

# Workshop 5: Numerical optimization

FIE463: Numerical Methods in Macroeconomics and Finance using Python

Richard Foltyn  
NHH Norwegian School of Economics

February 13, 2025

See GitHub repository for notebooks and data:

<https://github.com/richardfoltyn/FIE463-V25>

## 1 Consumption-leisure choice

In the lecture we analyzed how a household optimally chooses consumption and labor supply. In this exercise, we revisit this setting, imposing different assumptions on the household's utility function.

Assume that the household chooses *consumption*  $c$  and *leisure*  $\ell$  in order to maximize

$$\begin{aligned} \max_{c, \ell} \quad & \left\{ \log(c) + \psi \log(\ell) \right\} \\ \text{subject to} \quad & c = a + (1 - \ell)w \\ & c \geq 0 \\ & 0 \leq \ell \leq 1 \end{aligned}$$

The second line represents the budget constraint, where  $a$  is the level of assets and  $w$  is the wage rate. The last line imposes that leisure has to be between 0 and 1, where 1 represents the entire time endowment of the household. Consequently, for a given leisure choice, the household supplies the remaining time  $1 - \ell$  in the labor market and earns  $(1 - \ell)w$  in labor income.

### 1.1 Analytical solution

To solve the problem, we set up the Lagrangian,

$$\mathcal{L} = \log(c) + \psi \log(\ell) + \lambda(a + (1 - \ell)w - c)$$

which gives us the first-order conditions

$$\left. \begin{aligned} \frac{\partial \mathcal{L}}{\partial c} = \frac{1}{c} - \lambda &= 0 \\ \frac{\partial \mathcal{L}}{\partial \ell} = \frac{\psi}{\ell} - \lambda w &= 0 \end{aligned} \right\} \implies c = \frac{\ell w}{\psi}$$

This condition characterizes the optimal allocation of consumption and leisure for a given wage rate. Substituting this back into the budget constraint, we can solve for optimal leisure:

$$\ell^* = \frac{\psi}{1 + \psi} \frac{a + w}{w}$$

The optimal consumption level follows by plugging the above expression into the optimality condition we derived earlier.

$$c^* = \frac{1}{1 + \psi} (a + w)$$

## 1.2 Numerical solution

Instead of solving this problem analytically, we can use a numerical solver to obtain the optimal choices for a given set of parameters.

1. Write a function `util()` which takes leisure  $\ell$  and parameters as arguments and returns the corresponding utility level.

```
def util(l, a, w, psi):  
    """  
    Compute utility for given leisure choice and parameters.  
    """
```

Use the budget constraint to obtain consumption for a given leisure choice.

2. Assume that the model is parametrized by  $a = 0$ ,  $w = 1$  and  $\psi = 1$ . Use the univariate minimizer `minimize_scalar()` from the module `scipy.optimize` and solve for the optimal leisure choice. Use the `util()` function you defined previously as the objective.

*Hint:* Recall that you are using a *minimizer* to solve a maximization problem, hence you need to flip the sign of your objective function, e.g., by using a lambda expression.

*Hint:* Use the bounds argument to `minimize_scalar()` to restrict the set of admissible values for  $\ell$ .

3. You want to make the above implementation more flexible so you can experiment with different parameter values. Write a function `solve_cons_leisure()` which takes *only* the parameters as arguments and returns a tuple containing the optimal consumption and leisure choices:

```
def solve_cons_leisure(a, w, psi):  
    """  
    Solve for the optimal consumption/leisure choice for given parameters.  
    """
```

For this part, you should just reuse the code you wrote earlier by wrap it in a function.

4. You are interested in how the optimal leisure and consumption choices depend on the utility weight on leisure,  $\psi$ . Use the function you wrote in the last part to evaluate optimal consumption and leisure for a grid of 21 points for  $\psi$  that is uniformly spaced on the interval  $[0, 2]$ .

Create a graph with two panels, one for consumption and one for leisure, plotting the optimal choices against the values of  $\psi$ . What is the intuition for the slope of  $c$  and  $\ell$  as  $\psi$  changes?

5. Since this problem admits an analytical solution, compute the *exact* optimal choices using the equations from above for each value on the  $\psi$  grid you used in the previous part.

Adapt your graph to show both the numerical and the analytical solution. Use a legend to clearly label what is being plotted.

## 2 Consumption & labor supply with commuting

Recall the consumption & labor supply problem we studied in the lecture: the individual has preferences over consumption and leisure characterized by

$$u(c, h) = \frac{c^{1-\gamma} - 1}{1-\gamma} - \psi \frac{h^{1+1/\theta}}{1+1/\theta}$$

where  $c \geq 0$  is the amount consumed, and  $h \geq 0$  (for “hours”) is the amount of labor the individual wishes to work (the remaining time is then consumed as leisure). The individual does not like working, and therefore the amount of hours worked enters as a disutility term in the utility function. Note that for

the special case of  $\gamma = 1$ , the utility from consumption becomes  $\log(c)$ , the natural logarithm of  $c$ , so that

$$u(c, h) = \log(c) - \psi \frac{h^{1+1/\theta}}{1+1/\theta}$$

Unlike in the lecture, we assume that the individual can either work part-time for  $h < H$  hours in their neighborhood for a lower wage  $w_\ell$  or commute to the city to work full-time for  $h \geq H$  hours at a higher wage  $w_h$ . However, commuting to the city incurs a fixed commuting cost  $\kappa$ . The budget constraint is therefore given by

$$c = \begin{cases} a + w_\ell \cdot h & \text{if } h < H \\ a + w_h \cdot h - \kappa & \text{if } h \geq H \end{cases}$$

This is an example of a problem that is harder to solve analytically because of the discontinuous jump at the choice  $h = H$ . We therefore proceed to directly solve it numerically.

1. Implement a function to evaluate utility,

```
def util(c, h, gamma, psi, theta):
    """
    Compute the utility of a given consumption/labor supply choice.
    """
```

*Hint:* This function is exactly the same as the one we used in the lecture.

2. Write a function with the signature

```
def util_h(h, a, w_l, w_h, gamma, psi, theta, H, kappa):
    """
    Compute utility for given labor choice and parameters.
    """
```

which uses `util()` to return the utility associated with a hours choice  $h$ . Use the budget constraint to obtain the implied consumption level.

*Hint:* To get a vectorized function (which can deal with  $h$  arguments that are arrays), use `np.where()` instead of an `if` statement when dealing with the two cases in the budget constraint.

3. Use the following parameters to numerically solve the problem:

```
a = 0          # initial assets
w_l = 0.75     # low wage (neighborhood)
w_h = 3        # high wage (city)
gamma = 1.0    # Relative risk aversion
psi = 2.0      # weight on disutility of labor
theta = 0.5    # labor supply elasticity
H = 1.0       # cut-off for part-time work
kappa = 2.0    # commuting cost
```

First, solve the problem using grid search, evaluating utility on a grid of 10,000 points for  $h$  on the interval  $[0.1, 2]$ . Plot utility as a function of  $h$  for the values on the hours grid.

4. Use the scalar minimizer `minimize_scalar()` and the function `util_h()` you wrote to locate the optimum. Do you get a similar result as with grid search?
5. Lastly, use `util_h()` and a derivative-based minimizer implemented by `minimize()` to find the optimum.

*Hint:* Call this minimizer as follows:

```
res = minimize(
    lambda x: -util_h(x, a, w_l, w_h, gamma, psi, theta, H, kappa),
    x0 = x0,
    method='L-BFGS-B',
    bounds=((0, None),)
)
```

This selects the 'L-BFGS-B' algorithm, a derivative-based algorithm which also supports bounds. We impose these using the `bounds=((0, None),)` argument which imposes the  $h \geq 0$  lower bound.

You need to specify an initial guess `x0`: experiment with `x0=0.75` and `x0=1.1`. Do these yield the same result?

*Hint:* The `minimize()` returns a result object where the optimum is stored in the `x` attribute, which in this case is an array of length 1.