIOR DE LA TABLA 5.2.

En la figura 5.7, son mostrados los histogramas para los parámetros η , ψ y σ^2 .

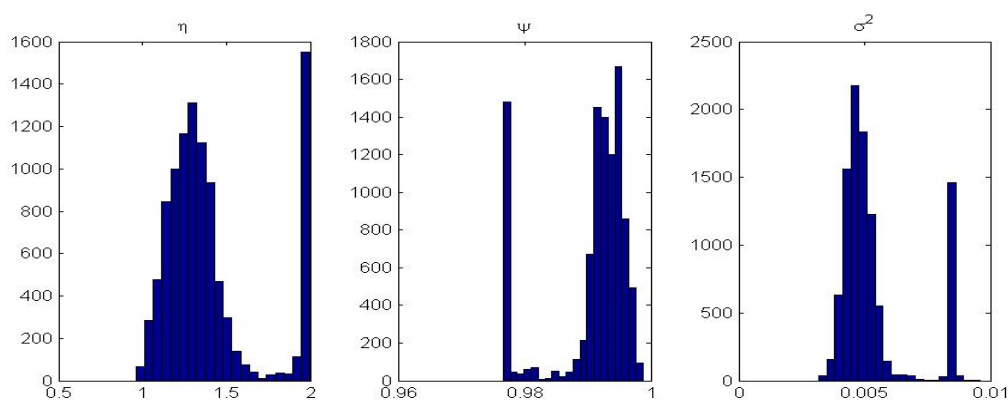


Figura 5.7: Histogramas de frecuencias del algoritmo Metropolis-Hastings con parámetros calibrados y los prior de la tabla 5.2.

5.2.3. Resultados adicionales

Un resultado importante del cálculo de la matriz de información es que para estimadores obtenidos utilizando el método de máxima verosimilitud, la matriz de covarianza de estos, esta dada por la inversa de la matriz de información. Este resultado esta dado de la siguiente proposición (ver [4]):

Proposición 5.2.1. *Sea $\hat{\theta}$ el estimador de máxima verosimilitud de θ , obtenido por maximizar la función log-verosimilitud. Entonces, cuando $n \rightarrow \infty$,*

$$\sqrt{n} (\hat{\theta} - \theta) \rightarrow N(0, \mathfrak{I}_{\theta}^{-1}),$$

donde \mathfrak{J}_θ es la matriz de información asintótica.

Ya que en los resultados de las implementaciones de los algoritmos muestreo de importancia y Metropolis-Hastings hemos obtenido la moda, el cual es el estimador que maximiza el posterior, y en el caso de la utilización de prior uniformes para los parámetros β , η , δ , ρ , ψ y σ^2 (ver tabla 5.1), este estimador resulta el estimador de máxima verosimilitud. Si denotamos la moda obtenida de la implementación del algoritmo muestreo de importancia a través de la utilización de prior uniformes por $\hat{\theta}_1$ y análogamente, denotamos por $\hat{\theta}_2$ a la moda obtenida del algoritmo Metropolis-Hastings a través de la utilización de prior uniformes (ver tablas 5.3 y 5.6); luego de obtener las evaluaciones de la matriz de información en estas modas, de sus diagonales principales puede ser extraída la varianza de cada parámetro.

En la siguiente tabla mostramos estos resultados, donde la primera columna representa los parámetros del modelo; la segunda, la desviación estándar de estos parámetros obtenidos de la diagonal principal de la matriz información inversa evaluada en $\hat{\theta}_1$ (Std_1) y en la tercera columna, la desviación estándar obtenida de la diagonal principal de la matriz información inversa evaluada en $\hat{\theta}_2$ (Std_2).

Parámetro	Std ₁	Std ₂
β	0.3584	1.3363
η	39.1361	67.3169
δ	2.7472	0.4424
ρ	8.0960	3.5220
ψ	0.1168	0.9564
σ^2	0.2231	0.0433

TABLA 5.9: DESVIACIONES ESTÁNDAR DE LOS PARÁMETROS A TRAVÉS DE LA INVERSA DE LA MATRIZ DE INFORMACIÓN.

Conclusión

En los resultados de la implementación de los algoritmos muestreo de importancia y Metropolis-Hastings no observamos una diferencia tan significativa como la que era esperada, teniendo en cuenta que el algoritmo Metropolis-Hastings se considera como un algoritmo de simulación más eficaz, esto dado que aunque en él a todos los valores les son asignados pesos por igual, no todos los candidatos son aceptados, debido a que posee una regla de aceptación y rechazo que permite tomar un valor si en este aumenta la densidad posterior y si por el contrario esta no aumenta, con cierta probabilidad se acepta o se rechaza. Mientras que en el algoritmo muestreo de importancia es corregido el hecho de que la función de importancia difiera del posterior, por los diferentes pesos que le son asignados a sus valores generados; más sin embargo, es de hacer notar que la escogencia del escalar c que permite re-escalar la inversa de la matriz de información resulta más crítica en este algoritmo, pues una escogencia errada puede causar que los vectores generados desde la función de importancia estén frecuentemente fuera del soporte y la simulación resulte ineficiente.

APÉNDICE A

A.1. Teorema de la función implícita

Teorema A.1.1. *Sea $A \subset \mathbb{R}^{k+n}$ un conjunto abierto, sea $f : A \rightarrow \mathbb{R}^n$ de clase C^r . Escribimos a f en la forma $f(x, y)$ para $x \in \mathbb{R}^k$ e $y \in \mathbb{R}^n$. Supongamos que $(x_0, y_0) \in A$ es tal que $f(x_0, y_0) = 0$ y $\det \frac{\partial f}{\partial y}(x_0, y_0) \neq 0$. Entonces:*

Existe $B \subset \mathbb{R}^k$ vecindad de x_0 y una única función continua $g : B \rightarrow \mathbb{R}^n$ tal que $g(x_0) = y_0$ y $f(x, g(x)) = 0$, $\forall x \in B$, donde la función g es de clase C^r . Además, para $x \in B$:

$$\frac{\partial f}{\partial x}(x, g(x)) + \frac{\partial f}{\partial y}(x, g(x)) \cdot Dg(x) = 0$$

Así, como la matriz $\frac{\partial f}{\partial y}$ de orden $n \times n$ es no-singular en el punto $(x_0, g(x_0))$, se cumple que:

$$Dg(x_0) = - \left[\frac{\partial f}{\partial y}(x_0, g(x_0)) \right]^{-1} \cdot \frac{\partial f}{\partial x}(x_0, g(x_0)).$$

A.2. Producto Kronecker

Definición A.2.1. *Sea $A = \{a_{ij}\}$ una matriz de orden $m \times n$ y $B = \{b_{ij}\}$ una matriz de orden $p \times q$. Se define el productor Kronecker entre A y B , el cual es denotado por $A \otimes B$, como la matriz de orden $mp \times nq$, dada por:*

$$A \otimes B = \begin{pmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{m2}B \end{pmatrix}$$

Propiedades del producto Kronecker:

1. $k \otimes A = A \otimes k = k A, \forall k \in \mathbb{R}, \forall A \in M_{m \times n}$.
2. $0 \otimes A = A \otimes 0 = 0, \forall A \in M_{m \times n}$.
3. $I_m \otimes I_p = I_{mp}$.
4. $(A + B) \otimes C = (A \otimes C) + (B \otimes C), \forall A, B \in M_{m \times n}$ y $\forall C \in M_{p \times q}$.
5. $(A \otimes B)' = A' \otimes B', \forall A \in M_{m \times n}, \forall B \in M_{p \times q}$.
6. $A \otimes (B \otimes C) = (A \otimes B) \otimes C, \forall A \in M_{m \times n}, \forall B \in M_{r \times s}$ y $\forall C \in M_{p \times q}$.
7. $(A \otimes B)(C \otimes D) = (AC) \otimes (BD), \forall A \in M_{m \times n}, \forall B \in M_{p \times q}, \forall C \in M_{n \times u}$ y $\forall D \in M_{q \times v}$.
8. $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}, \forall A \in M_{m \times m}$ y $\forall B \in M_{p \times p}$ matrices no singulares.

A.3. Operador vec

Definición A.3.1. Sea $A = a_{ij}$ una matriz de orden $m \times n$. Se define $vec(A)$ como el vector columna de dimensión mn formado de la siguiente manera:

$$vec(A) = \begin{pmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{12} \\ a_{22} \\ \vdots \\ a_{mn} \end{pmatrix}.$$

Propiedades del operador vec:

1. $vec(a') = vec(a) = a$, para cualquier vector columna a .
2. $vec(cA) = c vec(A), \forall c \in \mathbb{R}$ y $\forall A \in M_{m \times n}$.
3. $vec(A + B) = vec(A) + vec(B), \forall A, B \in M_{m \times n}$.

4. $\text{vec}(ABC) = (C' \otimes A) \text{vec}(B)$, $A \in M_{m \times n}$, $\forall \in M_{n \times p}$ y $\forall C \in M_{p \times q}$.
5. $\text{vec}(AB) = (B' \otimes I_m) \text{vec}(A)$, $A \in M_{m \times n}$ y $\forall \in M_{n \times p}$.
6. $\text{vec}(AB) = (B' \otimes A) \text{vec}(I_n)$, $A \in M_{m \times n}$ y $\forall \in M_{n \times p}$.

A.4. Derivadas de las ecuaciones del filtro de Kalman

Teorema A.4.1. *Sea $U : S \rightarrow \mathbb{R}^{m \times r}$ y $V : S \rightarrow \mathbb{R}^{r \times p}$ dos matrices de funciones definidas y diferenciables en un conjunto abierto $S \subset \mathbb{R}^{n \times q}$. Entonces, UV es diferenciable en S y la matriz jacobiana es la matriz de orden $mp \times nq$, tal que:*

$$\frac{\partial \text{vec}(UV)}{\partial (\text{vec}(X))'} = (V' \otimes I_m) \frac{\partial \text{vec}(U)}{\partial (\text{vec}(X))'} + (I_p \otimes U) \frac{\partial \text{vec}(V)}{\partial (\text{vec}(X))'}.$$

Haciendo uso de este teorema, y de las propiedades del productor Kronecker y del operador vec , las **ecuaciones derivadas del Filtro de Kalman** son las siguientes:

- $\frac{\partial}{\partial \phi'} (x_{t|t-1}) = \frac{\partial}{\partial \phi'} (Ax_{t-1|t-1})$

$$= \left(x'_{t-1|t-1} \otimes I_m \right) \frac{\partial}{\partial \phi'} \text{vec}(A) + A \frac{\partial}{\partial \phi'} (x_{t-1|t-1}).$$
- $\frac{\partial}{\partial \phi'} \text{vec}(\Sigma_{t|t-1}) = \frac{\partial}{\partial \phi'} \text{vec}(A\Sigma_{t-1|t-1}A' + BQB')$

$$= \frac{\partial}{\partial \phi'} \text{vec}(A\Sigma_{t-1|t-1}A') + \frac{\partial}{\partial \phi'} \text{vec}(BQB')$$

$$= (A \otimes I_m) \frac{\partial}{\partial \phi'} \text{vec}(A\Sigma_{t-1|t-1}) + (I_m \otimes (A\Sigma_{t-1|t-1})) \frac{\partial}{\partial \phi'} (A')$$

$$+ (B \otimes I_m) \frac{\partial}{\partial \phi'} \text{vec}(BQ) + (I_m \otimes (BQ)) \frac{\partial}{\partial \phi'} (B')$$

$$= (A \otimes I_m) \left[\left((\Sigma_{t-1|t-1})' \otimes I_m \right) \frac{\partial}{\partial \phi'} \text{vec}(A) + (I_m \otimes A) \frac{\partial}{\partial \phi'} \text{vec}(\Sigma_{t-1|t-1}) \right]$$

$$+ (I_m \otimes (A\Sigma_{t-1|t-1})) \frac{\partial}{\partial \phi'} (A') + (BQ' \otimes I_m) \frac{\partial}{\partial \phi'} \text{vec}(B) + (I_m \otimes (BQ)) \frac{\partial}{\partial \phi'} (B').$$

- $\frac{\partial}{\partial \phi'} (\epsilon_t) = \frac{\partial}{\partial \phi'} (z_t - Cx_{t|t-1})$

$$= \frac{\partial}{\partial \phi'} (z_t) - \frac{\partial}{\partial \phi'} (Cx_{t|t-1})$$

$$= -C \frac{\partial}{\partial \phi'} (x_{t|t-1}).$$
- $\frac{\partial}{\partial \phi'} \text{vec}(\Omega_{t|t-1}) = \frac{\partial}{\partial \phi'} \text{vec}(C\Sigma_{t|t-1}C' + R)$

$$= \frac{\partial}{\partial \phi'} \text{vec}(C\Sigma_{t|t-1}C') + \frac{\partial}{\partial \phi'} \text{vec}(R)$$

$$= (C \otimes I_p) \frac{\partial}{\partial \phi'} \text{vec}(C\Sigma_{t|t-1}) + (I_p \otimes (C\Sigma_{t|t-1})) \frac{\partial}{\partial \phi'} \text{vec}(C')$$

$$= (C \otimes I_p) \left[\left((\Sigma_{t|t-1})' \otimes I_p \right) \frac{\partial}{\partial \phi'} \text{vec}(C) + (I_m \otimes C) \frac{\partial}{\partial \phi'} \text{vec}(\Sigma_{t|t-1}) \right]$$

$$= (C \otimes I_p) (I_m \otimes C) \frac{\partial}{\partial \phi'} \text{vec}(\Sigma_{t|t-1})$$

$$= (C \otimes C) \frac{\partial}{\partial \phi'} \text{vec}(\Sigma_{t|t-1}).$$
- $\frac{\partial}{\partial \phi'} \text{vec}(\Omega_{t|t-1}^{-1}) = - \left((\Omega_{t|t-1}^{-1})' \otimes \Omega_{t|t-1}^{-1} \right) \frac{\partial}{\partial \phi'} \text{vec}(\Omega_{t|t-1}).$
- $\frac{\partial}{\partial \phi'} (x_{t|t}) = \frac{\partial}{\partial \phi'} \left(x_{t|t-1} + \Sigma_{t|t-1}C'\Omega_{t|t-1}^{-1}\epsilon_t \right)$

$$= \frac{\partial}{\partial \phi'} (x_{t|t-1}) + \frac{\partial}{\partial \phi'} \left(\Sigma_{t|t-1}C'\Omega_{t|t-1}^{-1}\epsilon_t \right)$$

$$\begin{aligned}
&= \frac{\partial}{\partial \phi'} (x_{t|t-1}) + \left(\left(\Omega_{t|t-1}^{-1} \epsilon_t \right)' \otimes I_m \right) \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1} C') + (\Sigma_{t|t-1} C') \frac{\partial}{\partial \phi'} \text{vec} \left(\Omega_{t|t-1}^{-1} \epsilon_t \right) \\
&= \frac{\partial}{\partial \phi'} (x_{t|t-1}) + \left(\left(\Omega_{t|t-1}^{-1} \epsilon_t \right)' \otimes I_m \right) (C \otimes I_m) \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) \\
&\quad + (\Sigma_{t|t-1} C') \left[(\epsilon_t' \otimes I_p) \frac{\partial}{\partial \phi'} \text{vec} (\Omega_{t|t-1}^{-1}) + \Omega_{t|t-1}^{-1} \frac{\partial}{\partial \phi'} \text{vec} (\epsilon_t) \right].
\end{aligned}$$

- $\frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t}) = \frac{\partial}{\partial \phi'} \text{vec} \left(\Sigma_{t|t-1} - \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} C \Sigma_{t|t-1} \right)$

$$\begin{aligned}
&= \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) - \frac{\partial}{\partial \phi'} \text{vec} \left(\Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} C \Sigma_{t|t-1} \right) \\
&= \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) - \left((C \Sigma_{t|t-1})' \otimes I_m \right) \frac{\partial}{\partial \phi'} \text{vec} \left(\Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} \right) \\
&\quad - \left(I_m \otimes \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} \right) \frac{\partial}{\partial \phi'} \text{vec} (C \Sigma_{t|t-1}) \\
&= \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) - \left((C \Sigma_{t|t-1})' \otimes I_m \right) \left[\left(\left(\Omega_{t|t-1}^{-1} \right)' C \otimes I_m \right) \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}) \right. \\
&\quad \left. + (I_p \otimes \Sigma_{t|t-1} C') \frac{\partial}{\partial \phi'} \text{vec} \left(\Omega_{t|t-1}^{-1} \right) \right] - \left(I_m \otimes \Sigma_{t|t-1} C' \Omega_{t|t-1}^{-1} \right) (I_m \otimes C) \frac{\partial}{\partial \phi'} \text{vec} (\Sigma_{t|t-1}).
\end{aligned}$$

APÉNDICE B

CÓDIGOS DESARROLLADOS EN MATLAB

B.1. Cálculo de la matriz de información y filtro de Kalman

B.1.1. gradiente.m

```
1 %La función gradiente calcula el gradiente numérico de la función
2 %que mapea el vector de parámetros profundos en el vector de
3 %parámetros de la forma reducida (representación espacio-estado)
4 %para el Modelo del Ciclo Económico Real de Hansens.
5 %Donde:
6 %Theta_num es un vector de entradas numéricas y A, B son
7 %las matrices de la representación espacio-estado del
8 %modelo en dicho vector.
9
10 function [H]=gradiente(A,B,Theta_num)
11
12 %% ==Definición de las expresiones simbólicas requeridas==
13
14 load Theta.mat %Expresiones simbólicas de los parámetros profundos
15
16 %Cargando las matrices simbólicas gamma0, gamma1, gamma2, gamma3:
17 load gamma0.mat
18 load gamma1.mat
```

```
19 load gamma2.mat
20 load gamma3.mat
21
22 [m,n]=size(gamma3);
23
24 %Definición de las matrices simbólicas A y B de dimensión mxm y ...
    mxn respect.
25 A_sim=sym('A',[m m]);
26 B_sim=sym('B',[m n]);
27
28 % Definicion del vector de parámetros Phi.
29 Phi=[A_sim(:);B_sim(:)];
30 B1=(gamma0-gamma1*A_sim)*A_sim-gamma2; % bloque de ecuaciones 1
31 B2=(gamma0-gamma1*A_sim)*B_sim-gamma3; % bloque de ecuaciones 2
32
33 % Definición de la función F (función implícita):
34 F=[B1(:);B2(:)];
35
36 %% =Cálculo de las derivadas de F (función implícita) resp. de ...
    Phi y Theta=
37
38 % Definición de la derivada de F respecto a Phi
39 F_Phi=jacobian(F,Phi);
40
41 % Definición de la derivada de F respecto a Theta
42 F_Theta=jacobian(F,Theta);
43
44 %% ==Sustitución de los parámetros simbólicos por los numéricos==
45 for i=1:length(Theta)
46 k=[findsym(Theta(i)), 'Theta_num(i)', ',';'];
```

```

47 eval(k);
48 end
49
50 vec_A_sim=A_sim(:);
51 vec_A=A(:);
52 for i=1:length(vec_A_sim)
53     k=[findsym(vec_A_sim(i)), 'vec_A(i)', ','];
54 eval(k);
55 end
56
57 vec_B_sim=B_sim(:);
58 vec_B=B(:);
59 for i=1:length(vec_B_sim)
60 k=[findsym(vec_B_sim(i)), 'vec_B(i)', ','];
61 eval(k);
62 end
63 %% =Evaluación numérica de las derivadas de F (función implícita)...
    resp. de Phi y Theta=
64
65 F_Phi=eval(F_Phi); %Sustitución numérica de la derivada F_Phi
66 F_Theta=eval(F_Theta); %Sustitución numérica de la derivada ...
    F_Theta
67
68 %% =====Cálculo numérico de H=====
69
70 H=-F_Phi\F_Theta; %Definición de H, según la ecuación (3.9).
71 return

```

B.1.2. Filtro_Kalman.m

```

1 % Esta función calcula la estimación de la función
2 %log-verosimilitud y la matriz de información para

```



```
3 %el modelo de ciclo económico real de Hansens
4 %en Theta_num, donde Z es el vector de observaciones.
5
6 function [Log_Verosimilitud,J_Theta]=Filtro_Kalman(Theta_num,Z)
7
8 disp('Modelo del Ciclo Económico Real de Hansens.');
```

```
9
10 [PP,QQ,RR,SS,NN,Z_ss,Sigma]=Uhlig_Hansens(Theta_num);
11
12 %% ==Cálculo de las matrices de la representación espacio-estado ...
    del modelo==
13
14 %A y B son las matrices de la ecuacion de estado y C es la matriz
15 %correspondiente a la ecuación de observación.
16
17 A=[PP zeros(1,5) NN*QQ;
18     RR zeros(5,5) NN*SS;
19     zeros(1,6) NN];
20
21 B=[QQ;
22     SS;
23     1];
24
25 C= [0 -1 1 0 0 0 0];
26
27 [m,n]=size(B);
28 B=B*Sigma.^0.5;
29 [p,T]=size(Z); %p dimensión de los vectores de observación y T el ...
    número de observaciones
30
```

```
31 R=zeros(p,1); %matriz de covarianza del ruido de la ecuación de ...
    estado.
32 Q=eye(n); %matriz de covarianza del ruido de la ecuación de ...
    observación.
33
34 %%==Derivadas de las matrices del sistema espacio-estado==
35
36 % Derivada de vec(A)
37 vec_A_Phi=[eye(m*m,m*m),zeros(m*m,m*n)];
38
39 % Derivada de vec(B)
40 vec_B_Phi=[zeros(m*n,m*m),eye(n*m,n*m)];
41
42 % Derivada de vec(A')
43 q(1)=1;
44
45 for i=2:m*m
46 if (q(i-1)+m>(i-1)*m*m)
47 q(i)=q(i-1)+m+1;
48 else
49 q(i)=q(i-1)+m+m*m;
50 end
51 end
52 o(q)=1;
53 o=reshape(o,m*m,m*m);
54 vec_tr_A_Phi=[o,zeros(m*m,m*n)];
55 clear q
56 clear o
57
58 % Derivada de vec(B')
```

```
59 q(1)=1;
60 for i=2:n*m
61 if (q(i-1)+n>(i-1)*m*n)
62 q(i)=q(i-1)+n+1;
63 else
64 q(i)=q(i-1)+n+m*n;
65 end
66 end
67 o(q)=1;
68 o=reshape(o,n*m,n*m);
69 vec_tr_B_Phi=[zeros(m*n,m*m),o];
70
71 %% =====Inicialización Algoritmo Filtro de Kalman=====
72 E=B*Q*B.';
73
74 %Calculando sigma inicial a través de dlyap1.mat:
75 sigma=dlyap1(A,E);
76 %=====
77 b=(p/2)*log(2*pi);
78 x=zeros(m,1);
79 xt=x;
80 sigmat=sigma;
81 N=m^2+m*n;
82 xt_Phi=zeros(m,N);
83 vec_sigmat_Phi=zeros(m*m,N);
84 w1=zeros(N);
85 w2=zeros(N);
86 L=0;
87
88 %Ecuaciones del filtro de Kalman y cálculo de las derivadas ...
```

```

requeridas
89
90 for j=1:T
91     omega=C*sigma*C'+R;
92     z=C*x;
93     e=(Z(j)-Z_ss)-z; %Innovaciones
94
95     %==Derivadas de las ecuaciones del filtro de Kalman==
96     x_Phi=kron(xt.',eye(m))*vec_A_Phi+A*xt_Phi;
97     vec_sigma_Phi=kron(A,eye(m))*(kron(sigmat.',eye(m))*vec_A_Phi+...
        kron(eye(m),A)*vec_sigmat_Phi)+(kron(eye(m),(A*sigmat))*...
        vec_tr_A_Phi)+(kron(eye(m),B*Q)*vec_tr_B_Phi)+(kron(B*Q.',...
        eye(m))*vec_B_Phi);
98     e_Phi=-C*x_Phi;
99     vec_omega_Phi=kron(C,C)*vec_sigma_Phi;
100    invers_omega=omega\eye(p); %Inversa de la matriz de covarianza...
        del error.
101    vec_omega_inv_Phi=-kron(invers_omega.',invers_omega)*...
        vec_omega_Phi;
102    xt_Phi=x_Phi+(kron((invers_omega*e).',eye(m))*kron(C,eye(m))*...
        vec_sigma_Phi)+(sigma*C.'*((kron(e.',eye(p))*...
        vec_omega_inv_Phi)+(invers_omega*e_Phi));
103    vec_sigmat_Phi=vec_sigma_Phi-kron((C*sigma).',eye(m))*(kron(...
        invers_omega.*C,eye(m))*vec_sigma_Phi+kron(eye(p),sigma*C...
        .')*vec_omega_inv_Phi)-kron(eye(m),sigma*C.*invers_omega)*...
        kron(eye(m),C)*vec_sigma_Phi;
104
105    %==Evaluaciones para la matriz J_Phi==
106    w1=w1+vec_omega_Phi.*kron(invers_omega,invers_omega)*...
        vec_omega_Phi;

```

```
107     w2=w2+e_Phi.'*invers_omega*e_Phi;
108
109     %%==Actualizaciones del Filtro de Kalman==
110     K=sigma*C.'*invers_omega; %Ganancia de Kalman.
111     sigmat=sigma-K*C*sigma;
112     xt=x+K*e;
113     sigma=A*sigmat*A.'+E;
114     x=A*xt;
115
116     %%==Evaluando la función de verosimilitud==
117     L=L+(b+(0.5)*log(det(omega))+0.5*e.'*invers_omega*e);
118 end
119 Log_Verosimilitud=-L;
120
121 %% ===Cálculo de la matriz de Información===
122
123 %Cálculo de la matriz de información del modelo en la forma ...
    reducida:
124 J_Phi=(1/2)*w1+w2;
125
126 %Estimación del gradiente de h calculada con
127 %gradiente.m:
128 [H]=gradiente(A,B,Theta_num);
129
130 %Cálculo de la matriz de información segun la ecuación (3.4):
131 J_Theta=H.'*J_Phi*H;
132 return
```

B.2. Algoritmos muestreo de importancia y Metropolis-Hastings

B.2.1. MetropolisHastings.m

```
1 %La siguiente función corresponde al algoritmo Metropolis-Hasting
2 %el cual es utilizado para obtener muestras de la densidad a
3 %posteriori para el vector de parámetros profundos del modelo
4 %del ciclo económico real de Hansens, dada las observaciones.
5
6 %Entradas: Theta_num: vector inicial (moda del posterior).
7 % Invers_J_Theta: inversa de la aproximación a la matriz Hessiana
8 % en Theta_num.
9 % N: número de simulaciones.
10 % c: escalar utilizado para re-escalar la inversa de la matriz
11 % de información en el vector inicial.
12
13 function [M,E,eps,MODA]=MetropolisHastings(Theta_num,...
14     Invers_J_Theta,N,c)
15 n=length(Theta_num);
16 H=Invers_J_Theta;
17
18 %Inicializando el arreglo donde se guardará la cadena de Markov ...
19     generada.
20
21 M=zeros(n,N);
22
23 %Evaluando la función de log-verosimilitud en Theta_num:
24 [loglike]=FiltroKalman(Theta_num);
25
26 %Evaluando el prior en Theta_num:
27 [logpri]=prior(Theta_num);
```

```
26
27 MODA=Theta_num;
28
29 loglikeprimax=loglike+logpri;
30 acceptance=0;
31 S=0;
32
33 for j=1:N;
34     %Generando un vector aleatorio desde la distribución normal
35     %con media Theta_num y matriz de covarianza c^2*H:
36     newTheta=Theta_num+sqrtm((c^2)*H)*normrnd(0,1,6,1);
37
38     %Calculando el prior en newTheta:
39     [newlogpri]=prior(newTheta);
40
41     %Guardando los vectores que conformarán la cadena:
42     M(:,j)=Theta_num;
43
44     %Suma de los vectores que conforman la cadena de Markov:
45     S=M(:,j)+S;
46
47     if newlogpri>-Inf;
48         [newloglike]=FiltroKalman(newTheta);
49
50         if newloglike== -Inf;
51             ratio=0;
52         else
53             ratio=exp((newloglike+newlogpri)-(loglike+logpri));
54         end
55
```

```
56         if rand<=ratio;
57             loglike=newloglike;
58             logpri=newlogpri;
59             Theta_num=newTheta;
60             acceptance=acceptance+1;
61
62             %Cálculando la MODA:
63             if newloglike+newlogpri>loglikeprimax;
64                 loglikeprimax=newloglike+newlogpri;
65                 MODA=newTheta;
66             end
67         end
68     end
69 end
70
71 %Cálculo de la tasa de aceptación:
72 G=(acceptance/N)*100;
73 fprintf('La tasa de aceptación del vector aleatorio generado es ...
74         de un %g por ciento',G)
75
76 %Cálculo de la media o valor esperado:
77 E=S/N;
78
79 %Cálculo de la desviación estándar:
80 eps=std(M');
```

B.2.2. MuestreoImportancia.m

```
1 %La siguiente función corresponde al algoritmo muestreo de
2 %importancia el cual es utilizado para obtener muestras de
3 %la densidad a posteriori para el vector de parámetros
4 %profundos del modelo del ciclo económico real de Hansens,
```



```
5 %dada las observaciones. El cual utiliza como función
6 %de importancia, la función de densidad de una
7 %t-distribución multivariada.
8
9 %Entradas: Theta_num: vector inicial.
10 % Invers_J_Theta: inversa de la aproximación a la matriz Hessiana
11 % en Theta_num.
12 % N: número de simulaciones.
13 % c: escalar utilizado para re-escalar la inversa de la matriz
14 % de información en el vector inicial.
15 % v: grados de libertad para la t-distribución multivariada.
16
17 function [M,E,eps,MODA]=MuestreoImportancia(Theta_num,...
    Invers_J_Theta,N,c,v)
18
19 n=length(Theta_num);
20 H=Invers_J_Theta;
21
22 %Inicializando el arreglo donde se guardarán los vectores ...
    generados:
23 M=zeros(n,N);
24
25 %Evaluando la función de log-verosimilitud en Theta_num:
26 [L]=FiltroKalman(Theta_num);
27
28 %Evaluando el prior en Theta_num:
29 [p]=prior(Theta_num);
30
31 Logposterior_max=L+p;
32 MODA=Theta_num;
```

```
33 Wp=0;
34 Ws=0;
35
36 for i=1:N
37     %generando un vector aleatorio desde la t-distribución
38     %multivariada con v grados de libertad, media Theta
39     %y matriz de covarianza c^2*H:
40     M(:,i)=(sqrtm((c^2)*H)*normrnd(0,1,6,1))/sqrt(chi2rnd(v)/v)+...
        Theta_num;
41
42     %evaluando la función log-verosimilitud en M(:,i)
43     [L]=FiltroKalman(M(:,i));
44
45     %evaluando el prior en M(:,i)
46     [p]=prior(M(:,i));
47
48     if p>-Inf && L>-Inf
49
50         %evaluando la función de densidad de la t-distribución
51         %en M(:,i) con media Theta_num, matriz de covarianza c^2*H
52         %y v grados de libertad desde t_MulPDF.m
53         q=t_MulPDF(M(:,i),Theta_num,(c^2)*H,v);
54
55         %Cálculo de los pesos w.
56         w=(exp(L)*exp(p))/q;
57         Wp=(w*M(:,i))+Wp;
58         Ws=w+Ws;
59
60         %Cálculando la MODA.
61         if L+p>Logposterior_max
```

```
62         Logposterior_max=L+p;
63         MODA=M(:,i);
64     end
65 end
66 end
67
68 %Cálculo de la media o valor esperado:
69 E=Wp/Ws;
70
71 %Cálculo de la desviación estándar:
72 eps=std(M');
```

REFERENCIAS

- [1] An introduction to log-linearizations. Fall 2000.
- [2] S. An and F. Schorfheide (2007) “Bayesian Analysis of DSGE Models”. *Econometric Reviews*, 26 (2-4), 113 – 172.
- [3] L. Bravo, I. LLantas, E. Pérez (2008). “Análisis de datos con técnicas Bayesianas”. XXI escuela venezolana de matemáticas.
- [4] P. Caines (1988). “Linear Stochastic Systems”. New York: Wiley.
- [5] P. Caraiani (2008). “Forecasting Romanian GDP using a small DSGE model”. Institute of Economic Forecasting Romanian.
- [6] N. Iskrev (2008). “Evaluating the information matrix in linearized DSGE models”. *Economics Letters* 99. 607 – 610.
- [7] R. Kalman (1060). “A New Approach to Linear Filtering and Prediction Problems”. Instituto de Investigación de Estudios Avanzados. Baltimore, Md.
- [8] H. Neudecker, A. Klein (2000). “A direct derivation of the exact Fisher information matrix of Gaussian vector state space models”. *Linear Algebra and its Applications*.
- [9] T. Rothenberg (1966) . “Efficient estimation with a priori information: A classic approach”.
- [10] R.H Shumway and D.S Stoffer (2006). “Time series analysis and its applications”. Springer Text in Statistics. Second Edition.
- [11] C.A. Sims (2002). Solvin linear rational expectations models. *Computational economics*.
- [12] H. Uhlig (1995). “A Toolkit for Analyzing Nonlinear Dynamic Stochastic Models Easily”. CentER, University of Tilburg and CEPR.