Computational Statistics

Course Instructor: Ming Lin

Computational Statistics

- **Time and Location:**
  - 16:40-18:20, Wednesday; Room 204, Zuying Building
  - 10:10-11:50, Friday; Room 204, Zuying Building
- **Course Instructor:** Ming Lin
  - Email: linming50@xmu.edu.cn
  - Office: A313, Economics Building
  - Office Hour: 13:00-16:00, Thursday
- **Teaching Assistant:** Jiajing Xue
  - Email: xuestatistics@163.com
- **QQ Group:** ......;

Computational Statistics

- **Prerequisties:** Calculus, Linear Algebra, Probability Theory, Mathematical Statistics

- **References:**

  – Computational Statistics , Geof H. Givens and Jennifer A. Hoeting, John Wiley & Sons, Inc., 2nd edition, 2013.

  – Numerical Optimization, Jorge Nocedal and Stephen J. Wright, Springer, 2nd edition, 2006.

  – Simulation, Sheldon M. Ross, Elsevier, 5th edition, 2013.

- **Homework:**
  - Homework assignments are due on Wednesday, **before the class.**
  - You need use Python to do programming.

- **Grade Policy:**
  - Attendance & Homework (written questions): 5%;
  - Homework (programming questions): 10%;
  - Course Project: 25%;
  - Midterm: 30%;
  - Final Exam: 30%.

- **Course Outline:**

  1. Optimization and Solving Nonlinear Equations
  2. EM Optimization Methods
  3. Numerical Integration
  4. Simulation and Monte Carlo Integration
  5. Markov Chain Monte Carlo
  6. Bootstrapping

1. Optimization and Solving Nonlinear Equations

- **Optimization Problem:** We want to find a point $\theta^* \in \Theta$ (for example, $\Theta = \mathbb{R}^p$) to maximize (or minimize) an *objective function* $g(\theta)$, denoted by

$$\theta^* = \arg\max_{\theta\in\Theta} g(\theta).$$

Under certain conditions, it is equivalent to solving equation $\nabla g(\theta) = 0$.

- **Maximum Likelihood Estimate (MLE):** Let $X_1, X_2, \cdots, X_n$ be a random sample following a distribution with probability density function (PDF) $f(x_1, \cdots, x_n; \theta)$, where θ is a $p \times 1$ vector.

  – For each given $x_1, \cdots, x_n$, $f(x_1, \cdots, x_n; \theta)$ considered as a function of the parameter θ is called the *likelihood function* and denoted by $l(\theta)$.

  – The MLE of θ is

$$\hat{\theta}_{MLE} = \arg\max_{\theta\in\Theta} l(\theta) = \arg\max_{\theta\in\Theta} \log l(\theta).$$

1. Optimization and Solving Nonlinear Equations

- We often use an iterative updating step

$$\theta^{(t+1)} = \theta^{(t)} + \alpha_t u^{(t)}$$

to search for θ^*, where $u^{(t)}$ is a $p \times 1$ direction vector and $\alpha_t > 0$ is the *step size*. Consider Taylor series expansion, we have

$$g(\theta^{(t+1)}) = g(\theta^{(t)} + \alpha_t u^{(t)}) \approx g(\theta^{(t)}) + \alpha_t \nabla g(\theta^{(t)})^T u^{(t)}.$$

When $\nabla g(\theta^{(t)})^T u^{(t)} > 0$ and α_t is not too large, we can have $g(\theta^{(t+1)}) > g(\theta^{(t)})$.

  – How to choose $u^{(t)}$?

  – How to decide α_t?

2. EM Optimization Methods

- Suppose we want to estimate parameter θ in model $f_{XY}(x, y; \theta)$, but we can not observe x.
  - In this case, the MLE of θ is

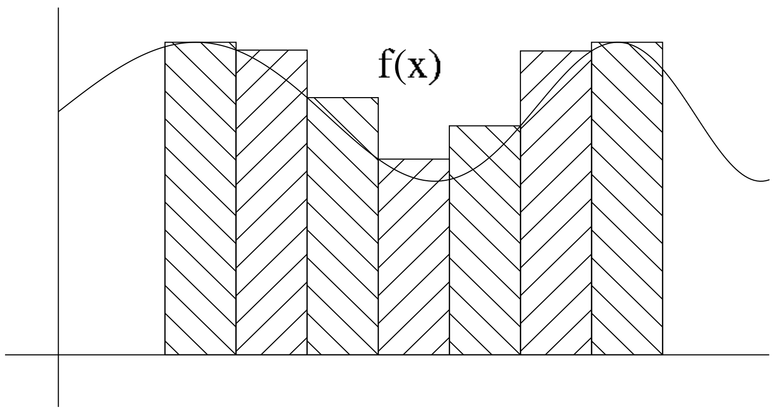
$$\begin{aligned}\theta_{MLE} &= \arg\max_{\theta} l(\theta) \\ &= \arg\max_{\theta} f_Y(y; \theta) = \arg\max_{\theta} \int f_{XY}(x, y; \theta)\, dx.\end{aligned}$$

  - The integration $\int f_{XY}(x, y; \theta)\, dx$ may be difficult to calculate.
  - The expectation-maximization (EM) algorithm provides a strategy to find the MLE in this case.

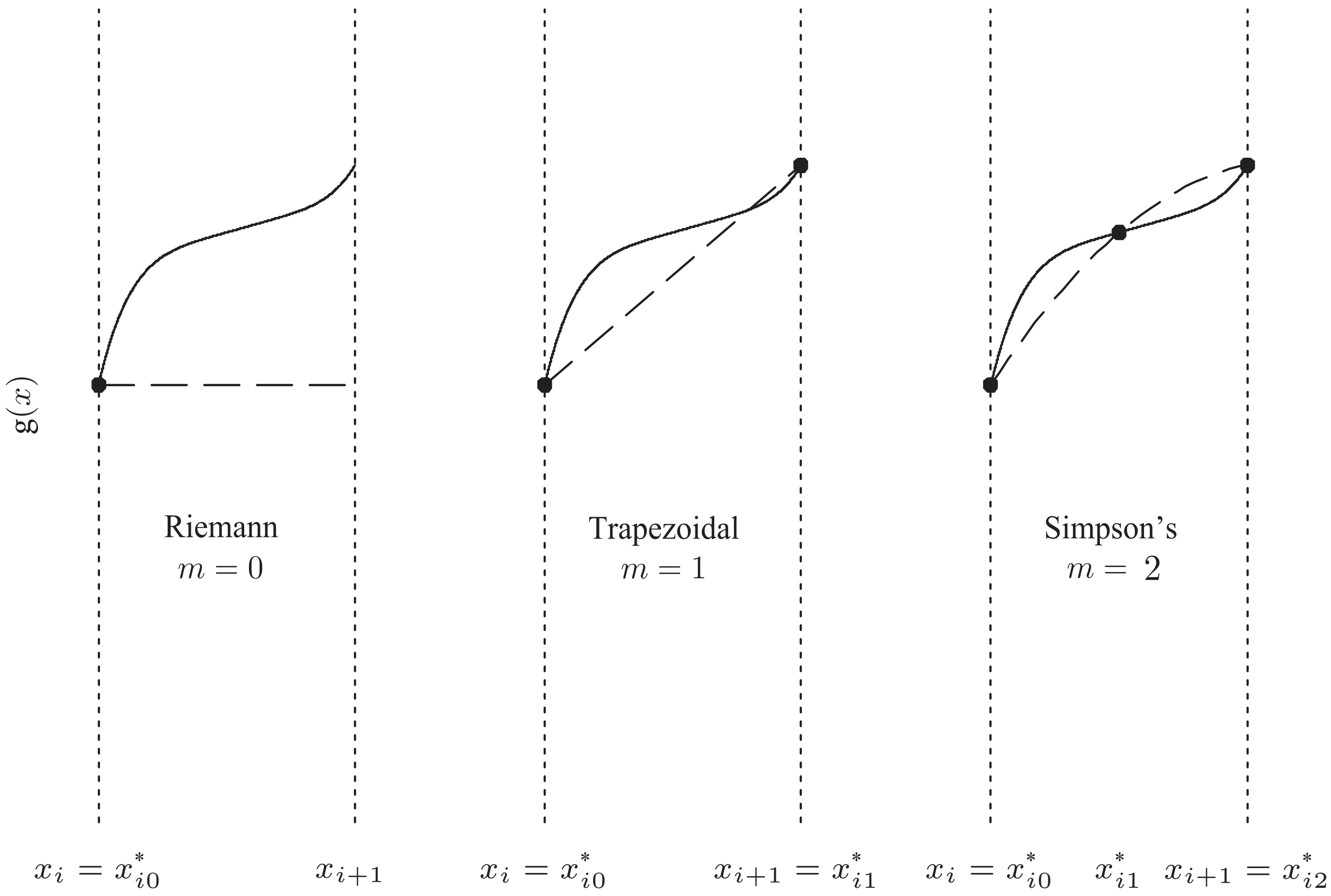
3. Numerical Integration

- Suppose we want to calculate integration $\int g(x)\,dx$ for some function $g(\cdot)$.
  - For example, when $g(x) = f_{XY}(x, y)$ for a given y, then $\int g(x)\,dx = \int f_{XY}(x, y)\,dx = f_Y(y)$.
  - **Numerical Integration:** Let $a = x_0 < x_1 < \cdots < x_{m-1} < x_m = b$ be a partition of interval $[a, b]$ and ξ_j is any point between x_{j-1} and x_j. Then

$$\widetilde{\Pi} \triangleq \sum_{j=1}^{m} g(\xi_j)(x_j - x_{j-1}) \to \int_a^b g(x)\,dx.$$



3. Numerical Integration

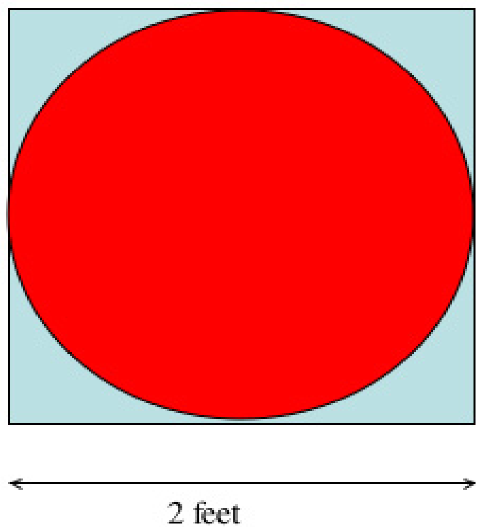


Riemann Rule, trapezoidal rule and Simpson's rule

4. Simulation and Monte Carlo Integration

- **Example: Calculating π.** A circle of radius 1 will have area equal to π, and a square drawn around that circle will have area 4. If we draw samples (x_j, y_j), $j = 1, \cdots, m$, uniformly distributed within the square, then

$$\frac{1}{m}\sum_{j=1}^{m} I(x_j^2 + y_j^2 < 1) \approx \pi/4.$$



4. Simulation and Monte Carlo Integration

- Calculate π using simulation:
  - Generate random samples (x_j, y_j), $j = 1, \cdots, m$, where $x_j \sim U(-1, 1)$ and $y_j \sim U(-1, 1)$.
  - Estimate π by

$$\widehat{\pi} = 4 \cdot \frac{1}{m} \sum_{j=1}^{m} I(x_j^2 + y_j^2 < 1),$$

where $I(\cdot)$ is the **indicator function**. Here

$$I(x_j^2 + y_j^2 < 1) = \begin{cases} 1, & \text{if } x_j^2 + y_j^2 < 1; \\ 0, & \text{if } x_j^2 + y_j^2 \geq 1. \end{cases}$$

4. Simulation and Monte Carlo Integration

- **Monte Carlo Methods:**
  - Wikipedia: Monte Carlo methods are a broad class of computational algorithms that rely on repeated **random sampling** to obtain numerical results.
  - It was named, by Stanislaw Ulam and Nicholas Metropolis, after the Monte Carlo Casino.

4. Simulation and Monte Carlo Integration

- **Monte Carlo Integration:** Suppose we want to calculate $\int g(x)\,dx$.
  - Generate random samples $x^{(1)}, \cdots, x^{(m)}$ from a *trial distribution* (or *sampling distribution*) with PDF $q(x)$.
  - Calculate

$$\begin{aligned}
\widehat{\Pi} &= \frac{1}{m}\sum_{j=1}^{m}\frac{g(x^{(j)})}{q(x^{(j)})} \\
&\to E\left(\frac{g(x^{(j)})}{q(x^{(j)})}\right) = \int \frac{g(x)}{q(x)}\,q(x)dx = \int g(x)\,dx.
\end{aligned}$$

- **Example: Calculating π.** We want to calculate integration

$$\pi = \int_{x^2+y^2<1} 1\, dxdy = \int I(x^2+y^2<1)\, dxdy.$$

  – Generate random samples (x_j, y_j), $j = 1, \cdots, m$, where $x_j \sim U(-1,1)$ and $y_j \sim U(-1,1)$. The PDF of the trial distribution is

$$q(x,y) = 1/4 \quad \text{for } -1 < x, y < 1.$$

  – Estimate π by

$$\begin{aligned} \widehat{\pi} &= \frac{1}{m} \sum_{j=1}^{m} \frac{I(x_j^2 + y_j^2 < 1)}{q(x_j, y_j)} \\ &= 4 \cdot \frac{1}{m} \sum_{j=1}^{m} I(x_j^2 + y_j^2 < 1). \end{aligned}$$

4. Simulation and Monte Carlo Integration

- The convergence rate of Monte Carlo integration is

$$\sqrt{\mathrm{Var}(\widehat{\Pi})} = \sqrt{\frac{1}{m}\mathrm{Var}\left(\frac{g(x^{(j)})}{q(x^{(j)})}\right)} = \frac{c}{\sqrt{m}}.$$

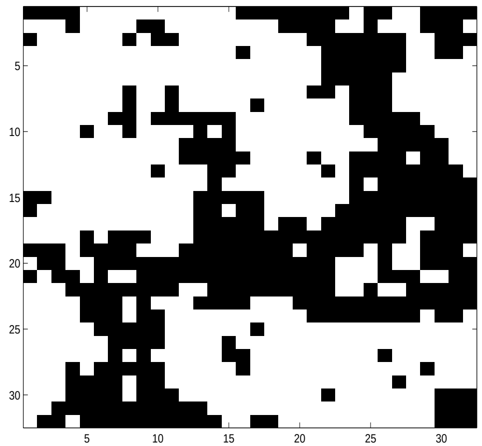
- In one-dimensional cases, the convergence rate of numerical integration is $1/m$ (or even faster).

- However, the convergence rate of numerical integration will decrease as the dimension of x increases.

5. Markov Chain Monte Carlo

- **Markov Chain Monte Carlo (MCMC):** We want to compute $E_f[g(X)] = \int g(x)f(x)\,dx$, where the dimension of x is large. The MCMC algorithm generates a sequence of random variables $x^{(1)}, \cdots, x^{(m)}$ (a Markov chain), so that $\frac{1}{m}\sum_{i=1}^{m} g(x^{(i)}) \xrightarrow{a.s.} E_f[g(x)]$.

- **Ising Model:** In a magnet field, the atomic spins on a $N \times N$ lattice space, $\mathcal{L} = \{(i,j) : i,j = 1,\cdots,N\}$, can be represented by a random matrix $\boldsymbol{X} = \{X_{i,j}\}_{N\times N}$. Each $X_{i,j}$ is either 1 or -1.



- The random matrix X follows a distribution with the form

$$\pi(\boldsymbol{x}) = P(\boldsymbol{X} = \boldsymbol{x}) = \frac{1}{S}\, e^{-U(\boldsymbol{x})/kT} \propto e^{-U(\boldsymbol{x})/kT},$$

  where $\boldsymbol{x} = \{x_{i,j}\}_{N\times N}$, k is the Boltzmann constant, T is the temperature, $S = \sum_{\boldsymbol{x}} e^{-U(\boldsymbol{x})/kT}$ is the normalizing constant.

  – The potential function is

$$U(\boldsymbol{x}) = -J \sum_{(i,j)\sim(i',j')} x_{i,j}\, x_{i',j'} + \sum_{i,j} h_{i,j}\, x_{i,j},$$

    where the symbol $(i,j) \sim (i',j')$ means that the two sites are neighbors, J is called the *interaction strength*, $\{h_{i,j}\}_{N\times N}$ is the magnetic field.

  – We want to calculate the *internal energy*, which is defined as

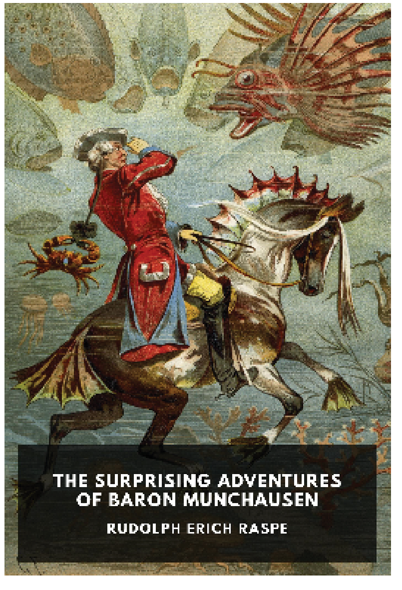
$$E\big[U(\boldsymbol{X})\big] = \sum_{\boldsymbol{x}} U(\boldsymbol{x})\pi(\boldsymbol{x}) = \frac{\sum_{\boldsymbol{x}} U(\boldsymbol{x}) e^{-U(\boldsymbol{x})/kT}}{\sum_{\boldsymbol{x}} e^{-U(\boldsymbol{x})/kT}}.$$

6. Bootstrapping

- Suppose we have one realization $x_1, \cdots, x_n$ of i.i.d. random variables $X_1, \cdots, X_n$ from a population P. We often want to know the distribution of a statistic $T(X_1, \cdots, X_n)$, especially for testing problems.
  - We only have one observation $T(x_1, \cdots, x_n)$, how to estimate the distribution of $T(X_1, \cdots, X_n)$?
  - **Bootstrap Method:**
    * For $b = 1, \cdots, B$,
      · Randomly draw samples $\{x_1^{*(b)}, \cdots, x_n^{*(b)}\}$ from $\{x_1, \cdots, x_n\}$ **with replacement**;
      · Compute $T^{*(b)} = T(x_1^{*(b)}, \cdots, x_n^{*(b)})$.
    * **Under certain conditions**, we can use $\{T^{*(1)}, \cdots, T^{*(B)}\}$ to estimate the distribution of $T(X_1, \cdots, X_n)$.

6. Bootstrapping

- The use of the term "**bootstrap**" derives from the phrase to *pull oneself up by one's bootstrap*.
  - Adventures of Baron Munchausen by Rudolph Erich Raspe: "The Baron had fallen to the bottom of a deep lake. Just when it looked like all was lost, he thought to pick himself up by his own bootstraps".



6. Bootstrapping

- **Principle of Bootstrap:** Suppose we have one realization $x_1, \cdots, x_n$ of i.i.d. random variables $X_1, \cdots, X_n$ from a population P. We are interested in the distribution of a statistic $T(X_{1:n})$.
  - Assume that the true population P is known.
    * Compute the exact distribution of $T(X_{1:n})$ using P.
    * Draw new sample sets $\{x_{1:n}^{(b)}\}$, $b = 1, \cdots, B$, from P, compute $T(x_{1:n}^{(b)})$, and use $T(x_{1:n}^{(1)}), \cdots, T(x_{1:n}^{(B)})$ to estimate the distribution of $T(X_{1:n})$.
  - The true population P is unknown in most cases.
    * Use limiting theories to develop the asymptotic distribution of $T(X_{1:n})$.
    * **Bootstrap:** Consider $\{x_1, \cdots, x_n\}$ as the "true" population. Draw new sample sets $\{x_1^{*(b)}, \cdots, x_n^{*(b)}\}$, $b = 1, \cdots, B$, from $\{x_1, \cdots, x_n\}$ randomly with replacement, compute $T(x_{1:n}^{*(b)})$, and use $T(x_{1:n}^{*(1)}), \cdots, T(x_{1:n}^{*(B)})$ to estimate the distribution of $T(X_{1:n})$.

Homework

1. Use the Monte Carlo method to estimate the value of π with sample sizes of $m = 10000$, 20000, and 40000. Repeat each experiment 500 times and report the root mean squared errors (RMSE) for each sample size.