Modern Computational Statistics

Fall 2019

Instructor: Cheng Zhang

Time: Monday 6:40-8:30pm, odd Wednesday 8:00-9:50am

Location: Classroom Building No.3, Room 504

Contact: chengzhang@math.pku.edu.cn

Office Hours: Wednesday 10:00-11:00am or by appointment, 1279 Science Building No.1

Web Page: `https://zcrabbit.github.io/courses/mcs-f19.html`. Visit this page regularly. It will contain homework assignments, lectures, etc.

Description and Objectives

Computational statistics is a branch of mathematical sciences focusing on efficient numerical methods for problems arising in statistics analysis. The goal of this course is to provide students an introduction to a variety of modern computational statistical techniques and the role of computation as a tool of discovery. Topics include numerical optimization in statistical inference including expectation-maximization (EM) algorithm, Fisher scoring, etc., numerical integration approaches including numerical quadrature and Monte Carlo methods, and approximate Bayesian inference methods including Markov chain Monte Carlo methods, variational inference and their scalable counterparts, with applications in statistical machine learning, computational biology and other related fields. Additional topics may vary. Coursework will include computer assignments.

Prerequisites

Some background in probability and statistical inference equivalent to two quarters of upper division or graduate coursework. Prior experience programming in **python** or R is helpful.

Assignments and Grading Policy

There will be 4 problem sets ($4 \times 15\% = 60\%$), and a final project (40%) which includes a midterm proposal (5%) and a final write-up (35%). There will be 7 free late days in total, use them in your own ways. Afterwards, late homework will be discounted by 25% for each additional day. Not accepted after 3 late days per problem set (PS). Late policy does not apply to the final project, please submit it on time. Discussing assignments verbally with classmates is allowed and encouraged. However, you should finish your work independently. Identified cheating incidents will be reported and will result in zero grades.

Computer and Technical Requirements

We will use python during the course. A good Python tutorial is available at `http://www.scipy-lectures.org/`. You may also find another shorter tutorial useful at `http://cs231n.github.io/python-numpy-tutorial/`. If you have never used Python before, I recommend using Anaconda Python 3.7 `https://www.continuum.io/`.

Recommended Texts

- Givens, G. H. and Hoeting, J. A. (2005) Computational Statistics, 2nd Edition, Wiley-Interscience.
- Gelman, A., Carlin, J., Stern, H., and Rubin, D. (2003). Bayesian Data Analysis, 2nd Edition, Chapman & Hall.
- Liu, J. (2001). Monte Carlo Strategies in Scientific Computing, Springer-Verlag.
- Lange, K. (2002). Numerical Analysis for Statisticians, Springer-Verlag, 2nd Edition.
- Hastie, T., Tibshirani, R. and Friedman, J. (2009). The Elements of Statistical Learning, 2nd Edition, Springer.
- Goodfellow, I., Bengio, Y. and Courville, A. (2016). Deep Learning, MIT Press.

Tentative Outline of Topics

1. Basic concepts in statistical computing: likelihood; exponential family; Bayesian inference; Markov chains, etc.
2. Modern optimization methods: Newton's Method; Fisher Scoring; Gradient Descent (GD); Stochastic Gradient Descent (SGD); Adaptive Stochastic Gradient Descent.
3. Numerical integration and Monte Carlo approximation: Basic quadrature; simple simulation methods; rejecting sampling; importance sampling; variance reduction techniques.
4. Markov chain Monte Carlo: Metropolis-Hasting; Gibbs sampling; slice sampling; Hamiltonian Monte Carlo; stochastic gradient MCMC.
5. Expectation Maximization: Minority Maximization (MM); EM algorithm and its variants; Variational Bayesian EM (VBEM).
6. Variational inference: mean-field; stochastic gradient optimization (control variate and the reparameterization trick); scalable approaches; choice of training objectives.
7. Applications in Bayesian deep learning: variational autoencoder (VAE); generative adversarial networks (GAN); flow based methods.