Utility Maximization in R

James Bang

6/3/2022

Analytic Solution

Equilibrium

The following code finds the first-order conditions for a utility maximization problem involving a two-good Cobb-Douglass utility function:

$$\max_{x} U(x_1, x_2) = x_1^{\alpha} x_2^{1-\alpha}$$
$$s.t. : p_1 x_1 + p_2 x_2 \leq I$$

The "Lagrangean" function for this problem is:

$$\mathcal{L} = x_1^{\alpha} x_2^{1-\alpha} + \lambda(I - p_1 x_1 - p_2 x_2)$$

The `Deriv` function in the `Deriv` package "solves" basic algorithmic derivatives (i.e. memorizes the rules so you don't have to):

```
library(Deriv)
L <- 'x_1^(alpha) * x_2^(1-alpha) + lambda*(I - p_1*x_1 - p_2*x_2)'
(fml <- c(Deriv(L, 'x_1'), Deriv(L, 'x_2'), Deriv(L, 'lambda')))
```

```
## [1] "alpha * x_1^(alpha - 1) * x_2^(1 - alpha) - lambda * p_1"
## [2] "x_1^alpha * (1 - alpha)/x_2^alpha - lambda * p_2"
## [3] "I - (p_1 * x_1 + p_2 * x_2)"
```

Using various "wildcard" characters and other "regular expression" (regex) syntax, I can apply some substitutions in this formula to make the output values more readable as equations for typesetting in Rmarkdown:

```
fmlrmd <- gsub('(\\d*\\.\\d*)', '{\\1}', fml)
fmlrmd <- gsub('(\\(.)', '{\\1', fmlrmd)
fmlrmd <- gsub('(.))', '\\1)}', fmlrmd)
fmlrmd <- gsub('(\\))($)', '\\1)}\\2', fmlrmd)
fmlrmd <- gsub('\\*', '', fmlrmd)
fmlrmd <- gsub('alpha', '\\\\alpha', fmlrmd)
(fmlrmd <- gsub('lambda', '\\\\lambda', fmlrmd))
```

```
## [1] "\\alpha  x_1^{(\\alpha - 1)}  x_2^{(1 - \\alpha)} - \\lambda  p_1"
## [2] "x_1^\\alpha  {(1 - \\alpha)}/x_2^\\alpha - \\lambda  p_2"
## [3] "I - {(p_1  x_1 + p_2  x_2)}"
```

I can reference the output of the (edited) derivatives inline within the equation editor using $` r command`$. Notice the use of backquotes inside the inline equation escape character ($). To produce the following full-line equation I used $$` r fmlrmd[1]`$$. Here are the first-order conditions:

$$\alpha x_1^{(\alpha-1)} x_2^{(1-\alpha)} - \lambda p_1 = 0 \text{ (w.r.t. good 1)}$$

$$x_1^{\alpha}(1-\alpha)/x_2^{\alpha} - \lambda p_2 = 0 \text{ (w.r.t. good 2)}$$

$$I - (p_1 x_1 + p_2 x_2) = 0 \text{ (w.r.t. } \lambda\text{)}$$

Next, we can define the marginal rate of substitution as MU_1/MU_2. The `Simplify` function in `Deriv` helps clean things up.

```
mu1 <- gsub('( - lambda)(.*)', '', fml[1])
mu2 <- gsub('( - lambda)(.*)', '', fml[2])
mrs <- paste0('(', mu1, ')', '/', '(', mu2, ')')
(mrs <- Simplify(mrs))
```

```
## [1] "alpha * x_2/(x_1 * (1 - alpha))"
```

Accordingly, we can substitute various brackets and escape characters to allow translation to Markdown equation syntax:

```
mrsrmd <- gsub('(\\d*\\.\\d*)', '{\\1}', mrs)
mrsrmd <- gsub('(\\(.)', '{\\1', mrsrmd)
mrsrmd <- gsub('(.))', '\\1)}', mrsrmd)
mrsrmd <- gsub('(\\))($)', '\\1}\\2', mrsrmd)
mrsrmd <- gsub('\\*', '', mrsrmd)
mrsrmd <- gsub('alpha', '\\\\alpha', mrsrmd)
(mrsrmd <- gsub('lambda', '\\\\lambda', mrsrmd))
```

```
## [1] "\\alpha  x_2/{(x_1  {(1 - \\alpha)})}"
```

$$\alpha x_2/(x_1(1-\alpha))$$

Note that in the (linearly-homogeneous) Cobb-Douglass example, setting $MRS_{1,2} = p_1/p_2$, we get the intuition that the preference parameter α represents the budget share of good 1, or:

$$\frac{\alpha}{1-\alpha} = \frac{p_1 x_1}{p_2 x_2}$$

This, combined with the budget constraint gives a linear system of two equations and two unknowns by:

$$x_1 = \frac{\alpha p_2 x_2}{(1-\alpha)p_1} \text{ (from MRS)}$$

$$p_1 x_1 + p_2 x_2 = I \text{ (Budget Constrant)}$$

Demand for good 1 is given by:

$$x_1 = \frac{\alpha I}{p_1}$$

Demand for good 2 is given by:

$$x_2 = \frac{(1-\alpha)I}{p_2}$$

Comparative Statics

Next, we can analyze what happens when various parameters (prices, income, budget-share preferences) change.

```
d1 <- 'alpha * I/p_1'
d.d1 <- c(Deriv(d1, 'alpha'), Deriv(d1, 'I'), Deriv(d1, 'p_1'))
```

Numerical example

Let's try solving a numerical example with specific parameters. Let $\alpha = 0.25$, $p_1 = 1.5$, $p_2 = 1$, and $I = 100$.

$$\max_x U(x_1, x_2) = x_1^{0.25} x_2^{0.75}$$
$$s.t. : 1.5x_1 + x_2 \leq 100$$

This yields the following Lagrangean:

$$\mathcal{L} = x_1^{0.25} x_2^{0.75} + \lambda(100 - 1.5x_1 - x_2)$$

Equilibrium

Let's orient ourselves to the solution using the analytical derivative.

The first-order conditions are:

$$0.25(x_2^{0.75}/x_1^{0.75}) - 1.5\lambda = 0$$
$$0.75(x_1^{0.25}/x_2^{0.25}) - \lambda = 0$$
$$100 - (1.5x_1 + x_2) = 0$$

The marginal rate of substitution is:

$$x_2^{0.25}(0.25(x_2^{0.75}/x_1^{0.75}) - 1.5\lambda)/(0.75x_1^{0.25})$$

The solution follows from $MRS_{1,2} = p_1/p_2$

$$\frac{0.25}{0.75} = \frac{1.5x_1}{x_2},$$

and the budget constraint:

$$100 = 1.5x_1 + x_2.$$

Numerical solution

```
u <- function(x) {
  x[1] ^ (alpha) * x[2] ^ (1 - alpha)
}
alpha = 0.25
p_1 = 1.5
p_2 = 1
I = 100
equilibrium <-
  constrOptim(
    c(0.01, 0.01),
```

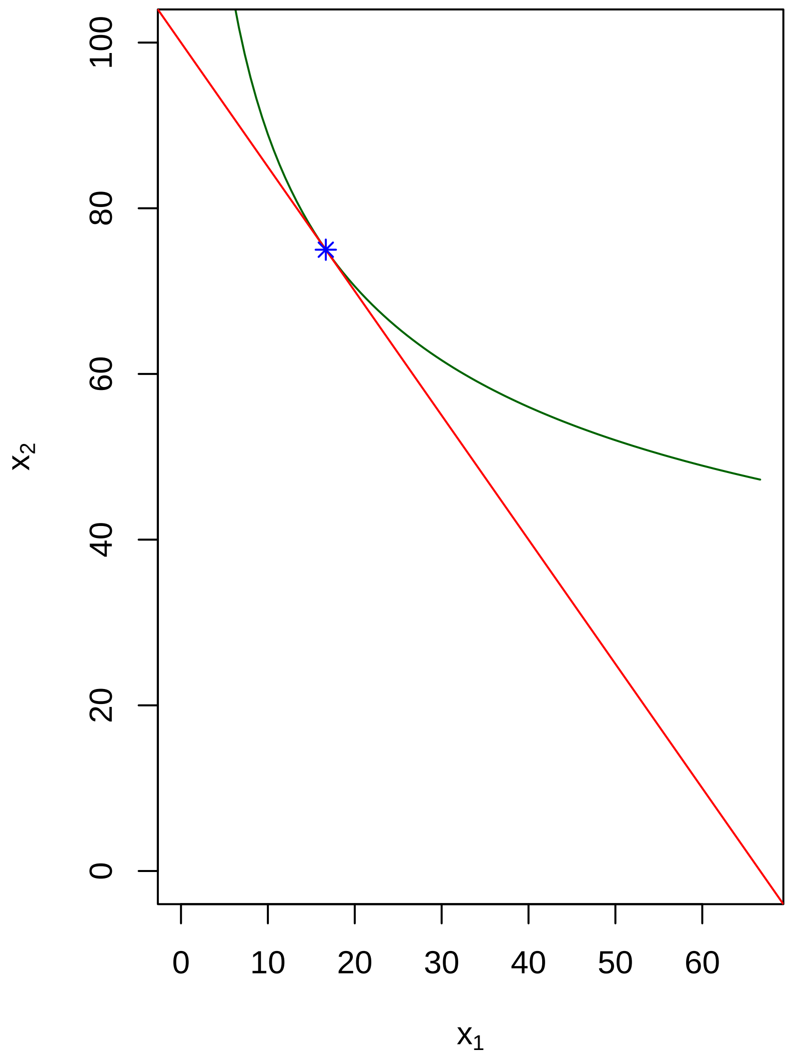
```r
    u,
    NULL,
    ui = c(-p_1,-p_2),
    ci = c(-I),
    control = list(fnscale = -1)
  )
equilibrium$lambda <-
  alpha * equilibrium$par[1] ^ (alpha - 1) * equilibrium$par[2] ^ (1 - alpha) / p_1
equilibrium[c(1, 2, 8)]
```

```
## $par
## [1] 16.66803 74.99796
##
## $value
## [1] 51.49418
##
## $lambda
## [1] 0.5148997
```

Note that the intuition from the analytical solution tells us that the "true" answer is $x_1 = 50/3 \approx 16.667$ and $x_2 = 75$ to ensure that this consumer spends 25% of their planned budget for the two goods on good 1, or one-third as much as they spend on good 2. Computational methods only give approximate answers, so it's good to double check!

We can also plot the equilibrium.

```r
curve(
  equilibrium$value ^ (1 / (1 - alpha)) / (x ^ (alpha / (1 - alpha))),
  0,
  I / p_1,
  xlab = expression(x[1]),
  ylab = expression(x[2]),
  xlim = c(0, I / p_1),
  ylim = c(0, I / p_2),
  col = 'dark green'
)
abline(
  I / p_2,
  -p_1 / p_2,
  col = 'red'
)
points(equilibrium$par[1],
       equilibrium$par[2],
       col = 'blue',
       pch = 8)
```



Comparative Statics