Problem Set 1

Hyoungchul Kim

2025-02-15

Part 1: Theory

In this problem set, you will use a Mirrlees-style model to characterize the optimal linear income tax with a lump-sump grant. You'll then solve for the optimal tax in a numerical simulation.

1.

$$\int_0^1 x^2 \, dx = \frac{1}{3}$$

Part 2: Numerical application

1.

2.

```r
# Given parameters
t <- 0.3  # Current tax rate
b <- 5000 # Lump-sum benefit
k <- 1    # Utility parameter

# Income distribution
income_distribution <- data.frame(
  type = c("Low", "Middle", "High"),
  proportion = c(0.3, 0.6, 0.1),
  earnings = c(20000, 50000, 200000)
```

```r
)

# Given compensated elasticity
elasticity_c <- 0.3

# Problem 1: Compute Implied Abilities
compute_w <- function(earnings, t, k) {
  earnings / (1 - t)
}
income_distribution$w <- compute_w(income_distribution$earnings, t, k)
print(income_distribution)
```

```
    type proportion earnings         w
1    Low        0.3    2e+04  28571.43
2 Middle        0.6    5e+04  71428.57
3   High        0.1    2e+05 285714.29
```

```r
# Problem 2: Compute Exogenous Expenditures
E <- sum(income_distribution$proportion * (t * income_distribution$earnings -
↪  b))
print(paste("Exogenous expenditures (E):", E))
```

```
[1] "Exogenous expenditures (E): 11800"
```

3.

```r
# Problem 3: Compute Earnings Choices
compute_earnings <- function(w, t, k) {
  w * ((1 - t)^(1 / k))
}
income_distribution$earnings_no_tax <-
↪  compute_earnings(income_distribution$w, 0, k)
print(income_distribution)
```

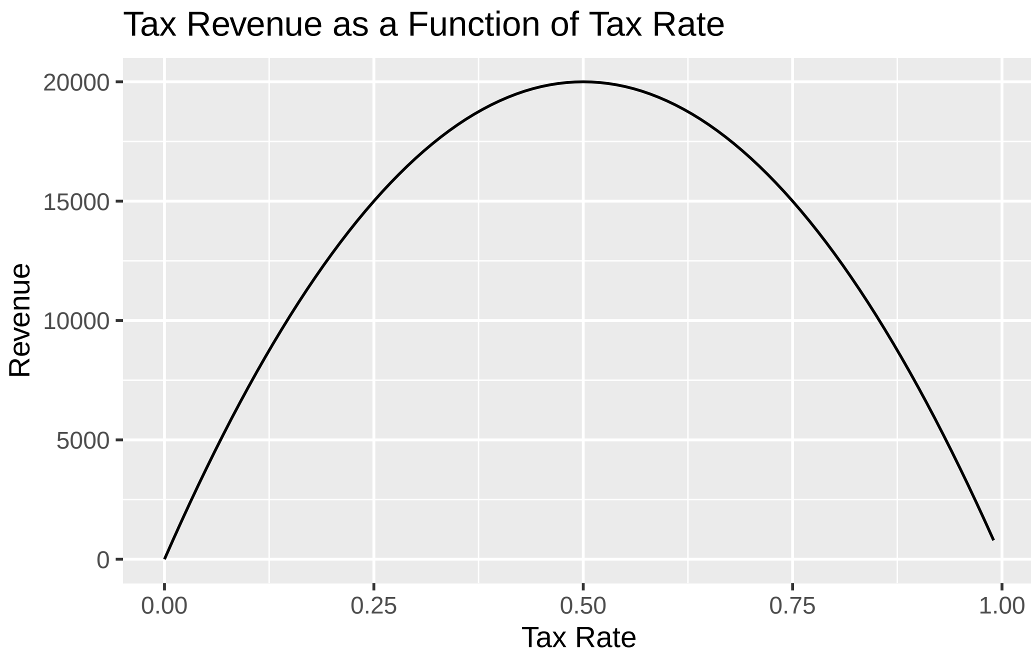
```
    type proportion earnings         w earnings_no_tax
1    Low        0.3    2e+04  28571.43        28571.43
2 Middle        0.6    5e+04  71428.57        71428.57
3   High        0.1    2e+05 285714.29       285714.29
```

4.

```r
# Problem 4: Find Revenue-Maximizing Tax Rate
compute_revenue <- function(t, income_distribution, k) {
  earnings <- compute_earnings(income_distribution$w, t, k)
  sum(income_distribution$proportion * t * earnings)
}
revenue_function <- function(t) -compute_revenue(t, income_distribution, k)
optimal_tax <- optimize(revenue_function, interval = c(0, 0.99))$minimum
print(paste("Revenue-maximizing tax rate:", optimal_tax))
```

```
[1] "Revenue-maximizing tax rate: 0.5"
```

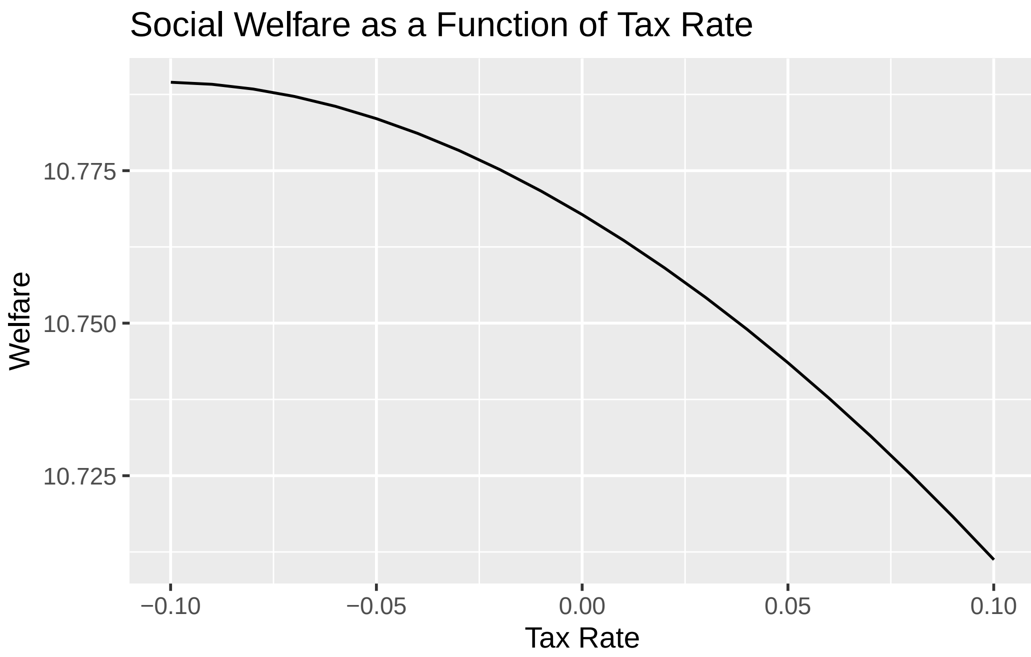
```r
t_values <- seq(0, 0.99, by = 0.01)
revenues <- sapply(t_values, compute_revenue, income_distribution, k)

ggplot(data.frame(t_values, revenues), aes(x = t_values, y = revenues)) +
  geom_line() +
  labs(title = "Tax Revenue as a Function of Tax Rate", x = "Tax Rate", y =
  ↪  "Revenue")
```



```r
# Problem 5: Compute Welfare-Maximizing Tax Rate
compute_social_welfare <- function(t, income_distribution, k, E) {
  earnings <- compute_earnings(income_distribution$w, t, k)
  b <- (sum(income_distribution$proportion * t * earnings) - E) /
  ↪  sum(income_distribution$proportion)
  utilities <- log(earnings - t * earnings + b - (1 / (1 + k)) * (earnings /
  ↪  income_distribution$w)^(1 + k))
  sum(income_distribution$proportion * utilities)
}
welfare_function <- function(t) -compute_social_welfare(t,
  ↪  income_distribution, k, E)
optimal_welfare_tax <- optimize(welfare_function, interval = c(0,
  ↪  0.99))$minimum
print(paste("Welfare-maximizing tax rate:", optimal_welfare_tax))
```

```
[1] "Welfare-maximizing tax rate: 5.30500266259972e-05"
```

```r
t_values_welfare <- seq(optimal_welfare_tax - 0.1, optimal_welfare_tax + 0.1,
  ↪  by = 0.01)
welfares <- sapply(t_values_welfare, compute_social_welfare,
  ↪  income_distribution, k, E)

ggplot(data.frame(t_values_welfare, welfares), aes(x = t_values_welfare, y =
  ↪  welfares)) +
  geom_line() +
  labs(title = "Social Welfare as a Function of Tax Rate", x = "Tax Rate", y
    ↪  = "Welfare")
```



5.

```r
tUpdate <- function(t_current, income_distribution, k, E) {
  lambda_function <- function(t) {
    earnings <- compute_earnings(income_distribution$w, t, k)
    sum(income_distribution$proportion * log(earnings - t * earnings +
     ↪  (sum(income_distribution$proportion * t * earnings) - E) /
     ↪  sum(income_distribution$proportion)))
  }
  optimal_t <- optimize(lambda_function, interval = c(0, 0.99))$minimum
  return(optimal_t)
}

# Iteratively update tax rate
iterations <- 10
t_values_iter <- numeric(iterations)
t_values_iter[1] <- t
for (i in 2:iterations) {
  t_values_iter[i] <- tUpdate(t_values_iter[i - 1], income_distribution, k,
 ↪  E)
}
```

```
Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced
```

```
Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value
```

```
Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced
```

```
Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value
```

```
Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced
```

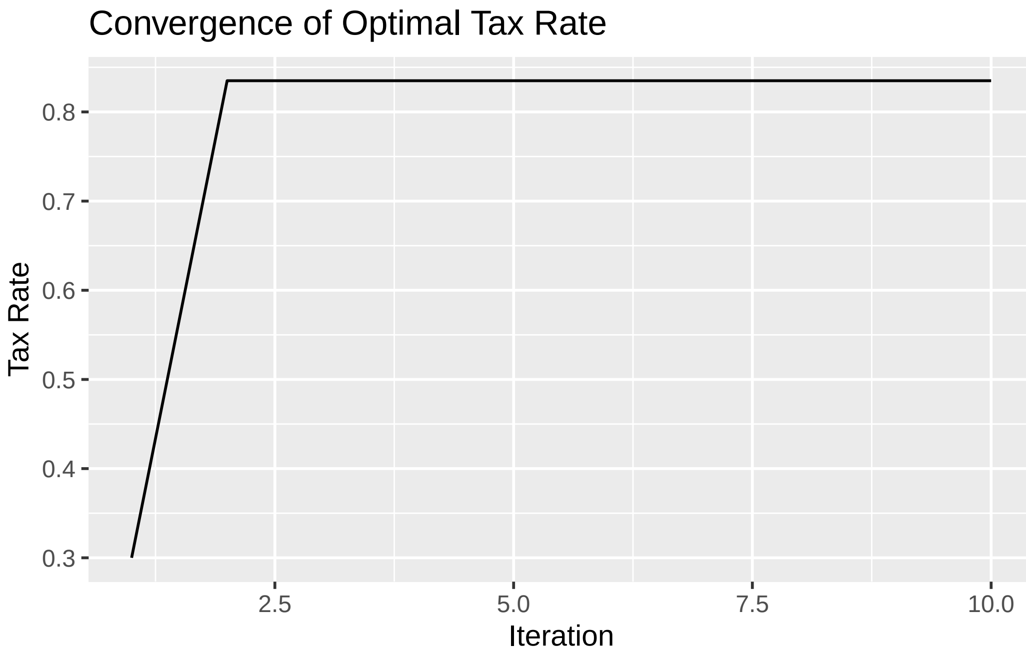
```
Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value
```

```
Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value

Warning in log(earnings - t * earnings + (sum(income_distribution$proportion *
: NaNs produced

Warning in optimize(lambda_function, interval = c(0, 0.99)): NA/Inf replaced by
maximum positive value
```

```r
print(t_values_iter)
```

```
 [1] 0.3000000 0.8349915 0.8349915 0.8349915 0.8349915 0.8349915 0.8349915
 [8] 0.8349915 0.8349915 0.8349915
```

```
ggplot(data.frame(iteration = 1:iterations, tax_rate = t_values_iter), aes(x
↪   = iteration, y = tax_rate)) +
  geom_line() +
  labs(title = "Convergence of Optimal Tax Rate", x = "Iteration", y = "Tax
  ↪   Rate")
```



Conclusion

Summarize key findings or insights.