Matthew Fitzgerald
Econ 204B
March 1, 2020

# Problem Set # 4

**Problem 1: Working in Academics**

Individuals have preferences over consumption $c$ and leisure $l$ and are endowed with one unit of time. Workers maximize

$$\mathbb{E}\int_0^\infty e^{-rt}[c_t + v(l_t)]dt$$

where $v$ is both increasing and concave with $v(1) = 0$. Suppose that there are three states of the world: unemployed, non-tenured, and tenured. Unemployed workers search for jobs and receive an unemployed utility flow of $b < \underline{w}$. They receive wage offers at a Poisson rate $\lambda_0$ from a distribution $F(w)$ where $F(\underline{w}) = 0$ and $F(\overline{w}) = 1$. If workers accept an offer, they enter the non-tenured state where they choose work intensity $n$ such that $n + l = 1$. Non-tenured workers face the tenure-clock. Their tenure decision arrives at a Poisson rate $\lambda_t$ and they have a probability $\alpha(n)$ that they receive tenure, otherwise they lose their job. Once tenured, workers are set for life. They receive their non-tenured wage $w$ forever and cannot lose their job.

**(a)** Write down the flow Bellman for each state. Show that $V_T \geq V_{NT}$.

Before we begin, make note that I will utilize the following integration-by-parts result in the writing of the flow Bellman equations (where appropriate).

$$\begin{aligned}
\int_{\underline{w}}^{\overline{w}}[V_{NT}(t) - V_U]dF(t) &= [V_{NT}(t) - V_U]F(t)\Big|_{\underline{w}}^{\overline{w}} - \int_{\underline{w}}^{\overline{w}} V'_{NT}(t)F(t)dt \\
&= V_{NT}(\overline{w}) - V_U - \int_{\underline{w}}^{\overline{w}} V'_{NT}(t)F(t)dt \\
&= \int_{\underline{w}}^{\overline{w}} V'_{NT}(t)[1 - F(t)]dt
\end{aligned}$$

Now let's start in discrete time and move to continuous. We'll start with the unemployed Bellman:

$$
\begin{aligned}
V_U &= bdt + \frac{\lambda_0 dt}{1+rdt} \int_{\underline{w}}^{\overline{w}} \max[V_{NT}(t), V_U] dF(t) + \frac{1-\lambda_0 dt}{1+rdt} V_U \\
\frac{1+rdt}{1+rdt} V_U - \frac{1}{1+rdt} V_U &= bdt + \frac{\lambda_0 dt}{1+rdt} \int_{\underline{w}}^{\overline{w}} \max[V_{NT}(t), V_U] dF(t) - \frac{\lambda_0 dt}{1+rdt} V_U \\
\frac{rdt}{1+rdt} V_U &= b + \frac{\lambda_0 dt}{1+rdt} \int_{\underline{w}}^{\overline{w}} \max[V_{NT}(t) - V_U, 0] dF(t) \\
V_U &= \frac{1+rdt}{r} b + \frac{\lambda_0}{r} \int_{\underline{w}}^{\overline{w}} \max[V_{NT}(t) - V_U, 0] dF(t)
\end{aligned}
$$

Now take $dt \to 0$

$$
\begin{aligned}
rV_U &= b + \lambda_0 \int_{\underline{w}}^{\overline{w}} \max[V_{NT}(t) - V_U, 0] dF(t) \\
rV_U &= b + \lambda_0 \int_{w_R}^{\overline{w}} [V_{NT}(t) - V_U] dF(t) + \lambda_0 \int_{\underline{w}}^{w_R} 0 dF(t) \\
rV_U &= b + \lambda_0 \int_{w_R}^{\overline{w}} [V_{NT}(t) - V_U] dF(t)
\end{aligned}
$$

Now using the integration by parts result from above we have

$$
rV_U = b + \lambda_0 \int_{w_R}^{\overline{w}} V'_{NT}(t)[1 - F(t)] dt
$$

Now for not tenured:

$$
\begin{aligned}
V_{NT}(w) &= \max_n \{ wdt + v(1-n)dt + \frac{\lambda_t dt}{1+rdt} [\alpha(n) V_T(w) + (1-\alpha(n)) V_U] + \frac{1-\lambda_t dt}{1+rdt} V_{NT}(w) \} \\
&\vdots \\
(r+\lambda_t) V_{NT}(w) &= \max_n \{ (1+rdt)w + (1+rdt)v(1-n) + (1+rdt)\lambda_t [\alpha(n) V_T(w) + (1-\alpha(n)) V_U] \}
\end{aligned}
$$

Now take $dt \to 0$

$$
(r+\lambda_t) V_{NT}(w) = \max_n \{ w + v(1-n) + \lambda_t [\alpha(n) V_T(w) + (1-\alpha(n)) V_U] \}
$$

Finally for tenured we have

$$
\begin{aligned}
V_T(w) &= wdt + \frac{1}{1+rdt} V_T(w) \\
rV_T(w) &= w(1+rdt) \\
rV_T(w) &= w
\end{aligned}
$$

Thus the flow Bellman equations for each state will be

$$rV_U = b + \lambda_0 \int_{WR}^{\overline{w}} V'_{NT}(t)[1 - F(t)]dt$$

$$(r + \lambda_t)V_{NT}(w) = \max_n \{w + v(1-n) + \lambda_t [\alpha(n)V_T(w) + (1-\alpha(n))V_U]\}$$

$$rV_T(w) = w$$

Now we can show that $V_T(w) \geq V_{NT}(w)$ for $w \geq w_R$.

*Proof.* Assume that $w \geq w_R$. Then note that $v(\cdot)$ is maximized at $n = 0$ where it is equal to 0. Now let's look at $[\alpha(n)V_T(w) + (1-\alpha(n))V_U]$. We have three cases: $V_T(w) > V_U$, $V_T(w) = V_U$, and $V_T(w) < V_U$. First we will show that $V_T(w) < V_U$ is not possible. Assume, for a contradiction, that $V_T(w) < V_U$. Then our RHS term is maximized when $\alpha(n) = 0$ so we have

$$\begin{aligned}
(r + \lambda_t)V_{NT}(w) &\leq w + \lambda_t V_U \\
(r + \lambda_t)V_{NT}(w) &\leq w + \lambda_t V_{NT}(w) && \text{(Since } V_U \leq V_{NT}(w) \text{ for } w \geq w_R\text{)} \\
rV_{NT}(w) + \lambda_t V_{NT}(w) &\leq w + \lambda_t V_{NT}(w) \\
V_{NT}(w) &\leq \frac{w}{r} = V_T(w) && (\spadesuit)
\end{aligned}$$

Note that $V_U \leq V_{NT}(w)$ for $w \geq w_R$. Then combining this with ($\spadesuit$) we have

$$V_U \leq V_{NT}(w) \leq V_T(w) \implies V_U \leq V_T(w)$$

But we assumed that $V_U > V_T(w)$, which is a contradiction. Now assume $V_T(w) > V_U$. Then $\alpha(n) = 1$ maximizes our RHS term and we have:

$$\begin{aligned}
(r + \lambda_t)V_{NT}(w) &\leq w + \lambda_t V_T(w) \\
(r + \lambda_t)V_{NT}(w) &\leq rV_T(w) + \lambda_t V_T(w) && \text{(plugging in } w = rV_T(w)\text{)} \\
(r + \lambda_t)V_{NT}(w) &\leq (r + \lambda_t)V_T(w) \\
V_{NT}(w) &\leq V_T(w)
\end{aligned}$$

as desired. Note that if $V_T(w) = V_U$ we have

$$(r + \lambda_t)V_{NT}(w) \leq w + \lambda_t V_T(w)$$

and thus this reduces to the case where $V_T(w) > V_U$. Therefore we have shown that $V_T(w) \geq V_{NT}(w)$ for $w \geq w_R$. $\square$

**(b)** Consider a world where $\alpha(n) = 1$. Derive the reservation wage and policy rule for work intensity.

With $\alpha(n) = 1$, there is no possibility of losing one's job, and therefore no need for both the tenure and non-tenure states. Thus we can simplify our flow Bellman equations to the following (subsuming both employed states into "$E$"):

$$\begin{aligned} rV_U &= b + \lambda_0 \, \mathbb{E}\left[\max\left(V_E - V_U, 0\right)\right] \\ &= b + \lambda_0 \int_{WR}^{\overline{w}} V_E'(t)[1 - F(t)]dt \end{aligned}$$

$$rV_E = w.$$

To find the reservation wage, we follow the usual steps

① find $rV_E(WR)$

$$rV_E(WR) = WR$$

② construct $rV_E(WR) = rV_U$ and then solve for $WR$

$$WR = b + \lambda_0 \int_{WR}^{\overline{w}} V_E'(t)[1 - F(t)]dt$$

③ find $V_E'(w)$ and plug in

$$V_E'(w) = \frac{1}{r} \qquad \Longrightarrow \qquad \boxed{WR = b + \frac{\lambda_0}{r} \int_{WR}^{\overline{w}} [1 - F(t)]dt}$$

Last, as alluded to earlier, since there is no possibility of losing one's job, the optimum level of work intensity will therefore be $n^* = 0$.

**(c)** Assume that $\alpha(n)$ is strictly increasing and concave. Derive the reservation wage and policy rule for work intensity.

First suppose that agents are optimally choosing their work intensity, $n^*$. Then, to find the reservation wage, we follow the usual steps (using the value functions in part (a)):

① find $V_{NT}(WR)$

$$(r+\lambda_t)V_{NT}(WR) = WR + v(1-n^*) + \lambda_t\left[\alpha(n)V_T(WR) + (1-\alpha(n^*))V_U\right]$$

$$(r+\lambda_t)V_{NT}(WR) - \lambda_t(1-\alpha(n^*))V_U = WR + v(1-n^*) + \lambda_t\alpha(n^*)\frac{WR}{r} \qquad \text{(note: } V_T(WR) = \tfrac{WR}{r}\text{)}$$

$$(r+\lambda_t\alpha(n^*))V_{NT}(WR) = WR + v(1-n^*) + \frac{WR\lambda_t\alpha(n^*)}{r} \qquad \text{(note } V_{NT}(WR) = V_U\text{)}$$

$$V_{NT}(WR) = \frac{1}{r+\lambda_t\alpha(n^*)}\left[WR\left(1+\frac{\lambda_t\alpha(n^*)}{r}\right) + v(1-n^*)\right]$$

$$V_{NT}(WR) = \frac{WR}{r} + \frac{v(1-n^*)}{r+\lambda_t\alpha(n^*)} \qquad \text{(simplifying)}$$

② construct $V_{NT}(WR) = V_U$ and then solve for $WR$

$$\frac{WR}{r} + \frac{v(1-n^*)}{r+\lambda_t\alpha(n^*)} = \frac{1}{r}\left(b + \lambda_0\int_{WR}^{\overline{w}} V'_{NT}(t)[1-F(t)]dt\right)$$

$$WR + \frac{rv(1-n^*)}{r+\lambda_t\alpha(n^*)} = b + \lambda_0\int_{WR}^{\overline{w}} V'_{NT}(t)[1-F(t)]dt \qquad \text{(multiply by } r\text{)}$$

$$WR = b + \lambda_0\int_{WR}^{\overline{w}} V'_{NT}(t)[1-F(t)]dt - \frac{rv(1-n^*)}{r+\lambda_t\alpha(n^*)}$$

③ find $V'_{NT}(w)$ and plug in

$$V_{NT}(w) = \frac{1}{r+\lambda_t}\left\{w + v(1-n) + \lambda_t\left[\alpha(n)V_T(w) + (1-\alpha(n))V_U\right]\right\}$$

$$\implies V'_{NT}(w) = \frac{1}{r+\lambda_t}\left\{1 + \lambda_t \alpha(n^*)V'_T(w)\right\} \qquad \text{(note: } V'_T(w) = \tfrac{1}{r}\text{)}$$

$$= \frac{r+\lambda_t\alpha(n^*)}{r(r+\lambda_t)}$$

And so putting this altogether the reservation wage is given by

$$\boxed{WR = b + \lambda_0 \frac{r+\lambda_t\alpha(n^*)}{r(r+\lambda_t)} \int_{WR}^{\overline{w}} [1-F(t)]dt - \frac{rv(1-n^*)}{r+\lambda_t\alpha(n^*)}}.$$

We can also find the optimal policy choice of work intensity, $n$. To do so, we maximize $V_{NT}(w)$ w.r.t. $n$:

$$\frac{\partial V_{NT}(w)}{\partial n} = \frac{1}{r+\lambda_t}\left\{-v'(1-n) + \lambda_t\left[\alpha'(n)V_T(w) - \alpha'(n)V_U\right]\right\} = 0$$

$$\implies \boxed{v'(1-n^*) = \lambda_t \alpha'(n^*)\left[V_T(w) - V_U\right]}.$$

This says that the optimal level of work intensity will set the marginal value of leisure equal to the instantaneous marginal increase in the probability of getting the "tenure-unemployment surplus."

**Problem 2: Value Function Iteration with Matrices**

This problem will have you improve the efficiency of the code you completed in Problem 3 of Problem Set 3. You previously completed code that performed value function iteration by looping through all possible values of $k$ in your capital grid. For each value of $k$ in your grid, the code found the optimal $k'$ to save for next period and computed the value of starting with $k$ in this period (under the assumption that you will continue to act optimally in the future). In this problem, we will remove the inner loop and instead implement the code using matrices.

Open the file "ValueFnIterationWithMatricesIncomplete.jl".

**(a)** This file implements value function iteration, but it is incomplete. Fill in the code where necessary and make sure it runs properly by comparing it to the code from "ValueFunctionIteration.jl".

**(b)** Write additional code in the script that plots the value function and the policy function against $k$ in separate graphs.

**(c)**. Time your completed code in “ValueFnIterationWithMatricesIncomplete.jl” and in “ValueFunctionIteration.jl” using the following parameters: grid points = 1000, $\alpha = 0.3$, $\delta = 0.5$. Plugging in the appropriate parameter values, you can time your functions in these scripts using:

```
@time val_fun_iter_mats(prod_fun,num_grid_points, alpha, delta)
@time val_fun_iter(prod_fun,num_grid_points, alpha, delta)
```

Note that you should run each @time line twice and record the time from the second run. How much faster is your new code?

**(d)**. What portion of code in “ValueFnIterationWithMatricesIncomplete.jl” has been taken out of the while loop compared to that in “ValueFunctionIteration.jl”? Why can we now remove this code from the while loop?

The portion of code that constructs the $k$ and $k'$ matrices, the consumption matrix, and the utility matrix. In the last problem set when we were doing value function iteration with loops, we needed to take one value of $k$ at a time in a for loop to fill in the value function. Now that we're doing this with matrices, we can construct the consumption and utility matrices for every combination of $k$ and $k'$, and these matrices will be the same for every run of the while loop. Thus we are able to do this process once and remove it from the while loop.