

Incendio Forestal

ExactasPrograma

Facultad de Ciencias Exactas y Naturales, UBA

Verano 2019

Incendio Forestal



 **INCENDIO DE COPAS**

Sistemas complejos

- micro motivos/interacciones, macro comportamientos/efectos
- feedback
- auto-organización, patrones emergentes
- transiciones al variar los parámetros

La computadora nos permite programar, simular explorar y detectar estas cosas.





La verdad sobre China: ¿El embotellamiento más largo del mundo duró 12 días?

Daniel Canal | 13-01-2015 - 05:40:58

Mito: En China el tráfico vehicular es tal que la peor congestión registrada en la historia ocurrió en Beijing y duró 12 días.

Realidad: Cuando el conductor de un Peugeot 404 pasa de segunda marcha a tercera del cuento "Autopista al sur", de Julio Cortázar, se produce un embotellamiento que dura lo suficiente para...

El 14 de agosto de 2010 empezó el embotellamiento más demorado en la historia, en la autopista 1 que conecta a Beijing, la capital china, con el Tibet. Este atasco se demoró 12 días en descongestionarse y se extendió a lo largo de 100 kilómetros. Lo particular es que el embotellamiento ocurrió sin ninguna razón aparente, solo porque había muchos autos en la vía, contrario al de París o a las autopistas como el de Chicago en 2011, que fueron consecuencia del mal clima.

Veredicto: Sí, el embotellamiento más demorado del mundo ocurrió en China en una carretera al norte de Beijing. Los chinos debieron vivir casi dos semanas en sus autos, y como el protagonista de Cortázar, cuando se empezaron a mover los carros y pasaron de segunda a tercera marcha, dejaron su vida...







¿Por qué incendios de bosques?

Fue uno de los primeros modelos (~ 1992), con el de avalanchas en pilas de arena.

Buscaba entender por qué los bosques tienen los tamaños que tienen, y se alternan sectores con árboles y sectores sin árboles.

Hay mil variantes, pero ninguna explica ningún bosque conocido. ¿Por qué no nos olvidamos del modelo, entonces?

Porque se aplica en otros problemas, paciencia...

Modelaremos ciclos de un año donde a cada posición del bosque le suceden cosas.

Terreno estará dividido en posiciones, y en cada una puede:

- tener un árbol
- no tener nada

Evolución se suceden una serie de etapas: Primavera, Verano y Otoño-Invierno

- 1 Primavera: en cada posición vacía, puede brotar un nuevo árbol (o no)
- 2 Verano: caen rayos en algunas posiciones incendiando dichos árboles. Cuando un árbol se incendia, propaga el fuego a los arboles de posiciones vecinas, y estos a su vez a su vecinos... y así, hasta que todo árbol incendiado incendió a todos sus vecinos y el incendio no puede propagarse más.
- 3 Otoño-Invierno: los árboles incendiados degradan y dejan la posición vacía, lista para que en la primavera vuelva a comenzar el ciclo.

Experimentando la dinámica



















Pre-requisitos:

- Una computadora donde poder correr `random.random()`
- Una tarjeta verde y una roja
- Determinar quienes son tus vecinos


















A simular

- Todos los que saquen menos de 0.8 se transforman en árboles
- Todos los que saquen menos de 0.3 se transforman prenden fuego
- A propagar
- Limpiar
- Otra vez























Etapas de la dinámica de un bosque - Año 1

Inicio						
Primavera						
Rayos						
Propagación						
						
Limpieza						






























Etapas de la dinámica de un bosque - Año 2

Inicio							
Primavera							
Rayos							
							
Propagación							
Limpieza							

Etapas de la dinámica de un bosque - Año 3

Inicio						
Primavera						
Rayos						
						
Propagación						
Limpieza						

Etapas de la dinámica de un bosque - Año 4

Inicio							
Primavera							
Rayos							
							
Propagación							
Limpieza							

Modelo de incendios de bosques

Queremos tomar los ingredientes básicos del problema real y construir un modelo.

- **Terreno:** tenemos n lugares, y en cada uno puede crecer un solo árbol.
- **Calidad del terreno:** cada primavera brota un árbol (si la posición está vacía) con probabilidad p .
- **Incendio:** en el verano, cae un rayo en cada posición con probabilidad f .
- **Propagación:** si un árbol tiene al lado uno prendido fuego, también se prende fuego.
- **Limpieza:** los árboles quemados se retiran dejando la posición vacía.

¿Cómo representar esto en la máquina?

- **Terreno:** tomamos una lista de n posiciones, una al lado de la otra, indexadas del 0 al $n - 1$.
- **Arboles:** representamos las posiciones como:
 - 0 si está vacía,
 - 1 si hay un árbol vivo,
 - 1 si hay un árbol prendido fuego.

Problemas interesantes a resolver hoy:

- 1 ¿Cuál es el valor de p que maximiza la cantidad de árboles que sobreviven cada año?
- 2 ¿Si cada posición pudiera tener y modificar su propio valor de p , qué pasaría?
- 3 ¿Y si modelamos un sistema dónde la propagación no sólo se da entre vecinos?

Ideas para el problema 1

Simulaciones: armamos una grilla de 100 lugares y tomemos un p arbitrario. Entonces, en el bosque...

- brota un árbol con probabilidad p en cada lugar vacío, cae un rayo con probabilidad 0,02, se propaga, limpiamos los quemados, y contamos cuántos quedan. En el bosque que queda...
- brota un árbol con probabilidad p en cada lugar vacío, cae un rayo con probabilidad 0,02, se propaga, limpiamos los quemados, y contamos cuántos quedan. En el bosque que queda...
- brota un árbol con probabilidad p en cada lugar vacío, cae un rayo con probabilidad 0,02, se propaga, limpiamos los quemados, y contamos cuántos quedan. En el bosque que queda...

Lo hacemos 1000 veces, y calculamos el promedio.

Ideas para el problema 1

Y esto lo hacemos explorando valores de p entre 0 y 1:

```
probs = []
promedios = []
valores_p = np.arange(0.0, 1.0, 0.01)
for i in range(len(valores_p)):
    ...COMPLETAR...
    probs.append(valores_p[i])
    promedios.append(...COMPLETAR...)
```

Es similar a la clase anterior, cuando había que verificar que un álbum estaba lleno.

Ideas para el problema 1

Ahora busquemos cuándo se alcanza el máximo graficando.

...Sí, haciendo un gráfico. Para cada probabilidad marcamos con un punto en el plano la cantidad de árboles sobrevivientes.

¿Puede Python ayudarnos con esta tarea? ¡Claro!

- Al igual que `random` o `numpy`, es un módulo que nos deja elegir una lista como eje x , una como eje y , y graficar.
- Para importarlo, hay que usar `import matplotlib.pyplot as plt`

Ejemplo de Gráfico

¿Qué dirían que hace este código?

```
v1 = []
v2 = []
v3 = []
numero = 0
numero_final = 10
while numero <= numero_final:
    v1.append(numero)
    v2.append(numero ** 2)
    v3.append(numero ** 3)
    numero = numero + 1
```

La lista `v1` tiene los números del 0 al 10, la lista `v2` tiene sus cuadrados, y la lista `v3` tiene sus cubos.

¡Grafiquémoslo!

Comandos para graficar

Estas líneas muestran el gráfico por pantalla

```
plt.plot(v1, v2, " . ")  
plt.plot(v1, v3, " . ")  
plt.show()
```

Adicionalmente, podemos definir el título, el nombre de los ejes, etc.

```
plt.title("titulo del grafico")  
plt.xlabel("tipos de p", fontsize = 16)  
plt.ylabel("media anual", color = "blue")  
plt.plot(probs, promedios)  
plt.show()
```

¡No se olviden de importar `random` y `numpy` además de `pyplot`!

¡A pensar en el resto! (¡Y no olviden enviarlo al terminar!)

Material <http://campus.exactas.uba.ar>

Formulario <http://cor.to/ep2019>