Financial Econometrics II

Homework #1 Part I: MATLAB Implementation of MLE

Due in class on January 22

NOTE: *The homework needs to be typped and only hard copies are accepted. Please keep a copy for yourself since HW won't be returned.*

This homework aims to show you the necessary steps involved in writing high quality programs such as those in the MATLAB. We focus on maximum likelihood estimation in the i.i.d. setup as an example, but the implementation can be easily generalized to ARCH/GARCH models.

Statement of the Problem: Suppose that we have i.i.d. observations (data) $X_1, X_2, ..., X_n$ from a distribution with density $f(x, \theta)$, where θ is the unknown parameter. For simplicity, here we assume both X_i and θ are scalars. Write a MATLAB program that implements the MLE estimation $\hat{\theta}$ defined by

$$\hat{\theta} = \arg\max_{\theta} L(\theta). \tag{1}$$

Here the likelihood function $L(\theta)$ is defined by

$$L(\theta) = \sum_{i=1}^{n} \log f(X_i, \theta).$$

Your program should be able able to out put the MLE $\hat{\theta}$, $L(\hat{\theta})$, the condifence interval, and the p-value for testing $H_0 : \theta = \theta_0$.

This is a general problem. If you are not very sophisticated in program, pay attention to the following issue when you implement:

- How your program takes inputs data $X_1, X_2, ..., X_n$?
- Do you code the density $f(x, \theta)$ inside your program, or pass it in as a parameter?
- Which MATLAB optimization function (check MATLAB Optimization Toolbox) should you use to do the maximization work?
- How to output the results?
- Can you modify your program to work for higher dimensional data?

It's better to pass it as a parameter; so that the customer can change the function

To help you develop programming skills, I have divided the above problem into smaller problems that are easier in the follows. This is what experienced programmer do when they attack a new problem. If you are experienced in programming, you can go directly to the above problem.

Step 1. Build a generic maximization program that can maximize any given function $g(\theta)$ over θ. Since MATLAB has several built in optimization routines in its Optimization Toolbox, you only need to learn how to make a function call.

- Pay attention to the input and output of the MATLAB optimization routine you want to use. Read the MATLAB Optimization Toolbox –available from Blackboard or MATLAB website.
- Write a main program (driver) to call the MATLAB optimization routine and test it with function that you know the answer. E.g.,
  - Maximizing univariate function $g(\theta) = -3\theta^2 + \theta - 6$ over θ.
  - Create another example of yours to test it.

Step 2. Write a function that can computes the likelihood function given the data $X_1, ..., X_n$

$$L(\theta) = \sum_{i=1}^{n} \log f(X_i, \theta).$$

Here your input for this function should be the data $X_1, X_2, ..., X_n$, and the density function $f(x, \theta)$ (if you want to pass the density $f(x, \theta)$ in as a parameter). This step is simple enough so you may not need to test it.

Step 3. Now combine the above two steps and write a program that

- Inputs data $X_1, X_2, ..., X_n$, the density $f(x, \theta)$;
- Outputs the MLE $\hat{\theta}$ and $L\left(\hat{\theta}\right)$
- Test your implementation using examples that you know the answer:
  - Generate 100 i.i.d. observations from normal distribution

    $$f(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$ (assume $\sigma = 0.1$ is known)

    Pass the data and the density (the only unknown parameter is μ) to your program to see if gives you the correct answer: $\hat{\mu} = n^{-1} \sum_{i=1}^{n} X_i$.

Step 4. Now add all the bells and whistles such as outputting p-value, confidence interval.