Demand Systems Adding Up

TheoryGuru applications

(c) Copyright 2016 by JMJ Economics

Load Economicreasoning package only if it is not already loaded

\[
\text{If}[\text{Length@Names["PLTools`"]} < 10, \\
\quad \text{Get["http://economicreasoning.com"]}]
\]

Notes

\[\n\nabla u[x], x, p, h, s, \epsilon H_i \text{ are each automatically recognized as vectors with length equal to the number of commodities. That number must not be less than one but is otherwise arbitrary.}
\]

In the Wolfram Language, \(x.y\) refers to the tensor DOT PRODUCT, NOT scalar multiplication. For TheoryGuru purposes, tensor means vector, so that the result of \(x.y\) is a scalar.

Setup

Consumer first-order conditions, as a vector equation

\[
\nabla u[x] = \lambda p;
\]
\[
nonsatiation = \{\lambda > 0, \text{income} > 0\};
\]

Differentiate it with respect to \(p_i\)

\[
dutility = \nabla u[x].h_i = 0;
\]
\[
(\ast \ h_i \text{ is a vector of impacts of } p_i \text{ on each Hicksian quantity choice } \ast)
\]
Definition of price elasticities

\[ \text{defineelas} = p.h_i = \text{income } s.eH_i; \]
(* \( eH_i \) is a vector of the (Hicksian) elasticity
of each quantity demanded with respect to \( p_i \) *)

Result: “Adding up”

TheoryGuru[{dutility, defineelas, nonsatiation}, s.eH_i = 0]
True

TheorySpace[]
Using MostRecentTheory.
{income, \( \lambda \), p.p, p.s, p.h_i, p.eH_i, s.s, s.h_i, s.eH_i, h_i.h_i, h_i.eH_i, eH_i.eH_i}
p, s, h_i, eH_i are interpreted as vectors.

Variable interpretations

Element-by-element notation with 4 goods