# Parameterization and Bayesian Modeling

Andrew Gelman

Dept of Statistics and Dept of Political Science, Columbia University, New York
(Visiting Sciences Po, Paris, for 2009–2010)

30 nov 2009

# Statistical computation and statistical modeling

# Statistical computation and statistical modeling

- The folk theorem

# Statistical computation and statistical modeling

- The folk theorem
- The Pinocchio principle

# Statistical computation and statistical modeling

- The folk theorem
- The Pinocchio principle
- Latent variables, transformations, and Bayes

# Statistical computation and statistical modeling

- The folk theorem
- The Pinocchio principle
- Latent variables, transformations, and Bayes
- Examples

# Statistical computation and statistical modeling

- The folk theorem
- The Pinocchio principle
- Latent variables, transformations, and Bayes
- Examples
  - Truncated and censored data

# Statistical computation and statistical modeling

- The folk theorem
- The Pinocchio principle
- Latent variables, transformations, and Bayes
- Examples
  - Truncated and censored data
  - Modeling continuous data using an underlying discrete distirbution

# Statistical computation and statistical modeling

- The folk theorem
- The Pinocchio principle
- Latent variables, transformations, and Bayes
- Examples
  - Truncated and censored data
  - Modeling continuous data using an underlying discrete distirbution
  - Modeling discrete data using an underlying continuous distirbution

# Statistical computation and statistical modeling

- The folk theorem
- The Pinocchio principle
- Latent variables, transformations, and Bayes
- Examples
  - Truncated and censored data
  - Modeling continuous data using an underlying discrete distirbution
  - Modeling discrete data using an underlying continuous distirbution
  - Parameter expansion for hierarchical models

# Statistical computation and statistical modeling

- The folk theorem
- The Pinocchio principle
- Latent variables, transformations, and Bayes
- Examples
  - Truncated and censored data
  - Modeling continuous data using an underlying discrete distirbution
  - Modeling discrete data using an underlying continuous distirbution
  - Parameter expansion for hierarchical models
  - Iterative algorithms and time processes

# Example 1: Truncated and censored data

# Example 1: Truncated and censored data

- Sample of size $N$ from a distribution $f(y|\theta)$. We only observe the measurments that are less than 200. Out of these $N$, we observe $n = 91$ cases where $y_i < 200$.

# Example 1: Truncated and censored data

- Sample of size $N$ from a distribution $f(y|\theta)$. We only observe the measurments that are less than 200. Out of these $N$, we observe $n = 91$ cases where $y_i < 200$.
- Goal is inference about $\theta$

# Example 1: Truncated and censored data

- Sample of size $N$ from a distribution $f(y|\theta)$. We only observe the measurments that are less than 200. Out of these $N$, we observe $n = 91$ cases where $y_i < 200$.
- Goal is inference about $\theta$
- Scenario 1: $N$ is unknown. Use the *truncated-data likelihood*:

$$p(\theta|y) \propto p(\theta)\,[1 - F(y|\theta)]^{-91} \prod_{i=1}^{91} f(y_i|\theta)$$

# Example 1: Truncated and censored data

- Sample of size $N$ from a distribution $f(y|\theta)$. We only observe the measurments that are less than 200. Out of these $N$, we observe $n = 91$ cases where $y_i < 200$.
- Goal is inference about $\theta$
- Scenario 1: $N$ is unknown. Use the *truncated-data likelihood*:

$$p(\theta|y) \propto p(\theta)\,[1 - F(y|\theta)]^{-91} \prod_{i=1}^{91} f(y_i|\theta)$$

- Scenario 2: $N$ is known. Use the *censored-data likelihood*:

$$p(\theta|y, N) \propto p(\theta) F(y|\theta)^{N-91} \prod_{i=1}^{91} f(y_i|\theta)$$

# Truncated and censored data: things get weird

# Truncated and censored data: things get weird

- Now suppose $N$ is unknown. A Bayesian has two options:

# Truncated and censored data: things get weird

- Now suppose $N$ is unknown. A Bayesian has two options:
- Use the truncated-data model, or

# Truncated and censored data: things get weird

- Now suppose $N$ is unknown. A Bayesian has two options:
- Use the truncated-data model, or
- Use the censored-data model, averaging over the unknown $N$:

$$p(\theta|y) \propto \sum_{N=91}^{\infty} p(N)p(\theta) \binom{N}{91} F(y|\theta)^{N-91} \prod_{i=1}^{91} f(y_i|\theta)$$

# Truncated and censored data: things get weird

- Now suppose $N$ is unknown. A Bayesian has two options:
- Use the truncated-data model, or
- Use the censored-data model, averaging over the unknown $N$:

$$p(\theta|y) \propto \sum_{N=91}^{\infty} p(N)p(\theta) \binom{N}{91} F(y|\theta)^{N-91} \prod_{i=1}^{91} f(y_i|\theta)$$

- If $p(N) \propto 1/N$—but only with this prior—the above summation reduces to the truncated-data model:

$$p(\theta|y) \propto p(\theta) \left[1 - F(y|\theta)\right]^{-91} \prod_{i=1}^{91} f(y_i|\theta)$$

# Truncated and censored data: things get weird

- Now suppose $N$ is unknown. A Bayesian has two options:
- Use the truncated-data model, or
- Use the censored-data model, averaging over the unknown $N$:

$$p(\theta|y) \propto \sum_{N=91}^{\infty} p(N)p(\theta) \left( \begin{array}{c} N \\ 91 \end{array} \right) F(y|\theta)^{N-91} \prod_{i=1}^{91} f(y_i|\theta)$$

- If $p(N) \propto 1/N$—but only with this prior—the above summation reduces to the truncated-data model:

$$p(\theta|y) \propto p(\theta) \left[1 - F(y|\theta)\right]^{-91} \prod_{i=1}^{91} f(y_i|\theta)$$

- ?!

# Example 2: Modeling continuous data using an underlying discrete distribution

A bimodal distribution modeled using a mixture:

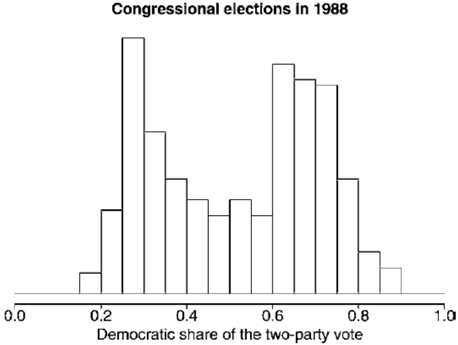


*Figure 1. Histogram of Democratic Share of the Two-Party Vote in Congressional Elections in 1988. Only districts that were contested by both major parties are shown here.*

# Separating into Republicans, Democrats, and open seats

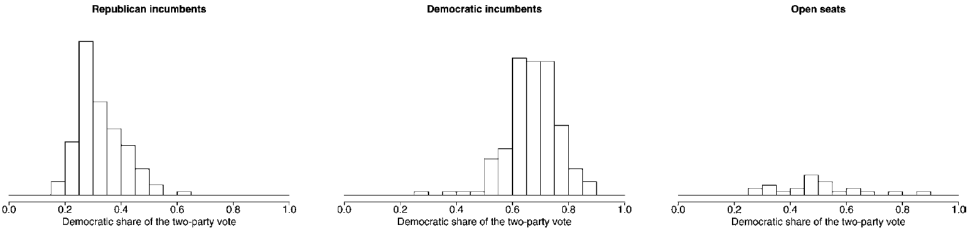


*Figure 2. Histogram of Democratic Share of the Two-Party Vote in Congressional Elections in 1988, in Districts With (a) Republican Incumbents, (b) Democratic Incumbents, and (c) Open Seats. Combined, the three distributions yield the bimodal distribution in Figure 1.*

# Separating into Republicans, Democrats, and open seats

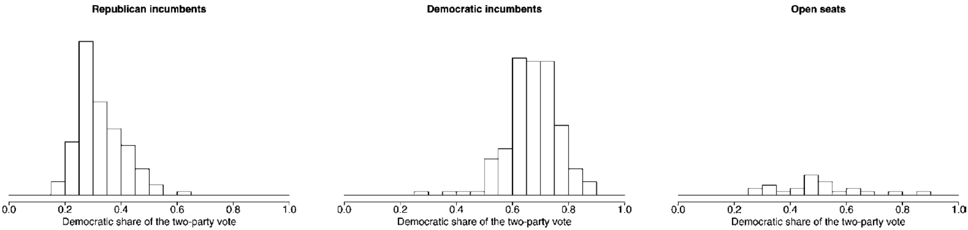


*Figure 2. Histogram of Democratic Share of the Two-Party Vote in Congressional Elections in 1988, in Districts With (a) Republican Incumbents, (b) Democratic Incumbents, and (c) Open Seats. Combined, the three distributions yield the bimodal distribution in Figure 1.*

- We took the mixture components seriously . . . and then they came to life!

# Example 3: Modeling discrete data using an underlying continuous distribution

# Example 3: Modeling discrete data using an underlying continuous distribution

- Logistic regression model for voting:

$$\text{Pr(vote Republican)} = \text{logit}^{-1}(X\beta)$$

# Example 3: Modeling discrete data using an underlying continuous distribution

- Logistic regression model for voting:

$$\text{Pr(vote Republican)} = \text{logit}^{-1}(X\beta)$$

- Latent variable interpretation:

$$z = X\beta + \epsilon; \text{ vote Republican if } z > 0$$

# Example 3: Modeling discrete data using an underlying continuous distribution

- Logistic regression model for voting:

$$\Pr(\text{vote Republican}) = \text{logit}^{-1}(X\beta)$$

- Latent variable interpretation:

$$z = X\beta + \epsilon;\ \text{vote Republican if } z > 0$$

- Take $z$ seriously as a continuous measure of Republican-ness

# Example 3: Modeling discrete data using an underlying continuous distribution

- Logistic regression model for voting:

$$\text{Pr(vote Republican)} = \text{logit}^{-1}(X\beta)$$

- Latent variable interpretation:

$$z = X\beta + \epsilon;\ \text{vote Republican if } z > 0$$

- Take $z$ seriously as a continuous measure of Republican-ness
- Can look at correlations with other z's, changes over time, ...

# Example 4: Parameter expansion for hierarchical models

# Example 4: Parameter expansion for hierarchical models

- Hierarchical regression with $M$ batches of coefficients:

$$y = \sum_{m=1}^{M} X^{(m)} \beta^{(m)} + \text{error}$$

# Example 4: Parameter expansion for hierarchical models

- Hierarchical regression with $M$ batches of coefficients:

$$y = \sum_{m=1}^{M} X^{(m)}\beta^{(m)} + \text{error}$$

- Exchangeable prior distributions: within each batch $m$,

$$\beta_j^{(m)} \sim \mathrm{N}(0, \sigma_m^2), \text{ for } j = 1, \ldots, J_m$$

# Example 4: Parameter expansion for hierarchical models

- Hierarchical regression with $M$ batches of coefficients:

$$y = \sum_{m=1}^{M} X^{(m)}\beta^{(m)} + \text{error}$$

- Exchangeable prior distributions: within each batch $m$,

$$\beta_j^{(m)} \sim \text{N}(0, \sigma_m^2), \text{ for } j = 1, \ldots, J_m$$

- Gibbs sampler for coefficients $\beta$ and hyperparameters $\sigma$ gets stuck near $\sigma \approx 0$

# Example 4: Parameter expansion for hierarchical models

- Hierarchical regression with $M$ batches of coefficients:

$$y = \sum_{m=1}^{M} X^{(m)} \beta^{(m)} + \text{error}$$

- Exchangeable prior distributions: within each batch $m$,

$$\beta_j^{(m)} \sim \mathrm{N}(0, \sigma_m^2), \text{ for } j = 1, \ldots, J_m$$

- Gibbs sampler for coefficients $\beta$ and hyperparameters $\sigma$ gets stuck near $\sigma \approx 0$
- Parameter expansion:

$$y = \sum_{m=1}^{M} \alpha_m X^{(m)} \beta^{(m)} + \text{error}$$

# Example 4: Parameter expansion for hierarchical models

- Hierarchical regression with $M$ batches of coefficients:

$$y = \sum_{m=1}^{M} X^{(m)} \beta^{(m)} + \text{error}$$

- Exchangeable prior distributions: within each batch $m$,

$$\beta_j^{(m)} \sim \mathrm{N}(0, \sigma_m^2), \text{ for } j = 1, \ldots, J_m$$

- Gibbs sampler for coefficients $\beta$ and hyperparameters $\sigma$ gets stuck near $\sigma \approx 0$
- Parameter expansion:

$$y = \sum_{m=1}^{M} \alpha_m X^{(m)} \beta^{(m)} + \text{error}$$

- Run Gibbs on $\alpha, \beta, \sigma$

# New models inspired by parameter expansion

# New models inspired by parameter expansion

- $y = \sum_{m=1}^{M} \alpha_m X^{(m)} \beta^{(m)} + \text{error}$

# New models inspired by parameter expansion

- $y = \sum_{m=1}^{M} \alpha_m X^{(m)} \beta^{(m)} + \text{error}$
- Factor-analysis model: $\alpha_m$ is the importance of factor $m$, and the $\beta_j^{(m)}$'s are the factor loadings

# New models inspired by parameter expansion

- $y = \sum_{m=1}^{M} \alpha_m X^{(m)} \beta^{(m)} + \text{error}$
- Factor-analysis model: $\alpha_m$ is the importance of factor $m$, and the $\beta_j^{(m)}$'s are the factor loadings
- Need informative prior distributions, for example $\beta_j^{(m)} \sim \mathrm{N}(1, 0.2^2)$

# New models inspired by parameter expansion

- $y = \sum_{m=1}^{M} \alpha_m X^{(m)} \beta^{(m)} + \text{error}$
- Factor-analysis model: $\alpha_m$ is the importance of factor $m$, and the $\beta_j^{(m)}$'s are the factor loadings
- Need informative prior distributions, for example $\beta_j^{(m)} \sim \mathrm{N}(1, 0.2^2)$
- The coefficients $\beta_j^{(m)}$ get a new life as latent factor loadings

# New prior distributions inspired by parameter expansion

# New prior distributions inspired by parameter expansion

- Simple hierarchical model: $y_{ij} \sim \mathrm{N}(\theta_j, \sigma_y^2)$, $\theta_j \sim \mathrm{N}(\mu, \sigma_\theta^2)$

# New prior distributions inspired by parameter expansion

- Simple hierarchical model: $y_{ij} \sim \mathrm{N}(\theta_j, \sigma_y^2),\ \theta_j \sim \mathrm{N}(\mu, \sigma_\theta^2)$
- What's a good prior distribution for $\sigma_\theta$?

# New prior distributions inspired by parameter expansion

- Simple hierarchical model: $y_{ij} \sim \mathrm{N}(\theta_j, \sigma_y^2),\ \theta_j \sim \mathrm{N}(\mu, \sigma_\theta^2)$
- What's a good prior distribution for $\sigma_\theta$?
- Redundant parameterization:
  $y_{ij} \sim \mathrm{N}(\mu + \alpha\eta_j, \sigma_y^2),\ \eta_j \sim \mathrm{N}(0, \sigma_\eta^2)$

# New prior distributions inspired by parameter expansion

- Simple hierarchical model: $y_{ij} \sim \mathrm{N}(\theta_j, \sigma_y^2), \ \theta_j \sim \mathrm{N}(\mu, \sigma_\theta^2)$
- What's a good prior distribution for $\sigma_\theta$?
- Redundant parameterization:
  $y_{ij} \sim \mathrm{N}(\mu + \alpha\eta_j, \sigma_y^2), \ \eta_j \sim \mathrm{N}(0, \sigma_\eta^2)$
  - Same model: $\theta_j = \mu + \alpha\eta_j$ for each $j$, and $\sigma_\theta = |\alpha|\sigma_\eta$

# New prior distributions inspired by parameter expansion

- Simple hierarchical model: $y_{ij} \sim \mathrm{N}(\theta_j, \sigma_y^2),\ \theta_j \sim \mathrm{N}(\mu, \sigma_\theta^2)$
- What's a good prior distribution for $\sigma_\theta$?
- Redundant parameterization:
  $y_{ij} \sim \mathrm{N}(\mu + \alpha\eta_j, \sigma_y^2),\ \eta_j \sim \mathrm{N}(0, \sigma_\eta^2)$
  - Same model: $\theta_j = \mu + \alpha\eta_j$ for each $j$, and $\sigma_\theta = |\alpha|\sigma_\eta$
  - Conditionally conjugate prior: normal distribution for $\alpha$ and inv-$\chi^2$ for $\sigma_\eta^2$

# New prior distributions inspired by parameter expansion

- Simple hierarchical model: $y_{ij} \sim \mathrm{N}(\theta_j, \sigma_y^2),\ \theta_j \sim \mathrm{N}(\mu, \sigma_\theta^2)$
- What's a good prior distribution for $\sigma_\theta$?
- Redundant parameterization:
  $y_{ij} \sim \mathrm{N}(\mu + \alpha\eta_j, \sigma_y^2),\ \eta_j \sim \mathrm{N}(0, \sigma_\eta^2)$
  - Same model: $\theta_j = \mu + \alpha\eta_j$ for each $j$, and $\sigma_\theta = |\alpha|\sigma_\eta$
  - Conditionally conjugate prior: normal distribution for $\alpha$ and inv-$\chi^2$ for $\sigma_\eta^2$
  - Put these together to get a half-$t$ prior for $\sigma_\alpha$

# New prior distributions inspired by parameter expansion

- Simple hierarchical model: $y_{ij} \sim \mathrm{N}(\theta_j, \sigma_y^2),\ \theta_j \sim \mathrm{N}(\mu, \sigma_\theta^2)$
- What's a good prior distribution for $\sigma_\theta$?
- Redundant parameterization:
  $y_{ij} \sim \mathrm{N}(\mu + \alpha\eta_j, \sigma_y^2),\ \eta_j \sim \mathrm{N}(0, \sigma_\eta^2)$
  - Same model: $\theta_j = \mu + \alpha\eta_j$ for each $j$, and $\sigma_\theta = |\alpha|\sigma_\eta$
  - Conditionally conjugate prior: normal distribution for $\alpha$ and inv-$\chi^2$ for $\sigma_\eta^2$
  - Put these together to get a half-$t$ prior for $\sigma_\alpha$
  - It really works!

# New prior distributions inspired by parameter expansion

- Simple hierarchical model: $y_{ij} \sim \mathrm{N}(\theta_j, \sigma_y^2),\ \theta_j \sim \mathrm{N}(\mu, \sigma_\theta^2)$
- What's a good prior distribution for $\sigma_\theta$?
- Redundant parameterization:
  $y_{ij} \sim \mathrm{N}(\mu + \alpha\eta_j, \sigma_y^2),\ \eta_j \sim \mathrm{N}(0, \sigma_\eta^2)$
  - Same model: $\theta_j = \mu + \alpha\eta_j$ for each $j$, and $\sigma_\theta = |\alpha|\sigma_\eta$
  - Conditionally conjugate prior: normal distribution for $\alpha$ and inv-$\chi^2$ for $\sigma_\eta^2$
  - Put these together to get a half-$t$ prior for $\sigma_\alpha$
  - It really works!
- Similar ideas for covariance matrices

# Example 5: Iterative algorithms and time processes

# Example 5: Iterative algorithms and time processes

- Iterative simulation turns an intractable spatial process into a tractable space-time process

# Example 5: Iterative algorithms and time processes

- Iterative simulation turns an intractable spatial process into a tractable space-time process
- MCMC algorithms such as hybrid sampling (in which "position" and "momentum" variables evlove over time)

# Example 5: Iterative algorithms and time processes

- Iterative simulation turns an intractable spatial process into a tractable space-time process
- MCMC algorithms such as hybrid sampling (in which "position" and "momentum" variables evlove over time)
- Simulation of "agent models" to determine equlibrium states in economics

# Example 5: Iterative algorithms and time processes

- Iterative simulation turns an intractable spatial process into a tractable space-time process
- MCMC algorithms such as hybrid sampling (in which "position" and "momentum" variables evlove over time)
- Simulation of "agent models" to determine equlibrium states in economics
- Connection to the Folk Theorem

# Example 5: Iterative algorithms and time processes

- Iterative simulation turns an intractable spatial process into a tractable space-time process
- MCMC algorithms such as hybrid sampling (in which "position" and "momentum" variables evlove over time)
- Simulation of "agent models" to determine equlibrium states in economics
- Connection to the Folk Theorem
- The Traveling Salesman Problem

# Example 5: Iterative algorithms and time processes

- Iterative simulation turns an intractable spatial process into a tractable space-time process
- MCMC algorithms such as hybrid sampling (in which "position" and "momentum" variables evlove over time)
- Simulation of "agent models" to determine equlibrium states in economics
- Connection to the Folk Theorem
- The Traveling Salesman Problem
- Mapping the Gibbs sampler or Metropolis algorithm to a real-time learning process

# Example 5: Iterative algorithms and time processes

- Iterative simulation turns an intractable spatial process into a tractable space-time process
- MCMC algorithms such as hybrid sampling (in which "position" and "momentum" variables evlove over time)
- Simulation of "agent models" to determine equlibrium states in economics
- Connection to the Folk Theorem
- The Traveling Salesman Problem
- Mapping the Gibbs sampler or Metropolis algorithm to a real-time learning process
  - Burn-in = "learning"

# Example 5: Iterative algorithms and time processes

- Iterative simulation turns an intractable spatial process into a tractable space-time process
- MCMC algorithms such as hybrid sampling (in which "position" and "momentum" variables evlove over time)
- Simulation of "agent models" to determine equlibrium states in economics
- Connection to the Folk Theorem
- The Traveling Salesman Problem
- Mapping the Gibbs sampler or Metropolis algorithm to a real-time learning process
  - Burn-in = "learning"
  - Moving through the distribution = tracing of variation

# Summary

# Summary

- Often we add latent variables that improve computation but do not change the likelihood function

# Summary

- Often we add latent variables that improve computation but do not change the likelihood function
  - New parameters are sometimes partially identified

# Summary

- Often we add latent variables that improve computation but do not change the likelihood function
  - New parameters are sometimes partially identified
  - Sometimes completely nonidentified

# Summary

- Often we add latent variables that improve computation but do not change the likelihood function
  - New parameters are sometimes partially identified
  - Sometimes completely nonidentified
- The Pinocchio principle

# Summary

- Often we add latent variables that improve computation but do not change the likelihood function
  - New parameters are sometimes partially identified
  - Sometimes completely nonidentified
- The Pinocchio principle
- New classes of prior distributions