Asset Pricing I for Masters students at Uni Basel

Yvan Lengwiler

Fall 2024

A cautionary note: The content of this syllabus is currently still provisional and it may change in the course of the semester.

Content

This class is for masters students who are interested in Finance. The aim of this course is to empower you to "do finance" in a hands-on and relevant way. So you will get your hands dirty. In my mind, this is a good thing. It is a marketable skill to be able to develop useful code in finance. That should help you get productive quicker should you decide to enter the financial industry.

Before we do that, some IT business must be resolved. Please make sure that the `R` language is installed on your laptop. You probably also want to install `Rstudio` (a specialized development environment for the R language). Please bring your laptop to the first class and make sure that you have wireless internet connectivity.

We will start by exploring **statistical characteristics of equity prices**. We will explore the distribution of returns at high frequency (daily observation) and at lower (monthly) frequency and ponder about the reasons for the qualitative differences that we can observe.

We will also study the phenomenon that equities sometimes experience volatility spikes that last for some time. This is often modelled as a "generalized auto-regressive conditionally heteroscadastic process," or GARCH.

We will learn how to generate **Monte-Carlo simulations**. This is a method that is akin to a "Swiss Army Knife" of quantitative finance. It is almost never the best tool, but it is very widely applicable and often good enough. It pays to be proficient with this method.

After that, we study the mechanics of diversification. This is a fundamental staple of investment management. We will explore how this idea is used in practive and you will be asked to develop a computer program that can compute an **efficient frontier** with different constraints.

Finally, we will learn about the **term structure of interest rates**. It is difficult to overestimate the importance of interest rates. They affect all aspects of the economy. Understanding them is a key ingredient of any investment or risk management task.

There is a need to describe the relationship between interest rates of different maturities

with just a few parameters. One method that is commonly used — by central banks or institutional investors alike — is the Nelson-Siegel model, see e.g. Federal Reserve (link), ECB (link), SNB (link).

Now, a big ***CAVEAT*** is in order here: This is the first time I will give this course. It is possible that this list of topics is too much. If that is the case, I will cut some material. It is more important that you get a few methods and insight correctly than throwing a bunch of stuff at you that you cannot digest. So do not worry if you feel that this appears to be more material than you can chew.

Relationship between *Asset Pricing I* and *Asset Pricing II*

I know that I and II are consecutive numbers, so one might believe that you need to do I before doing II. But this is not so. Instead, these course both tackle asset pricing but with rather different aims.

Asset Pricing I aims to provide a hands-on approach to learning asset pricing. The idea is to empower you to do practical things in this domain.

Asset Pricing II aims to prepare you for academic research in asset pricing theory. It is much more geared towards students pursueing a PhD.

Requirements

Required: *Math and Statistics*

It is assumed that you know statistics and econometrics at the level of a Bachelor of economics. The minimum skills required are

- 23346 “Statistics”
- 10135 “Mathematics I”
- 10172 “Introduction to Econometrics”

More is better.

Recommended: *Finance*

Your previous knowledge of Finance is maybe of lesser importance, although you should be comfortable with discounting or computing the present value of a cash flow stream. If you have passed

- 10146 “Finance Theory I”

you are more than qualified.

Required: *Computer Science*

You should have had at least some exposure to programming, preferrably in R. If you have some gaps in computer science topics, you should participate in the introductory course on programming that the group of Dietmar Maringer offers at the beginning of the semester.

- Crash course: Working with scientific software (link)

By the way, I do not mind at all if you use ChatGPT or Copilot etc. to help you code. That is true for the work in the semester and for the exam. You can use all the help you can get from the internet, *but you may not accept help from a human being* in the exam! Moreover, you have to be able to fully explain what your program does.

Organization

This lecture is organized a little bit differently than other courses. Before each lecture, you will have to prepare. To do this, we will provide handouts that you should work through. These handouts will contain the leading questions that should help digest the papers. These questions are not to be answered simply with yes or no, but are meant to encourage you to reflect. Moreover, the handouts also explain the programming problems that you should master.

During contact time, we will go through the handouts together. You will be able to show your solutions and share your thoughts with the class. This is actually easier to achieve via Zoom because we can share the screen. For this reason, the lectures will be mostly via Zoom. An exception will be the very first lecture on Sep 19. This is an opportunity to meet your fellow colleagues in person and form potential study groups.

We encourage you to collaborate with your colleagues for exercises and weekly tasks. A forum will be provided on ADAM for you to exchange contact information, look for collaboration partners, and ask for help from fellow students. You are welcome to make an apportment as well for face-to-face time with me.

Important: All materials will be made available on ADAM or on Github. The Zoom meeting is for sharing your insights or points that remained unclear. Each week, individual students will be asked to comment briefly on specific "guiding questions" to get the discussion going, so prepare carefully. Also, note that Zoom sessions will be in English, unless all participants are German-speaking.

Groups: Because an interactive Zoom session does not work when the group is too large, the class will be split into groups, depending on how many attend. You will be assigned randomly to one of the groups. If you have a preference, let us know as soon as possible. Zoom dial-in information will also be provided on ADAM.

Zoom Settings: Turn on your video camera throughout the meeting. It is very difficult to have a discussion with a symbol image. Turn off your microphone to mute to avoid noise. Turn on your microphone only when you want to speak.

Important preparation for the first lecture!

The first lecture will be *sur place* in Basel. This might be slightly inconvenient for the students that join from Bern, but it will provide an opportunity for you to get to know your colleages and potentially form working groups. It is generally advisable to work in groups (not at the exam, of course ;-)

For the first lecture, bring your laptops with you and make sure that *eduroam* wifi

works for you so that you can be online during class.[1] You will need it.

Install `R` and possibly `Rstudio` on your laptop before the first lecture.[2] You have to have root access (an admin account) to your laptop because we will install more software (namely `git`) in class.

Teaching staff

The lecturer for the course is Yvan Lengwiler (yvan.lengwiler@unibas.ch). There will also be a teaching assistant, but this person is not nominated yet.

If you have any questions or comments, please contact us. If you would like a more in-depth discussion, please make an appointment.

Administrative issues

The course will take place in the first half of the semester, on Thursdays from Sep 19 to Oct 31, in the time span of 2:15 to 6:00 pm. The first session on Sep 19 will be at the University of Basel. The final exam on Oct 24 or Oct 31 (depending on our progress) will be either in the form of a take-home exam or will take place at the University of Basel (I have not decided that yet). The other meetings will be through Zoom.

Grading and credits

The course provides 3 credits upon successful completion.

1/3 of the final grade will be determined by your participation during the lectures and the work you hand in. Please note: If you are unsuccessful in solving a problem, that is totally ok. What we require is only that you seriously try. You have to be able to show that you tried and explain where and why you failed, or how you got stuck. Failure to achieve a solution is completely acceptable. It happens to all of us.

2/3 of the final grade will be determined by a written final exam. The tasks of the final exam will be formulated in English, but can be answered in English or German. The task will likely contain some programming work.

[1] Alternatively, you could tether through your smartphone if you have a good mobile data plan.

[2] You can find the `R` language at `https://cran.r-project.org/`. `Rstudio` is a common integrated development environment (IDE) for `R` and is available at https://posit.co/products/open-source/rstudio/, the download link should be here https://posit.co/download/rstudio-desktop/. Be careful: there exists a free as well as a few non-free versions.