BIOSTATISTICS 235
Statistical Computing - Basic Principles and Applications

Description:

The objective of this course is to prepare the student for effective use of the ever increasing computing power and introduce basic concepts and methods of computing as tools for statistical work and research. The three main themes are software design, numerical methods and simulation methods. Topics include software development, algorithms and data structures, enumeration problems, some computational biology algorithms, function approximation, matrix computations, linear and nonlinear systems, the EM algorithm, simulation experiments, Gibbs sampling and Markov chain Monte-Carlo, symbolic computation and document preparation (\LaTeX). Most topics will be implemented in actual working programs and documents.

Prerequisites: BIOS 160 and 161, basic statistical methods (example: BIOS 162 and 163), and familiarity with at least one computer system and programming language (R, SAS/IML, FORTRAN, PASCAL, etc.).

Instructor: Bahjat Qaqish, 966-7271, Room 3105-B, bahjat.qaqish@unc.edu
Programming languages used in the course: C++, MATLAB, R.
Computer system: Sun workstations (running UNIX).
Class directory: /distrib/bios235/
Website: ftp://ftp.bios.unc.edu/distrib/bios235/

Outline:

Introduction to UNIX, introduction to C++, compiling and running programs.

Computer organization: bits, bytes, words, number systems, floating point representations (IEEE 754), floating point errors.

Elements of programming: data structures and algorithms, software design and development basics, program performance, worst case, best case, average case, profiling, examples (finding an array minimum, sorting algorithms).

Project maintenance with “make”, basic ideas of data abstraction, top-down design, bottom-up design, layering, modularization, testing, debugging, documentation.

Enumeration problems: Spearman’s correlation, the Wilcoxon rank sum test, 2 \times 2 tables, R \times C tables.

Dynamic programming: basic ideas, application to string matching of DNA sequences. Computations for hidden Markov chains.

Solving non-linear equations in one variable: bisection, Newton-Raphson, secant, error analysis, convergence rates, linear and quadratic convergence.

Numerical integration and differentiation: open and closed rules (trapezoidal, Simpson’s, ...), globally-adaptive quadrature, Gaussian quadrature, Gauss-Legendre quadrature, Gauss-Hermite quadrature, adaptive Gauss-Hermite quadrature, applications to generalized mixed models.

Matrix computations: linear spaces, vector and matrix norms, projectors, orthogonal projectors, QR decomposition, QR via the Gram-Schmidt algorithm, Householder reflectors, QR via Householder reflectors, SVD, eigenvalues and eigenvectors, Cholesky decomposition, back substitution, forward substitution, the least-squares problem, error analysis, condition number, accuracy and stability (general), accuracy and stability of least-squares algorithms, other LU factorizations (Doolittle, Crout), Gaussian elimination, applications to linear regression, iterative methods for linear systems (Jacobi, Gauss-Seidel), iteration and convergence for linear systems.

Solving non-linear systems: Newton-Raphson in several dimensions, applications to maximum-likelihood estimation, Fisher scoring, conjugate gradient.

The EM algorithm: definition, properties, advantages and disadvantages, computing the observed information, acceleration, examples, applications to mixture models, misclassified data, generalized mixed models.

Simulation: overview, random number generation and testing, design of simulation experiments, generating uni- and multi-variate, continuous and discrete, accept-reject methods, Markov-Chain Monte Carlo, coalescence, coupling from the past, the perfect sampler, Gibbs sampling, Metropolis algorithm, Metropolis-Hastings algorithm, importance sampling, Monte-Carlo integration, variance reduction, Rao-Blackwellization, sampling with and without replacement from a finite population, random permutations, parametric and non-parametric bootstrap sampling.

Example simulation experiment: The robustness of the one-sample $t$ confidence interval.

Example simulation experiment: Simulating a stochastic process.

Introduction to symbolic computation (Mathematica).

Introduction to document preparation in LaTeX.

Linking C code to R.

Linking C code to MATLAB.

REFERENCES

Algorithms:


**Numerical methods:**


**Matrix analysis:**


**Matrix computations:**


Mathematics.


C:


C++ (Introductory):


C++ (Intermediate and Advanced):


Software development:


General statistical computing:


**Simulation:**


**Other topics:**


**Internet resources:**

The Association of C & C++ Users: http://www.accu.org/
Blitz++ matrix library: http://www.oonumerics.org/blitz/
GSL, GNU Scientific Library: http://www.gnu.org/software/gsl/
The Matrix Template Library: http://www.osl.iu.edu/research/mtl/
National Institute of Standards and Technology: http://math.nist.gov/
Netlib collection of mathematical software, papers, and databases: http://www.netlib.org/
R Project for Statistical Computing: http://www.r-project.org/