Mathematical Modeling
A tutorial

Tamás Biró

Eötvös Loránd University

Network Theory and Computer Modeling in the Study of Religion
August 31, 2016
A note on myself

June 1997: two exams on the same day

- János Kertész: Computer simulations (physics major)
- József Schweitzer: Jewish liturgy (Hebrew major)

Anything common in these two topics?

HOPEFULLY…
A note on myself

June 1997: two exams on the same day

- János Kertész: Computer simulations (physics major)
- József Schweitzer: Jewish liturgy (Hebrew major)

Anything common in these two topics?

HOPEFULLY...
A note on myself

Theses on

- Analysis of DNA sequences using text analysis methods (physics major, supervisor: Tamás Vicsek)
- Modeling Mathematically the Statistical Properties of Written Texts (theoretical linguistics major)

*From physics to linguistics: was it a big step?*

**NO!**
A note on myself

Theses on

- Analysis of DNA sequences using text analysis methods (physics major, supervisor: Tamás Vicsek)

- Modeling Mathematically the Statistical Properties of Written Texts (theoretical linguistics major)

From physics to linguistics: was it a big step? NO!
Overview

1. Computation ≠ computers
2. Mathematics and computer simulations as methodologies
3. On differential equations
4. Differential equations for dynamic systems
5. Conclusions
Overview

1. Computation ≠ computers

2. Mathematics and computer simulations as methodologies

3. On differential equations

4. Differential equations for dynamic systems

5. Conclusions
Let us create a calculating machine

The machine has to be able to sum up (two) numbers.

Input: Tamás Biró
Output: István Czachesz
Programmer: Luther Martin
Processing units: everybody else

Only rule type allowed for each processing unit:

\[
\text{if } \text{you hear } X_1 \quad \text{[and } X_2 \quad \text{[and } X_3 \ldots \text{]} \quad \text{],}
\]
\[
\text{then } \text{say } Y_1 \text{ to } Z_1 \quad \text{[and say } Y_2 \text{ to } Z_2 \quad \text{[and } Y_3 \text{ to } Z_3 \text{[…]} \text{] }
\]

20 minutes for the project!
Let us create a calculating machine

The machine has to be able to sum up (two) numbers.

*Input:* Tamás Biró

*Output:* István Czachesz

*Programmer:* Luther Martin

*Processing units:* everybody else

Only rule type allowed for each processing unit:

\[
\text{if you hear } X_1 \quad \text{[and } X_2 \quad \text{[and } X_3 \ldots] \quad \text{],}
\]

\[
\text{then say } Y_1 \text{ to } Z_1 \quad \text{[and say } Y_2 \text{ to } Z_2 \quad \text{[and } Y_3 \text{ to } Z_3 \ldots]\quad \text{]} \]

20 minutes for the project!
Let us create a calculating machine

The machine has to be able to sum up (two) numbers.

**Input:** Tamás Biró  
**Output:** István Czachesz  
**Programmer:** Luther Martin  
**Processing units:** everybody else

Only rule type allowed for each processing unit:

\[
\text{if } \text{you hear } X_1 \text{ [and } X_2 \text{ [and } X_3 \ldots ] \text{],} \\
\text{then } \text{say } Y_1 \text{ to } Z_1 \text{ [and say } Y_2 \text{ to } Z_2 \text{ [and } Y_3 \text{ to } Z_3 \ldots ] \text{]}
\]

20 minutes for the project!
What have we learned?

- Computation ≠ computers!
- Seemingly intelligent processes can be automated.
- Computational resources: memory (number of processing units) and time.
- Human resources: the time to create the program.
- Need to precisely define everything. Bugs and debugging.
- Continuous time vs. discrete time ticks.
- . . .
What have we learned?

- Computation $\neq$ computers!
- Seemingly intelligent processes can be automated.
- Computational resources: memory (# of processing units) and time.
- Human resources: the time to create the program.
- Need to precisely define everything. Bugs and debugging.
- Continuous time vs. discrete time ticks.
- ...
David Marr: Three levels of analysis

1. **Computational level:** What does the system do? What is the function (i.e., mapping input onto output) performed by the system? E.g., summation; face recognition; ritual performance.

2. **Algorithmic/representational level:** How is it performed? Representations, and manipulations of those representation. E.g., summation digit-by-digit.

3. **Implementational/physical level:** How is this algorithm physically realized? E.g., in silico; wetware; workshop participants.
Overview

1. Computation $\neq$ computers

2. Mathematics and computer simulations as methodologies

3. On differential equations

4. Differential equations for dynamic systems

5. Conclusions
Methodologies in physics

A. Experimental physics: data collection
   (exploratory research vs. hypothesis testing)

B. Theoretical physics: mathematics for modeling the word/nature
   + [Thought experiments]
   + Computer simulations (e.g., Kertész and Vicsek)

Analogy in other disciplines?
“Complicatedness” of theories

0. **Thought experiments**: handled mentally.

1. **Mathematical models**: handled analytically.

2. **Computer simulations**: can be more complex than mathematically tractable models, but simpler than real life.

Are we happy with

- *Level of abstraction?*
- *Simplifications?*

3. **Experiments**: complexities of real life controlled.

4. **Observations**: complexities of real life at their best.
Numerical solution vs. analytic solution

\[ 1 + 2 + 3 + \ldots + 98 + 99 + 100 = ? \]

- **Numerical solution**: go and compute it with sheer force. For more complex problems: often an approximate solution, only.

- **Analytic solution**: clever math provides a closed formula. Exact solution with pencil and paper
  \[ \rightarrow \text{on the condition that an analytic solution exists!} \]

\[ 1 + 2 + \ldots + 99 + 100 = \frac{100 \times (100 + 1)}{2} = 50 \times 101 = 5050 \]
Numerical solution vs. analytic solution

1 + 2 + 3 + ... + 98 + 99 + 100 = ?

- **Numerical solution**: go and compute it with sheer force. For more complex problems: often an approximate solution, only.

- **Analytic solution**: clever math provides a closed formula. Exact solution with pencil and paper → on the condition that an analytic solution exists!

\[
1 + 2 + \ldots + 99 + 100 = \frac{100 \times (100 + 1)}{2} = 50 \times 101 = 5050
\]
Issues with computer simulations

- Helps you better understand your theory/hypothesis.
- Forces you to formulate details of theory/hypothesis precisely.
- Faster. Can also be applied to past/remote/unreal conditions. Etc.
- Level of optimal abstractions:
  - If too simple: no connection to reality? *What do the results tell us?*
  - If too complex, too many parameters: easy to tweak the model. *What do the results tell us?*

→ Possible answer:
  - Understand the behavior of the model as a function of its parameters.
  - Seek results that are not too dependent on parameter setting.
Overview

1. Computation ≠ computers
2. Mathematics and computer simulations as methodologies
3. On differential equations
4. Differential equations for dynamic systems
5. Conclusions
Derivatives


\[ y = f(x) \]

\[ f(x) = x \sin(x^2) + 1 \Rightarrow f'(x) = \sin(x^2) + 2x^2 \cos(x^2) \]
## Derivatives

<table>
<thead>
<tr>
<th>$f(x)$</th>
<th>$f'(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c$</td>
<td>0</td>
</tr>
<tr>
<td>$x$</td>
<td>1</td>
</tr>
<tr>
<td>$x^2$</td>
<td>$2x$</td>
</tr>
<tr>
<td>$x^3$</td>
<td>$3x^2$</td>
</tr>
<tr>
<td>$c \cdot f(x)$</td>
<td>$c \cdot f'(x)$</td>
</tr>
<tr>
<td>$f(x) + g(x)$</td>
<td>$f'(x) + g'(x)$</td>
</tr>
<tr>
<td>$e^x$</td>
<td>$e^x$</td>
</tr>
<tr>
<td>$\sin(x)$</td>
<td>$\cos(x)$</td>
</tr>
<tr>
<td>$\cos(x)$</td>
<td>$-\sin(x)$</td>
</tr>
<tr>
<td>$\ldots$</td>
<td>$\ldots$</td>
</tr>
</tbody>
</table>
Differential equations

What is $f(x)$, if

$$f'(x) = 2x$$

Solution:

$$f(x) = x^2$$

$$f(x) = x^2 + c$$
Differential equations

What is $f(x)$, if

\[ f'(x) = 2x \]

Solution:

\[
\begin{align*}
f(x) &= x^2 \\
f(x) &= x^2 + c
\end{align*}
\]
What is \( f(x) \), if

\[ f'(x) = f(x) \]

Solution:

\[ f(x) = e^x \]

\[ f(x) = c \cdot e^x \]
Differential equations

What is \( f(x) \), if

\[
f'(x) = f(x)
\]

Solution:

\[
\begin{align*}
f(x) &= e^x \\
f(x) &= c \cdot e^x
\end{align*}
\]
Differential equations

What is \( f(x) \), if

\[
    f''(x) = -f(x)
\]

Solution:

\[
    f(x) = \sin(x) \\
    f(x) = \cos(x) \\
    f(x) = c_1 \cdot \sin(x) + c_2 \cdot \cos(x)
\]

Thus far: analytic solutions
Differential equations

What is \( f(x) \), if

\[
f''(x) = -f(x)
\]

Solution:

\[
\begin{align*}
f(x) &= \sin(x) \\
f(x) &= \cos(x) \\
f(x) &= c_1 \cdot \sin(x) + c_2 \cdot \cos(x)
\end{align*}
\]

Thus far: analytic solutions
Differential equations

What is $f(x)$, if

\[ f''(x) = -f(x) \]

Solution:

\[
\begin{align*}
  f(x) & = \sin(x) \\
  f(x) & = \cos(x) \\
  f(x) & = c_1 \cdot \sin(x) + c_2 \cdot \cos(x)
\end{align*}
\]

Thus far: analytic solutions
Numerical solutions for differential equations

What is $f(x)$, if

$$f''(x) + x \cdot f'(x) - 2 \cdot x^2 \cdot f(x) + \cos(x^3) - 15 = 0$$

Solution:

Use computers to solve this problem.
Numerical solutions: e.g., using step-by-step approximations.
NB: various sources of errors.
Numerical solutions for differential equations

What is \( f(x) \), if

\[ f''(x) + x \cdot f'(x) - 2 \cdot x^2 \cdot f(x) + \cos(x^3) - 15 = 0 \]

Solution:

Use computers to solve this problem.
Numerical solutions: e.g., using step-by-step approximations.
NB: various sources of errors.
Overview

1. Computation ≠ computers
2. Mathematics and computer simulations as methodologies
3. On differential equations
4. Differential equations for dynamic systems
5. Conclusions
Population dynamics

$y(t)$: size of the population at time $t$.

$\Delta y(t) = y(t + 1) - y(t)$: population growth at time $t$.

Suppose that population growth is equal to population size:

\[
y(t + 1) - y(t) = y(t)
\]
\[
y(t + 1) = 2y(t)
\]

Then: $y(1) = 2y(0)$, $y(2) = 2y(1) = 4y(0)$, $y(3) = 2y(2) = 8y(0), \ldots$, $y(t) = 2^t y(0)$. 
Population dynamics

\( y(t) \): size of the population at time \( t \).

\( \Delta y(t) = y(t + 1) - y(t) \): population growth at time \( t \).

Suppose that population growth is equal to population size:

\[
\begin{align*}
  y(t + 1) - y(t) &= y(t) \\
  \Delta y(t) &= y(t) \\
  \frac{dy}{dt} = y'(t) &= y(t)
\end{align*}
\]

And so: \( y(t) = e^t \).
Dynamic system

1, 2, ... n: the components of the dynamics system.

\(y_1(t), y_2(t), \ldots y_n(t)\):
“value” of each component in the dynamics system at time \(t\).

The equations defining the dynamic system (discrete time!):

\[
\begin{align*}
y_1(t + 1) &= \ldots y_1(t) + \ldots y_2(t) + \ldots y_n(t) + \ldots t + \ldots \\
y_2(t + 1) &= \ldots y_1(t) + \ldots y_2(t) + \ldots y_n(t) + \ldots t + \ldots \\
\vdots \\
y_n(t + 1) &= \ldots y_1(t) + \ldots y_2(t) + \ldots y_n(t) + \ldots t + \ldots 
\end{align*}
\]

So what functions are \(y_1(t), y_2(t), \ldots y_n(t)\)?
Solve those differential equations either numerically, or analytically.
Dynamic system

1, 2, \ldots n: the components of the dynamics system.

\( y_1(t), y_2(t), \ldots y_n(t): \) “value” of each component in the dynamics system at time \( t \).

The equations defining the dynamic system \textbf{(continuous time!)}:

\[
\begin{align*}
y_1'(t) &= \ldots y_1(t) + \ldots y_2(t) + \ldots y_n(t) + \ldots t + \ldots \\
y_2'(t) &= \ldots y_1(t) + \ldots y_2(t) + \ldots y_n(t) + \ldots t + \ldots \\
&\quad \vdots \\
y_n'(t) &= \ldots y_1(t) + \ldots y_2(t) + \ldots y_n(t) + \ldots t + \ldots 
\end{align*}
\]

So what functions are \( y_1(t), y_2(t), \ldots y_n(t) \)?
Solve those differential equations either numerically, or analytically.
Overview

1. Computation $\neq$ computers
2. Mathematics and computer simulations as methodologies
3. On differential equations
4. Differential equations for dynamic systems
5. Conclusions
Conclusions

And now: I am expected to provide smart conclusions!

But what if

- you gave them
- or postpone them to the end of the day/end of the week?

Anyway…
Conclusions

And now: I am expected to provide smart conclusions!

But what if

- you gave them
- or postpone them to the end of the day/end of the week?

Anyway...
Thank you for your attention!

Tamás Biró:
tamas.biro@btk.elte.hu

Tools for Optimality Theory
http://www.birot.hu/OTKit/

Work supported by:

European Union
Marie Curie Actions