Syllabus for Math 623 (Computational Finance), Winter 2018.

- **Instructor**: Sergey Nadtochiy (sergeyn@umich.edu, 2854 EH).

- **Grader**: Zeyu Zhang (zeyuz@umich.edu).

- **Synopsis**: This is a course on computational methods in finance and financial modeling. Using financial mathematics (like many branches of applied mathematics) in practice involves three tasks. First, one needs to develop mathematical models that accurately describe the real-life phenomena that one wishes to study – in the present case, probabilistic models for the evolution of prices, interest rates, and other relevant quantities. Once a model is chosen, the second task is to derive theoretical equations, or formulas, which establish relations between various objects in the financial markets: for example, the prices of derivative securities (options, bonds, etc), and the risk profiles of investment portfolios, as functions of risk factors. Finally, one needs to design and implement numerical methods to perform computations based on these formulas and equations. This course is concerned with the latter task, and it has three components. In the first part, we will study the lattice (or, tree) methods, which correspond to the models based on discrete time Markov chains (e.g. the binomial model). We will discuss the pricing and hedging of financial derivatives in such models, using the arbitrage theory, or, more specifically, the risk-neutral pricing. We will, then, proceed to analyze the diffusion-based models of financial mathematics (including, e.g., the Black-Scholes model) and the associated Partial Differential Equations (PDEs). We will discuss the finite difference methods, which provide numerical approximations for solutions to these PDEs. Both explicit and implicit schemes will be studied, the concepts of stability and convergence will be introduced, and a connection between the finite difference schemes and lattice methods will be established. After that, we will turn to the Monte Carlo simulations – the most general computational method for probabilistic equations. This method is based on generating a large number of paths of the underlying stochastic processes, in order to approximate the expectations of certain functions of these paths (which, e.g., may determine prices, portfolio weights, default probabilities, etc.). In addition to the standard Monte Carlo algorithms, we will study the variance reduction techniques, which are often necessary to obtain accurate results. The computational methods presented in this course will be illustrated using the popular models of equity markets (e.g. Black-Scholes, Heston), fixed income (e.g. Vasicek, CIR, Hull-White, Heath-Jarrow-Morton) and credit risk (e.g. Merton, Black-Cox, reduced-form models). We will also address the issue of calibration – i.e. the problem of finding the values of the model’s parameters that are consistent with the observed data. In the homeworks, you will implement these models and the associated computational methods in MatLab.

- **Prerequisites**: good understanding of the concepts from Probability Theory (probability measures, random variables, expectations, cumulative distribution and probability density functions, conditional probabilities and independence, law of large numbers, central limit theorem), Stochastic Analysis (stochastic processes, martingales, Brownian motion, stochastic integration, Ito’s formula, SDEs), Differential Equations (ODEs, as well as elliptic and parabolic PDEs), Mathematical Finance (arbitrage theory, binomial models, Black-Scholes and other diffusion-based models), basic Numerical Methods (numerical methods for systems of linear equations, interpolation methods, numerical integration, bisection method, Newton’s method), and Computer Programming (MatLab).

- **Textbooks**:  
  - **Required**: Monte Carlo Methods in Financial Engineering (Glasserman 2004). In addition, lecture notes will be posted in the Files section of canvas.

• Classes: Tue/Thu, 8:30-10am, except Spring Break. First class on Jan 4, last class on Apr 17. Lecture room: 3088 EH.

• Attendance: you are strongly encouraged to attend the lectures.

• Coursework: Homeworks and final exam.

  – There will be approximately 5 homeworks (hwks) in the course. You will be given at least one week to complete each hwk. The submission deadlines are specified when the hwk is assigned. Hwks and their solutions will be posted in the Files section of canvas. Each submission deadline will be specified via an announcement, when a hwk is assigned. The parts of completed hwk assignments that do not include MatLab code should be submitted either electronically, via the Assignments section of canvas, or in class, or placed into the outside dropbox on the door of room 2854 EH. In case you choose to put your assignment in the dropbox, you should do so during the time period specified when the hwk is assigned (typically two hours before the deadline), to ensure that it is not lost. The instructor is not responsible for any hwks that are stolen from the dropbox. If the assignment requires you to submit MatLab code, the corresponding MatLab files must be submitted electronically, via the Assignments section of canvas.

  – Final exam is closed-book, but a cheat sheet is allowed, provided its contents comply with the rules (the rules will be posted on canvas before each exam). There will be no makeup exams.

     * Final exam is scheduled at 8-10am on Apr 26, in the usual lecture room (3088 EH).

  – There will be no hwks during the first two weeks of the course. All material posted during that time, will also appear on the instructor’s personal webpage: http://www-personal.umich.edu/ sergeyn/

• Grading: Hwks – 50%, Final – 50%. The final grades will be distributed according to the table contained in the file named ‘grade_options’ and located in the Files section of canvas. The first column of the table represents the grade, the second column shows the minimum score (in percent) needed to achieve this grade. Note that the distribution of final grades may be curved but only in your favor. In other words, the numbers in the second column may be decreased but they will not be increased.

• Instructor’s office hours: Tue 10:15–11:15am and 3–4pm, Wed 10–11am, in 2854 EH.

• Tentative course schedule.

  – Lattice (tree) methods.

     * One-period binomial models: no-arbitrage theorem in this model (example of arbitrage via wrong interest rate, computation of risk-neutral probabilities, presence of dividend), arbitrage (or, risk-neutral) pricing in this model (vanilla options), completeness of this model.

     * Multi-period tree models: recombining tree (via geometrically uniform grid), computation of risk-neutral probabilities, recursive relation for the prices of vanilla options via no-arbitrage, trinomial model and its incompleteness.

     * Pricing exotic options on trees: barrier (UOP), american put (formulation of price via optimal stopping, DPP, example).
* **Implied trees**: calibration in general and to vanilla options, Arrow-Debreu securities, calibration to Arrow-Debreu prices, Arrow-Debreu prices via vanilla options, numerical issues.

– **PDE methods**.

* **Diffusion models and associated PDEs**: Black-Scholes PDE (derivation via no-arbitrage, delta-hedging and Ito’s lemma; or via FTAP and Ito’s lemma; explicit solution to Black-Scholes PDE), local volatility models and associated pricing PDE (implied smile; general diffusions; brief derivation of the pricing PDE), examples (Shifted-Lognormal, CEV, Vasicek, CIR).

* **Finite-difference methods for parabolic PDEs**: introduction via ODEs, explicit and implicit methods for parabolic PDEs (boundary conditions, convergence, stability, solution to linear systems (iterative methods), examples of Black-Scholes and Local Vol models, using European and Barrier options), connection to lattice (tree) methods (in terms of pricing formulas).

* **Other payoffs**: American options (free-boundary problem), Asian options in Black-Scholes model (dimension reduction via change of variables), forward starting options, change of numeraire.

* **Calibration to European options**: Local Vol models (inverse problems, example of LV Mixture models).

– **Monte-Carlo (Probabilistic) methods**.

* **Motivation via multifactor models**: curse of dimensionality, Stochastic volatility models (Heston (pricing European options via PDE), SABR (expansion for IV)), models for multiple underlying processes (spread options, basket options, structural models for default by Cox-Merton).

* **Foundations of MC**: LLN, CLT, rates of convergence (e.g. via variance estimate), examples (e.g. pricing options in BS).

* **Generation of random variables and vectors**: reduction to uniform distribution, Box-Muller method, pseudo-random numbers (low discrepancy numbers: Halton sequence), modeling correlation (copula, example of CDS and CDO, multivariate Gaussian via Cholecky decomposition).

* **Numerical solutions to SDEs**: exact simulation and standard Euler methods (definitions, approximation error, examples of GBM, CIR and OU processes), Milstein’s algorithm, multidimensional SDEs (Heston, structural CDO, basket options), hedging and estimating sensitivities via MC, approximation with a lattice (moment matching, example of BDT model).

* **Variance reduction**: antithetic variables (definition, applicability criterion, examples of pricing vanilla options in BS model and/or Asian options in Bachelier model), control variate (definition, example of basket options: geometric vs arithmetic average payoffs), importance sampling (definition, examples of pricing deep OTM options and VaR), other methods (moment matching, stratified sampling).

* **Dynamic programming via MC**: Longstaff-Schwartz.

* **Term structure models (market-based models for derivatives)**: short rate models (Vasicek, Hull-White extended Vasicek, calibration to yield curve), interest rate derivatives (caplets, caps, swaptions), modeling yield curve directly (HJM, BJM, Black’s model).

**Academic Integrity**: The LSA undergraduate academic community, like all communities, functions best when its members treat one another with honesty, fairness, respect, and trust. The College holds all members of its community to high standards of scholarship and integrity. To accomplish its mission of providing an optimal educational environment and developing leaders of
society, the College promotes the assumption of personal responsibility and integrity and prohibits all forms of academic dishonesty and misconduct. Academic dishonesty may be understood as any action or attempted action that may result in creating an unfair academic advantage for oneself or an unfair academic advantage or disadvantage for any other member or members of the academic community. Conduct, without regard to motive, that violates the academic integrity and ethical standards of the College community cannot be tolerated. The College seeks vigorously to achieve compliance with its community standards of academic integrity. Violations of the standards will not be tolerated and will result in serious consequences and disciplinary action.