Financial Analytics - Syllabus

Audience

This course is designed for graduate students interested in pursuing a career in financial services, with an emphasis on business analytics. It is offered as a selective in the MS in Business Analytics Program, and can also be taken by graduate students from other programs in order to complete their course requirements.

Description

An introduction to methods and tools useful in decision-making in the financial industry, including: macroeconomic event studies, analysis of term structures, Morningstar equity data, style analysis, credit card receivables, trading analytics, execution algorithms, etc.

Course credits

The successful completion of this course will earn the student 3 credit-hours.

Pre-requisites

While the catalog description and syllabus do not list pre-requisites, I strongly urge students to complete Data Analysis (MBC 638 or equivalent), Managerial Finance (MBC 633 or equivalent) and Business Analytics (SCM 651 or equivalent) before registering for Financial Analytics.

Additional Course Description

This course is designed to provide students with approaches to analyze various types of financial data sets, and to make meaningful decisions based on statistics obtained from the data. The course covers various areas in the financial industry, from analyzing personal data (credit card receivables) to studying global relations between macroeconomic events to managing risk and return in multi-asset portfolios. Students will be exposed to a wide range of techniques including non-linear estimation, portfolio analytics, risk measurement, extreme value analysis, forecasting and predictive techniques, and financial modeling. Students will be expected to complete several assignments projects in various areas, applying methods learned in the course to problems faced by decision makers. These projects will involve working in teams of two to complete each assignment. A final project will apply techniques to the construction of an online application.

Analytics

> *Science alone of all the subjects contains within itself the lesson of the danger of belief in the infallibility of the greatest teachers of the preceding generation.* - Richard Feynman

By its very nature the science of data analytics is disruptive. That means, among many other things, that much attention should be paid to the scale, range, invalid, at yet not understand, outlying, and emerging trends. This is as true within the finance domain of knowledge as any other.

Throughout the course, students will learn the core of ideas of programming software development to implement financial analyses (functions, objects, data structures, flow control, input and output, debugging,

logical design and abstraction) through writing code. Students will learn how to set up stochastic simulations, manage data analyses, employ numerical optimization algorithms, diagnose their limitations, and work with and filter large data sets. Since code is also an important form of communication among analysts, students will learn how to comment and organize code, as well as document work product.

This course will be taught in the `R` programming language. Asynchronous materials will be extended with guided hands-on live sessions labs. Assignments will be formally assigned on the day of the live session and due by close of business (6PM eastern time US) on the day before each next live session. This will allow the course learning community to review, revise, and resubmit work product on a timely basis as the course proceeds to build new and accumulate more skills and capabilities.

Resources

The main resources for the course are two books:

1. [Teetor] Paul Teetor. 2011. *R Cookbook*, O'Reilly.
2. [Foote] William G. Foote. 2017. *Financial Engineering Analytics: A Topical Manual Using R*, manuscript.

The live sessions will expand on key aspects of each chapter in Foote and prepare the student for the weekly assignment. R scripts, `R Markdown`own source files, and data sets accompany each week.

The content and `R` implementation in Foote derive from several sources. Among the recommended sources are:

1. [Ruppert and Matteson] David Ruppert and David S. Matteson. 2014. *Statistics and Data Analysis for Financial Engineering, Second Edition*, Springer.
2. [Brealey, et al.] Richard Brealey, Stewart Myers, and Franklyn Allen, 2015. *Principles of Corporate Finance*, various editions, McGraw-Hill.
3. [McNeil, et al.] Alexander McNeil, Rudiger Frey, and Paul Embrechts, 2015. *Quantitative Risk Management: concepts, Techniques, and Tools* Princeton, 2014.
4. [Hastie, et al.] James Hastie, Robert Tibshirani, and Jerome Friedman. 2009. *Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*, Springer Science & Business Media, 2009
5. Numerous articles on various financial model topics

Ruppert and Matteson is a comprehensive text on the use of statistical and operational research techniques in financial engineering. McNeil, et al. is a far more rigorously mathematical treatment of the modern theory and practice of market, credit, and operational risk management. Brealey, et al. is a standard reference and course text in finance that has numerous worked examples and insightful explanations. Hastie, et al. is among the standards in statistical learning and data science.

In addition to Teetor's book, here are many `R` books useful for managing implementation of models in this course. Three particularly useful `R` books include:

1. Phil Spector, *Data Manipulation with R*
2. Norman Matloff, *The Art of R Programming: A Tour of Statistical Software Design*
3. John Taveras, *R for Excel Users* at https://www.rforexcelusers.com/book/.

You will ultimately need all three (and whatever else you can get your hands on) in your professional work. John Taveras' book is an excellent bridge and compendium of Excel and R practices.

Several excellent online statistics resources exist. Among them is https://www.openintro.org/stat/textbook.php?stat_book=isrs developed at Johns Hopkins University.

R and RStudio

R is a free, open-source programming language for statistical computing. All of our work in this class can and will be done using R. You will need regular, reliable access to a computer running an up-to-date version of R. If this is a problem, let the professors know right away.

RStudio is a free, open-source R programming environment. It contains a built-in code editor, many features to make working with R easier, and works the same way across different operating systems. Use of RStudio is required for the course, and strongly recommended in general.

There are many online resources for learning about it and working with R and RStudio, in addition to the texts:

- The official introduction to R from the CRAN website, *An Introduction to R*, available at https://cran.r-project.org/doc/manuals/R-intro.pdf.
- John Verzani, *simpleR*, at https://cran.r-project.org/doc/contrib/Verzani-SimpleR.pdf,
- The *Google* R *Style Guide*, at https://google.github.io/styleguide/Rguide.xml, offers rules for naming, spacing, etc., which are generally good ideas for reproducibility and dissemination of analytical products.
- *Quick-R* at http://www.statmethods.net/. This site is primarily aimed at those who already know a commercial statistics package like SAS, SPSS or Stata, but it's very clear and well-organized, and others may find it useful as well.
- Patrick Burns, *The R Inferno* at http://www.burns-stat.com/pages/Tutor/R_inferno.pdf. "If you are using R and you think you're in hell, this is a map for you." - *Patrick Burns* at http://www.burns-stat.com/documents/books/the-r-inferno/
- Thomas Lumley, *R Fundamentals and Programming Techniques* (large PDF file at http://faculty.washington.edu/tlumley/Rcourse/R-fundamentals.pdf)
- Rstudio, *Building Shiny Applications* at http://shiny.rstudio.com/, with tutorials, galleries, example code, and articles. We will be building basic applications with Shiny, a package that will embed your analytical implementations, allow for user interaction and graphical display.

Assignment Formatting

All assignments must be turned in electronically, through the learning management system, by each student. All assignments will involve writing a combination of code and actual prose. You must submit your assignment in a format which allows for the combination of the two, and the automatic execution of all your code. The easiest way to do this is to use R Markdown. R Markdown also allows the use of interactive modeling through Shiny applications.

Work submitted as Word files, unformatted plain text, etc., are not acceptable at any time during the course. Each assignment will require the submission of at least one R Markdownown script and the pdf or html file that the R Markdown script generates. When using data sets, this course will only use csv (comma separated variable files generated by Excel or in text files. If the submission uses a csv file, that file must also be submitted with the R Markdown script and generated pdf or html output files. The student may also submit an R script file, suitably commented, that represents the R code chunks in the R Markdown script.

Managing the data base of submitted assignments throughout the course will be aided by standards including file name construction for assignment submission. To this end, every file submitted must have a file name which includes the student's name, course identifier, and clearly indicates the type of assignment (project) and its number (week). Here is the format we will use: yourName_courseidentifier_Assignment#.ext, where # is the week number and ext is the file name extension. For example: wgfoote_FIN654_Assignment1.Rmd, where the file extension Rmd is the extension that RStudio uses for R Markdownown documents. File extensions R, PDF, and CSV are the other three admissible file types.

Other Matters

Grading

Equal weight will be given to each of the projects undertaken. Students, in teams of two, will both be given the same score for a completed project.

Grades will be assigned based on an A-F scale.

Grades for assignments will follow this general rubric:

- **Words:** The text is laid out cleanly, with clear divisions and transitions between sections and subsections. The writing itself is well-organized, free of grammatical and other mechanical errors, divided into complete sentences, logically grouped into paragraphs and sections, and easy to follow from the presumed level of knowledge.
- **Numbers:** All numerical results or summaries are reported to suitable precision, and with appropriate measures of uncertainty attached when applicable.
- **Pictures:** All figures and tables shown are relevant to the argument for ultimate conclusions. Figures and tables are easy to read, with informative captions, titles, axis labels and legends, and are placed near the relevant pieces of text.
- **Code:** The code is formatted and organized so that it is easy for others to read and understand. It is indented, commented, and uses meaningful names. It only includes computations which are actually needed to answer the analytical questions, and avoids redundancy. Code borrowed from the notes, from books, or from resources found online is explicitly acknowledged and sourced in the comments. Functions or procedures not directly taken from the notes have accompanying tests which check whether the code does what it is supposed to. All code runs, and the `R Markdown` file `knits` to `pdf_document` output, or other output agreed with the instructor.
- **Modeling:** Model specifications are described clearly and in appropriate detail. There are clear explanations of how estimating the model helps to answer the analytical questions, and rationales for all modeling choices. If multiple models are compared, they are all clearly described, along with the rationale for considering multiple models, and the reasons for selecting one model over another, or for using multiple models simultaneously.
- **Inference:** The actual estimation and simulation of model parameters or estimated functions is technically correct. All calculations based on estimates are clearly explained, and also technically correct. All estimates or derived quantities are accompanied with appropriate measures of uncertainty.
- **Conclusions:** The substantive, analytical questions are all answered as precisely as the data and the model allow. The chain of reasoning from estimation results about the model, or derived quantities, to substantive conclusions is both clear and convincing. Contingent answers (for example, ???if X, then Y , but if A, then B, else C???) are likewise described as warranted by the model and data. If uncertainties in the data and model mean the answers to some questions must be imprecise, this too is reflected in the conclusions.
- **Sources:** All sources used, whether in conversation, print, online, or otherwise are listed and acknowledged where they used in code, words, pictures, and any other components of the analysis.

Course Specific Policies

Students are expected to behave in a professional and courteous manner at all times when interacting with all members of the course learning community. Respect for others is demonstrated through attendance,

meaningful participation, and punctuality. Every effort should be made to be present for the entirety of each session especially since weekly assignments will be made conditional on content in live sessions.

All projects must be completed and submitted by the due dates and times set out. This will allow the entire class to review and revise submissions in a timely fashion. Submissions to the 2SU learning management system are based on eastern time. Late submissions will result in student inability to accumulate the knowledge needed to advance to the next week's coverage of course topics. Late submission will also delay necessary instructor feedback to the student in a timely fashion. As the course continues to layer on more skills and capabilities, a late submission with inaccurate or incorrect implementations of financial applications will only deprecate the student's ability to successfully complete future assignments.

Academic Integrity

Syracuse University's Academic Integrity Policy holds students accountable for the integrity of the work they submit. Students should be familiar with the policy and know that it is their responsibility to learn about course-specific expectations, as well as about university policy. The university policy governs appropriate citation and use of sources, the integrity of work submitted in exams and assignments, and the veracity of signatures on attendance sheets and other verification of participation in class activities. The policy also prohibits students from submitting the same written work in more than one class without receiving written authorization in advance from both instructors. The standard sanction for a first offense by a graduate student is suspension or expulsion. For more information and the complete policy, see http://academicintegrity.syr.edu/academic-integrity-policy/.

In this course, all sources, whether verbal, online, in print, or other, must be cited following prevailing business and academic requirements and practice.

Disability-Related Accommodations

If you believe that you need accommodations for a disability, please contact the Office of Disability Services (ODS), http://disabilityservices.syr.edu, located in Room 309 of 804 University Avenue, or call (315) 443-4498, TDD: (315) 443-1371 for an appointment to discuss your needs and the process for requesting accommodations. ODS is responsible for coordinating disability-related accommodations and will issue students with documented Disabilities Accommodation Authorization Letters, as appropriate. Since accommodation may require early planning and generally are not provided retroactively, please contact ODS as soon as possible.

Religious Observances Policy

SU religious observances policy, found at http://supolicies.syr.edu/emp_ben/religious_observance.htm, recognizes the diversity of faiths represented among the campus community and protects the rights of students, faculty, and staff to observe religious holidays according to their tradition. Under the policy, students are provided an opportunity to make up any examination, study, or work requirements that may be missed due to a religious observance provided they notify their instructors before the end of the second week of classes for regular session classes and by the submission deadline for flexibly formatted classes. For fall and spring semesters, an online notification process is available through `MySlice > Student Services > Enrollment > My Religious Observances`.

Tentative Course Outline

Course outline and contents are subject to change by the instructor. Topics may have coverage in asynchronous and synchronous materials, as readings, or in assignments.

1. R Warm-Ups for Finance. `R` computations, data structures, financial, probability, and statistics calculations, visualization. Documentation with `R Markdown`.

- **Lab 1:** Basic `R` practices and practice
- **Project 1:** Commercial loan analysis

2. More R Warm-Ups. Functions, loops, control bootstrapping, simulation, and more visualization.

- **Lab 2:** Intermediate `R` practice (especially functions)
- **Project 2:** Financial planning simulation

3. Stylized Facts of Financial Markets. Data from FRED, Yahoo, and other sources. Empirical characteristics of economic and financial time series. Boostrapping confidence intervals.

- **Lab 3:** Dependence and Correlation
- **Project 3:** Systemic failure (and success) for economic scenarios

4. Term Structure of Interest Rates. Bond pricing, forward and yield curves. Estimating Non-linear regression splines. Applications.

- **Lab 4:** Term structure twists and turns
- **Project 4:** Predicting bond prices for collateral management

5. Market Risk. Quantile (i.e., Value at Risk) and coherent (i.e., Expected Shortfall) risk measures.

- **Lab 5:** Measuring equity price risk
- **Project 5:** Price risk across asset classes.

6. Credit Risk. Hazard rate models, Markov transition probabilities Risk measures, Laplace simulation with FFT.

- **Lab 6:** Simulating credit markov chains
- **Project 6:** Rating receivables

7. Operational Risk and Extreme Finance. Generate frequency and severity of operational loss distributions. Estimating operational risk distribution parameters. Simulating loss distributions.

- **Lab 7:** Simulating operational loss distributions
- **Project 7:** Managing emergent risks

8. Measuring Volatility. Measuring volatility. GARCH estimation. GARCH simulation. Measuring Value at Risk (VaR) and Expected Shortfall (ES).

- **Lab 8:** Estimating and simulating volatility
- **Project 8:** Simulating cross-asset correlation volatility

9. Portfolio Optimization. Combining risk management with portfolio allocations. Optimizing allocations. Simulating the efficient frontier.

- **Lab 9:** The new portfolio frontier
- **Project 9:** Allocations across risk budgets

10. Aggregating Enterprise Risks. Enterprise risk management analytics and application. Workflow to build an online application. Introduction to `Shiny` and `ShinyDashboard`. Building a simple app. Using `R Markdown` and `Shiny`.

- **Lab 10:** Financial apps examples
- **Project 10:** Pick a project and build an `app`

