Problem Set 2

Quantitative Economics, Fall 2024

September 9, 2025

This problem set consists of four problems. You have two weeks to solve them and submit your solutions (Dec. 5th 11:59 PM). You can work in teams of up to three students.

Problem 1: Iterative solver for nonlinear equations

In this problem you will have to code up a simple function that we can use to solve nonlinear equations in one unknown by using an iterative method. Consider the following problem:

Let $f : R \to R$. We want to find x such that

$$0 = f(x).$$

Our task is thus to find a root (because there are possibly many) of $f(x)$. Notice that it is the same as

$$x = f(x) + x$$

or, if we define $g(x) := f(x) + x$

$$x = g(x).$$

We look for a fixed point of $g(x)$: a value of x such that $g(x)$ is equal to x. It may or may not exist. Sometimes we can prove a fixed point exists.[1] Sometimes we can even show it is unique and we will converge to it from any guess.[2] Suppose that we start with some $x = x_0$ and we calculate $x_1 = g(x_0)$. We can then check if x_1 is close to x_0. If it is, i.e. $\frac{|x_1 - x_0|}{1+|x_0|} < \epsilon$ for some small $\epsilon > 0$ we (approximately) found a fixed point. If not, then we calculate $x_2 = g(x_1)$ and check again. We proceed in this fashion using the formula $x_{n+1} = g(x_n)$.

You task is to write a *function* that takes five arguments (three mandatory and two optional):

1. a one-dimensional *function* f,
2. a starting guess x_0,
3. an extra parameter α to be explained shortly below,
4. a tolerance parameter ϵ (optional),
5. a maximum number of iterations *maxiter* (optional).

[1] Brouwer's, Kakutani's and Tarski's fixed point theorems are are often used in economics.

[2] See Contraction Mapping Theorem by Stefan Banach. We will talk about it in detail.

This function is supposed to return:

1. an integer $flag$ equal to 0 if the solution has been found before the maximum number of iterations has been reached
2. the point that is the solution (or NaN if it has not been found),
3. the value of that solution (or NaN if it has not been found),
4. an absolute value of a difference between the solution x and $g(x)$,
5. a vector of all $\{x_0.x_1, x_2, \cdots\}$
6. a vector of all residuals $\{|x_1 - x_0|, |x_2 - x_1|, \ldots\}$.

We will actually work with a modified version of the algorithm:

$$x_{n+1} = (1 - \alpha)\, g(x_n) + \alpha x_n.$$

The parameter α is responsible for dampening: if it's close to 1, then we will update our guesses only slowly.

The optional parameters should be equal to $\epsilon = 10^{-6}$ and $maxiter = 1000$ be default (i.e. unless specified otherwise).

Once you write the function, you will test how it works with

$$f(x) = x^4 - e^x + 1, \quad h(x) = x - (e^x - 1)^{1/4}.$$

Region of convergence...

First, pl.

Graphically verify that these three functions have exactly the same three roots. For each of these three functions, rewrite the problem of finding the root as a fixed point problem. For each of these three functions, create a plot showing $g(x)$ and x on the same graph. Start from $x_0 = 0$ and $\alpha = 0$. Does it work for different values of α and/or different starting points? How many roots does $f(x)$ have? Can you find them all using your function? Once you write the function, test it by find the root of $f(x) = (x+1)^{\frac{1}{3}} - x$ starting from $x_0 = 1$ and using $\alpha = 0$. Does it work for different values of α and/or different starting points? Convince yourself that finding the root of $f(x)$ is the same as finding the root of $h(x) = x^3 - x - 1$. Can you find α such that the algorithm converges to the root starting from $x_0 = 1$?[3]

[3] What if you start from a point slightly different from the root of $f(x)$?

Problem 2: Some linear algebra

Let

$$\mathbf{A} = \begin{bmatrix} 1 & -1 & 0 & \alpha-\beta & \beta \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \alpha \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

The system $\mathbf{Ax} = \mathbf{b}$ is relatively easy to solve by hand.

Steps to follow:

1. Your first task is to write a function that takes α and β as arguments and returns the exact solution. You will have to do algebra by hand to find it – start from the last equation.
2. Next, write a function that takes α, β as arguments and return the exact solution, the solution obtained by using the backslash operator, the relative residual, and the condition number.
3. Use your functions to create the table that shows x_1 (both exact and obtained by using the backslash operator), the condition number, and the relative residual for $\alpha = 0.1$ and $\beta = 1, 10, 100, \ldots, 10^{12}$.

Problem 3: Internal rate of return

Net present value of an investment is calculated as

$$NPV = \sum_{t=0}^{T} \left(\frac{1}{1+r} \right)^{t} C_t.$$

In the notation above, $\{C_t\}_{t=0}^{\infty}$ is a sequence of cash flows. Negative values of C_t denote costs associated an the investment (outflows); positive values of C_t are gains (inflows). r denotes the rate of return used to discount these cash flows. We call $r > -1$[4] that satisfies

$$NPV(r) = 0$$

the *internal rate of return* (IRR). Such a value of r makes the discounted present value of costs incurred equal the discounted present value of gains related to an investment. The IRR is an indicator of the profitability of an investment. Write a function `internal_rate(C)` that takes a vector of numbers C and calculates the internal rate of return.

Steps to follow:

1. Create a function `NPV(r,C)`. This function should return the net present value as a function of the sequence of cash flows $\{C_t\}_{t=0}^{\infty}$ and some rate of return r.
2. Use a nonlinear equation solver to find the root of `NPV_rate(r,C)` for a given value of C. You can either write a wrapper function `wrapped_NPV(r)` = `NPV(r,C)` or write `r` → `NPV(r,C)` when calling the solver.
3. Put everything inside a function `internal_rate(C)`.

[4] $r = -1$ means that the gross rate of return, $1 + r$, is 0: for one unit invested today we get zero units tomorrow - we lose everything.

There are some things to consider. You are free to use any numerical solver. It might be the case that the root does not exist. It can happen, for example, when C_t has the same sign for all t. In this case your code should return a warning message instead of a number. Similarly, if your solver did not manage to find any root, your code should display a warning. Use information from the solver output. It is also possible that there are multiple roots. For now, you can ignore this issue.

Keep in mind that experimenting with arbitrary numbers in $\{C_t\}_{t=0}^{T}$ usually does not make too much sense. A first step is to test whether your code returns a correct answer for $C_0 < 0$ and $C_1 = C_2 = ... = C_T > 0$. Given these conditions, there is always a single unique solution for IRR.

Example

$$\texttt{internal\_rate}([-5, 0, 0, 2.5, 5]) = 0.11735$$

Problem 4: Production

A firm uses two inputs x_1 and x_2 to produce a single output y. The production function is given by

$$f(x_1, x_2) = \left(\alpha x_1^{\frac{\sigma-1}{\sigma}} + (1-\alpha)\, x_2^{\frac{\sigma-1}{\sigma}}\right)^{\frac{\sigma}{\sigma-1}},$$

where $\alpha \in (0,1)$ and $\sigma \geq 0$. This function is know as CES (constant elasticity of substitution) production function, and the parameter σ represents the elasticity of substitution. Notice $\sigma = 1$ corresponds to the Cobb-Douglas production function (i.e. $f(x_1, x_2) = x_1^{\alpha} x_2^{1-\alpha}$).[5]

The firm minimizes its expenditure on inputs x_1 and x_2 subject to the production target y. The problem can be written as

$$\min_{x_1, x_2} \quad w_1 x_1 + w_2 x_2$$
$$\text{s.t.} f(x_1, x_2) = y,$$

where w_1 and w_2 are the prices of inputs x_1 and x_2, respectively. We also require that $x_1, x_2 \geq 0$ - these are nonnegativity constraints on inputs.

The value of $w_1 x_1 + w_2 x_2$ at the optimum is called the cost function, $C(w_1, w_2, y)$.

Steps to follow:

1. Write a function that takes α, σ and x_1 and x_2 as arguments and creates a contour plot of the production function $f(x_1, x_2)$ for $x_1 \geq 0, x_2 \geq 0$.

[5] You can check it by taking the limit of the CES production function as $\sigma \to 1$.

2. Create the above plot for $\alpha = 0.5$ and $\sigma = 0.25, 1, 4$. Your answer should be a single plot with three panels (each one for a different value of σ). You do not need to write a function for this part, it is enough to write a script that calls the function you wrote in the first part.

3. Write a function that takes α, σ, w_1, w_2, and y as arguments and returns the cost function and x_1 and x_2. Inside this function, you will have to solve a constrained minimization problem (because you have nonnegativity constraints). You can use whatever package you want. Make sure that you test that optimization works correctly and finds the correct solution. Note that the $\sigma = 1$ case might need a special treatment.

4. Plot the cost function and the input demand functions (x_1 and x_2) for three different values of σ: $\sigma = 0.25, 1, 4$ as a function of w_1 for $w_2 = 1$ and $y = 1$. Set α to 0.5. Your answer should be a single plot with three panels (cost, x_1, and x_2) and three lines in each panel (each one for a different value of σ). You do not need to write a function for this part, it is enough to write a script that first calls the functions you wrote in the first part and then plots the results.