



Modelos dinámicos y computacionales en Economía

Introducción a NetLogo

Licenciatura en Economía, FCEA, UDELAR

17 de octubre de 2023

Contenido de la clase:

- Presentación de Netlogo
- Principios del diseño
- Interfaz
- Clases / objetos en NetLogo
- Ejemplo: Pila de arena
- Ejemplo: Fire model

Modelos

- ¿qué es un modelo?

"...Since all models are wrong the scientist cannot obtain a "correct" one by excessive elaboration. On the contrary following William of Occam he should seek an economical description of natural phenomena. Just as the ability to devise simple but evocative models is the signature of the great scientist so overelaboration and overparameterization is often the mark of mediocrity.

Since all models are wrong the scientist must be alert to what is importantly wrong. It is inappropriate to be concerned about mice when there are tigers abroad".¹

- ¿qué es un modelo basado en agentes?
- ¿qué es un agente?

¹Box, G. E. (1976). Science and statistics. *Journal of the American Statistical Association*, 71(356), 791-799.

Presentación de NetLogo

- Es un lenguaje de programación y un entorno de desarrollo (IDE), diseñado por Uri Wilensky de Northwestern University.
- Es uno de los entornos de programación más utilizados para Modelos Basados en Agentes (ABM).
- Es el más fácil de aprender.
- Especialmente diseñado para estudiar la dinámica de sistemas complejos.

Principios del diseño

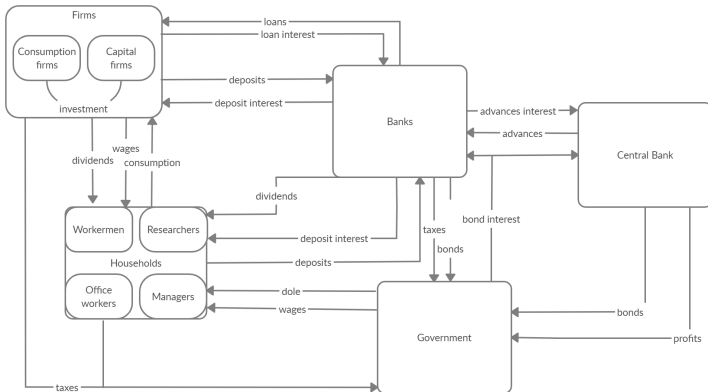
- Bajo coste de aprendizaje
 - Pueden construirse modelos simples en poco tiempo
 - Interfaz gráfica intuitiva
 - Programación Orientada a Objetos
- Alta performance
 - El lenguaje permite modelos a gran escala.
 - Permite leer, modificar y publicar modelos.
 - Elimina las diferencias entre el investigador y el programador.
 - Permite desarrollo e investigación interactivas.
 - Permite compartir modelos.

¿Qué tan "complicado" puede llegar a ser?

- Decenas de miles de "agentes", "zonas" y "links".
- Decisiones complejas
- Muchos tipos de agentes
- Modelos de ciudades enteras
- Herramientas adicionales (extensiones) permiten la integración con otros software.
- Tipos de análisis
 - Modelos espaciales
 - Análisis de políticas
 - Propagación de información (marketing, redes)
 - Sistemas de apoyo en la toma de decisiones

¿Qué tan "complicado" puede llegar a ser?

Ejemplo: Esquema de interacciones en un ABM macroeconómico.



¿Cómo podemos utilizar el análisis de sistemas complejos?

- Los sistemas complejos pueden ser muy difíciles de predecir, controlar y manejar.
- Sin embargo, en muchos casos el objetivo de las políticas públicas consiste precisamente en procurar predecir, controlar y manejar estos sistemas.
- El análisis de sistemas complejos y los modelos basados en agentes nos permiten usar un "simulador de vuelo" más que una predicción certera.

Aspectos a tomar en cuenta:

- Tipping points: puntos donde el sistema puede cambiar de régimen -fase- a partir de un pequeño cambio en los parámetros.
- Irreversibilidad: las posibilidades futuras se encuentran determinadas por elecciones pasadas.
- Alta sensibilidad a las condiciones iniciales: "el presente determina el futuro, pero una aproximación del presente no determina aproximadamente el futuro" (Lorenz).
- No linealidad: los insumos no afectan a los resultados de forma lineal.
- Individuos diversos y heterogéneos.
- Los individuos se encuentran interconectados y afectan a las decisiones de los demás.

Una tercera forma de hacer ciencia

Axelrod (1997)

- Dos formas tradicionales de hacer ciencia:
 - Inducción: desde lo particular a una teoría general
 - Deducción: desde los principios fundamentales a una teoría.
- Una tercera forma:
 - "Generativa". Utiliza principios fundamentales para generar un set de datos que permita avanzar hacia una teoría general.

Bottom-up / Up-bottom

- Si conocemos las reglas de comportamiento individuales, ¿podemos determinar el patrón agregado?
 - Este tipo de modelización nos permite un método para hacerlo.
- ¿Si conocemos el patrón agregado y queremos conocer las reglas de comportamiento individual?
 - Podemos proponer reglas y ver si generan el fenómeno agregado observado.

- Tutoriales:
 - En el sitio Web
 - En **Ayuda / Guía del Usuario Netlogo**
 - En **Ayuda / Diccionario Netlogo**
 - En **Ayuda / Comunidad de Usuarios de Netlogo**
- En la pantalla principal vemos tres pestañas:
 - Ejecutar (interfaz)
 - Información
 - Código
- Al abrir un modelo, en la interfaz veremos siempre²:
 - Botón "SETUP"
 - Botón "GO"
 - Selector de velocidad
 - Opción de añadir otros elementos para la interfaz

²Ver ejemplo de **Archivo / Biblioteca de Modelos / IABM Textbook / chapter 8 / "Run example"**

Algunos comandos para los agentes "turtles"

- create-turtles (crt): **create-turtles 100**
- ask: ask turtles [set color blue set shape "car"]
- ask turtles [setxy random-xcor random-ycor]
- ask: ask patches [set pcolor white]
- forward (fd), backward (bk)
- repeat
- color, size, xcor, ycor
- pen-down (pd), pen-up (pu)
- clear-all (ca)
- inspect / watch
- die

Importante: los comandos los podemos ingresar desde la línea de comandos o en la pestaña "Código".

Zonas "patches"

- Podemos usar la opción "Inspect" (botón derecho del mouse o **inspect patch 0 0**)
- Cambiamos el color con **pcolor**:
ask patch 0 0 [set pcolor blue]
ask patches [set pcolor green]
- Turtles pueden acceder directamente a la info de los patches.
- Comandos:
 - **setxy, facexy**
 - **random-xcor (pxcor), random-ycor (pycor)**

Ejemplo: Pila de arena

“Who could ever calculate the path of a molecule? How do we know that the creations of worlds are not determined by falling grains of sand?”

– Victor Hugo, *Les Miserables*

Los fenómenos complejos que se observan, indican que la naturaleza se encuentra en un estado críticamente auto-organizado³.

En **Archivo / Biblioteca de Modelos / IABM Textbook / chapter 8 / "Sandpile simple"**

- ¿Cómo se distribuyen las avalanchas?
- ¿cuál es el estado crítico relevante en este modelo?

³Bak, P., Tang, C., & Wiesenfeld, K. (1987). Self-organized criticality: An explanation of the $1/f$ noise. *Physical Review Letters*, 59(4), 381.

Ejemplo: Fire model

En **Archivo / Biblioteca de Modelos / IABM Textbook / chapter 3 / Fire extensions / "Fire simple"**

- ¿Cuáles son los parámetros relevantes?
- ¿A partir de qué valores cambia el comportamiento del modelo?
- ¿Cómo cambia este modelo si modificamos el criterio por el cual se prende fuego un árbol?
 - ① Que observe a los ocho vecinos a su alrededor.
 - ② Que los “contagie” con una probabilidad del 70%.
- ¿Cómo podemos medirlo?

Ejemplo: Fire model (cont.)

BehaviorSpace

- Cada vez que encontremos un modelo con componentes estocásticos, debemos correr varias veces el modelo para poder caracterizarlo.
 - Para encontrar su comportamiento “promedio”.
 - Para encontrar si existe un comportamiento anómalo o errático que es inherente al modelo.
- NetLogo tiene disponible la herramienta BehaviorSpace, que permite correr un modelo repetidas veces.
- Con los mismos valores de los parámetros o con distintos valores.

Ejemplo: Fire model (cont.)

BehaviorSpace

Experimento 1

- Analizamos la cantidad de árboles quemados, para una densidad entre 40% y 80%.

Experimento 2

- Analizamos la cantidad de árboles quemados, considerando:
 - una densidad entre 50% y 80%.
 - Una probabilidad de propagación entre el 65% y el 100%.