

# Good Practices for Computational Modeling

Kyle A. Joyce

Department of Political Science  
University of California, Davis

August 14, 2014

# Practice of Agent-Based Modeling

- Validity
- Usability
- Extendability

# Practice of Agent-Based Modeling

## Validity

- Does the program correctly implement the model?
- How do you know if a result is a consequence of the model or due to a programming error?

# Practice of Agent-Based Modeling

## Usability

- You want others to be able to understand how your model works and to be able to replicate the results.

# Practice of Agent-Based Modeling

## Extendability

- You want a future user (including your future self) to be able to modify the program to examine new questions.

# Good Computing Practices

- Use long names for variables instead of short names (and make sure they are sensible).

# Good Computing Practices

- Use long names for variables instead of short names (and make sure they are sensible).
- List all variables at the beginning of the program.

# Good Computing Practices

- Use long names for variables instead of short names (and make sure they are sensible).
- List all variables at the beginning of the program.
- Write helpful comments.



# Good Computing Practices

- Use long names for variables instead of short names (and make sure they are sensible).
- List all variables at the beginning of the program.
- Write helpful comments.
- Write subroutines (and use switches) to add new features; do not delete code.

# Good Computing Practices

- Use long names for variables instead of short names (and make sure they are sensible).
- List all variables at the beginning of the program.
- Write helpful comments.
- Write subroutines (and use switches) to add new features; do not delete code.
- Document each version of the code (use a version control system).

# Good Computing Practices

- Use long names for variables instead of short names (and make sure they are sensible).
- List all variables at the beginning of the program.
- Write helpful comments.
- Write subroutines (and use switches) to add new features; do not delete code.
- Document each version of the code (use a version control system).
- Use another program to analyze the simulated data.

# Good Agent-Based Modeling Practices

- Keep the model simple

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)
- Have tunable dials



# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)
- Have tunable dials
- Implement key modeling choices in several ways

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)
- Have tunable dials
- Implement key modeling choices in several ways
- Check the parameters

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)
- Have tunable dials
- Implement key modeling choices in several ways
- Check the parameters
- Be your harshest critic

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)
- Have tunable dials
- Implement key modeling choices in several ways
- Check the parameters
- Be your harshest critic
- Keep a lab notebook

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)
- Have tunable dials
- Implement key modeling choices in several ways
- Check the parameters
- Be your harshest critic
- Keep a lab notebook
- Distribute your code

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)
- Have tunable dials
- Implement key modeling choices in several ways
- Check the parameters
- Be your harshest critic
- Keep a lab notebook
- Distribute your code
- Prove your results

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)
- Have tunable dials
- Implement key modeling choices in several ways
- Check the parameters
- Be your harshest critic
- Keep a lab notebook
- Distribute your code
- Prove your results
- Inspect simulated data carefully

# Good Agent-Based Modeling Practices

- Keep the model simple
- Focus on the science, not the computer (no need to talk about hardware, software, etc.)
- Be explicit about the assumptions
- Include “sanity checks” (build in implications that must be true)
- Have tunable dials
- Implement key modeling choices in several ways
- Check the parameters
- Be your harshest critic
- Keep a lab notebook
- Distribute your code
- Prove your results
- Inspect simulated data carefully
- Check micro-level behavior as well as macro-level patterns.