Expanded graphical models:
Inference, Model comparison, Model checking, Fake-data debugging, and Model understanding

Andrew Gelman
Dept of Statistics and Dept of Political Science, Columbia University, New York
1 Aug 2011
Generalized graphical models

- Expand graphical modeling to include:
  - Predictive model checking
  - Fake-data simulation
  - Scaffolding

- Common features:
  - Small changes to an existing fitted model
  - Comparisons of nodes between models
Goals

« (applied) Building confidence in our computations and our models
« (methodological) Being able to do this routinely
« (theoretical): A unified framework for model building, model fitting, and model checking
« (computational): Implementing in a Bayesian computing environment such as stan
6 challenges in statistical modeling

- Setting up a realistic (i.e., complicated) model
- Regularization or partial pooling
- Fitting the model
- Checking the fit to data
- Confidence building
- Understanding the fitted model
The models we’re fitting

WHO SUPPORTS HEALTH CARE REFORM?

18 to 29

30 to 34

45 to 64

65+

INCOME

UNDER $20,000

$20,000 – $40,000

$40,000 – $75,000

OVER $75,000

*Support for increased federal health care spending for the uninsured, based on the 2004 Annenberg survey.
Models for deep interactions

- Main effects, 2-way, 3-way, etc.
- Example: predicting public opinion given 4 age categories, 5 income categories, 50 states
- Also, group-level predictors (linear trends for age and income, previous voting patterns for states)
- Need a richer modeling language than this:
  - `bglmer (y ~ z.age*z.inc*rvote.st + (z.age*z.inc | st) + (z.age*rvote.st | inc) + (z.inc*rvote.st | age) + (z.age | inc*st) + (z.inc | age*st) + (z.st | age*inc) + (1 | age*inc*st), family=binomial(link="logistic"))`
  - No easy way to write this in Bugs or to program it oneself!
Example 1: a normal distribution is fit to the following data:
Example 1 of 3: checking a fit to a univariate dataset

20 replicated datasets under the model:
Example 2: checking a model fit to data with time ordering

```r
> plot (y, type="l")
> lines (y.rep)
```
Example 3: checking a model with three-way structure

Data and 7 replications:
Theoretical framework for predictive checking

- All our models are wrong
- What aspects of our models don’t fit the data?
- Data and replicated data: \( \theta \rightarrow y, y^{\text{rep}} \)
- Posterior predictive distribution, \( p(y^{\text{rep}}|y) \)
- Computation:
  - Simulate \( \theta \) from the posterior distribution, \( p(\theta|y) \)
  - Simulate \( y^{\text{rep}} \) from the predictive distribution, \( p(y^{\text{rep}}|\theta, y) \)
  - Compare \( y \) to the replicated datasets \( y^{\text{rep}} \)

- The generalized graphical model:

\[
M \rightarrow \text{theta} \rightarrow y \\
\downarrow \\
\downarrow \\
\downarrow \\
y.\text{rep}
\]
A posterior predictive check requires:

- Set of conditioning variables $\theta$
- Set of fixed design variables $X$ (e.g., sample size)
- Test variable $T(y)$ (more generally, $T(X, y, \theta)$)

Simulating posterior predictive replications is a fundamental operation in graphical models.

Requires a new node, $y^{\text{rep}}$, whose distribution is implied by the existing model.
Fake-data debugging

- Sample $\theta^{\text{pretend}}$ from the prior distribution $p(\theta)$
- Sample $y$ from the model $p(y|\theta^{\text{pretend}})$
- Perform Bayesian inference, simulations from $p(\theta|y)$
- Check calibration of posterior means, predictive intervals, etc. compared to $\theta^{\text{pretend}}$ (Cook, Gelman, and Rubin, 2007)
- Fake-data simulation is a fundamental operation in graphical models
- $\theta^{\text{pretend}}$ is a new node

$M \rightarrow \theta^{\text{pretend}} \rightarrow y$
\[
\begin{array}{c}
\text{\}/}\text{}/
\text{\}/}
\text{\}/}
\text{\}/}
\text{theta}
\end{array}
\]
Generalized graphical models

- Step 0 (already done): Expressing a statistical model as a graph; Bayesian computation on the graph
- Step 1: Graph of models
  - Each model is a node of this super-graph
  - Two models are connected if they differ by only one feature (adding/removing a variable, allowing a parameter to vary by group, adding/removing a grouping factor, changing a probability distribution or link function, ...)
- Step 2: Integrated graph
  - Nodes within models are linked within a larger graph
  - All models coexist
  - Analogy to computational method of parallel tempering
Automatic posterior predictive checking

- Example in Bugs:
  
  for (i in 1:n){
    y[i] ~ dnorm (y.hat[i], tau.y)
    y.rep[i] <- dnorm (y.hat[i], tau.y)
  }

- But $y^{rep}$ should be included automatically

- Implicit graphical structure for model checking: $y \leftarrow \theta \rightarrow y^{rep}$
Predictive checking and fake-data debugging

- Ideal of model checking or debugging in Stan, Bugs, etc.:
  - On/off switch for each node: is it conditioned on or averaged over?
  - Specify a test summary (numerical or graphical) of data and parameters
  - Various off-the-shelf test summaries will be available
- Design of data collection is integrated with graphical modeling
The network of models

- Each node is itself a graphical model
- Common parameters in neighboring models are linked
- Computations in the network:
  - Inference within a model
  - Inference among models (model comparison, averaging, and expansion)
  - Model checking
  - Fake-data debugging
  - Model understanding (exploratory model analysis)
Summary and future directions

- Generalized graphical models:
  \[
  M \rightarrow \theta \rightarrow y \quad \text{or} \quad y_{\text{rep}} \rightarrow \theta_{\text{rep}} \rightarrow y_{\text{rep}}
  \]

- All these quantities—\(\theta, y, y_{\text{rep}}\)—exist together
- Model checking can be done systematically
- All is completely Bayesian—there is no “double use of data”!
- A theoretical and computational unification of different aspects of statistical practice