Bayesian Data Analysis, class 8b

Andrew Gelman

Chapter 10: Overview of computation
Discussion of homework due beginning of Class 8b

- Theory problem
- Computing problem
- Applied problem
Discussion of homework due beginning of Class 8b

- Theory problem
- Computing problem
- Applied problem
Discussion of homework due beginning of Class 8b

- Theory problem
- Computing problem
- Applied problem
Discussion of homework due beginning of Class 8b

- Theory problem
- Computing problem
- Applied problem
How many simulation draws do you need to compute the 2.5% and 97.5% quantiles of $\theta$ to an accuracy of $0.1sd(\theta|y)$?

Lots more than you’d need to accurately specify $\theta$.
How many simulation draws do you need to compute the 2.5% and 97.5% quantiles of $\theta$ to an accuracy of $0.1\text{sd}(\theta|y)$?

Lots more than you’d need to accurately specify $\theta$. 

Andrew Gelman  
Bayesian Data Analysis, class 8b
How many simulation draws do you need to compute the 2.5% and 97.5% quantiles of $\theta$ to an accuracy of $0.1\text{sd}(\theta|y)$? Lots more than you’d need to accurately specify $\theta$. 
Computing problem

- Fit a hierarchical and a non-hierarchical model to the dogs data
Fit a hierarchical and a non-hierarchical model to the dogs data
Applied problem

- Estimating average birthweight in the population using poststratification
Applied problem

- Estimating average birthweight in the population using poststratification
10. Overview of computation

- Numerical integration
- Distributional approximations
- Direct simulation, rejection sampling, importance sampling
- Computing and debugging
- How many simulation draws are needed?
10. Overview of computation

- Numerical integration
  - Distributional approximations
  - Direct simulation, rejection sampling, importance sampling
  - Computing and debugging
  - How many simulation draws are needed?
10. Overview of computation

- Numerical integration
- Distributional approximations
  - Direct simulation, rejection sampling, importance sampling
  - Computing and debugging
- How many simulation draws are needed?
10. Overview of computation

- Numerical integration
- Distributional approximations
- Direct simulation, rejection sampling, importance sampling
- Computing and debugging
- How many simulation draws are needed?
10. Overview of computation

- Numerical integration
- Distributional approximations
- Direct simulation, rejection sampling, importance sampling
- Computing and debugging
- How many simulation draws are needed?
10. Overview of computation

- Numerical integration
- Distributional approximations
- Direct simulation, rejection sampling, importance sampling
- Computing and debugging
- How many simulation draws are needed?
10. Overview of computation

- Numerical integration
- Distributional approximations
- Direct simulation, rejection sampling, importance sampling
- Computing and debugging
- How many simulation draws are needed?
10. Overview of computation

- Numerical integration
- Distributional approximations
- Direct simulation, rejection sampling, importance sampling
- Computing and debugging
- How many simulation draws are needed?
10.1. Numerical integration

\[ E(h(\theta)|y) = \int h(\theta)p(\theta|y)d\theta \]

- Approximation using simulation
- Deterministic methods
10.1. Numerical integration

\[ E(h(\theta)|y) = \int h(\theta)p(\theta|y)d\theta \]

- Approximation using simulation
- Deterministic methods
10.1. Numerical integration

\[ E(h(\theta)|y) = \int h(\theta)p(\theta|y)d\theta \]

Approximation using simulation

Deterministic methods
10.1. Numerical integration

- $E(h(\theta) | y) = \int h(\theta)p(\theta | y) d\theta$
- Approximation using simulation
- Deterministic methods
10.2. Distributional approximations

- Simpler model
- Set hyperparameters to fixed values
- Quick imputation of missing data
- Network of models
10.2. Distributional approximations

- Simpler model
  - Set hyperparameters to fixed values
  - Quick imputation of missing data
  - Network of models
10.2. Distributional approximations

- Simpler model
- Set hyperparameters to fixed values
- Quick imputation of missing data
- Network of models
10.2. Distributional approximations

- Simpler model
- Set hyperparameters to fixed values
- Quick imputation of missing data
- Network of models
10.2. Distributional approximations

- Simpler model
- Set hyperparameters to fixed values
- Quick imputation of missing data
- Network of models
10.3. Direct simulation and rejection sampling

- Standard models
- Sampling on grid
10.3. Direct simulation and rejection sampling

- Standard models
- Sampling on grid
10.3. Direct simulation and rejection sampling

- Standard models
- Sampling on grid
Approximation $Mg(\theta)$ should dominate $p(\theta|y)$

For $s = 1, \ldots, S$:
- Draw $\theta$ from density proportional to $g$
- Accept with probability $\frac{p(\theta|y)}{Mg(\theta)}$

When is this difficult?
Approximation $M_g(\theta)$ should dominate $p(\theta|y)$

For $s = 1, \ldots, S$:
- Draw $\theta$ from density proportional to $g$
- Accept with probability $\frac{p(\theta|y)}{M_g(\theta)}$

When is this difficult?
Rejection sampling

- Approximation $Mg(\theta)$ should dominate $p(\theta|y)$
- For $s = 1, \ldots, S$:
  - Draw $\theta$ from density proportional to $g$
  - Accept with probability $\frac{p(\theta|y)}{Mg(\theta)}$
- When is this difficult?
Rejection sampling

- Approximation $Mg(\theta)$ should dominate $p(\theta|y)$
- For $s = 1, \ldots, S$:
  - Draw $\theta$ from density proportional to $g$
  - Accept with probability $\frac{p(\theta|y)}{Mg(\theta)}$
- When is this difficult?
Approximation $Mg(\theta)$ should dominate $p(\theta | y)$

For $s = 1, \ldots, S$:
- Draw $\theta$ from density proportional to $g$
- Accept with probability $\frac{p(\theta | y)}{Mg(\theta)}$

When is this difficult?
Rejection sampling

- Approximation $Mg(\theta)$ should dominate $p(\theta|y)$
- For $s = 1, \ldots, S$:
  - Draw $\theta$ from density proportional to $g$
  - Accept with probability $\frac{p(\theta|y)}{Mg(\theta)}$
- When is this difficult?
10.4. Importance sampling

- Goal: $E(h(\theta)|y) = \frac{\int h(\theta)q(\theta|y)d\theta}{\int q(\theta|y)d\theta}$

- If we had draws $\theta^s$ from $q$:
  
  est. of $E(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)$

- Instead we have draws $\theta^s$ from $g$:

- When does importance sampling work well?
- When does importance sampling fail?
10.4. Importance sampling

- **Goal:** \( E(h(\theta)|y) = \frac{\int h(\theta)q(\theta|y)d\theta}{\int q(\theta|y)d\theta} \)

- If we had draws \( \theta^s \) from \( q \):

  \[
  \text{est. of } E(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)
  \]

- Instead we have draws \( \theta^s \) from \( g \):
  - **Importance weights**, \( w(\theta^s) = \frac{g(\theta^s)}{q(\theta^s)} \)
  - \( w(\theta^s) \) is the ratio of the density functions.

  \[
  \text{est. of } E(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s) w(\theta^s)
  \]

- When does importance sampling work well?
- When does importance sampling fail?
10.4. Importance sampling

- Goal: \( E(h(\theta) | y) = \frac{\int h(\theta) q(\theta | y) d\theta}{\int q(\theta | y) d\theta} \)
- If we had draws \( \theta^s \) from \( q \):

\[
est. \text{ of } E(h(\theta) | y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)\]

- Instead we have draws \( \theta^s \) from \( g \):
  - Importance weights, \( w(\theta^s) = \frac{q(\theta^s | y)}{g(\theta^s)} \)
  - \( \text{est. of } E(h(\theta) | y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s) w(\theta^s) \)
- When does importance sampling work well?
- When does importance sampling fail?
10.4. Importance sampling

- Goal: \( \mathbb{E}(h(\theta) | y) = \frac{\int h(\theta)q(\theta | y)d\theta}{\int q(\theta | y)d\theta} \)

- If we had draws \( \theta^s \) from \( q \):
  
  \[
  \text{est. of } \mathbb{E}(h(\theta) | y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)
  \]

- Instead we have draws \( \theta^s \) from \( g \):
  
  \[
  \text{Importance weights, } w(\theta^s) = \frac{q(\theta^s | y)}{g(\theta^s)}
  \]
  
  \[
  \text{est. of } \mathbb{E}(h(\theta) | y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)w(\theta^s)
  \]

- When does importance sampling work well?
- When does importance sampling fail?
10.4. Importance sampling

- Goal: \( E(h(\theta)|y) = \frac{\int h(\theta)q(\theta|y)d\theta}{\int q(\theta|y)d\theta} \)

- If we had draws \( \theta^s \) from \( q \):

\[
\text{est. of } E(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)
\]

- Instead we have draws \( \theta^s \) from \( g \):
  
  - *Importance weights*, \( w(\theta^s) = \frac{q(\theta^s|y)}{g(\theta^s)} \)
  
  - est. of \( E(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)w(\theta^s) \)

- When does importance sampling work well?

- When does importance sampling fail?
10.4. Importance sampling

- **Goal:** \( E(h(\theta)|y) = \frac{\int h(\theta)q(\theta|y)d\theta}{\int q(\theta|y)d\theta} \)

- If we had draws \( \theta^s \) from \( q \):

\[
\text{est. of } E(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)
\]

- Instead we have draws \( \theta^s \) from \( g \):
  - Importance weights, \( w(\theta^s) = \frac{q(\theta^s|y)}{g(\theta^s)} \)
  - est. of \( E(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} \frac{h(\theta^s)w(\theta^s)}{\sum_{s=1}^{S} w(\theta^s)} \)

- When does importance sampling work well?
- When does importance sampling fail?
10.4. Importance sampling

▶ Goal: \( \text{E}(h(\theta)|y) = \frac{\int h(\theta)q(\theta|y)d\theta}{\int q(\theta|y)d\theta} \)

▶ If we had draws \( \theta^s \) from \( q \):

\[
\text{est. of } \text{E}(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)
\]

▶ Instead we have draws \( \theta^s \) from \( g \):
   ▶ Importance weights, \( w(\theta^s) = \frac{q(\theta^s|y)}{g(\theta^s)} \)
   ▶ est. of \( \text{E}(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)w(\theta^s) \)

▶ When does importance sampling work well?
▶ When does importance sampling fail?
10.4. Importance sampling

- **Goal:**
  \[ E(h(\theta)|y) = \frac{\int h(\theta)q(\theta|y)d\theta}{\int q(\theta|y)d\theta} \]

- If we had draws \( \theta^s \) from \( q \):
  \[
  \text{est. of } E(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)
  \]

- Instead we have draws \( \theta^s \) from \( g \):
  - **Importance weights**, \( w(\theta^s) = \frac{q(\theta^s|y)}{g(\theta^s)} \)
  - **est. of** \( E(h(\theta)|y) : \frac{1}{S} \sum_{s=1}^{S} h(\theta^s)w(\theta^s) \)

- When does importance sampling work well?
- When does importance sampling fail?
10.5. How many simulation draws are needed?

- How many simulation draws are needed to ...
  - ...locate a posterior distribution?
  - ...compute the posterior mean to a desired precision?
  - ...compute posterior quantiles and intervals?
  - ...compute the posterior standard deviation?

- Estimating rare events
10.5. How many simulation draws are needed?

- How many simulation draws are needed to...
  - ...locate a posterior distribution?
  - ...compute the posterior mean to a desired precision?
  - ...compute posterior quantiles and intervals?
  - ...compute the posterior standard deviation?
- Estimating rare events
10.5. How many simulation draws are needed?

- How many simulation draws are needed to...
  - ...locate a posterior distribution?
  - ...compute the posterior mean to a desired precision?
  - ...compute posterior quantiles and intervals?
  - ...compute the posterior standard deviation?

- Estimating rare events
10.5. How many simulation draws are needed?

- How many simulation draws are needed to...
  - ...locate a posterior distribution?
  - ...compute the posterior mean to a desired precision?
  - ...compute posterior quantiles and intervals?
  - ...compute the posterior standard deviation?
- Estimating rare events
10.5. How many simulation draws are needed?

- How many simulation draws are needed to...
  - locate a posterior distribution?
  - compute the posterior mean to a desired precision?
  - compute posterior quantiles and intervals?
  - compute the posterior standard deviation?

- Estimating rare events
  - What is the probability that your vote is decisive?
  - Combining simulations and analytic probabilities
10.5. How many simulation draws are needed?

- How many simulation draws are needed to . . .
  - . . . locate a posterior distribution?
  - . . . compute the posterior mean to a desired precision?
  - . . . compute posterior quantiles and intervals?
  - . . . compute the posterior standard deviation?

- Estimating rare events
  - What is the probability that your vote is decisive?
  - Combining simulations and analytic probabilities
10.5. How many simulation draws are needed?

- How many simulation draws are needed to . . .
  - . . . locate a posterior distribution?
  - . . . compute the posterior mean to a desired precision?
  - . . . compute posterior quantiles and intervals?
  - . . . compute the posterior standard deviation?

- Estimating rare events
  - What is the probability that your vote is decisive?
  - Combining simulations and analytic probabilities
10.5. How many simulation draws are needed?

- How many simulation draws are needed to . . .
  - . . . locate a posterior distribution?
  - . . . compute the posterior mean to a desired precision?
  - . . . compute posterior quantiles and intervals?
  - . . . compute the posterior standard deviation?

- Estimating rare events
  - What is the probability that your vote is decisive?
  - Combining simulations and analytic probabilities
10.5. How many simulation draws are needed?

- How many simulation draws are needed to . . .
  - . . . locate a posterior distribution?
  - . . . compute the posterior mean to a desired precision?
  - . . . compute posterior quantiles and intervals?
  - . . . compute the posterior standard deviation?

- Estimating rare events
  - What is the probability that your vote is decisive?
  - Combining simulations and analytic probabilities
10.6. Computing environments

- Programming it yourself (in Matlab, R, Py, C, F, etc.)
- Special-purpose programs
- Bugs/Jags
- Stan
- Others
10.6. Computing environments

- Programming it yourself (in Matlab, R, Py, C, F, etc.)
- Special-purpose programs
- Bugs/Jags
- Stan
- Others
10.6. Computing environments

- Programming it yourself (in Matlab, R, Py, C, F, etc.)
- Special-purpose programs
  - Bugs/Jags
  - Stan
  - Others
10.6. Computing environments

- Programming it yourself (in Matlab, R, Py, C, F, etc.)
- Special-purpose programs
- Bugs/Jags
  - Stan
- Others
10.6. Computing environments

- Programming it yourself (in Matlab, R, Py, C, F, etc.)
- Special-purpose programs
- Bugs/Jags
- Stan
- Others
10.6. Computing environments

- Programming it yourself (in Matlab, R, Py, C, F, etc.)
- Special-purpose programs
- Bugs/Jags
- Stan
- Others
Computing

- What you should be able to do
  - Simulation from standard distributions
  - Vector and matrix operations
  - Generic optimizers
  - Transformations, numerical, and analytic derivatives
  - Graphics
- Fake-data debugging
- The folk theorem and the Pinocchio principle
Computing

▶ What you should be able to do
  ▶ Simulation from standard distributions
  ▶ Vector and matrix operations
  ▶ Generic optimizers
  ▶ Transformations, numerical, and analytic derivatives
  ▶ Graphics
  ▶ Fake-data debugging
  ▶ The folk theorem and the Pinocchio principle
Computing

- What you should be able to do
  - Simulation from standard distributions
  - Vector and matrix operations
  - Generic optimizers
  - Transformations, numerical, and analytic derivatives
  - Graphics
  - Fake-data debugging
  - The folk theorem and the Pinocchio principle
Computing

- What you should be able to do
  - Simulation from standard distributions
  - Vector and matrix operations
  - Generic optimizers
  - Transformations, numerical, and analytic derivatives
  - Graphics
  - Fake-data debugging
  - The folk theorem and the Pinocchio principle
Computing

- What you should be able to do
  - Simulation from standard distributions
  - Vector and matrix operations
  - Generic optimizers
  - Transformations, numerical, and analytic derivatives
  - Graphics
  - Fake-data debugging
  - The folk theorem and the Pinocchio principle
Computing

- What you should be able to do
  - Simulation from standard distributions
  - Vector and matrix operations
  - Generic optimizers
  - Transformations, numerical, and analytic derivatives
  - Graphics
- Fake-data debugging
- The folk theorem and the Pinocchio principle
Computing

What you should be able to do

- Simulation from standard distributions
- Vector and matrix operations
- Generic optimizers
- Transformations, numerical, and analytic derivatives
- Graphics

- Fake-data debugging
- The folk theorem and the Pinocchio principle
Computing

► What you should be able to do
  ▶ Simulation from standard distributions
  ▶ Vector and matrix operations
  ▶ Generic optimizers
  ▶ Transformations, numerical, and analytic derivatives
  ▶ Graphics

► Fake-data debugging
  ▶ The folk theorem and the Pinocchio principle
Computing

- What you should be able to do
  - Simulation from standard distributions
  - Vector and matrix operations
  - Generic optimizers
  - Transformations, numerical, and analytic derivatives
  - Graphics
- Fake-data debugging
- The folk theorem and the Pinocchio principle
10.7. Debugging Bayesian computing

Simple models that can be fit successfully

Complex models that cannot be fit, or that give nonsensical results
Computing tips and tricks

- Programming
  - Write scripts, don't type into the console
  - Modular: create functions and subroutines
- Graphics
- Debugging

Andrew Gelman
Bayesian Data Analysis, class 8b
Computing tips and tricks

- **Programming**
  - Write scripts, don’t type into the console
  - Modular: create functions and subroutines

- **Graphics**

- **Debugging**
Computing tips and tricks

- **Programming**
  - Write scripts, don’t type into the console
  - Modular: create functions and subroutines

- **Graphics**
  - Multiple graphs per page
  - Label all graphs

- **Debugging**
Computing tips and tricks

- **Programming**
  - Write scripts, don’t type into the console
  - Modular: create functions and subroutines

- **Graphics**
  - Multiple graphs per page
  - Label all graphs

- **Debugging**
Computing tips and tricks

- **Programming**
  - Write scripts, don't type into the console
  - Modular: create functions and subroutines

- **Graphics**
  - Multiple graphs per page
  - Label all graphs

- **Debugging**
Computing tips and tricks

- Programming
  - Write scripts, don’t type into the console
  - Modular: create functions and subroutines
- Graphics
  - Multiple graphs per page
  - Label all graphs
- Debugging
  - Work with smaller datasets
  - Strip down the model
  - Fixed parameter values, then strong priors, then weak priors
  - Network of models
Computing tips and tricks

➤ Programming
  ➤ Write scripts, don’t type into the console
  ➤ Modular: create functions and subroutines

➤ Graphics
  ➤ Multiple graphs per page
  ➤ Label all graphs

➤ Debugging
  ➤ Work with smaller datasets
  ➤ Strip down the model
  ➤ Fixed parameter values, then strong priors, then weak priors
  ➤ Network of models
Computing tips and tricks

- **Programming**
  - Write scripts, don’t type into the console
  - Modular: create functions and subroutines

- **Graphics**
  - Multiple graphs per page
  - Label all graphs

- **Debugging**
  - Work with smaller datasets
  - Strip down the model
  - Fixed parameter values, then strong priors, then weak priors
  - Network of models
Computing tips and tricks

- **Programming**
  - Write scripts, don't type into the console
  - Modular: create functions and subroutines

- **Graphics**
  - Multiple graphs per page
  - Label all graphs

- **Debugging**
  - Work with smaller datasets
  - Strip down the model
  - Fixed parameter values, then strong priors, then weak priors
  - Network of models
Computing tips and tricks

- **Programming**
  - Write scripts, don’t type into the console
  - Modular: create functions and subroutines

- **Graphics**
  - Multiple graphs per page
  - Label all graphs

- **Debugging**
  - Work with smaller datasets
  - Strip down the model
  - Fixed parameter values, then strong priors, then weak priors
  - Network of models
Computing tips and tricks

- Programming
  - Write scripts, don’t type into the console
  - Modular: create functions and subroutines

- Graphics
  - Multiple graphs per page
  - Label all graphs

- Debugging
  - Work with smaller datasets
  - Strip down the model
  - Fixed parameter values, then strong priors, then weak priors
    - Network of models
Computing tips and tricks

- **Programming**
  - Write scripts, don’t type into the console
  - Modular: create functions and subroutines

- **Graphics**
  - Multiple graphs per page
  - Label all graphs

- **Debugging**
  - Work with smaller datasets
  - Strip down the model
  - Fixed parameter values, then strong priors, then weak priors
  - Network of models
Summary of Chapter 10

- Simulation and numerical integration
- Importance sampling
- Normalizing factors
- How many simulation draws are needed?
- Computing environments
Summary of Chapter 10

- Simulation and numerical integration
  - Importance sampling
  - Normalizing factors
  - How many simulation draws are needed?
  - Computing environments
Summary of Chapter 10

- Simulation and numerical integration
- Importance sampling
- Normalizing factors
- How many simulation draws are needed?
- Computing environments
Summary of Chapter 10

- Simulation and numerical integration
- Importance sampling
- Normalizing factors
- How many simulation draws are needed?
- Computing environments
Summary of Chapter 10

- Simulation and numerical integration
- Importance sampling
- Normalizing factors
- How many simulation draws are needed?
- Computing environments
Summary of Chapter 10

- Simulation and numerical integration
- Importance sampling
- Normalizing factors
- How many simulation draws are needed?
- Computing environments
Homework due beginning of class 9b

All assignments are at http://www.stat.columbia.edu/~gelman/bda.course/homeworks.pdf

- Computing and theory problem: Program a Metropolis algorithm, understand overdispersed Poisson model
- Computing problem: Rejection sampling and importance sampling
- Applied problem: Political attitudes and social networks in General Social Survey
Homework due beginning of class 9b

All assignments are at http://www.stat.columbia.edu/~gelman/bda.course/homeworks.pdf

- Computing and theory problem: Program a Metropolis algorithm, understand overdispersed Poisson model
- Computing problem: Rejection sampling and importance sampling
- Applied problem: Political attitudes and social networks in General Social Survey
Homework due beginning of class 9b

* All assignments are at [http://www.stat.columbia.edu/~gelman/bda.course/homeworks.pdf](http://www.stat.columbia.edu/~gelman/bda.course/homeworks.pdf)
  * Computing and theory problem: Program a Metropolis algorithm, understand overdispersed Poisson model
  * Computing problem: Rejection sampling and importance sampling
  * Applied problem: Political attitudes and social networks in General Social Survey
Homework due beginning of class 9b

► All assignments are at http://www.stat.columbia.edu/~gelman/bda.course/homeworks.pdf
  ► Computing and theory problem: Program a Metropolis algorithm, understand overdispersed Poisson model
  ► Computing problem: Rejection sampling and importance sampling
  ► Applied problem: Political attitudes and social networks in General Social Survey
Homework due beginning of class 9b

▶ All assignments are at http://www.stat.columbia.edu/~gelman/bda.course/homeworks.pdf
  ▶ Computing and theory problem: Program a Metropolis algorithm, understand overdispersed Poisson model
  ▶ Computing problem: Rejection sampling and importance sampling
  ▶ Applied problem: Political attitudes and social networks in General Social Survey