Distance Learning Course Applied Numerical Methods

Foundations for Computational Finance, Engineering and Data Sciences

Contents

Part D. Orthogonal Polynomials

Elementary Theory

- Fundamental recurrence formulae
- Kernel polynomials
- Zeroes of orthogonal polynomials
- Rodrigues' formula

Special Cases of Orthogonal Polynomials

- Legendre polynomials and Legendre series
- Hermite polynomials
- Laguerre polynomials
- Generating functions

Applications of Orthogonal Polynomials

- Gaussian quadrature
- Solution of differential equations
- Bessel functions

Part E. Numerical Differentiation

General Considerations

- Taylor's theorem
- The different ways to calculate derivatives
- Gradient, Jacobian and Hessian
- Multiprecision data types

Finite Difference Method

- Forward, backward and centred approximations
- Truncation and roundoff errors
- Scalar and vector cases
- Approximating the gradient
- Catastophic cancellation

The Complex Step Method

- Semiautomatic differentiation
- Motivation: scalar functions for first derivative
- Resolving subtractive (catastrophic) cancellation
- Approximating second derivative
- Extending the method to compute gradient and Hessian
- Directional derivatives and Jacobian
- Computing Fréchet derivatives

Applying the Complex Step Method

- Computing sensitivities in finance, engineering and science
- First and second-order option greeks
- Combining the method with analytical solution
- Choice: Complex Method versus Automatic Differentiation
- Engineering applications

Automatic Differentiation (AD)

- Introduction and motivation
- Forward mode
- Reverse mode
- Using the *DiffSharp* package in C# and F#

IEEE 754

- Overview of IEEE 754
- Numerics and IEEE 754
- Rounding rules and exception handling
- Normal, subnormal and infinite numbers; NaN
- Machine precision
- Rounding and cancellation errors

Part F. Numerical Quadrature

Introduction to numerical integration

- Rectangle, Trapezoidal and Romberg's methods
- Gaussian rules
- Composite rules
- Adaptive quadrature
- Extrapolation to the limit

Part G. The Solution of Nonlinear Equations

Introduction

- Overview of iterative methods
- Fixed point iteration
- Linear and quadratic convergence
- Newton's method

Advanced and Other Methods

- Roots of polynomials
- Newton-Raphson method for nonlinear systems of equations
- Bracketing and Bisection
- Homotopy and continuation methods

Part H. The Finite Difference Method (FDM) in a Nutshell

(discussed in more detail in FDM course)

Ordinary Differential Equations (ODEs)

- Scalar ODEs and systems of ODEs
- Stiff and non-stiff ODEs
- Numerical solutions of ODEs
- Boost C++ *odeint* library

Partial Differential Equations (PDEs)

- Types of PDEs
- Explicit and implicit finite difference methods
- The heat equation as model
- Stability and convergence of finite difference schemes

FDM for Heat Equation

- Statement of initial boundary value problem
- Types of boundary conditions
- Crank-Nicolson method
- Alternating Direction Explicit (ADE) method
- Method of Lines (MOL)
- The FD algorithm structure

Part I. Introduction to Probability

Probability '101'

- Sample space and events
- Axioms of probability
- Conditional probability
- Independent events

Random Variables

- Distribution functions
- Discrete and continuous random variables
- Probability mass and probability density functions
- Multiple random variables

Statistical Distributions

- Discrete and continuous distributions
- Univariate and bivariate normal probability distributions
- Uniform distribution
- Chi squared and non-central chi squared distributions
- Poisson distribution
- C++11 and Boost C++ support

Random Number Generation

- Random number engines
- Linear congruential
- Mersenne Twister
- Nonlinear generators
- C++11 and Boost C++ support

Part J. Optimisation

Overview

- What is optimisation?
- Univariate and multivariate optimisation
- Constrained and unconstrained optimisation
- Application areas

Univariate Optimisation

- Local and global minima
- Three-point interval search
- Fibonacci and Golden-Mean search methods
- Brent's method

Multivariate Unconstrained Optimisation

- Line search methods and trust region methods
- Step length and Wolfe conditions
- Steepest Descent Method
- Newton's Method and its variants

Multivariate Constrained Optimisation

- Lagrange multipliers
- Penalty Method
- Kuhn-Tucker conditions
- Barriers

Least-Squares Problems

- Linear least-squares problems
- Gauss-Newton method
- Levenberg-Marquardt method
- Orthogonal distance regression

An Introduction to Genetic Algorithms (GA)

- What is a genetic algorithm?
- A simple genetic algorithm
- Schemata
- Reproduction, crossover and mutation
- Objective functions and fitness
- Classifier systems

Differential Evolution (DE)

- Overview
- Derivative-based and derivative-free methods
- Global optimisers versus local optimisers
- Parameter representation
- Problem domains

DE Algorithm Details

- Initialisation
- Base vector selection
- Differential mutation
- Recombination
- Selection
- Termination criteria

Examples of DE and Generalisations

- DE for least-squares problems
- Simulated annealing
- Computing implied volatility using least-squares DE
- Computing default probability
- Adaptive DE
- Constrained optimisation

Software Libraries

- C++11
- Boost C++ libraries
- MathNet (C#)
- Python numerical and scientific libraries
- DiffSharp and Adept libraries

Part K Mathematical and Algorithmic Foundations for Machine Learning (ML)

The main goal of this module is to introduce a number of important algorithms in ML by discussing them in detail, how they are based on numerical and statistical methods and representative code examples in C++, Python and C#. In this way we make this popular topic accessible to a wide audience. We wish to provide insights and a stepping stone into algorithms as used in packages such as *Scikit* and *TensorFlow*. In general, we wish to understand what is happening inside these computational *black boxes*.

Essential Numerical Methods

- What is approximation?
- Local and global minima, finding them
- Underflow, overflow and conditioning
- Approximating derivatives and gradients

Essential Statistical Methods

- Activation functions
- Maximum Likelihood Estimation (MLE)
- Information Theory
- Bayesian Statistics

Mathematical Foundations of Markov Chains

- Stochastic matrices (left, right, double)
- Substochastic matrices
- Nonnegative and positive matrices
- Perron-Frobenius theorem
- Spectral analysis of stochastic matrices

An Introduction to Markov Chains

- The Markov property
- State space, stochastic matrices and state diagrams
- Continuous and discrete time Markov chains
- Transition probability and transition rate (intensity) matrix
- Kolmogorov forward equation
- Steady-state analysis and limiting distributions
- Hitting times

Training Models

- Linear and polynomial regression
- Gradient Descent (GD), (Mini-)Batch GD and Stochastic Gradient Descent (SGD)
- Regularised linear models
- Logistic and Softmax regression
- Cross Entropy

Your Trainer

Daniel J. Duffy started the company Datasim in 1987 to promote C++ as a new object-oriented language for developing applications in the roles of developer, architect and requirements analyst to help clients design and analyse software systems for Computer Aided Design (CAD), process control and hardware-software systems, logistics, holography (optical technology) and computational finance. He used a combination of top-down functional decomposition and bottom-up object-oriented programming techniques to create stable and extendible applications (for a discussion, see Duffy 2004 where we have grouped applications into domain categories). Previous to Datasim he worked on engineering applications in oil and gas and semiconductor industries using a range of numerical methods (for example, the finite element method (FEM)) on mainframe and mini-computers.

Daniel Duffy has BA (Mod), MSc and PhD degrees in pure and applied mathematics and has been active in promoting partial differential equation (PDE) and finite difference methods (FDM) for applications in computational finance. He was responsible for the introduction of the Fractional Step (Soviet Splitting) method and the Alternating Direction Explicit (ADE) method in computational finance. He is also the originator of the exponential fitting method for time-dependent partial differential equations.

He is also the originator of two very popular C++ online courses (both C++98 and C++11/14) on www.quantnet.com in cooperation with Quantnet LLC and Baruch College (CUNY), NYC. He also trains developers and designers around the world on-site and on line. He can be contacted dduffy@datasim.nl for queries, information and course venues, in-company course and course dates