Certificate in Quantitative Investment

http://cqi.sg/

August 2013

Contents

02 The Key Success Factors
03 The Demand of Good Financial Engineers
05 Our Uniqueness
05 ·····Mathematics
05 ·····Programming
06 ·····Quantitative Investment
07 Introduction to Quantitative Investment
09 Core Curriculum
09 ·····Syllabus
10 ·····Mathematics
10 ·····Statistics
11 ·····Stochastic Control
12 ·····Monte Carlo Methods & Optimization Methods
13 ·····Programming
13 ·····Professional Programming
14 ·····OOP and Design Patterns
15 ·····Algorithm Design & Analysis
16 ·····Portfolio & Risk Management
16 ·····Quantitative Equity Portfolio Management
17 ·····Introduction to Algorithmic Trading Strategies
19 ·····Risk Management
20 Easy-To-Understand Presentations
21 Testimonials
22 Life-Time Continuous Learning
23 Team

The Key Success Factors

- Mathematics is what translates a trading idea or intuition into well-defined meaningful symbols.
- Programming is what translates the mathematical symbols into lines of code for trading research and execution.
- Financial insights spark the imagination to create new innovative trading ideas.

The Demand of Good Financial Engineers

It is very difficult to find people who can produce effective code for mathematical models. In general, mathematicians do not write good code and computer programmers do not understand the (advanced) mathematics. Yet, computer language is the only language we use to turn an idea into a product that people can use. I foresee that the truly valuable talents are those who have creative ideas and are able to implement them. They are the most-sought after in many engineering disciplines, e.g., finance.

Speaking from personal experience, I have been working exclusively in the financial industry, Algorithmic Trading in particular. This field is where mathematics and computer science meet. As an algo-trader, my job is to develop mathematical models and write computer software for automatic execution. In addition, I lead a team of mathematicians to design trading models and a team of programmers to build these systems. From these years of job experience, I find it exceedingly difficult to hire the perfect candidates to work in this algorithmic trading industry.

I have hired some very good statisticians and mathematicians from the top schools. They produce good research and design very sophisticated mathematical models. Unfortunately, they are not able to program to the professional standard their models that are ready to be used by other people. Often the programmers need to translate their prototypes in Matlab / R into C++ / C# / Java. On the other hand, it is unrealistic to expect the programmers to develop these mathematical models simply because they do not have the training. In summary, my dilemma is: mathematicians cannot differentiate between inheritance and interface; programmers do not know about hidden Markov Chain.

It occurs very surprisingly to me that given the prevalence of computer programming in the industries, such as finance and bio-chemistry, universities have not been placing enough emphasize and proper training on their students. In my opinion, in a scientific corporation programming skill is as essential as speaking English. It is a way to communicate ideas (models) in a form that other people can actually use, i.e., a product. Our school curriculums have a number of drawbacks.

First, programming courses are not mandatory for many students, even science students. For instance, statistics students can graduate without ever taking a course in computer programming. They might very well be proficient in Matlab, R, and other specialized software. However, these are not real programming languages. These have little use in an industrial production environment. The students are still not trained in terms of object-oriented concepts, debugging skills, software engineering principles, team collaboration, quick adaptation of new tools and technologies.

Secondly, the professors, instructors, or lecturers teaching the programming courses usually have little industrial programming experience. Speaking from my personal experience, I thought I was a good programmer when I graduated with a PhD in computer science and after spending many years programming for my thesis and homework. It turned out that I was very naïve and ignorant. I was proved to know nothing about industrial programming on my first job. Looking back, the professors teaching programming in universities are probably in the stage where I once was. Most have never delivered a real product (not hands-on anyway).

Therefore, as we are now in the era of a technology driven world, the truly valuable talents are those who can have creative ideas and are able to implement them. I foresee that, gradually, the schools begin to recognize that having good programming skill is as essential as having good communication skill on the job. I am looking forward to changes in the academic curriculums. More emphases are placed on computer programming training across all majors. At the least, all engineering students must be proficient in one modern programming language. Equally important, these programming courses should be taught by experienced professionals rather than academic people who are trained to write journal papers.

In conclusion, we need a new course to teach numerical programming. This is a course to educate science students (not just computer science students) how to code mathematical models. That includes a modern programming language, software engineering methodology, debugging, algorithm design and analysis, effective implementation, and design pattern.

Haksun Li
CEO, Numerical Method Incorporation Limited

Our Uniqueness

Mathematics

We differentiate ourselves from the traditional master's degrees in financial mathematics or financial engineering. Firstly, these programs take too long a time (e.g., 3 months for a semester) to touch only the surface of the subjects. For instance, the standard topics are: options pricing, stochastic calculus and data analysis. However, you do not really need to do a degree program to learn them. Reading the right books is more efficient and effective: Financial Calculus: An Introduction to Derivative Pricing by Baxter and Rennie, Introduction To Stochastic Calculus With Applications by Klebaner, and Statistical Analysis of Financial Data in S-Plus by Carmona, etc. (They are not the standard textbooks used in universities.) The point is that the knowledge is easy and does not require a teacher. On the other hand, some topics are very useful in mathematical trading and yet very hard to self-learn them. If you would like to challenge yourself, try to study cointegration (the theory not just the R package "urca") by reading Likelihood-Based Inference in Cointegrated Vector Autoregressive Models, Johansen, or try to solve an optimal asset allocation problem with jumps by reading Applied Stochastic Control of Jump Diffusions, Oksendal and Sulem. Our courses are designed to make these more useful yet rather inaccessible mathematics concepts easy to understand and thus accessible to you when designing your trading strategy. More importantly, the focus of our mathematics courses is to teach mathematical thinking, namely how to translate a trading intuition into solid equations, rather than a set of formulas or mechanical computational rules.

Programming

Secondly, most graduates from these university programs cannot code professionally even though they may have gone through a year-long programming training. For example, if these students think that they can code in C++, think again after you read Scott Meyers' books. From our interviewing experience, most junior programmers have not read the three books, hence not being able to code. Learning a (natural) language is not about learning words and grammars. Similarly, learning a programming language (Java / C# / C++ / etc.) is not about learning the constructs and syntax. A professional programmer writes not only functional code that machines can read but elegant code that humans can read too.Writing elegant code is an art like painting or composing. The problem with bad spaghetti code is that there is no way to tell whether the code works or not maybe other than on a few toy examples. The consequence in trading could mean losses of millions of dollars. There are some basic skills to elegant coding like debugging, testing, software design, design patterns, algorithm design and analysis. All these essential programming knowledge, especially the most important programming skill, hitting F7 / F8 in NetBeans or F5 / F6 in Eclipse (debugging), is completely absent in school curriculums. The reason is simple: even computer science students are not trained to do professional programming; PhDs write for their research the code that no one reads or uses; professors write papers and ask students to code for them. Our courses are designed to teach professional programming from the basic to the advanced techniques. The focus is on writing code that is solidly objected-oriented, unified / consistent, and testable.

Quantitative Investment

Last but not least, the traditional university syllabuses focus on options pricing. Our personal opinion is that exotic derivative business is an evening industry since the housing bubble bursted in 2008. The money has shifted from exotic to flow or vanilla options. However, the flow business is a sales business. It does not take a lot of mathematics. How difficult is the Black-Scholes formula? OK. You may fit some volatility surface. But the money comes from customers willing to trade with you; again, it is a sales business. The desk probably makes more money by hiring a 22 year old cheerleader from USC rather than a 40 year old math PhD from MIT. The future of quantitative finance is uncertain. All banking professionals are searching for a new direction or the next gold mine. We believe, however, that there is always demand for wealth management as there are always wealthy people who are reluctant to put billions of dollars under their (big) mattess. Our courses do not teach any off-the-shelf profitable trading strategies (no one will) nor do we teach any get-rich-quick schemes (only scammers would). Our courses are designed to survey some of the sophisticated mathematical trading ideas from the academic world. From these published papers, we learn how to think mathematically, become equipped with the essential mathematics knowledge at fingertips to use and understand them, and get well versed in programming. In other words, our education objective is to train all-rounded would-be super-stars in the mathematical wealth management business.

Introduction to Quantitative Investment

This course introduces students to quantitative investment. A “quant” portfolio manager or a trader usually starts with an intuition or a vague trading idea. Using mathematics, s/he turns the intuition into a mathematical trading model for analysis, back testing and refinement. When the quantitative investment model proves to be likely profitable after passing rigorous statistical tests, the portfolio manager implements the model on a computer system for automatic execution. In short, quantitative investment is the process where ideas are turned into mathematical models and then coded into computer programs for systematic trading. It is a science where mathematics and computer science meet. In this course, students study investment strategies from the popular academic literature and learn the fundamental mathematics and IT aspects of this emerging field. After satisfactorily completing this course, the students will have an overview of the necessary quantitative, computing, and programming skills in quantitative investment.

Course Outline

There are a total of 8 lectures, each running for 3 hours.

Session	Topics
1	Technical analysis: linear trading rules
lab	Programming a hidden Markov chain, a trend following strategy
2	Construct baskets for trading
lab	Programming a cointegration model; basket creation; parameter sensitivity analysis
3	Optimal basket trading
lab	Programming a trading strategy; parameter calibration
4	Portfolio optimization & risk management
lab	Programming to optimize a portfolio

Recommended For

- Portfolio managers who wish to apply their mathematical and statistical strengths in the trading arena
- Algorithmic traders who seek a deeper appreciation of mathematics and programming
- Regulators, risk managers and auditors who need a good understanding of the nature of quantitative analysis
- Anyone who aspires to become a quantitative trader

Preferred Background

- Some experience in trading is preferred but not essential
- University level mathematics and statistics
- Programming experience

Trainer : Dr. Haksun Li

Dr. Haksun Li is the CEO of Numerical Method Inc., a quantitative trading research and analytic consulting company, which serves brokerage houses and funds all over the world, multinational corporations, very high net worth individuals and gambling groups. Prior to this, Dr. Li was a quantitative trader/quantitative analyst with multiple investment banks. He has worked in New York, London, Tokyo, Singapore and Hong Kong. Dr. Li has a B.S. and M.S. in Pure and Financial Mathematics from the University of Chicago, an M.S. and a Ph.D. in Computer Science & Engineering from the University of Michigan, Ann Arbor. Dr. Haksun Li is/was an adjunct professor with multiple universities. He taught at the National University of Singapore (Mathematics), Nanyang Technological University (Business School), Fudan University (Economics), as well as Hong Kong University of Science and Technology (Mathematics).

Language of Instruction

English

Schedule

September, 2013 - October, 2013

For Enquiries:

Please email : sales@numericalmethod.com

Venue

- Beijing
- China, PRC
- Online class available

Fees

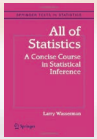
RMB 20,000.00 for classroom participation or
USD 3,200.00 for online participation.

Core Curriculum

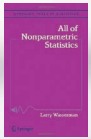
3 core modules: mathematics, programming, portfolio and risk management.
Each module consists of 3 courses.
Each course has 12 classes and runs for 3 months.
Each class consists of theoretical and when applicable practical sessions (hands-on experiments) and runs for 3 hours.

Syllabus

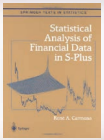
Mathematics	Programming	Portfolio & Risk Management
Statistics	Professional Programming	Quantitative Equity Portfolio Management
Stochastic Control	OOP and Design Patterns	Introduction to Algorithmic Trading Strategies
Monte Carlo Methods & Optimization Methods	Algorithm Design & Analysis	Risk Management



1

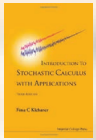


2



3

4



5

6

Statistics

Session	Topics
1	Basic statistics, distributions, hypothesis testing, maximum likelihood
2	Linear regression
3	Univariate time series analysis, white noise, AR, MA, ARMA, ARIMA, and GARCH
4	Kalman filter, dynamic system model
5	Multivariate time series and cointegration
6	Longitudinal data analysis
7	Measures (real analysis) and probability
8	Brownian motion calculus, Ito's lemma
9	Calculus for martingales and semi-martingales
10	Extreme value theory
11	Markov chain
12	Bootstrapping

Recommended Readings

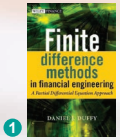
1. All of Statistics: A Concise Course in Statistical Inference, Larry Wasserman
2. All of Nonparametric Statistics, Larry Wasserman
3. Statistical Analysis of Financial Data in S-Plus, René Carmona
4. Likelihood-Based Inference in Cointegrated Vector Autoregressive Models, Søren Johansen
5. Introduction To Stochastic Calculus With Applications (3rd Edition), Fima C. Klebaner
6. Financial Calculus: An Introduction to Derivative Pricing, Martin Baxter, Andrew Rennie

Stochastic Control

Session	Topics
1	ODE
2	PDE
3	Dynamic programming
4	The Hamilton-Jacobi-Bellman equation
5	Problems with perfect and imperfect information
6	Discounted problems
7	Undiscounted problems
8	Stochastic jump and diffusion processes
9	Stochastic calculus for jump-diffusions
10	Stochastic dynamic programming
11	Computational stochastic control methods
12	Viscosity solutions of variational inequalities

Recommended Readings

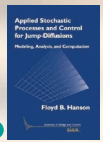
1. Finite Difference Methods in Financial Engineering: A Partial Differential Equation Approach, Daniel J. Duffy
2. Dynamic Programming & Optimal Control, Vol. I, Dimitri P. Bertsekas
3. Dynamic Programming and Optimal Control, Vol. II, 4th Edition: Approximate Dynamic Programming, Dimitri P. Bertsekas
4. Applied Stochastic Processes and Control for Jump-Diffusions: Modeling, Analysis, and Computation, Floyd B. Hanson
5. Applied Stochastic Control of Jump Diffusions, Bernt Øksendal, Agnès Sulem



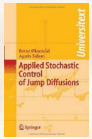
1

2

3



4

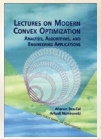


5

1

2

3



4

Monte Carlo Methods and Optimization Methods

Session	Topics
1	Generate random numbers
2	Generate random processes
3	Variance reduction techniques
4	Quasi-Monte Carlo
5	Discretization methods
6	Estimating sensitivities
7	Sequential importance sampling and resampling
8	Linear programming
9	Quadratic programming
10	Second-order conic programming
11	Mixed-Integer (non-)linear programming
12	(Stochastic) linear quadratic regulator problems

Recommended Readings

1. Monte Carlo Methods in Financial Engineering, Paul Glasserman
2. Practical Optimization: Algorithms and Engineering Applications, Andreas Antoniou, Wu-Sheng Lu
3. Convex Optimization – Boyd and Vandenberghe
4. Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications, Aharon Ben-Tal, Arkadi Nemirovski

Professional Programming

Session	Topics
1	"Hello World!", IDE and debugging
2	Language basics, classes and objects, interfaces and inheritance
3	Using libraries
4	Exceptions, I/O, concurrency
5	Writing testable code
6	Data structures: list, stack, queue, set and map
7	Sorting and searching
8	API, interface, class library design
9	Effective programming
10	Numerical programming
11	Web programming
12	Android programming

Recommended Readings

- The Java Tutorials
  http://docs.oracle.com/javase/tutorial/

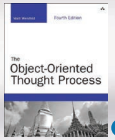
1. Data Structures and Algorithms Made Easy: Data Structure and Algorithmic Puzzles, Second Edition, Narasimha Karumanchi
2. Java Concurrency in Practice, Brian Goetz, Tim Peierls, Joshua Bloch, Joseph Bowbeer, David Holmes, Doug Lea
3. Effective Java (2nd Edition), Joshua Bloch
4. Effective Unit Testing: A guide for Java Developers, Lasse Koskela

1

2

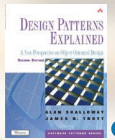
3

4



1

2



3

OOP and Design Patterns

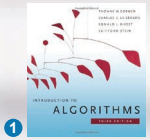
Session	Topics
1	Class design and objects
2	Interface, inheritance, composition
3	Introduction to design pattern
4	Creational patterns
5	Structural patterns
6	Behavioral patterns 1
7	Behavioral patterns 2
8	Introduction to trading systems
9	Trading strategy programming 1
10	Trading strategy programming 2
11	Backtesting 1
12	Backtesting 2

Recommended Readings

1. The Object-Oriented Thought Process (4th Edition), Matt Weisfeld
2. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides
3. Design Patterns Explained: A New Perspective on Object-Oriented Design (2nd Edition), Alan Shalloway, James R. Trott

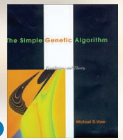
Algorithm Design and Analysis

Session	Topics
1	Introduction to algorithm analysis
2	Divide-and-conquer
3	Probabilistic analysis and randomized algorithms
4	Dynamic programming
5	Greedy algorithms
6	Amortized analysis
7	Trees
8	Graphs
9	Number-theoretic algorithms
10	Approximation algorithms
11	Genetic algorithm theory
12	P and NP



1

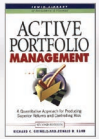
2



3

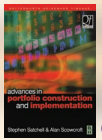
Recommended Readings

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein
2. Introduction of the Theory of Complexity, Daniel Plerre Bovet, Plerlulgi Crescenzl, D. Bovet
3. The Simple Genetic Algorithm: Foundations and Theory, Michael D. Vose



1

2



3

4

Quantitative Equity Portfolio Management

Session	Topics
1	Consensus expected, exceptional, residual returns and risk
2	Expected returns and valuation
3	Information and forecasting
4	Fundamental vs. economic factor models
5	Portfolio construction
6	Transaction cost, liquidity, neutrality, blacklist, market impact, tax and all that
7	Performance analysis
8	Portfolio optimization using SOCP and global optimization methods
9	Advanced portfolio optimization 1 (Bayesian approach, etc.)
10	Advanced portfolio optimization 2 (moment selection, etc.)
11	Universal portfolio
12	Estimation of covariance matrices

Recommended Readings

1. Active Portfolio Management: A Quantitative Approach for Producing Superior Returns and Controlling Risk, Richard Grinold, Ronald Kahn
2. Quantitative Equity Portfolio Management: An Active Approach to Portfolio Construction and Management, Ludwig B Chincarini, Daehwan Kim
3. Advances in Portfolio Construction and Implementation, Alan Scowcroft
4. Bayesian portfolio optimization with time-varying factor models, Feng Zhao

Introduction to Algorithmic Trading Strategies

Session	Topics
1	Overview of algorithmic trading
2	Hidden Markov trading model
3	Programming a hidden Markov chain
4	Pairs trading by cointegration
5	Programming a cointegration model; parameter sensitivity analysis
6	Optimal pairs trading by stochastic control
7	Pairs trading by stochastic spread methods
8	Programming a Kalman filter; parameter calibration
9	Technical analysis: linear trading rules
10	Portfolio optimization
11	Strategy & portfolio optimization
12	Risk management

Recommended Readings

1. Numerical Method's Collection of Quantitative/Algorithmic Trading Literature
   http://www.numericalmethod.com/trac/numericalmethod/wiki/Trading/Literature

Student Projects

2. The quantitative trading strategies group projects for FE8827, 2011, Nanyang Technological University.
   http://numericalmethod.com/papers/course1/ntu2011/index.html

3. The quantitative trading strategies group projects for QF5205, Fall 2011, National University of Singapore.
   http://numericalmethod.com/papers/course1/nus2011/index.html

Risk Management

Session	Topics
1	Risk overview: market, liquidity, credit, operational, and all that
2	Quantitative risk analysis
3	Value-at-Risk (VaR)
4	VaR for linear portfolios
5	VaR for non-linear portfolios
6	Forecasting volatilities and correlations
7	Backtesting VaR models
8	Building risk management systems
9	Basics of multivariate modeling, regression analysis, discriminant analysis
10	VaR using multivariate Normal distribution, non-Normal multivariate distributions, PCA
11	Extreme value theory (EVT)
12	Managing risk using EVT

Recommended Readings

1. Modelling Extremal Events: for Insurance and Finance, Paul Embrechts, Claudia Klüppelberg, Thomas Mikosch

Easy-To-Understand Presentations

The major problem with learning mathematics is that mathematical books or papers are written for the experts in the fields rather than laymen who want to learn the subjects. Moreover, the demand for rigorous proofs in mathematical writings often hides the intuitions behind the symbols. In contrast, our courses are designed to make difficult concepts easy to understand for students. We make advanced mathematics easy to grasp at the sacrifice of rigorousness.

Testimonials

Student Feedback

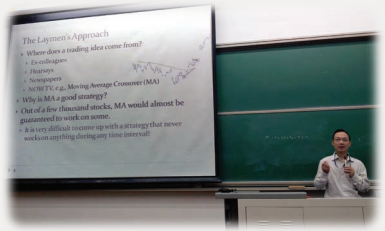
- The Hong Kong University of Science & Technology, 2013
- The National University of Singapore, 2012
- Nanyang Technological University, 2012
- The Hong Kong University of Science & Technology, 2012
- Nanyang Technological University, 2011
- 复旦大学 (Fudan University), 2011

"...Thank you for offering this course. This is the first time I see where mathematics, computer science and trading can be fit together that well."

Paul

"...This is one of the most useful courses I have taken in this semester. "

sofdjl



"...I am new to trading and hence this course was a good refresher on the techniques and methodologies used by algo traders. I learnt quite a lot from this course and I enjoyed the lectures as they were very interactive."

Alex

"...This is so far the most useful course that I have attended in first year of the programme, out of 11 courses."

Daniel

"...Doctor Haksun is pretty funny and (smart) obviously very good at what he does. I am glad we get to meet. Thanks for the wonderful class and spending the last month with me."

Simon

Life-Time Continuous Learning

- All our teaching videos are available online for students for real-time lectures or review.
- We invite experts from all over the world to give (sometimes free), online, optional, supplementary seminars to keep students updated of the current development in the financial industry.

Team

Dr. Ng Kah Hwa

Dr. Ng Kah Hwa is the Executive Director of Quant Management Consulting (Shanghai) Ltd and the Managing Partner of Financial Engineering and Risk Management Solutions LLP (FEnRM) in Singapore. He is also the Special Appointment Senior Research Fellow at the Research Center of Financial Engineering at Shanghai Jiao Tong University. Previously he was the Senior Vice President at DBS Bank in-charge of Financial Engineering Unit, Structuring Structured Products and Asset Securitization. He has also worked at JP Morgan in Investment Banking and OCBC in Treasury and Foreign Exchange Trading. Dr Ng has served as a Council Member of the Academic Advisory Council, of the Professional Risk Managers International Association (PRMIA) and was a member of the Fitch Academic Advisory Board.

In 1999-2008, Dr. Ng was at the National University of Singapore, as Associate Professor in Finance at the NUS Business School, the Director of NUS Centre for Financial Engineering and the Director of the MSc Financial Engineering Program. He was also a Director at the NUS Risk Management Institute. He has also served as the Program Director of the BSc Quantitative Finance program at the NUS Department of Mathematics. He has also worked at the Nanyang Technological University (NTU) and served as the Head of the Division of Banking and Finance and the Division of Actuarial Science and Insurance. He has taught undergraduate and graduate financial engineering, finance and banking courses at both NTU and NUS.

Dr. Ng has extensive experience in risk management, derivatives and structured products, and worked with numerous renowned financial institutions in treasury, investment banking and capital markets. He has provided training in risk management and structured products to financial institutions and central bank. He is also a consultant to numerous financial institutions and companies on financial risk management and quantitative investment management.

Dr. Ng has a Ph.D. in Finance at Columbia Business School in New York, Master in International Business at Sophia University, Tokyo and BSc Honours degree in Mathematics from the National University of Singapore.

Dr. Daehwan Kim

Dr. Daehwan Kim received PhD in Economics from Harvard University in 2000. Dr. Kim worked as Financial Economist for FOLIOfn, Inc., a Vienna, Virginia-based brokerage specialized in portfolio trading, and as Senior Portfolio Manager for First Private Investment Management, a quantitative asset management firm based in Frankfurt, Germany. Currently, Dr. Kim is a professor of economics at Konkuk University, Seoul, Korea. His current appointment also includes an Academic Adviser at First Private and a Director at Maunakai Capital Partners, a Hong Kong-based asset manager. Dr. Kim is the author of Quantitative Equity Portfolio Management (New York: McGraw Hill, 2006), which he co-authored with Professor Ludwig Chincarini at the University of San Francisco.

Javed Ashraf

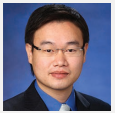
Javed Ashraf is the Managing Principal at Strode's Capital. Prior to that, he was the head quantitative researcher in credit relative value in a hedge fund. Javed was the head of US multi asset quantitative research team at Union Bank of Switzerland. He developed and ran algorithmic investments for equity derivatives. Javed has a M.Sc. in Financial Mathematics from the University of Chicago, an MBA from the City University, London UK, and a B.Eng. in Computer & Electronic Engineering from the University of Salford UK.

Dr. Ernie Chan

Dr. Ernie Chan is the Managing Member of QTS Capital Management, LLC., a commodity pool operator. Ernie has worked for various investment banks (Morgan Stanley, Credit Suisse, Maple) and hedge funds (Mapleridge, Millennium Partners, MANE) since 1997. He received his Ph.D. in physics from Cornell University and was a member of IBM's Human Language Technologies group before joining the financial industry. He was a co-founder and principal of EXP Capital Management, LLC., a Chicago-based investment firm. He is also the author of "Quantitative Trading: How to Build Your Own Algorithmic Trading Business" and "Algorithmic Trading: Winning Strategies and Their Rationale".

Dr. Haksun Li

Dr. Haksun Li is the CEO of Numerical Method Inc., an algorithmic trading research and analytic consulting company, which serves brokerage houses and funds all over the world, multinational corporations, very high net worth individuals and gambling groups. Prior to this, Dr. Li was a quantitative trader/quantitative analyst with multiple investment banks. He has worked in New York, London, Tokyo, Singapore and Hong Kong. Dr. Li has a B.S. and M.S. in Pure and Financial Mathematics from the University of Chicago, an M.S. and a Ph.D. in Computer Science & Engineering from the University of Michigan, Ann Arbor. Dr. Haksun Li is/was an adjunct professor with multiple universities. He has taught at the National University of Singapore (Mathematics), Nanyang Technological University (Business School), Fudan University (Economics), as well as Hong Kong University of Science and Technology (Mathematics).



Dr. Wei Zhen

Dr. Wei Zhen most recently held a senior Investment Strategist role in Bank of American Merrill Lynch (Hong Kong), where he advised world's largest asset managers on macro, fundamental and quantitative equity investment strategies. Prior to joining Merrill Lynch, Zhen was a Fixed Income Strategist at Nomura International (HK) and Lehman Brothers (Asia), covering cross-asset products and strategies for Asia Pacific clients. Zhen received his Ph.D. degree in Statistics and Masters of Science degree in Financial Mathematics from Stanford University, California. He graduated from Peking University, China, with a Bachelor's degree in Statistics and a minor in Computer Science.

Dr. Kevin Sun

Dr. Kevin Sun is a seasoned statistician who specializes in applying statistical methods to finance. Kevin was a quantitative analyst at an investment bank, where he created mathematical models for quantitative trading. Kevin has a B.S. and a M.S. in pure mathematics from the University of New South Wales, a M.S. in financial mathematics and a Ph.D. in statistics from Stanford University.

Ramesh Kadambi

Ramesh Kadambi is a Vice President at Sumitomo Capital Markets as a Senior Quant Developer. Ramesh has worked in the financial industry since 1994. He began his career at Cooper Neff Technologies as a Mathematical Programmer, and then moved on to BNP Paribas, NASDAQ, Union Bank of Switzerland, Pipeline Trading and Citadel Securities. Ramesh has worked in various technical areas of the financial industry. Along the way Ramesh has implemented systems ranging from back office to High Frequency/Low Latency front office trading. Ramesh interests include design patterns, High Frequency Trading (HFT), application of GPUs to HFT, quantitative libraries and back testing frameworks. Ramesh received Masters degrees in the fields of Financial Mathematics (University of Chicago), Computer Science and Mechanical Engineering (Villanova University).

Dr. Ken Yiu

Dr. Ken Yiu is an expert in software design and architecture. He is the chief architect of SuanShu and AlgoQuant. Ken worked at an investment bank, where he led a team to design and build an automatic trading system for high frequency trading. Ken has a B.Eng., and M.Phil., and a Ph.D. in Computer Science and Engineering from the Hong Kong University of Science and Technology.

Dr. Ding Hao

Dr. Ding Hao is a statistician whose interests lie in time series modeling and forecasting. He received his B.S. degree in Mathematics from TsingHua University, Beijing and his Ph.D degree in Financial Engineering from the Chinese University of Hong Kong.

by Numerical Method Incorporation Limited