ECON6080/8080 Assignment 2

Total marks: 10 marks. Marks will depend on accuracy and clarity. Speed is not an issue.

Use Python to answer all questions.

Q1 [2 marks]

The Newton-Raphson method is easily generalized to the multi-dimensional case. For a C^1 function $f : \mathbb{R}^N \to \mathbb{R}^N$, a sequence of candidate solutions are calculated starting from $x_0 \in \mathbb{R}^N$ and, for each n, the new iterate is computed as

$$x_{n+1} = x_n - [J(x_n)]^{-1} f(x_n),$$

where $J(x_n)$ is Jacobian of f evaluated at x_n. I.e. we have $\mathbb{R}^1$-valued functions $f^1, f^2, ..., f^N$ such that

$$f(x) = \begin{bmatrix} f^1(x) \\ f^2(x) \\ \vdots \\ f^N(x) \end{bmatrix},$$

and the (i, j)-element of Jacobian matrix at x is given by

$$\frac{\partial f^i(x)}{\partial x_j}.$$

Write a user-defined function that implements a multi-dimensional Newton-Raphson method. You can assume that the Jacobian function is supplied by a user. Choose a couple of test functions and test your code using them.

Note that it is well known that computing the inverse matrix is more expensive than solving a system of linear equations. Hence, it is recommended that you solve

$$J(x_n) \times d = f(x_n)$$

for $d \in \mathbb{R}^N$ to obtain the term $[J(x_n)]^{-1} f(x_n)$ in the above updating formula, instead of directly inverting J. `numpy.linalg.solve()` function solves a system of linear equations.

Q2 [3 marks]

Consider the two-period consumption-saving problem in the lecture 3 notebook.

We assume the following:

- Log utility: $u(c) = \ln c$,
- $\beta = 1 + R_2 = 1$,
- $Y_1 = 1$,
- $Y_2 = 0$ with probability ϵ and $Y_2 = 1/(1-\epsilon)$ with probability $1-\epsilon$.

In this setting, higher ϵ implies more risk in the mean-preserving spread sense. The goal here is to confirm the precautionary saving behavior — optimal savings increase with ϵ.

Compute the optimal saving for $\epsilon = 0, 0.01, 0.02, \cdots, 0.9$ (equally spaced, 91 values between 0 and 0.9) and plot the optimal saving against ϵ.

To find the optimal saving given ϵ, it is recommended to use

$$0 = \beta \mathbb{E}_1 [\frac{u'(Y_2 + (1 + R_2)x)}{u'(Y_1 - x)}](1 + R_2) - 1$$

for a root-finding purpose. You can use either SciPy routine(s) or your own function for a root-finding. You may also use a tight convergence criterion.

You may use a brute-force method by constructing a loop over values of ϵ and compute a root for each ϵ. **For this question, however, I would like you to try to make your code faster than**

the brute-force method. Out of 3 marks, 1 mark will be awarded specifically when a faster method is successfully implemented. Describe how you (try to) make your code faster. (Hint: Look at the method in Q1)

Q3 [2.5 marks]

To find a good enough α in the line search strategy, we often use the *Wolfe conditions*:

1. $f(x_n + \alpha_n p_n) \leq f(x_n) + c_1 \alpha_n p_n^T \nabla f(x_n)$ for small $c_1 \in (0, 1)$, and
2. $p_n^T \nabla f(x_n + \alpha_n p_n) \geq c_2 p_n^T \nabla f(x_n)$ for $c_2 \in (c_1, 1)$.

We take for granted that p_n is such that $p_n^T \nabla f(x_n) < 0$. I.e. the search direction p_n always decreases the objective function at least locally.

In this question, we first examine a case in which requiring only the first condition can result in a convergence to a point that is not even a local minimizer of f, and then the second condition can avoid such a problem.

The function f we use is

$$f(x) = \begin{cases} \ln x & \text{if } x \geq 1, \text{ and} \\ \frac{1}{2}(x^2 - 1) & \text{if } x < 1. \end{cases}$$

There is a unique local (and hence global) minimizer, $x = 0$.

Consider sequences $\{x_n\}$, $\{p_n\}$, and $\{\alpha_n\}$ such that

- $p_n = -1$ for all $n = 0, 1, ...$;
- $\alpha_n = \frac{1}{2^{n+1}}$ for all $n = 0, 1, ...$; and
- $x_0 = 2$ and $x_{n+1} = x_n + \alpha_n p_n$ for $n = 0, 1, ...$

Demonstrate that (1) $x_n \in (1, 2]$ for all n, that (2) $x_n \downarrow 1$ as $n \to \infty$, and that (3) the first Wolfe condition is satisfied for all n. (Hint: f is a concave function for $x \geq 1$.)

Now let us examine what happens when we impose the second condition. Suppose again that we start from $x_0 = 2$ and $p_0 = -1$. Under the second condition, what is the maximal value $x_1 = x_0 + \alpha_0 p_0$ can take?

Q4 [2.5 marks]

Consider a seller that faces a large, fixed number of buyers. There are two types of buyers, high type and low type, and high type buyers have stronger preferences for quality of goods.

The high type's utility when she purchases a good with quality q at price p is given by:

$$\theta_H \times q - p,$$

and the low type's is given by:

$$\theta_L \times q - p,$$

where $\theta_H > \theta_L > 0$ are parameters. Buyers decide whether to buy one unit of good or not. When the seller sells one unit of good of quality q at price p, its profit (per unit) is given by

$$p - q^2.$$

I.e. quality-improvement cost is a quadratic function.

The seller chooses type-specific pair of quality and price, (q_H, p_H) for the high type and (q_L, p_L) for the low type. For simplicity, we assume that there are equal number of high-type buyers and low-type buyers.

We assume that the seller's choice must satisfy buyers' *individual rationality* constraints, which requires that the buyers' utility from purchasing a good cannot be lower than their outside option:

$$\begin{aligned} \theta_H \times q_H - p_H &\geq U_H, \\ \theta_L \times q_L - p_L &\geq U_L, \end{aligned}$$

where U_H and U_L are exogenous parameters. One natural choice may be $U_H = U_L = 0$, because not purchasing a good (i.e. obtaining zero quality at zero price) is always an option for buyers.

Another assumption we make is that the seller cannot distinguish buyers' type (i.e. a buyer's type is private information for herself). Hence, the seller must choose (q_H, p_H) and (q_L, p_L) so that high type buyers (weakly) prefer (q_H, p_H) to (q_L, p_L) and low type buyers (weakly) prefer (q_L, p_L) to (q_H, p_H). This requirement leads to the *incentive compatibility constraints*:

$$\begin{aligned} \theta_H \times q_H - p_H &\geq \theta_H \times q_L - p_L, \\ \theta_L \times q_L - p_L &\geq \theta_L \times q_H - p_H \end{aligned}$$

The seller's profit maximization problem is therefore:

$$\max_{q_H, p_H, q_L, p_L} \frac{1}{2}(p_H - q_H^2) + \frac{1}{2}(p_L - q_L^2)$$

subject to the two incentive compatibility constraints and the two individual rationality constraints. (The objective function here is expressed in per-buyer term.)

Suppose we set parameters $(\theta_H, \theta_L, U_H, U_L) = (2, 1, 0, 0)$. Compute a solution to the above problem. Is a solution *separating*, i.e. $(q_H, p_H) \neq (q_L, p_L)$, or *pooling*, i.e. $(q_H, p_H) = (q_L, p_L)$? Do the individual rationality constraints bind? What about the incentive compatibility constraints? See how the answers change when you

1. Increase the gap θ_H, or

2. Increase U_H.

(You do not have to try a large number of parameter specifications. Just try two or three different values for each of the two parameters.)