Econometric Estimation of the "Constant Elasticity of Substitution" Function in R: Package micEconCES

Arne Henningsen
Institute of Food and
Resource Economics
University of Copenhagen

Géraldine Henningsen[*]
Risø National Laboratory
for Sustainable Energy
Technical University of Denmark

Abstract

The Constant Elasticity of Substitution (CES) function is popular in several areas of economics, but it is rarely used in econometric analysis because it cannot be estimated by standard linear regression techniques. We discuss several existing approaches and propose a new grid-search approach for estimating the traditional CES function with two inputs as well as nested CES functions with three and four inputs. Furthermore, we demonstrate how these approaches can be applied in R using the add-on package **micEconCES** and we describe how the various estimation approaches are implemented in the **micEconCES** package. Finally, we illustrate the usage of this package by replicating some estimations of CES functions that are reported in the literature.

Keywords: constant elasticity of substitution, CES, nested CES, R.

1. Introduction

The so-called Cobb-Douglas function (Douglas and Cobb 1928) is the most widely used functional form in economics. However, it imposes strong assumptions on the underlying functional relationship, most notably that the elasticity of substitution[1] is always one. Given these restrictive assumptions, the Stanford group around Arrow, Chenery, Minhas, and Solow (1961) developed the Constant Elasticity of Substitution (CES) function as a generalisation of the Cobb-Douglas function that allows for any (non-negative constant) elasticity of substitution. This functional form has become very popular in programming models (e.g. general equilibrium models or trade models), but it has been rarely used in econometric analysis. Hence, the parameters of the CES functions used in programming models are mostly guess-

---

[*]Senior authorship is shared.

[1]For instance, in production economics, the elasticity of substitution measures the substitutability between inputs. It has non-negative values, where an elasticity of substitution of zero indicates that no substitution is possible (e.g. between wheels and frames in the production of bikes) and an elasticity of substitution of infinity indicates that the inputs are perfect substitutes (e.g. electricity from two different power plants).

timated and calibrated, rather than econometrically estimated. However, in recent years, the CES function has gained in importance also in econometric analyses, particularly in macroeconomics (e.g. Amras 2004; Bentolila and Gilles 2006) and growth theory (e.g. Caselli 2005; Caselli and Coleman 2006; Klump and Papageorgiou 2008), where it replaces the Cobb-Douglas function.[2] The CES functional form is also frequently used in micro-macro models, i.e. a new type of model that links microeconomic models of consumers and producers with an overall macroeconomic model (see for example Davies 2009). Given the increasing use of the CES function in econometric analysis and the importance of using sound parameters in economic programming models, there is definitely demand for software that facilitates the econometric estimation of the CES function.

The R package **micEconCES** (Henningsen and Henningsen 2011) provides this functionality. It is developed as part of the "micEcon" project on R-Forge (`http://r-forge.r-project.org/projects/micecon/`). Stable versions of this package are available for download from the Comprehensive R Archive Network (CRAN, `http://CRAN.R-Project.org/package=micEconCES`).

The paper is structured as follows. In the next section, we describe the classical CES function and the most important generalisations that can account for more than two independent variables. Then, we discuss several approaches to estimate these CES functions and show how they can be applied in R. The fourth section describes the implementation of these methods in the R package **micEconCES**, whilst the fifth section demonstrates the usage of this package by replicating estimations of CES functions that are reported in the literature. Finally, the last section concludes.

2. Specification of the CES function

The formal specification of a CES production function[3] with two inputs is

$$y = \gamma \left(\delta x_1^{-\rho} + (1 - \delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}}, \tag{1}$$

where y is the output quantity, x_1 and x_2 are the input quantities, and γ, δ, ρ, and ν are parameters. Parameter $\gamma \in (0, \infty)$ determines the productivity, $\delta \in (0, 1)$ determines the

---

[2]The *Journal of Macroeconomics* even published an entire special issue titled "The CES Production Function in the Theory and Empirics of Economic Growth" (Klump and Papageorgiou 2008).

[3]The CES functional form can be used to model different economic relationships (e.g. as production function, cost function, or utility function). However, as the CES functional form is mostly used to model production technologies, we name the dependent (left-hand side) variable "output" and the independent (right-hand side) variables "inputs" to keep the notation simple.

optimal distribution of the inputs, $\rho \in (-1, 0) \cup (0, \infty)$ determines the (constant) elasticity of substitution, which is $\sigma = 1/(1 + \rho)$, and $\nu \in (0, \infty)$ is equal to the elasticity of scale.[4]

The CES function includes three special cases: for $\rho \to 0$, σ approaches 1 and the CES turns to the Cobb-Douglas form; for $\rho \to \infty$, σ approaches 0 and the CES turns to the Leontief production function; and for $\rho \to -1$, σ approaches infinity and the CES turns to a linear function if ν is equal to 1.

As the CES function is non-linear in parameters and cannot be linearised analytically, it is not possible to estimate it with the usual linear estimation techniques. Therefore, the CES function is often approximated by the so-called "Kmenta approximation" (Kmenta 1967), which can be estimated by linear estimation techniques. Alternatively, it can be estimated by non-linear least-squares using different optimisation algorithms.

To overcome the limitation of two input factors, CES functions for multiple inputs have been proposed. One problem of the elasticity of substitution for models with more than two inputs is that the literature provides three popular, but different definitions (see e.g. Chambers 1988): While the Hicks-McFadden elasticity of substitution (also known as direct elasticity of substitution) describes the input substitutability of two inputs i and j along an isoquant given that all other inputs are constant, the Allen-Uzawa elasticity of substitution (also known as Allen partial elasticity of substitution) and the Morishima elasticity of substitution describe the input substitutability of two inputs when all other input quantities are allowed to adjust. The only functional form in which all three elasticities of substitution are constant is the plain n-input CES function (Blackorby and Russel 1989), which has the following specification:

$$y = \gamma \left(\sum_{i=1}^{n} \delta_i x_i^{-\rho} \right)^{-\frac{\nu}{\rho}} \tag{2}$$
$$\text{with } \sum_{i=1}^{n} \delta_i = 1,$$

where n is the number of inputs and $x_1, \ldots, x_n$ are the quantities of the n inputs. Several scholars have tried to extend the Kmenta approximation to the n-input case, but Hoff (2004) showed that a correctly specified extension to the n-input case requires non-linear parameter restrictions on a Translog function. Hence, there is little gain in using the Kmenta approximation in the n-input case.

The plain n-input CES function assumes that the elasticities of substitution between any two inputs are the same. As this is highly undesirable for empirical applications, multiple-input CES functions that allow for different (constant) elasticities of substitution between different pairs of inputs have been proposed. For instance, the functional form proposed by Uzawa

---

[4]Originally, the CES function of Arrow *et al.* (1961) could only model constant returns to scale, but later Kmenta (1967) added the parameter ν, which allows for decreasing or increasing returns to scale if $\nu < 1$ or $\nu > 1$, respectively.

(1962) has constant Allen-Uzawa elasticities of substitution and the functional form proposed by McFadden (1963) has constant Hicks-McFadden elasticities of substitution.

However, the n-input CES functions proposed by Uzawa (1962) and McFadden (1963) impose rather strict conditions on the values for the elasticities of substitution and thus, are less useful for empirical applications (Sato 1967, p. 202). Therefore, Sato (1967) proposed a family of two-level nested CES functions. The basic idea of nesting CES functions is to have two or more levels of CES functions, where each of the inputs of an upper-level CES function might be replaced by the dependent variable of a lower-level CES function. Particularly, the nested CES functions for three and four inputs based on Sato (1967) have become popular in recent years. These functions increased in popularity especially in the field of macro-econometrics, where input factors needed further differentiation, e.g. issues such as Grilliches' capital-skill complementarity (Griliches 1969) or wage differentiation between skilled and unskilled labour (e.g. Acemoglu 1998; Krusell, Ohanian, Ríos-Rull, and Violante 2000; Pandey 2008).

The nested CES function for four inputs as proposed by Sato (1967) nests two lower-level (two-input) CES functions into an upper-level (two-input) CES function: $y = \gamma[\delta\,CES_1 + (1-\delta)\,CES_2]^{-\nu/\rho}$, where $CES_i = \gamma_i \left(\delta_i x_{2i-1}^{-\rho_i} + (1-\delta_i)\,x_{2i}^{-\rho_i}\right)^{-\nu_i/\rho_i}$, $i = 1, 2$, indicates the two lower-level CES functions. In these lower-level CES functions, we (arbitrarily) normalise coefficients γ_i and ν_i to one, because without these normalisations, not all coefficients of the (entire) nested CES function can be identified in econometric estimations; an infinite number of vectors of non-normalised coefficients exists that all result in the same output quantity, given an arbitrary vector of input quantities (see footnote 6 for an example). Hence, the final specification of the four-input nested CES function is as follows:

$$y = \gamma \left[\delta \left(\delta_1 x_1^{-\rho_1} + (1-\delta_1)x_2^{-\rho_1}\right)^{\rho/\rho_1} + (1-\delta)\left(\delta_2 x_3^{-\rho_2} + (1-\delta_2)x_4^{-\rho_2}\right)^{\rho/\rho_2}\right]^{-\nu/\rho}. \quad (3)$$

If $\rho_1 = \rho_2 = \rho$, the four-input nested CES function defined in equation 3 reduces to the plain four-input CES function defined in equation 2.[5]

---

[5]In this case, the parameters of the four-input nested CES function defined in equation 3 (indicated by the superscript n) and the parameters of the plain four-input CES function defined in equation 2 (indicated by the superscript p) correspond in the following way: where $\rho^p = \rho_1^n = \rho_2^n = \rho^n$, $\delta_1^p = \delta_1^n\,\delta^n$, $\delta_2^p = (1-\delta_1^n)\,\delta^n$, $\delta_3^p = \delta_2^n\,(1-\delta^n)$, $\delta_4^p = (1-\delta_2^n)\,(1-\delta^n)$, $\gamma^p = \gamma^n$, $\delta_1^n = \delta_1^p/(\delta_1^p + \delta_2^p)$, $\delta_2^n = \delta_3^p/(\delta_3^p + \delta_4^p)$, and $\delta^n = \delta_1^p + \delta_2^p$.

In the case of the three-input nested CES function, only one input of the upper-level CES function is further differentiated:[6]

$$y = \gamma \left[\delta \left(\delta_1 x_1^{-\rho_1} + (1 - \delta_1) x_2^{-\rho_1} \right)^{\rho/\rho_1} + (1 - \delta) x_3^{-\rho} \right]^{-\nu/\rho}. \tag{4}$$

For instance, x_1 and x_2 could be skilled and unskilled labour, respectively, and x_3 capital. Alternatively, Kemfert (1998) used this specification for analysing the substitutability between capital, labour, and energy. If $\rho_1 = \rho$, the three-input nested CES function defined in equation 4 reduces to the plain three-input CES function defined in equation 2.[7]

The nesting of the CES function increases its flexibility and makes it an attractive choice for many applications in economic theory and empirical work. However, nested CES functions are not invariant to the nesting structure and different nesting structures imply different assumptions about the separability between inputs (Sato 1967). As the nesting structure is theoretically arbitrary, the selection depends on the researcher's choice and should be based on empirical considerations.

The formulas for calculating the Hicks-McFadden and Allen-Uzawa elasticities of substitution for the three-input and four-input nested CES functions are given in appendices B.3 and C.3, respectively. Anderson and Moroney (1994) showed for n-input nested CES functions that the Hicks-McFadden and Allen-Uzawa elasticities of substitution are only identical if the nested technologies are all of the Cobb-Douglas form, i.e. $\rho_1 = \rho_2 = \rho = 0$ in the four-input nested CES function and $\rho_1 = \rho = 0$ in the three-input nested CES function.

Like in the plain n-input case, nested CES functions cannot be easily linearised. Hence, they have to be estimated by applying non-linear optimisation methods. In the following section, we will present different approaches to estimate the classical two-input CES function as well as n-input nested CES functions using the R package **micEconCES**.

3. Estimation of the CES production function

---

[6] Papageorgiou and Saam (2005) proposed a specification that includes the additional term $\gamma_1^{-\rho}$:

$$y = \gamma \left[\delta \gamma_1^{-\rho} \left(\delta_1 x_1^{-\rho_1} + (1 - \delta_1) x_2^{-\rho_1} \right)^{\rho/\rho_1} + (1 - \delta) x_3^{-\rho} \right]^{-\nu/\rho}.$$

However, adding the term $\gamma_1^{-\rho}$ does not increase the flexibility of this function as γ_1 can be arbitrarily normalised to one; normalising γ_1 to one changes γ to $\gamma \left(\delta \gamma_1^{-\rho} + (1 - \delta) \right)^{-(\nu/\rho)}$ and changes δ to $\left(\delta \gamma_1^{-\rho} \right) \big/ \left(\delta \gamma_1^{-\rho} + (1 - \delta) \right)$, but has no effect on the functional form. Hence, the parameters γ, γ_1, and δ cannot be (jointly) identified in econometric estimations (see also explanation for the four-input nested CES function above equation (3)).

[7] In this case, the parameters of the three-input nested CES function defined in equation 4 (indicated by the superscript n) and the parameters of the plain three-input CES function defined in equation 2 (indicated by the superscript p) correspond in the following way: where $\rho^p = \rho_1^n = \rho^n$, $\delta_1^p = \delta_1^n \, \delta^n$, $\delta_2^p = (1 - \delta_1^n) \, \delta^n$, $\delta_3^p = 1 - \delta^n$, $\gamma^p = \gamma^n$, $\delta_1^n = \delta_1^p/(1 - \delta_3^p)$, and $\delta^n = 1 - \delta_3^p$.

Tools for economic analysis with CES function are available in the R package **micEconCES** (Henningsen and Henningsen 2011). If this package is installed, it can be loaded with the command

```
> library("micEconCES")
```

We demonstrate the usage of this package by estimating a classical two-input CES function as well as nested CES functions with three and four inputs. For this, we use an artificial data set cesData, because this avoids several problems that usually occur with real-world data.

```
> set.seed(123)
> cesData <- data.frame(x1 = rchisq(200, 10), x2 = rchisq(200,
+     10), x3 = rchisq(200, 10), x4 = rchisq(200, 10))
> cesData$y2 <- cesCalc(xNames = c("x1", "x2"), data = cesData,
+     coef = c(gamma = 1, delta = 0.6, rho = 0.5, nu = 1.1))
> cesData$y2 <- cesData$y2 + 2.5 * rnorm(200)
> cesData$y3 <- cesCalc(xNames = c("x1", "x2", "x3"), data = cesData,
+     coef = c(gamma = 1, delta_1 = 0.7, delta = 0.6, rho_1 = 0.3,
+         rho = 0.5, nu = 1.1), nested = TRUE)
> cesData$y3 <- cesData$y3 + 1.5 * rnorm(200)
> cesData$y4 <- cesCalc(xNames = c("x1", "x2", "x3", "x4"), data = cesData,
+     coef = c(gamma = 1, delta_1 = 0.7, delta_2 = 0.6, delta = 0.5,
+         rho_1 = 0.3, rho_2 = 0.4, rho = 0.5, nu = 1.1), nested = TRUE)
> cesData$y4 <- cesData$y4 + 1.5 * rnorm(200)
```

The first line sets the "seed" for the random number generator so that these examples can be replicated with exactly the same data set. The second line creates a data set with four input variables (called x1, x2, x3, and x4) that each have 200 observations and are generated from random χ^2 distributions with 10 degrees of freedom. The third, fifth, and seventh commands use the function cesCalc, which is included in the **micEconCES** package, to calculate the deterministic output variables for the CES functions with two, three, and four inputs (called y2, y3, and y4, respectively) given a CES production function. For the two-input CES function, we use the coefficients $\gamma = 1$, $\delta = 0.6$, $\rho = 0.5$, and $\nu = 1.1$; for the three-input nested CES function, we use $\gamma = 1$, $\delta_1 = 0.7$, $\delta = 0.6$, $\rho_1 = 0.3$, $\rho = 0.5$, and $\nu = 1.1$; and for the four-input nested CES function, we use $\gamma = 1$, $\delta_1 = 0.7$, $\delta_2 = 0.6$, $\delta = 0.5$, $\rho_1 = 0.3$, $\rho_2 = 0.4$, $\rho = 0.5$, and $\nu = 1.1$. The fourth, sixth, and eighth commands generate the stochastic output variables by adding normally distributed random errors to the deterministic output variable.

As the CES function is non-linear in its parameters, the most straightforward way to estimate the CES function in R would be to use nls, which performs non-linear least-squares estimations.

```
> cesNls <- nls(y2 ~ gamma * (delta * x1^(-rho) + (1 - delta) *
+     x2^(-rho))^(-phi/rho), data = cesData, start = c(gamma = 0.5,
+     delta = 0.5, rho = 0.25, phi = 1))
> print(cesNls)

Nonlinear regression model
  model:  y2 ~ gamma * (delta * x1^(-rho) + (1 - delta) * x2^(-rho))^(-phi/rho)
   data:  cesData
 gamma  delta    rho    phi
1.0239 0.6222 0.5420 1.0858
 residual sum-of-squares: 1197


Number of iterations to convergence: 6
Achieved convergence tolerance: 8.17e-06
```

While the `nls` routine works well in this ideal artificial example, it does not perform well in many applications with real data, either because of non-convergence, convergence to a local minimum, or theoretically unreasonable parameter estimates. Therefore, we show alternative ways of estimating the CES function in the following sections.

3.1. Kmenta approximation

Given that non-linear estimation methods are often troublesome—particularly during the 1960s and 1970s when computing power was very limited—Kmenta (1967) derived an approximation of the classical two-input CES production function that could be estimated by ordinary least-squares techniques.

$$\ln y = \ln \gamma + \nu\, \delta \ln x_1 + \nu\, (1 - \delta) \ln x_2 \tag{5}$$
$$- \frac{\rho\, \nu}{2}\, \delta\, (1 - \delta)\, (\ln x_1 - \ln x_2)^2$$

While Kmenta (1967) obtained this formula by logarithmising the CES function and applying a second-order Taylor series expansion to $\ln \left(\delta x_1^{-\rho} + (1 - \delta) x_2^{-\rho} \right)$ at the point $\rho = 0$, the same formula can be obtained by applying a first-order Taylor series expansion to the entire logarithmised CES function at the point $\rho = 0$ (Uebe 2000). As the authors consider the latter approach to be more straight-forward, the Kmenta approximation is called—in contrast to Kmenta (1967, p. 180)—first-order Taylor series expansion in the remainder of this paper.

The Kmenta approximation can also be written as a restricted translog function (Hoff 2004):

$$\ln y = \alpha_0 + \alpha_1 \ln x_1 + \alpha_2 \ln x_2 \tag{6}$$
$$+ \frac{1}{2}\, \beta_{11}\, (\ln x_1)^2 + \frac{1}{2}\, \beta_{22}\, (\ln x_2)^2 + \beta_{12}\, \ln x_1 \ln x_2,$$

where the two restrictions are

$$\beta_{12} = -\beta_{11} = -\beta_{22}. \tag{7}$$

If constant returns to scale are to be imposed, a third restriction

$$\alpha_1 + \alpha_2 = 1 \tag{8}$$

must be enforced. These restrictions can be utilised to test whether the linear Kmenta approximation of the CES function (5) is an acceptable simplification of the translog functional form.[8] If this is the case, a simple *t*-test for the coefficient $\beta_{12} = -\beta_{11} = -\beta_{22}$ can be used to check if the Cobb-Douglas functional form is an acceptable simplification of the Kmenta approximation of the CES function.[9]

The parameters of the CES function can be calculated from the parameters of the restricted translog function by:

$$\gamma = \exp(\alpha_0) \tag{9}$$

$$\nu = \alpha_1 + \alpha_2 \tag{10}$$

$$\delta = \frac{\alpha_1}{\alpha_1 + \alpha_2} \tag{11}$$

$$\rho = \frac{\beta_{12}(\alpha_1 + \alpha_2)}{\alpha_1 \cdot \alpha_2} \tag{12}$$

The Kmenta approximation of the CES function can be estimated by the function `cesEst`, which is included in the **micEconCES** package. If argument `method` of this function is set to `"Kmenta"`, it (a) estimates an unrestricted translog function (6), (b) carries out a Wald test of the parameter restrictions defined in equation (7) and eventually also in equation (8) using the (finite sample) *F*-statistic, (c) estimates the restricted translog function (6, 7), and finally, (d) calculates the parameters of the CES function using equations (9−12) as well as their covariance matrix using the delta method.

The following code estimates a CES function with the dependent variable `y2` (specified in argument `yName`) and the two explanatory variables `x1` and `x2` (argument `xNames`), all taken from the artificial data set `cesData` that we generated above (argument `data`) using the Kmenta approximation (argument `method`) and allowing for variable returns to scale (argument `vrs`).

```
> cesKmenta <- cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
+      method = "Kmenta", vrs = TRUE)
```

---

[8]Note that this test does *not* check whether the *non-linear* CES function (1) is an acceptable simplification of the translog functional form, or whether the *non-linear* CES function can be approximated by the Kmenta approximation.

[9]Note that this test does *not* compare the Cobb-Douglas function with the (non-linear) CES function, but only with its linear approximation.

Summary results can be obtained by applying the `summary` method to the returned object.

```
> summary(cesKmenta)
```

```
Estimated CES function with variable returns to scale
```

```
Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, method = "Kmenta")
```

```
Estimation by the linear Kmenta approximation
Test of the null hypothesis that the restrictions of the Translog
function required by the Kmenta approximation are true:
P-value = 0.2269042
```

```
Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  0.89834    0.14738   6.095 1.09e-09 ***
delta  0.68126    0.04029  16.910  < 2e-16 ***
rho    0.86321    0.41286   2.091   0.0365 *
nu     1.13442    0.07308  15.523  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 2.498807
Multiple R-squared: 0.7548401
```

```
Elasticity of Substitution:
           Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.5367     0.1189   4.513 6.39e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The Wald test indicates that the restrictions on the Translog function implied by the Kmenta approximation cannot be rejected at any reasonable significance level.

To see whether the underlying technology is of the Cobb-Douglas form, we can check if the coefficient $\beta_{12} = -\beta_{11} = -\beta_{22}$ significantly differs from zero. As the estimation of the Kmenta approximation is stored in component `kmenta` of the object returned by `cesEst`, we can obtain summary information on the estimated coefficients of the Kmenta approximation by:

```
> coef(summary(cesKmenta$kmenta))
```

```
                    Estimate Std. Error     t value      Pr(>|t|)
eq1_(Intercept) -0.1072116 0.16406442 -0.6534723 5.142216e-01
eq1_a_1            0.7728315 0.05785460 13.3581693 0.000000e+00
eq1_a_2            0.3615885 0.05658941  6.3896839 1.195467e-09
eq1_b_1_1         -0.2126387 0.09627881 -2.2085723 2.836951e-02
eq1_b_1_2          0.2126387 0.09627881  2.2085723 2.836951e-02
eq1_b_2_2         -0.2126387 0.09627881 -2.2085723 2.836951e-02
```

Given that $\beta_{12} = -\beta_{11} = -\beta_{22}$ significantly differs from zero at the 5% level, we can conclude that the underlying technology is not of the Cobb-Douglas form. Alternatively, we can check if the parameter ρ of the CES function, which is calculated from the coefficients of the Kmenta approximation, significantly differs from zero. This should—as in our case—deliver similar results (see above).

Finally, we plot the fitted values against the actual dependent variable (y) to check whether the parameter estimates are reasonable.

```
> compPlot(cesData$y2, fitted(cesKmenta), xlab = "actual values",
+     ylab = "fitted values")
```

Figure 1 shows that the parameters produce reasonable fitted values.
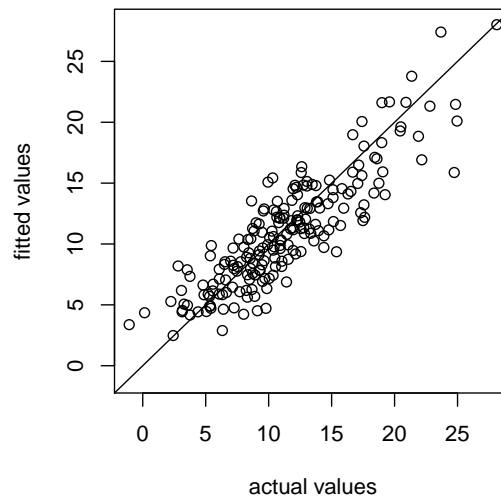

Figure 1: Fitted values from the Kmenta approximation against `y`

However, the Kmenta approximation encounters several problems. First, it is a truncated Taylor series and the remainder term must be seen as an omitted variable. Second, the Kmenta

approximation only converges to the underlying CES function in a region of convergence that is dependent on the true parameters of the CES function (Thursby and Lovell 1978).

Although, Maddala and Kadane (1967) and Thursby and Lovell (1978) find estimates for ν and δ with small bias and mean squared error (MSE), results for γ and ρ are estimated with generally considerable bias and MSE (Thursby and Lovell 1978; Thursby 1980). More reliable results can only be obtained if $\rho \to 0$, and thus, $\sigma \to 1$ which increases the convergence region, i.e. if the underlying CES function is of the Cobb-Douglas form. This is a major drawback of the Kmenta approximation as its purpose is to facilitate the estimation of functions with non-unitary σ.

3.2. Gradient-based optimisation algorithms

Levenberg-Marquardt

Initially, the Levenberg-Marquardt algorithm (Marquardt 1963) was most commonly used for estimating the parameters of the CES function by non-linear least-squares. This iterative algorithm can be seen as a maximum neighbourhood method which performs an optimum interpolation between a first-order Taylor series approximation (Gauss-Newton method) and a steepest-descend method (gradient method) (Marquardt 1963). By combining these two non-linear optimisation algorithms, the developers want to increase the convergence probability by reducing the weaknesses of each of the two methods.

In a Monte Carlo study by Thursby (1980), the Levenberg-Marquardt algorithm outperforms the other methods and gives the best estimates of the CES parameters. However, the Levenberg-Marquardt algorithm performs as poorly as the other methods in estimating the elasticity of substitution (σ), which means that the estimated σ tends to be biased towards infinity, unity, or zero.

Although the Levenberg-Marquardt algorithm does not live up to modern standards, we include it for reasons of completeness, as it is has proven to be a standard method for estimating CES functions.

To estimate a CES function by non-linear least-squares using the Levenberg-Marquardt algorithm, one can call the `cesEst` function with argument `method` set to `"LM"` or without this argument, as the Levenberg-Marquardt algorithm is the default estimation method used by `cesEst`. The user can modify a few details of this algorithm (e.g. different criteria for convergence) by adding argument `control` as described in the documentation of the R function `nls.lm.control`. Argument `start` can be used to specify a vector of starting values, where the order must be γ, δ_1, δ_2, δ, ρ_1, ρ_2, ρ, and ν (of course, all coefficients that are not in the model must be omitted). If no starting values are provided, they are determined automatically (see section 4.7). For demonstrative purposes, we estimate all three (i.e. two-input,

three-input nested, and four-input nested) CES functions with the Levenberg-Marquardt algorithm, but in order to reduce space, we will proceed with examples of the classical two-input CES function only.

```
> cesLm2 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE)
> summary(cesLm2)


Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE)


Estimation by non-linear least-squares using the 'LM' optimizer
assuming an additive error term
Convergence achieved after 4 iterations
Message: Relative error in the sum of squares is at most `ftol'.

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  1.02385    0.11562   8.855   <2e-16 ***
delta  0.62220    0.02845  21.873   <2e-16 ***
rho    0.54192    0.29090   1.863   0.0625 .
nu     1.08582    0.04569  23.765   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.446577
Multiple R-squared: 0.7649817

Elasticity of Substitution:
           Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6485     0.1224     5.3 1.16e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


> cesLm3 <- cesEst("y3", c("x1", "x2", "x3"), cesData, vrs = TRUE)
> summary(cesLm3)
```

```
Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y3", xNames = c("x1", "x2", "x3"), data = cesData,
    vrs = TRUE)


Estimation by non-linear least-squares using the 'LM' optimizer
assuming an additive error term
Convergence achieved after 5 iterations
Message: Relative error in the sum of squares is at most `ftol'.

Coefficients:
        Estimate Std. Error t value Pr(>|t|)
gamma    0.94558    0.08279  11.421  < 2e-16 ***
delta_1  0.65861    0.02439  27.000  < 2e-16 ***
delta    0.60715    0.01456  41.691  < 2e-16 ***
rho_1    0.18799    0.26503   0.709 0.478132
rho      0.53071    0.15079   3.519 0.000432 ***
nu       1.12636    0.03683  30.582  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 1.409937
Multiple R-squared: 0.8531556


Elasticities of Substitution:
              Estimate Std. Error t value Pr(>|t|)
E_1_2 (HM)     0.84176    0.18779   4.483 7.38e-06 ***
E_(1,2)_3 (AU) 0.65329    0.06436  10.151  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
HM = Hicks-McFadden (direct) elasticity of substitution
AU = Allen-Uzawa (partial) elasticity of substitution




> cesLm4 <- cesEst("y4", c("x1", "x2", "x3", "x4"), cesData, vrs = TRUE)
> summary(cesLm4)
```

```
Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y4", xNames = c("x1", "x2", "x3", "x4"), data = cesData,
    vrs = TRUE)


Estimation by non-linear least-squares using the 'LM' optimizer
assuming an additive error term
Convergence achieved after 8 iterations
Message: Relative error in the sum of squares is at most `ftol'.


Coefficients:
        Estimate Std. Error t value Pr(>|t|)
gamma    1.22760    0.12515   9.809  < 2e-16 ***
delta_1  0.78093    0.03442  22.691  < 2e-16 ***
delta_2  0.60090    0.02530  23.753  < 2e-16 ***
delta    0.51154    0.02086  24.518  < 2e-16 ***
rho_1    0.37788    0.46295   0.816 0.414361
rho_2    0.33380    0.22616   1.476 0.139967
rho      0.91065    0.25115   3.626 0.000288 ***
nu       1.01872    0.04355  23.390  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 1.424439
Multiple R-squared: 0.7890757


Elasticities of Substitution:
                  Estimate Std. Error t value Pr(>|t|)
E_1_2 (HM)          0.7258     0.2438   2.976  0.00292 **
E_3_4 (HM)          0.7497     0.1271   5.898 3.69e-09 ***
E_(1,2)_(3,4) (AU)  0.5234     0.0688   7.608 2.79e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
HM = Hicks-McFadden (direct) elasticity of substitution
AU = Allen-Uzawa (partial) elasticity of substitution
```

Finally, we plot the fitted values against the actual values y to see whether the estimated parameters are reasonable. The results are presented in figure 2.

```
> compPlot(cesData$y2, fitted(cesLm2), xlab = "actual values",
+     ylab = "fitted values", main = "two-input CES")
> compPlot(cesData$y3, fitted(cesLm3), xlab = "actual values",
+     ylab = "fitted values", main = "three-input nested CES")
> compPlot(cesData$y4, fitted(cesLm4), xlab = "actual values",
+     ylab = "fitted values", main = "four-input nested CES")
```
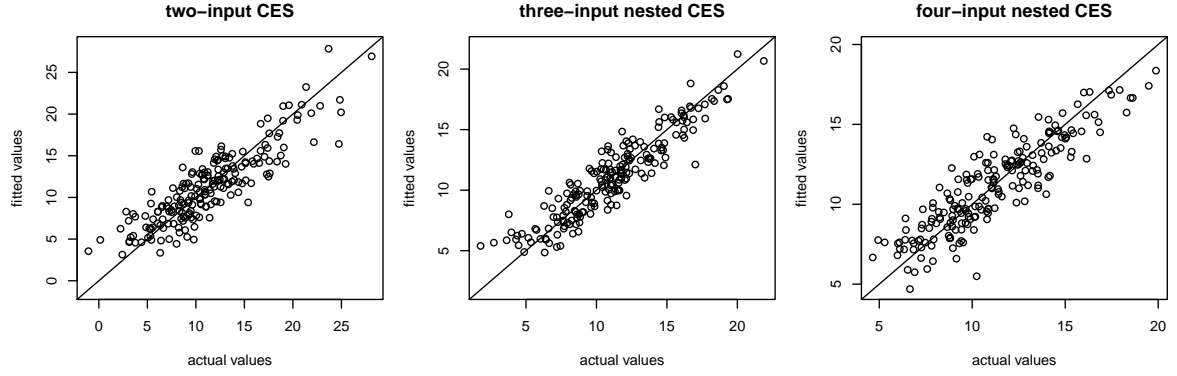

Figure 2: Fitted values from the LM algorithm against actual values

Several further gradient-based optimisation algorithms that are suitable for non-linear least-squares estimations are implemented in R. Function `cesEst` can use some of them to estimate a CES function by non-linear least-squares. As a proper application of these estimation methods requires the user to be familiar with the main characteristics of the different algorithms, we will briefly discuss some practical issues of the algorithms that will be used to estimate the CES function. However, it is not the aim of this paper to thoroughly discuss these algorithms. A detailed discussion of iterative optimisation algorithms is available, e.g. in Kelley (1999) or Mishra (2007).

Conjugate Gradients

One of the gradient-based optimisation algorithms that can be used by `cesEst` is the "Conjugate Gradients" method based on Fletcher and Reeves (1964). This iterative method is mostly applied to optimisation problems with many parameters and a large and possibly sparse Hessian matrix, because this algorithm does not require that the Hessian matrix is stored or inverted. The "Conjugated Gradient" method works best for objective functions that are approximately quadratic and it is sensitive to objective functions that are not well-behaved and have a non-positive semi-definite Hessian, i.e. convergence within the given number of iterations is less likely the more the level surface of the objective function differs from spherical (Kelley 1999). Given that the CES function has only few parameters and the objective function is not approximately quadratic and shows a tendency to "flat surfaces" around the

minimum, the "Conjugated Gradient" method is probably less suitable than other algorithms for estimating a CES function. Setting argument `method` of `cesEst` to `"CG"` selects the "Conjugate Gradients" method for estimating the CES function by non-linear least-squares. The user can modify this algorithm (e.g. replacing the update formula of Fletcher and Reeves (1964) by the formula of Polak and Ribière (1969) or the one based on Sorenson (1969) and Beale (1972)) or some other details (e.g. convergence tolerance level) by adding a further argument `control` as described in the "Details" section of the documentation of the R function `optim`.

```
> cesCg <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "CG")
> summary(cesCg)


Estimated CES function with variable returns to scale


Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, method = "CG")


Estimation by non-linear least-squares using the 'CG' optimizer
assuming an additive error term
Convergence NOT achieved after 401 function and 101 gradient calls


Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  1.03534    0.11680   8.865   <2e-16 ***
delta  0.62077    0.02827  21.956   <2e-16 ***
rho    0.48693    0.28518   1.707   0.0877 .
nu     1.08060    0.04567  23.664   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 2.446998
Multiple R-squared: 0.7649009


Elasticity of Substitution:
            Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6725     0.1290   5.214 1.85e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Although the estimated parameters are similar to the estimates from the Levenberg-Marquardt algorithm, the "Conjugated Gradient" algorithm reports that it did not converge. Increasing the maximum number of iterations and the tolerance level leads to convergence. This confirms a slow convergence of the "Conjugate Gradients" algorithm for estimating the CES function.

```
> cesCg2 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "CG",
+     control = list(maxit = 1000, reltol = 1e-05))
> summary(cesCg2)
```

```
Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, method = "CG", control = list(maxit = 1000, reltol = 1e-05))

Estimation by non-linear least-squares using the 'CG' optimizer
assuming an additive error term
Convergence achieved after 1874 function and 467 gradient calls

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  1.02385    0.11562   8.855  <2e-16 ***
delta  0.62220    0.02845  21.873  <2e-16 ***
rho    0.54192    0.29091   1.863  0.0625 .
nu     1.08582    0.04569  23.765  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.446577
Multiple R-squared: 0.7649817

Elasticity of Substitution:
            Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6485     0.1224     5.3 1.16e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Newton

Another algorithm supported by `cesEst` that is probably more suitable for estimating a CES function is an improved Newton-type method. As with the original Newton method, this algorithm uses first and second derivatives of the objective function to determine the direction of the shift vector and searches for a stationary point until the gradients are (almost) zero. However, in contrast to the original Newton method, this algorithm does a line search at each iteration to determine the optimal length of the shift vector (step size) as described in Dennis and Schnabel (1983) and Schnabel, Koontz, and Weiss (1985). Setting argument `method` of `cesEst` to `"Newton"` selects this improved Newton-type method. The user can modify a few details of this algorithm (e.g. the maximum step length) by adding further arguments that are described in the documentation of the R function `nlm`. The following commands estimate a CES function by non-linear least-squares using this algorithm and print summary results.

```
> cesNewton <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "Newton")
> summary(cesNewton)

Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, method = "Newton")

Estimation by non-linear least-squares using the 'Newton' optimizer
assuming an additive error term
Convergence achieved after 25 iterations

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  1.02385    0.11562   8.855   <2e-16 ***
delta  0.62220    0.02845  21.873   <2e-16 ***
rho    0.54192    0.29091   1.863   0.0625 .
nu     1.08582    0.04569  23.765   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.446577
Multiple R-squared: 0.7649817
```

```
Elasticity of Substitution:
            Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6485      0.1224      5.3 1.16e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Broyden-Fletcher-Goldfarb-Shanno

Furthermore, a quasi-Newton method developed independently by Broyden (1970), Fletcher (1970), Goldfarb (1970), and Shanno (1970) can be used by `cesEst`. This so-called BFGS algorithm also uses first and second derivatives and searches for a stationary point of the objective function where the gradients are (almost) zero. In contrast to the original Newton method, the BFGS method does a line search for the best step size and uses a special procedure to approximate and update the Hessian matrix in every iteration. The problem with BFGS can be that although the current parameters are close to the minimum, the algorithm does not converge because the Hessian matrix at the current parameters is not close to the Hessian matrix at the minimum. However, in practice, BFGS proves robust convergence (often superlinear) (Kelley 1999). If argument `method` of `cesEst` is `"BFGS"`, the BFGS algorithm is used for the estimation. The user can modify a few details of the BFGS algorithm (e.g. the convergence tolerance level) by adding the further argument `control` as described in the "Details" section of the documentation of the R function `optim`.

```
> cesBfgs <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "BFGS")
> summary(cesBfgs)


Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, method = "BFGS")


Estimation by non-linear least-squares using the 'BFGS' optimizer
assuming an additive error term
Convergence achieved after 73 function and 15 gradient calls


Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  1.02385    0.11562   8.855   <2e-16 ***
delta  0.62220    0.02845  21.873   <2e-16 ***
```

```
rho    0.54192    0.29091   1.863    0.0625 .
nu     1.08582    0.04569  23.765    <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 2.446577
Multiple R-squared: 0.7649817


Elasticity of Substitution:
           Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6485     0.1224     5.3 1.16e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3.3. Global optimisation algorithms

Nelder-Mead

While the gradient-based (local) optimisation algorithms described above are designed to find local minima, global optimisation algorithms, which are also known as direct search methods, are designed to find the global minimum. These algorithms are more tolerant to objective functions which are not well-behaved, although they usually converge more slowly than the gradient-based methods. However, increasing computing power has made these algorithms suitable for day-to-day use.

One of these global optimisation routines is the so-called Nelder-Mead algorithm (Nelder and Mead 1965), which is a downhill simplex algorithm. In every iteration, $n + 1$ vertices are defined in the n-dimensional parameter space. The algorithm converges by successively replacing the "worst" point by a new vertex in the multi-dimensional parameter space. The Nelder-Mead algorithm has the advantage of a simple and robust algorithm, and is especially suitable for residual problems with non-differentiable objective functions. However, the heuristic nature of the algorithm causes slow convergence, especially close to the minimum, and can lead to convergence to non-stationary points. As the CES function is easily twice differentiable, the advantage of the Nelder-Mead algorithm simply becomes its robustness. As a consequence of the heuristic optimisation technique, the results should be handled with care. However, the Nelder-Mead algorithm is much faster than the other global optimisation algorithms described below. Function `cesEst` estimates a CES function with the Nelder-Mead algorithm if argument `method` is set to `"NM"`. The user can tweak this algorithm (e.g. the reflection factor, contraction factor, or expansion factor) or change some other details

(e.g. convergence tolerance level) by adding a further argument `control` as described in the "Details" section of the documentation of the R function `optim`.

```
> cesNm <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "NM")
> summary(cesNm)


Estimated CES function with variable returns to scale


Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, method = "NM")


Estimation by non-linear least-squares using the 'Nelder-Mead' optimizer
assuming an additive error term
Convergence achieved after 265 iterations


Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  1.02399    0.11564   8.855   <2e-16 ***
delta  0.62224    0.02845  21.872   <2e-16 ***
rho    0.54212    0.29095   1.863   0.0624 .
nu     1.08576    0.04569  23.763   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 2.446577
Multiple R-squared: 0.7649817


Elasticity of Substitution:
           Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6485     0.1223     5.3 1.16e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Simulated Annealing

The Simulated Annealing algorithm was initially proposed by Kirkpatrick, Gelatt, and Vecchi (1983) and Cerny (1985) and is a modification of the Metropolis-Hastings algorithm. Every iteration chooses a random solution close to the current solution, while the probability of the

choice is driven by a global parameter T which decreases as the algorithm moves on. Unlike other iterative optimisation algorithms, Simulated Annealing also allows T to increase which makes it possible to leave local minima. Therefore, Simulated Annealing is a robust global optimiser and it can be applied to a large search space, where it provides fast and reliable solutions. Setting argument `method` to `"SANN"` selects a variant of the "Simulated Annealing" algorithm given in Bélisle (1992). The user can modify some details of the "Simulated Annealing" algorithm (e.g. the starting temperature T or the number of function evaluations at each temperature) by adding a further argument `control` as described in the "Details" section of the documentation of the R function `optim`. The only criterion for stopping this iterative process is the number of iterations and it does not indicate whether the algorithm converged or not.

```
> cesSann <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "SANN")
> summary(cesSann)


Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, method = "SANN")


Estimation by non-linear least-squares using the 'SANN' optimizer
assuming an additive error term


Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  1.01104    0.11477   8.809   <2e-16 ***
delta  0.63414    0.02954  21.469   <2e-16 ***
rho    0.71252    0.31440   2.266   0.0234 *
nu     1.09179    0.04590  23.784   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 2.449907
Multiple R-squared: 0.7643416


Elasticity of Substitution:
            Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.5839     0.1072   5.447 5.12e-08 ***
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As the Simulated Annealing algorithm makes use of random numbers, the solution generally depends on the initial "state" of R's random number generator. To ensure replicability, `cesEst` "seeds" the random number generator before it starts the "Simulated Annealing" algorithm with the value of argument `random.seed`, which defaults to 123. Hence, the estimation of the same model using this algorithm always returns the same estimates as long as argument `random.seed` is not altered (at least using the same software and hardware components).

```
> cesSann2 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "SANN")
> all.equal(cesSann, cesSann2)
```

```
[1] TRUE
```

It is recommended to start this algorithm with different values of argument `random.seed` and to check whether the estimates differ considerably.

```
> cesSann3 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "SANN", random.seed = 1234)
> cesSann4 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "SANN", random.seed = 12345)
> cesSann5 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "SANN", random.seed = 123456)
> m <- rbind(cesSann = coef(cesSann), cesSann3 = coef(cesSann3),
+     cesSann4 = coef(cesSann4), cesSann5 = coef(cesSann5))
> rbind(m, stdDev = sd(m))
```

```
              gamma       delta       rho          nu
cesSann   1.011041588 0.63413533 0.7125172 1.091787653
cesSann3  1.020815431 0.62383022 0.4716324 1.082790909
cesSann4  1.022048135 0.63815451 0.5632106 1.086868475
cesSann5  1.010198459 0.61496285 0.5284805 1.093646831
stdDev    0.006271878 0.01045467 0.1028802 0.004907647
```

If the estimates differ remarkably, the user can try increasing the number of iterations, which is 10,000 by default. Now we will re-estimate the model a few times with 100,000 iterations each.

```
> cesSannB <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "SANN", control = list(maxit = 1e+05))
> cesSannB3 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "SANN", random.seed = 1234, control = list(maxit = 1e+05))
> cesSannB4 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "SANN", random.seed = 12345, control = list(maxit = 1e+05))
> cesSannB5 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "SANN", random.seed = 123456, control = list(maxit = 1e+05))
> m <- rbind(cesSannB = coef(cesSannB), cesSannB3 = coef(cesSannB3),
+     cesSannB4 = coef(cesSannB4), cesSannB5 = coef(cesSannB5))
> rbind(m, stdDev = sd(m))
```

	gamma	delta	rho	nu
cesSannB	1.033763293	0.62601393	0.56234286	1.082088383
cesSannB3	1.038618559	0.62066224	0.57456297	1.079761772
cesSannB4	1.034458497	0.62048736	0.57590347	1.081348376
cesSannB5	1.023286165	0.62252710	0.52591584	1.086867351
stdDev	0.006525824	0.00256597	0.02332247	0.003058661

Now the estimates are much more similar—only the estimates of ρ still differ somewhat.

Differential Evolution

In contrary to the other algorithms described in this paper, the Differential Evolution algorithm (Storn and Price 1997; Price, Storn, and Lampinen 2006) belongs to the class of evolution strategy optimisers and convergence cannot be proven analytically. However, the algorithm has proven to be effective and accurate on a large range of optimisation problems, inter alia the CES function (Mishra 2007). For some problems, it has proven to be more accurate and more efficient than Simulated Annealing, Quasi-Newton, or other genetic algorithms (Storn and Price 1997; Ali and Törn 2004; Mishra 2007). Function `cesEst` uses a Differential Evolution optimiser for the non-linear least-squares estimation of the CES function, if argument `method` is set to `"DE"`. The user can modify the Differential Evolution algorithm (e.g. the differential evolution strategy or selection method) or change some details (e.g. the number of population members) by adding a further argument `control` as described in the documentation of the R function `DEoptim.control`. In contrary to the other optimisation algorithms, the Differential Evolution method requires finite boundaries for the parameters. By default, the bounds are $0 \leq \gamma \leq 10^{10}$; $0 \leq \delta_1, \delta_2, \delta \leq 1$; $-1 \leq \rho_1, \rho_2, \rho \leq 10$; and $0 \leq \nu \leq 10$. Of course, the user can specify own lower and upper bounds by setting arguments `lower` and `upper` to numeric vectors.

```
> cesDe <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "DE",
+       control = list(trace = FALSE))
> summary(cesDe)


Estimated CES function with variable returns to scale


Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
     vrs = TRUE, method = "DE", control = list(trace = FALSE))


Estimation by non-linear least-squares using the 'DE' optimizer
assuming an additive error term


Coefficients:
       Estimate Std. Error t value Pr(>|t|)
gamma  1.01905    0.11510   8.854   <2e-16 ***
delta  0.62368    0.02835  21.999   <2e-16 ***
rho    0.52300    0.28832   1.814   0.0697 .
nu     1.08753    0.04570  23.799   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 2.446653
Multiple R-squared: 0.7649671


Elasticity of Substitution:
           Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6566     0.1243   5.282 1.28e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Like the "Simulated Annealing" algorithm, the Differential Evolution algorithm makes use of random numbers and `cesEst` "seeds" the random number generator with the value of argument `random.seed` before it starts this algorithm to ensure replicability.

```
> cesDe2 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "DE",
+       control = list(trace = FALSE))
> all.equal(cesDe, cesDe2)


[1] TRUE
```

When using this algorithm, it is also recommended to check whether different values of argument `random.seed` result in considerably different estimates.

```
> cesDe3 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "DE",
+     random.seed = 1234, control = list(trace = FALSE))
> cesDe4 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "DE",
+     random.seed = 12345, control = list(trace = FALSE))
> cesDe5 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "DE",
+     random.seed = 123456, control = list(trace = FALSE))
> m <- rbind(cesDe = coef(cesDe), cesDe3 = coef(cesDe3), cesDe4 = coef(cesDe4),
+     cesDe5 = coef(cesDe5))
> rbind(m, stdDev = sd(m))
```

	gamma	delta	rho	nu
cesDe	1.01905185	0.623675438	0.5230009	1.08753121
cesDe3	1.04953088	0.620315935	0.5445222	1.07557854
cesDe4	1.01957930	0.621812211	0.5526993	1.08766039
cesDe5	1.02662390	0.623304969	0.5762994	1.08555290
stdDev	0.01431211	0.001535595	0.0220218	0.00574962

These estimates are rather similar, which generally indicates that all estimates are close to the optimum (minimum of the sum of squared residuals). However, if the user wants to obtain more precise estimates than those derived from the default settings of this algorithm, e.g. if the estimates differ considerably, the user can try to increase the maximum number of population generations (iterations) using control parameter `itermax`, which is 200 by default. Now we will re-estimate this model a few times with 1,000 population generations each.

```
> cesDeB <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "DE",
+     control = list(trace = FALSE, itermax = 1000))
> cesDeB3 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "DE",
+     random.seed = 1234, control = list(trace = FALSE, itermax = 1000))
> cesDeB4 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "DE",
+     random.seed = 12345, control = list(trace = FALSE, itermax = 1000))
> cesDeB5 <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "DE",
+     random.seed = 123456, control = list(trace = FALSE, itermax = 1000))
> rbind(cesDeB = coef(cesDeB), cesDeB3 = coef(cesDeB3), cesDeB4 = coef(cesDeB4),
+     cesDeB5 = coef(cesDeB5))
```

	gamma	delta	rho	nu
cesDeB	1.023852	0.6221982	0.5419226	1.08582

```
cesDeB3 1.023853 0.6221982 0.5419226 1.08582
cesDeB4 1.023852 0.6221982 0.5419226 1.08582
cesDeB5 1.023852 0.6221982 0.5419226 1.08582
```

The estimates are now virtually identical.

The user can further increase the likelihood of finding the global optimum by increasing the number of population members using control parameter NP, which is 10 times the number of parameters by default and should not have a smaller value than this default value (see documentation of the R function DEoptim.control).

3.4. Constraint parameters

As a meaningful analysis based on a CES function requires that the function is consistent with economic theory, it is often desirable to constrain the parameter space to the economically meaningful region. This can be done by the Differential Evolution (DE) algorithm as described above. Moreover, function **cesEst** can use two gradient-based optimisation algorithms for estimating a CES function under parameter constraints.

L-BFGS-B

One of these methods is a modification of the BFGS algorithm suggested by Byrd, Lu, Nocedal, and Zhu (1995). In contrast to the ordinary BFGS algorithm summarised above, the so-called L-BFGS-B algorithm allows for box-constraints on the parameters and also does not explicitly form or store the Hessian matrix, but instead relies on the past (often less than 10) values of the parameters and the gradient vector. Therefore, the L-BFGS-B algorithm is especially suitable for high dimensional optimisation problems, but—of course—it can also be used for optimisation problems with only a few parameters (as the CES function). Function **cesEst** estimates a CES function with parameter constraints using the L-BFGS-B algorithm if argument **method** is set to **"L-BFGS-B"**. The user can tweak some details of this algorithm (e.g. the number of BFGS updates) by adding a further argument **control** as described in the "Details" section of the documentation of the R function **optim**. By default, the restrictions on the parameters are $0 \leq \gamma \leq \infty$; $0 \leq \delta_1, \delta_2, \delta \leq 1$; $-1 \leq \rho_1, \rho_2, \rho \leq \infty$; and $0 \leq \nu \leq \infty$. The user can specify own lower and upper bounds by setting arguments **lower** and **upper** to numeric vectors.

```
> cesLbfgsb <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE,
+     method = "L-BFGS-B")
> summary(cesLbfgsb)

Estimated CES function with variable returns to scale
```

```
Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, method = "L-BFGS-B")


Estimation by non-linear least-squares using the 'L-BFGS-B' optimizer
assuming an additive error term
Convergence achieved after 35 function and 35 gradient calls
Message: CONVERGENCE: REL_REDUCTION_OF_F <= FACTR*EPSMCH


Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  1.02385    0.11562   8.855   <2e-16 ***
delta  0.62220    0.02845  21.873   <2e-16 ***
rho    0.54192    0.29090   1.863   0.0625 .
nu     1.08582    0.04569  23.765   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 2.446577
Multiple R-squared: 0.7649817


Elasticity of Substitution:
           Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6485     0.1224     5.3 1.16e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

PORT routines

The so-called PORT routines (Gay 1990) include a quasi-Newton optimisation algorithm that allows for box constraints on the parameters and has several advantages over traditional Newton routines, e.g. trust regions and reverse communication. Setting argument `method` to `"PORT"` selects the optimisation algorithm of the PORT routines. The user can modify a few details of the Newton algorithm (e.g. the minimum step size) by adding a further argument `control` as described in section "Control parameters" of the documentation of R function `nlminb`. The lower and upper bounds of the parameters have the same default values as for the L-BFGS-B method.

```
> cesPort <- cesEst("y2", c("x1", "x2"), cesData, vrs = TRUE, method = "PORT")
> summary(cesPort)
```

Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, method = "PORT")

Estimation by non-linear least-squares using the 'PORT' optimizer
assuming an additive error term
Convergence achieved after 27 iterations
Message: relative convergence (4)

Coefficients:
       Estimate Std. Error t value Pr(>|t|)
gamma   1.02385   0.11562   8.855   <2e-16 ***
delta   0.62220   0.02845 21.873   <2e-16 ***
rho   0.54192   0.29091   1.863   0.0625 .
nu   1.08582   0.04569 23.765   <2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.446577
Multiple R-squared: 0.7649817

Elasticity of Substitution:
            Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6485   0.1224   5.3 1.16e-07 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

### 3.5. Technological change

Estimating the CES function with time series data usually requires an extension of the CES functional form in order to account for technological change (progress). So far, accounting for technological change in CES functions basically boils down to two approaches:

- Hicks-neutral technological change

$$y = \gamma \, e^{\lambda t} \left( \delta x_1^{-\rho} + (1 - \delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}}, \tag{13}$$

where $\gamma$ is (as before) an efficiency parameter, $\lambda$ is the rate of technological change, and $t$ is a time variable.

- factor augmenting (non-neutral) technological change

$$y = \gamma \left( \left( x_1 \, e^{\lambda_1 t} \right)^{-\rho} + \left( x_2 \, e^{\lambda_2 t} \right)^{-\rho} \right)^{-\frac{\nu}{\rho}}, \tag{14}$$

where $\lambda_1$ and $\lambda_2$ measure input-specific technological change.

There is a lively ongoing discussion about the proper way to estimate CES functions with factor augmenting technological progress (e.g. Klump, McAdam, and Willman 2007; Luoma and Luoto 2010; León-Ledesma, McAdam, and Willman 2010). Although many approaches seem to be promising, we decided to wait until a state-of-the-art approach emerges before including factor augmenting technological change into **micEconCES**. Therefore, **micEconCES** only includes Hicks-neutral technological change at the moment.

When calculating the output variable of the CES function using `cesCalc` or when estimating the parameters of the CES function using `cesEst`, the name of time variable ($t$) can be specified by argument `tName`, where the corresponding coefficient ($\lambda$) is labelled `lambda`.[10] The following commands (i) generate an (artificial) time variable `t`, (ii) calculate the (deterministic) output variable of a CES function with 1% Hicks-neutral technological progress in each time period, (iii) add noise to obtain the stochastic "observed" output variable, (iv) estimate the model, and (v) print the summary results.

```
> cesData$t <- c(1:200)
> cesData$yt <- cesCalc(xNames = c("x1", "x2"), data = cesData,
+     tName = "t", coef = c(gamma = 1, delta = 0.6, rho = 0.5,
+         nu = 1.1, lambda = 0.01))
> cesData$yt <- cesData$yt + 2.5 * rnorm(200)
> cesTech <- cesEst("yt", c("x1", "x2"), data = cesData, tName = "t",
+     vrs = TRUE, method = "LM")
> summary(cesTech)


Estimated CES function with variable returns to scale
```

---

[10]If the Differential Evolution (DE) algorithm is used, parameter λ is by default restricted to the interval $[-0.5, 0.5]$, as this algorithm requires finite lower and upper bounds of all parameters. The user can use arguments `lower` and `upper` to modify these bounds.

```
Call:
cesEst(yName = "yt", xNames = c("x1", "x2"), data = cesData,
    tName = "t", vrs = TRUE, method = "LM")


Estimation by non-linear least-squares using the 'LM' optimizer
assuming an additive error term
Convergence achieved after 5 iterations
Message: Relative error in the sum of squares is at most `ftol'.


Coefficients:
        Estimate Std. Error t value Pr(>|t|)
gamma  0.9932082  0.0362520  27.397  < 2e-16 ***
lambda 0.0100173  0.0001007  99.506  < 2e-16 ***
delta  0.5971397  0.0073754  80.964  < 2e-16 ***
rho    0.5268406  0.0696036   7.569 3.76e-14 ***
nu     1.1024537  0.0130233  84.653  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 2.550677
Multiple R-squared: 0.9899389


Elasticity of Substitution:
            Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)  0.65495    0.02986   21.94   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Above, we have demonstrated how Hicks-neutral technological change can be modelled in the two-input CES function. In case of more than two inputs—regardless of whether the CES function is "plain" or "nested"—Hicks-neutral technological change can be accounted for in the same way, i.e. by multiplying the CES function with $e^{\lambda t}$. Functions `cesCalc` and `cesEst` can account for Hicks-neutral technological change in all CES specifications that are generally supported by these functions.

3.6. Grid search for ρ

The objective function for estimating CES functions by non-linear least-squares often shows a tendency to "flat surfaces" around the minimum—in particular for a wide range of values for the substitution parameters (ρ_1, ρ_2, ρ). Therefore, many optimisation algorithms have

problems in finding the minimum of the objective function, particularly in case of *n*-input nested CES functions.

However, this problem can be alleviated by performing a grid search, where a grid of values for the substitution parameters (ρ_1, ρ_2, ρ) is pre-selected and the remaining parameters are estimated by non-linear least-squares holding the substitution parameters fixed at each combination of the pre-defined values. As the (nested) CES functions defined above can have up to three substitution parameters, the grid search over the substitution parameters can be either one-, two-, or three-dimensional. The estimates with the values of the substitution parameters that result in the smallest sum of squared residuals are chosen as the final estimation result.

The function `cesEst` carries out this grid search procedure, if argument `rho1`, `rho2`, or `rho` is set to a numeric vector. The values of these vectors are used to specify the grid points for the substitution parameters ρ_1, ρ_2, and ρ, respectively. The estimation of the other parameters during the grid search can be performed by all the non-linear optimisation algorithms described above. Since the "best" values of the substitution parameters (ρ_1, ρ_2, ρ) that are found in the grid search are not known, but estimated (as the other parameters, but with a different estimation method), the covariance matrix of the estimated parameters also includes the substitution parameters and is calculated as if the substitution parameters were estimated as usual. The following command estimates the two-input CES function by a one-dimensional grid search for ρ, where the pre-selected values for ρ are the values from -0.3 to 1.5 with an increment of 0.1 and the default optimisation method, the Levenberg-Marquardt algorithm, is used to estimate the remaining parameters.

```
> cesGrid <- cesEst("y2", c("x1", "x2"), data = cesData, vrs = TRUE,
+     rho = seq(from = -0.3, to = 1.5, by = 0.1))
> summary(cesGrid)

Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, rho = seq(from = -0.3, to = 1.5, by = 0.1))

Estimation by non-linear least-squares using the 'LM' optimizer
and a one-dimensional grid search for coefficient 'rho'
assuming an additive error term
Convergence achieved after 4 iterations
Message: Relative error in the sum of squares is at most `ftol'.
```

```
Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  1.01851    0.11506   8.852    <2e-16 ***
delta  0.62072    0.02819  22.022    <2e-16 ***
rho    0.50000    0.28543   1.752    0.0798 .
nu     1.08746    0.04570  23.794    <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 2.44672
Multiple R-squared: 0.7649542


Elasticity of Substitution:
            Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6667     0.1269   5.255 1.48e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

A graphical illustration of the relationship between the pre-selected values of the substitution parameters and the corresponding sums of the squared residuals can be obtained by applying the `plot` method.[11]

```
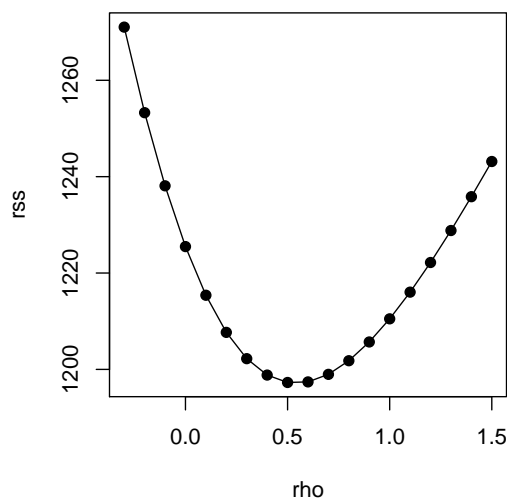> plot(cesGrid)
```

This graphical illustration is shown in figure 3.

As a further example, we estimate a four-input nested CES function by a three-dimensional grid search for ρ_1, ρ_2, and ρ. Preselected values are -0.6 to 0.9 with an increment of 0.3 for ρ_1, -0.4 to 0.8 with an increment of 0.2 for ρ_2, and -0.3 to 1.7 with an increment of 0.2 for ρ. Again, we apply the default optimisation method, the Levenberg-Marquardt algorithm.

```
> ces4Grid <- cesEst(yName = "y4", xNames = c("x1", "x2", "x3",
+     "x4"), data = cesData, method = "LM", rho1 = seq(from = -0.6,
+     to = 0.9, by = 0.3), rho2 = seq(from = -0.4, to = 0.8, by = 0.2),
+     rho = seq(from = -0.3, to = 1.7, by = 0.2))
> summary(ces4Grid)


Estimated CES function with constant returns to scale
```

---

[11]This `plot` method can only be applied if the model was estimated by grid search.

Figure 3: Sum of squared residuals depending on ρ

```
Call:
cesEst(yName = "y4", xNames = c("x1", "x2", "x3", "x4"), data = cesData,
    method = "LM", rho1 = seq(from = -0.6, to = 0.9, by = 0.3),
    rho2 = seq(from = -0.4, to = 0.8, by = 0.2), rho = seq(from = -0.3,
        to = 1.7, by = 0.2))
```

```
Estimation by non-linear least-squares using the 'LM' optimizer
and a three-dimensional grid search for coefficients 'rho_1', 'rho_2', 'rho'
assuming an additive error term
Convergence achieved after 4 iterations
Message: Relative error in the sum of squares is at most `ftol'.
```

```
Coefficients:
        Estimate Std. Error t value Pr(>|t|)
gamma    1.28086    0.01632  78.482  < 2e-16 ***
delta_1  0.78337    0.03237  24.197  < 2e-16 ***
delta_2  0.60272    0.02608  23.111  < 2e-16 ***
delta    0.51498    0.02119  24.302  < 2e-16 ***
rho_1    0.30000    0.45684   0.657 0.511382
rho_2    0.40000    0.23500   1.702 0.088727 .
rho      0.90000    0.24714   3.642 0.000271 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 1.425583
Multiple R-squared: 0.7887368


Elasticities of Substitution:
                  Estimate Std. Error t value Pr(>|t|)
E_1_2 (HM)         0.76923    0.27032   2.846  0.00443 **
E_3_4 (HM)         0.71429    0.11990   5.958 2.56e-09 ***
E_(1,2)_(3,4) (AU) 0.52632    0.06846   7.688 1.49e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
HM = Hicks-McFadden (direct) elasticity of substitution
AU = Allen-Uzawa (partial) elasticity of substitution
```

Naturally, for a three-dimensional grid search, plotting the sums of the squared residuals against the corresponding (pre-selected) values of ρ_1, ρ_2, and ρ, would require a four-dimensional graph. As it is (currently) not possible to account for more than three dimensions in a graph, the `plot` method generates three three-dimensional graphs, where each of the three substitution parameters (ρ_1, ρ_2, ρ) in turn is kept fixed at its optimal value. An example is shown in figure 4.

```
> plot(ces4Grid)
```

The results of the grid search algorithm can be used either directly, or as starting values for a new non-linear least-squares estimation. In the latter case, the values of the substitution parameters that are between the grid points can also be estimated. Starting values can be set by argument `start`.

```
> cesStartGrid <- cesEst("y2", c("x1", "x2"), data = cesData, vrs = TRUE,
+     start = coef(cesGrid))
> summary(cesStartGrid)

Estimated CES function with variable returns to scale

Call:
cesEst(yName = "y2", xNames = c("x1", "x2"), data = cesData,
    vrs = TRUE, start = coef(cesGrid))


Estimation by non-linear least-squares using the 'LM' optimizer
assuming an additive error term
Convergence achieved after 4 iterations
```

negative sums of squared residuals

Figure 4: Sum of squared residuals depending on ρ_1, ρ_2, and ρ

```
Message: Relative error in the sum of squares is at most `ftol'.


Coefficients:
       Estimate Std. Error t value Pr(>|t|)
gamma  1.02385    0.11562   8.855   <2e-16 ***
delta  0.62220    0.02845  21.873   <2e-16 ***
rho    0.54192    0.29090   1.863   0.0625 .
nu     1.08582    0.04569  23.765   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 2.446577
Multiple R-squared: 0.7649817


Elasticity of Substitution:
            Estimate Std. Error t value Pr(>|t|)
E_1_2 (all)   0.6485     0.1224     5.3 1.16e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


> ces4StartGrid <- cesEst("y4", c("x1", "x2", "x3", "x4"), data = cesData,
+     start = coef(ces4Grid))
> summary(ces4StartGrid)


Estimated CES function with constant returns to scale


Call:
cesEst(yName = "y4", xNames = c("x1", "x2", "x3", "x4"), data = cesData,
    start = coef(ces4Grid))


Estimation by non-linear least-squares using the 'LM' optimizer
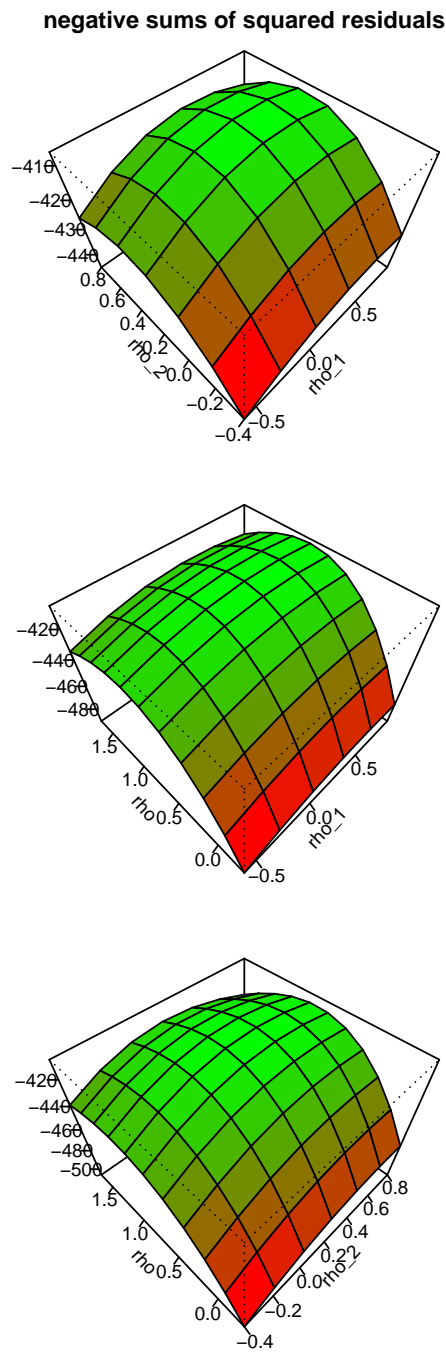assuming an additive error term
Convergence achieved after 6 iterations
Message: Relative error in the sum of squares is at most `ftol'.


Coefficients:
         Estimate Std. Error t value Pr(>|t|)
gamma     1.28212    0.01634  78.442  < 2e-16 ***
delta_1   0.78554    0.03303  23.783  < 2e-16 ***
```

```
delta_2  0.60130     0.02573  23.374  < 2e-16 ***
delta    0.51224     0.02130  24.049  < 2e-16 ***
rho_1    0.41742     0.46878   0.890 0.373239
rho_2    0.34464     0.22922   1.504 0.132696
rho      0.93762     0.25067   3.741 0.000184 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 1.425085
Multiple R-squared: 0.7888844


Elasticities of Substitution:
                  Estimate Std. Error t value Pr(>|t|)
E_1_2 (HM)         0.70551    0.23333   3.024   0.0025 **
E_3_4 (HM)         0.74369    0.12678   5.866 4.46e-09 ***
E_(1,2)_(3,4) (AU)  0.51610    0.06677   7.730 1.08e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
HM = Hicks-McFadden (direct) elasticity of substitution
AU = Allen-Uzawa (partial) elasticity of substitution
```

4. Implementation

The function `cesEst` is the primary user interface of the **micEconCES** package (Henningsen and Henningsen 2011). However, the actual estimations are carried out by internal helper functions, or functions from other packages.

4.1. Kmenta approximation

The estimation of the Kmenta approximation (5) is implemented in the internal function `cesEstKmenta`. This function uses `translogEst` from the **micEcon** package (Henningsen 2010) to estimate the unrestricted translog function (6). The test of the parameter restrictions defined in equation (7) is performed by the function `linear.hypothesis` of the **car** package (Fox 2009). The restricted translog model (6, 7) is estimated with function `systemfit` from the **systemfit** package (Henningsen and Hamann 2007).

4.2. Non-linear least-squares estimation

The non-linear least-squares estimations are carried out by various optimisers from other packages. Estimations with the Levenberg-Marquardt algorithm are performed by function `nls.lm` of the **minpack.lm** package (Elzhov and Mullen 2009), which is an R interface to the Fortran package **MINPACK** (Moré, Garbow, and Hillstrom 1980). Estimations with the "Conjugate Gradients" (CG), BFGS, Nelder-Mead (NM), Simulated Annealing (SANN), and L-BFGS-B algorithms use the function `optim` from the **stats** package (R Development Core Team 2011). Estimations with the Newton-type algorithm are performed by function `nlm` from the **stats** package (R Development Core Team 2011), which uses the Fortran library **UNCMIN** (Schnabel *et al.* 1985) with a line search as the step selection strategy. Estimations with the Differential Evolution (DE) algorithm are performed by function `DEoptim` from the **DEoptim** package (Mullen, Ardia, Gil, Windover, and Cline 2011). Estimations with the PORT routines use function `nlminb` from the **stats** package (R Development Core Team 2011), which uses the Fortran library **PORT** (Gay 1990).

4.3. Grid search

If the user calls `cesEst` with at least one of the arguments `rho1`, `rho2`, or `rho` being a vector, `cesEst` calls the internal function `cesEstGridRho`, which implements the actual grid search procedure. For each combination (grid point) of the pre-selected values of the substitution parameters (ρ_1, ρ_2, ρ), on which the grid search should be performed, the function `cesEstGridRho` consecutively calls `cesEst`. In each of these internal calls of `cesEst`, the parameters on which no grid search should be performed (and which are not fixed by the user), are estimated given the particular combination of substitution parameters, on which the grid search should be performed. This is done by setting the arguments of `cesEst`, for which the user has specified vectors of pre-selected values of the substitution parameters (`rho1`, `rho2`, and/or `rho`), to the particular elements of these vectors. As `cesEst` is called with arguments `rho1`, `rho2`, and `rho` being all single scalars (or `NULL` if the corresponding substitution parameter is neither included in the grid search nor fixed at a pre-defined value), it estimates the CES function by non-linear least-squares with the corresponding substitution parameters (ρ_1, ρ_2, and/or ρ) fixed at the values specified in the corresponding arguments.

4.4. Calculating output and the sum of squared residuals

Function `cesCalc` can be used to calculate the output quantity of the CES function given input quantities and coefficients. A few examples of using `cesCalc` are shown in the beginning of section 3, where this function is applied to generate the output variables of an artificial data set for demonstrating the usage of `cesEst`. Furthermore, the `cesCalc` function is called by the internal function `cesRss` that calculates and returns the sum of squared residuals,

which is the objective function in the non-linear least-squares estimations. If at least one substitution parameter (ρ_1, ρ_2, ρ) is equal to zero, the CES functions are not defined. In this case, `cesCalc` returns the limit of the output quantity for ρ_1, ρ_2, and/or ρ approaching zero. In case of nested CES functions with three or four inputs, function `cesCalc` calls the internal functions `cesCalcN3` or `cesCalcN4` for the actual calculations.

We noticed that the calculations with `cesCalc` using equations (1), (3), or (4) are imprecise if at least one of the substitution parameters (ρ_1, ρ_2, ρ) is close to 0. This is caused by rounding errors that are unavoidable on digital computers, but are usually negligible. However, rounding errors can become large in specific circumstances, e.g. in CES functions with very small (in absolute terms) substitution parameters, when first very small (in absolute terms) exponents (e.g. $-\rho_1$, $-\rho_2$, or $-\rho$) and then very large (in absolute terms) exponents (e.g. ρ/ρ_1, ρ/ρ_2, or $-\nu/\rho$) are applied. Therefore, for the traditional two-input CES function (1), `cesCalc` uses a first-order Taylor series approximation at the point $\rho = 0$ for calculating the output, if the absolute value of ρ is smaller than, or equal to, argument `rhoApprox`, which is $5 \cdot 10^{-6}$ by default. This first-order Taylor series approximation is the Kmenta approximation defined in (5).[12] We illustrate the rounding errors in the left panel of figure 5, which has been created by following commands.

```
> rhoData <- data.frame(rho = seq(-2e-06, 2e-06, 5e-09), yCES = NA,
+     yLin = NA)
> for (i in 1:nrow(rhoData)) {
+     cesCoef <- c(gamma = 1, delta = 0.6, rho = rhoData$rho[i],
+         nu = 1.1)
+     rhoData$yLin[i] <- cesCalc(xNames = c("x1", "x2"), data = cesData[1,
+         ], coef = cesCoef, rhoApprox = Inf)
+     rhoData$yCES[i] <- cesCalc(xNames = c("x1", "x2"), data = cesData[1,
+         ], coef = cesCoef, rhoApprox = 0)
+ }
> rhoData$yCES <- rhoData$yCES - rhoData$yLin[rhoData$rho == 0]
> rhoData$yLin <- rhoData$yLin - rhoData$yLin[rhoData$rho == 0]
> plot(rhoData$rho, rhoData$yCES, type = "l", col = "red", xlab = "rho",
+     ylab = "y (normalised, red = CES, black = linearised)")
> lines(rhoData$rho, rhoData$yLin)
```

---

[12]The derivation of the first-order Taylor series approximations based on Uebe (2000) is presented in appendix A.1.

Figure 5: Calculated output for different values of ρ

The right panel of figure 5 shows that the relationship between ρ and the output y can be rather precisely approximated by a linear function, because it is nearly linear for a wide range of ρ values.[13]

In case of nested CES functions, if at least one substitution parameter (ρ_1, ρ_2, ρ) is close to zero, `cesCalc` uses linear interpolation in order to avoid rounding errors. In this case, `cesCalc` calculates the output quantities for two different values of each substitution parameter that is close to zero, i.e. zero (using the formula for the limit for this parameter approaching zero) and the positive or negative value of argument `rhoApprox` (using the same sign as this parameter). Depending on the number of substitution parameters (ρ_1, ρ_2, ρ) that are close to zero, a one-, two-, or three-dimensional linear interpolation is applied. These interpolations are performed by the internal functions `cesInterN3` and `cesInterN4`.[14]

When estimating a CES function with function `cesEst`, the user can use argument `rhoApprox` to modify the threshold to calculate the dependent variable by the Taylor series approximation or linear interpolation. Argument `rhoApprox` of `cesEst` must be a numeric vector, where the first element is passed to `cesCalc` (partly through `cesRss`). This might not only affect the fitted values and residuals returned by `cesEst` (if at least one of the estimated substitution parameters is close to zero), but also the estimation results (if one of the substitution parameters was close to zero in one of the steps of the iterative optimisation routines).

---

[13]The commands for creating the right panel of figure 5 are not shown here, because they are the same as the commands for the left panel of this figure except for the command for creating the vector of ρ values.

[14]We use a different approach for the nested CES functions than for the traditional two-input CES function, because calculating Taylor series approximations of nested CES functions (and their derivatives with respect to coefficients, see the following section) is very laborious and has no advantages over using linear interpolation.

4.5. Partial derivatives with respect to coefficients

The internal function `cesDerivCoef` returns the partial derivatives of the CES function with respect to all coefficients at all provided data points. For the traditional two-input CES function, these partial derivatives are:

$$\frac{\partial y}{\partial \gamma} = e^{\lambda t} \left(\delta x_1^{-\rho} + (1 - \delta)x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \tag{15}$$

$$\frac{\partial y}{\partial \lambda} = \gamma \, t \, \frac{\partial y}{\partial \gamma} \tag{16}$$

$$\frac{\partial y}{\partial \delta} = - \gamma \, e^{\lambda t} \frac{\nu}{\rho} \left(x_1^{-\rho} - x_2^{-\rho} \right) \left(\delta x_1^{-\rho} + (1 - \delta)x_2^{-\rho} \right)^{-\frac{\nu}{\rho} - 1} \tag{17}$$

$$\frac{\partial y}{\partial \rho} = \gamma \, e^{\lambda t} \frac{\nu}{\rho^2} \, \ln \left(\delta x_1^{-\rho} + (1 - \delta)x_2^{-\rho} \right) \left(\delta x_1^{-\rho} + (1 - \delta)x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \tag{18}$$

$$+ \gamma \, e^{\lambda t} \frac{\nu}{\rho} \left(\delta \ln(x_1)x_1^{-\rho} + (1 - \delta) \ln(x_2)x_2^{-\rho} \right) \left(\delta x_1^{-\rho} + (1 - \delta)x_2^{-\rho} \right)^{-\frac{\nu}{\rho} - 1}$$

$$\frac{\partial y}{\partial \nu} = - \gamma \, e^{\lambda t} \frac{1}{\rho} \, \ln \left(\delta x_1^{-\rho} + (1 - \delta)x_2^{-\rho} \right) \left(\delta x_1^{-\rho} + (1 - \delta)x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \tag{19}$$

These derivatives are not defined for $\rho = 0$ and are imprecise if ρ is close to zero (similar to the output variable of the CES function, see section 4.4). Therefore, if ρ is zero or close to zero, we calculate these derivatives by first-order Taylor series approximations at the point $\rho = 0$ using the limits for ρ approaching zero:[15]

$$\frac{\partial y}{\partial \gamma} = e^{\lambda t} \, x_1^{\nu \, \delta} \, x_2^{\nu \, (1-\delta)} \exp \left(-\frac{\rho}{2} \, \nu \, \delta \, (1 - \delta) \, (\ln x_1 - \ln x_2)^2 \right) \tag{20}$$

$$\frac{\partial y}{\partial \delta} = \gamma \, e^{\lambda t} \, \nu \, (\ln x_1 - \ln x_2) \, x_1^{\nu \, \delta} \, x_2^{\nu(1-\delta)} \tag{21}$$

$$\left(1 - \frac{\rho}{2} \left[1 - 2 \, \delta + \nu \, \delta(1 - \delta) \, (\ln x_1 - \ln x_2) \right] (\ln x_1 - \ln x_2) \right)$$

$$\frac{\partial y}{\partial \rho} = \gamma \, e^{\lambda t} \, \nu \, \delta \, (1 - \delta) \, x_1^{\nu \, \delta} \, x_2^{\nu(1-\delta)} \left(-\frac{1}{2} \, (\ln x_1 - \ln x_2)^2 \right. \tag{22}$$

$$\left. + \frac{\rho}{3}(1 - 2 \, \delta) \, (\ln x_1 - \ln x_2)^3 + \frac{\rho}{4} \, \nu \, \delta(1 - \delta) \, (\ln x_1 - \ln x_2)^4 \right)$$

$$\frac{\partial y}{\partial \nu} = \gamma \, e^{\lambda t} \, x_1^{\nu \, \delta} \, x_2^{\nu(1-\delta)} \left(\delta \ln x_1 + (1 - \delta) \ln x_2 \right. \tag{23}$$

$$\left. - \frac{\rho}{2} \, \delta(1 - \delta) \, (\ln x_1 - \ln x_2)^2 \left[1 + \nu \, (\delta \ln x_1 + (1 - \delta) \ln x_2) \right] \right)$$

If ρ is zero or close to zero, the partial derivatives with respect to λ are calculated also with equation 16, but now $\partial y/\partial \gamma$ is calculated with equation 20 instead of equation 15.

The partial derivatives of the nested CES functions with three and four inputs with respect to the coefficients are presented in appendices B.2 and C.2, respectively. If at least one

---

[15]The derivations of these formulas are presented in appendix A.2.

substitution parameter (ρ_1, ρ_2, ρ) is exactly zero or close to zero, `cesDerivCoef` uses the same approach as `cesCalc` to avoid large rounding errors, i.e. using the limit for these parameters approaching zero and potentially a one-, two-, or three-dimensional linear interpolation. The limits of the partial derivatives of the nested CES functions for one or more substitution parameters approaching zero are also presented in appendices B.2 and C.2. The calculation of the partial derivatives and their limits are performed by several internal functions with names starting with `cesDerivCoefN3` and `cesDerivCoefN4`.[16] The one- or more-dimensional linear interpolations are (again) performed by the internal functions `cesInterN3` and `cesInterN4`.

Function `cesDerivCoef` has an argument `rhoApprox` that can be used to specify the threshold levels for defining when ρ_1, ρ_2, and ρ are "close" to zero. This argument must be a numeric vector with exactly four elements: the first element defines the threshold for $\partial y / \partial \gamma$ (default value $5 \cdot 10^{-6}$), the second element defines the threshold for $\partial y / \partial \delta_1$, $\partial y / \partial \delta_2$, and $\partial y / \partial \delta$ (default value $5 \cdot 10^{-6}$), the third element defines the threshold for $\partial y / \partial \rho_1$, $\partial y / \partial \rho_2$, and $\partial y / \partial \rho$ (default value 10^{-3}), and the fourth element defines the threshold for $\partial y / \partial \nu$ (default value $5 \cdot 10^{-6}$).

Function `cesDerivCoef` is used to provide argument `jac` (which should be set to a function that returns the Jacobian of the residuals) to function `nls.lm` so that the Levenberg-Marquardt algorithm can use analytical derivatives of each residual with respect to all coefficients. Furthermore, function `cesDerivCoef` is used by the internal function `cesRssDeriv`, which calculates the partial derivatives of the sum of squared residuals (RSS) with respect to all coefficients by:

$$\frac{\partial \text{RSS}}{\partial \theta} = -2 \sum_{i=1}^{N} \left(u_i \ \frac{\partial y_i}{\partial \theta} \right), \tag{24}$$

where N is the number of observations, u_i is the residual of the ith observation, $\theta \in \{\gamma, \delta_1, \delta_2, \delta, \rho_1, \rho_2, \rho, \nu\}$ is a coefficient of the CES function, and $\partial y_i / \partial \theta$ is the partial derivative of the CES function with respect to coefficient θ evaluated at the ith observation as returned by function `cesDerivCoef`. Function `cesRssDeriv` is used to provide analytical gradients for the other gradient-based optimisation algorithms, i.e. Conjugate Gradients, Newton-type, BFGS, L-BFGS-B, and PORT. Finally, function `cesDerivCoef` is used to ob-

---

[16]The partial derivatives of the nested CES function with three inputs are calculated by: `cesDerivCoefN3Gamma` ($\partial y / \partial \gamma$), `cesDerivCoefN3Lambda` ($\partial y / \partial \lambda$), `cesDerivCoefN3Delta1` ($\partial y / \partial \delta_1$), `cesDerivCoefN3Delta` ($\partial y / \partial \delta$), `cesDerivCoefN3Rho1` ($\partial y / \partial \rho_1$), `cesDerivCoefN3Rho` ($\partial y / \partial \rho$), and `cesDerivCoefN3Nu` ($\partial y / \partial \nu$) with helper functions `cesDerivCoefN3B1` (returning $B_1 = \delta_1 x_1^{-\rho_1} + (1 - \delta_1) x_2^{-\rho_1}$), `cesDerivCoefN3L1` (returning $L_1 = \delta_1 \ln x_1 + (1 - \delta_1) \ln x_2$), and `cesDerivCoefN3B` (returning $B = \delta B_1^{\rho/\rho_1} + (1 - \delta) x_3^{-\rho}$). The partial derivatives of the nested CES function with four inputs are calculated by: `cesDerivCoefN4Gamma` ($\partial y / \partial \gamma$), `cesDerivCoefN4Lambda` ($\partial y / \partial \lambda$), `cesDerivCoefN4Delta1` ($\partial y / \partial \delta_1$), `cesDerivCoefN4Delta2` ($\partial y / \partial \delta_2$), `cesDerivCoefN4Delta` ($\partial y / \partial \delta$), `cesDerivCoefN4Rho1` ($\partial y / \partial \rho_1$), `cesDerivCoefN4Rho2` ($\partial y / \partial \rho_2$), `cesDerivCoefN4Rho` ($\partial y / \partial \rho$), and `cesDerivCoefN4Nu` ($\partial y / \partial \nu$) with helper functions `cesDerivCoefN4B1` (returning $B_1 = \delta_1 x_1^{-\rho_1} + (1 - \delta_1) x_2^{-\rho_1}$), `cesDerivCoefN4L1` (returning $L_1 = \delta_1 \ln x_1 + (1 - \delta_1) \ln x_2$), `cesDerivCoefN4B2` (returning $B_2 = \delta_2 x_3^{-\rho_2} + (1 - \delta_2) x_4^{-\rho_2}$), `cesDerivCoefN4L2` (returning $L_2 = \delta_2 \ln x_3 + (1 - \delta_2) \ln x_4$), and `cesDerivCoefN4B` (returning $B = \delta B_1^{\rho/\rho_1} + (1 - \delta) B_2^{\rho/\rho_2}$).

tain the gradient matrix for calculating the asymptotic covariance matrix of the non-linear least-squares estimator (see section 4.6).

When estimating a CES function with function `cesEst`, the user can use argument `rhoApprox` to specify the thresholds below which the derivatives with respect to the coefficients are approximated by Taylor series approximations or linear interpolations. Argument `rhoApprox` of `cesEst` must be a numeric vector of five elements, where the second to the fifth element of this vector are passed to `cesDerivCoef`. The choice of the threshold might not only affect the covariance matrix of the estimates (if at least one of the estimated substitution parameters is close to zero), but also the estimation results obtained by a gradient-based optimisation algorithm (if one of the substitution parameters is close to zero in one of the steps of the iterative optimisation routines).

4.6. Covariance matrix

The asymptotic covariance matrix of the non-linear least-squares estimator obtained by the various iterative optimisation methods is calculated by:

$$\hat{\sigma}^2 \left(\left(\frac{\partial y}{\partial \theta} \right)^\top \frac{\partial y}{\partial \theta} \right)^{-1} \tag{25}$$

(Greene 2008, p. 292), where $\partial y/\partial \theta$ denotes the $N \times k$ gradient matrix (defined in equations (15) to (19) for the traditional two-input CES function and in appendices B.2 and C.2 for the nested CES functions), N is the number of observations, k is the number of coefficients, and $\hat{\sigma}^2$ denotes the estimated variance of the residuals. As equation (25) is only valid asymptotically, we calculate the estimated variance of the residuals by

$$\hat{\sigma}^2 = \frac{1}{N} \sum_{i=1}^{N} u_i^2, \tag{26}$$

i.e. without correcting for degrees of freedom.

4.7. Starting values

If the user calls `cesEst` with argument `start` set to a vector of starting values, the internal function `cesEstStart` checks if the number of starting values is correct and if the individual starting values are in the appropriate range of the corresponding parameters. If no starting values are provided by the user, function `cesEstStart` determines the starting values automatically. The starting values of δ_1, δ_2, and δ are set to 0.5. If the coefficients ρ_1, ρ_2, and ρ are estimated (not fixed as, e.g. during grid search), their starting values are set to 0.25, which generally corresponds to an elasticity of substitution of 0.8. The starting value of ν is set to 1, which corresponds to constant returns to scale. If the CES function includes a time

variable, the starting value of λ is set to 0.015, which corresponds to a technological progress of 1.5% per time period. Finally, the starting value of γ is set to a value so that the mean of the residuals is equal to zero, i.e.

$$\gamma = \frac{\sum_{i=1}^{N} y_i}{\sum_{i=1}^{N} CES_i}, \tag{27}$$

where CES_i indicates the (nested) CES function evaluated at the input quantities of the ith observation and with coefficient γ equal to one, all "fixed" coefficients (e.g. ρ_1, ρ_2, or ρ) equal to their pre-selected values, and all other coefficients equal to the above-described starting values.

4.8. Other internal functions

The internal function `cesCoefAddRho` is used to add the values of ρ_1, ρ_2, and ρ to the vector of coefficients, if these coefficients are fixed (e.g. during grid search for ρ) and hence, are not included in the vector of estimated coefficients.

If the user selects the optimisation algorithm Differential Evolution, L-BFGS-B, or PORT, but does not specify lower or upper bounds of the coefficients, the internal function `cesCoefBounds` creates and returns the default bounds depending on the optimisation algorithm as described in sections 3.3 and 3.4.

The internal function `cesCoefNames` returns a vector of character strings, which are the names of the coefficients of the CES function.

The internal function `cesCheckRhoApprox` checks argument `rhoApprox` of functions `cesEst`, `cesDerivCoef`, `cesRss`, and `cesRssDeriv`.

4.9. Methods

The **micEconCES** package makes use of the "S3" class system of the R language introduced in Chambers and Hastie (1992). Objects returned by function `cesEst` are of class `"cesEst"` and the **micEconCES** package includes several methods for objects of this class. The `print` method prints the call, the estimated coefficients, and the estimated elasticities of substitution. The `coef`, `vcov`, `fitted`, and `residuals` methods extract and return the estimated coefficients, their covariance matrix, the fitted values, and the residuals, respectively.

The `plot` method can only be applied if the model is estimated by grid search (see section 3.6). If the model is estimated by a one-dimensional grid search for ρ_1, ρ_2, or ρ, this method plots a simple scatter plot of the pre-selected values against the corresponding sums of the squared residuals by using the commands `plot.default` and `points` of the **graphics** package (R Development Core Team 2011). In case of a two-dimensional grid search, the plot method draws a perspective plot by using the command `persp` of the **graphics** package

(R Development Core Team 2011) and the command `colorRampPalette` of the **grDevices** package (R Development Core Team 2011) (for generating a colour gradient). In case of a three-dimensional grid search, the `plot` method plots three perspective plots by holding one of the three coefficients ρ_1, ρ_2, and ρ constant in each of the three plots.

The `summary` method calculates the estimated standard error of the residuals ($\hat{\sigma}$), the covariance matrix of the estimated coefficients and elasticities of substitution, the R^2 value as well as the standard errors, t-values, and marginal significance levels (P-values) of the estimated parameters and elasticities of substitution. The object returned by the `summary` method is of class `"summary.cesEst"`. The `print` method for objects of class `"summary.cesEst"` prints the call, the estimated coefficients and elasticities of substitution, their standard errors, t-values, and marginal significance levels as well as some information on the estimation procedure (e.g. algorithm, convergence). The `coef` method for objects of class `"summary.cesEst"` returns a matrix with four columns containing the estimated coefficients, their standard errors, t-values, and marginal significance levels, respectively.

5. Replication studies

In this section, we aim at replicating estimations of CES functions published in two journal articles. This section has three objectives: first, to highlight and discuss the problems that occur when using real-world data by basing the econometric estimations in this section on real-world data, which is in contrast to the previous sections; second, to confirm the reliability of the **micEconCES** package by comparing `cesEst`'s results with results published in the literature. Third, to encourage reproducible research, which should be afforded higher priority in scientific economic research (e.g. Buckheit and Donoho 1995; Schwab, Karrenbach, and Claerbout 2000; McCullough, McGeary, and Harrison 2008; Anderson, Greene, McCullough, and Vinod 2008; McCullough 2009).

5.1. Sun, Henderson, and Kumbhakar (2011)

Our first replication study aims to replicate some of the estimations published in Sun, Henderson, and Kumbhakar (2011). This article is itself a replication study of Masanjala and Papageorgiou (2004). We will re-estimate a Solow growth model based on an aggregate CES production function with capital and "augmented labour" (i.e. the quantity of labour multiplied by the efficiency of labour) as inputs. This model is given in Masanjala and Papageorgiou

(2004, eq. 3):[17]

$$y_i = A \left[\frac{1}{1-\alpha} - \frac{\alpha}{1-\alpha} \left(\frac{s_{ik}}{n_i + g + \delta} \right)^{\frac{\sigma-1}{\sigma}} \right]^{-\frac{\sigma}{\sigma-1}} \tag{28}$$

Here, y_i is the steady-state aggregate output (Gross Domestic Product, GDP) per unit of labour, s_{ik} is the ratio of investment to aggregate output, n_i is the population growth rate, subscript i indicates the country, g is the technology growth rate, δ is the depreciation rate of capital goods, A is the efficiency of labour (with growth rate g), α is the distribution parameter of capital, and σ is the elasticity of substitution. We can re-define the parameters and variables in the above function in order to obtain the standard two-input CES function (eq. 1), where: $\gamma = A$, $\delta = \frac{1}{1-\alpha}$, $x_1 = 1$, $x_2 = (n_i + g + \delta)/s_{ik}$, $\rho = (\sigma - 1)/\sigma$, and $\nu = 1$. Please note that the relationship between the coefficient ρ and the elasticity of substitution is slightly different from this relationship in the original two-input CES function: in this case, the elasticity of substitution is $\sigma = 1/(1 - \rho)$, i.e. the coefficient ρ has the opposite sign. Hence, the estimated ρ must lie between minus infinity and (plus) one. Furthermore, the distribution parameter α should lie between zero and one, so that the estimated parameter δ must have—in contrast to the standard CES function—a value larger than or equal to one.

The data used in Sun *et al.* (2011) and Masanjala and Papageorgiou (2004) are actually from Mankiw, Romer, and Weil (1992) and Durlauf and Johnson (1995) and comprise cross-sectional data on the country level. This data set is available in the R package **AER** (Kleiber and Zeileis 2009) and includes, amongst others, the variables `gdp85` (per capita GDP in 1985), `invest` (average ratio of investment (including government investment) to GDP from 1960 to 1985 in percent), and `popgrowth` (average growth rate of working-age population 1960 to 1985 in percent). The following commands load this data set and remove data from oil producing countries in order to obtain the same subset of 98 countries that is used by Masanjala and Papageorgiou (2004) and Sun *et al.* (2011).

```
> data("GrowthDJ", package = "AER")
> GrowthDJ <- subset(GrowthDJ, oil == "no")
```

Now we will calculate the two "input" variables for the Solow growth model as described above and in Masanjala and Papageorgiou (2004) (following the assumption of Mankiw *et al.* (1992) that $g + \delta$ is equal to 5%):

```
> GrowthDJ$x1 <- 1
> GrowthDJ$x2 <- (GrowthDJ$popgrowth + 5)/GrowthDJ$invest
```

---

[17]In contrast to equation 3 in Masanjala and Papageorgiou (2004), we decomposed the (unobservable) steady-state output per unit of *augmented* labour in country i ($y_i^* = Y_i/(A\,L_i)$ with Y_i being aggregate output, L_i being aggregate labour, and A being the efficiency of labour) into the (observable) steady-state output per unit of labour ($y_i = Y_i/L_i$) and the (unobservable) efficiency component (A) to obtain the equation that is actually estimated.

The following commands estimate the Solow growth model based on the CES function by non-linear least-squares (NLS) and print the summary results, where we suppress the presentation of the elasticity of substitution (σ), because it has to be calculated with a non-standard formula in this model.

```
> cesNls <- cesEst("gdp85", c("x1", "x2"), data = GrowthDJ)
> summary(cesNls, ela = FALSE)


Estimated CES function with constant returns to scale


Call:
cesEst(yName = "gdp85", xNames = c("x1", "x2"), data = GrowthDJ)


Estimation by non-linear least-squares using the 'LM' optimizer
assuming an additive error term
Convergence achieved after 23 iterations
Message: Relative error in the sum of squares is at most `ftol'.


Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma  646.141    549.993   1.175   0.2401
delta    3.977      2.239   1.776   0.0757 .
rho     -0.197      0.166  -1.187   0.2354
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1


Residual standard error: 3313.748
Multiple R-squared: 0.6016277
```

Now, we will calculate the distribution parameter of capital (α) and the elasticity of substitution (σ) manually.

```
> cat("alpha =", (coef(cesNls)["delta"] - 1)/coef(cesNls)["delta"],
+      "\n")


alpha = 0.7485641


> cat("sigma =", 1/(1 - coef(cesNls)["rho"]), "\n")


sigma = 0.835429
```

These calculations show that we can successfully replicate the estimation results shown in Sun *et al.* (2011, Table 1: $\alpha = 0.7486$, $\sigma = 0.8354$).

As the CES function approaches a Cobb-Douglas function if the coefficient ρ approaches zero and `cesEst` internally uses the limit for ρ approaching zero if ρ equals zero, we can use function `cesEst` to estimate Cobb-Douglas functions by non-linear least-squares (NLS), if we restrict coefficient ρ to zero:

```
> cdNls <- cesEst("gdp85", c("x1", "x2"), data = GrowthDJ, rho = 0)
> summary(cdNls, ela = FALSE)

Estimated CES function with constant returns to scale

Call:
cesEst(yName = "gdp85", xNames = c("x1", "x2"), data = GrowthDJ,
    rho = 0)


Estimation by non-linear least-squares using the 'LM' optimizer
assuming an additive error term
Coefficient 'rho' was fixed at 0
Convergence achieved after 7 iterations
Message: Relative error in the sum of squares is at most `ftol'.


Coefficients:
        Estimate Std. Error t value Pr(>|t|)
gamma 1288.0797    543.1772   2.371 0.017722 *
delta    2.4425      0.6955   3.512 0.000445 ***
rho      0.0000      0.1609   0.000 1.000000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 3342.308
Multiple R-squared: 0.5947313

> cat("alpha =", (coef(cdNls)["delta"] - 1)/coef(cdNls)["delta"],
+      "\n")

alpha = 0.590591
```

As the deviation between our calculated α and the corresponding value published in Sun *et al.* (2011, Table 1: $\alpha = 0.5907$) is very small, we also consider this replication exercise to be successful.

If we restrict ρ to zero and assume a multiplicative error term, the estimation is in fact equivalent to an OLS estimation of the logarithmised version of the Cobb-Douglas function:

```
> cdLog <- cesEst("gdp85", c("x1", "x2"), data = GrowthDJ, rho = 0,
+     multErr = TRUE)
> summary(cdLog, ela = FALSE)


Estimated CES function with constant returns to scale

Call:
cesEst(yName = "gdp85", xNames = c("x1", "x2"), data = GrowthDJ,
    multErr = TRUE, rho = 0)


Estimation by non-linear least-squares using the 'LM' optimizer
assuming a multiplicative error term
Coefficient 'rho' was fixed at 0
Convergence achieved after 8 iterations
Message: Relative error in the sum of squares is at most `ftol'.


Coefficients:
      Estimate Std. Error t value Pr(>|t|)
gamma 965.2337   120.4003   8.017 1.08e-15 ***
delta   2.4880     0.3036   8.195 2.51e-16 ***
rho     0.0000     0.1056   0.000        1
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 0.6814132
Multiple R-squared: 0.5973597


> cat("alpha =", (coef(cdLog)["delta"] - 1)/coef(cdLog)["delta"],
+     "\n")


alpha = 0.5980698
```

Again, we can successfully replicate the α shown in Sun *et al.* (2011, Table 1: $\alpha = 0.5981$).[18]

---

[18]This is indeed a rather complex way of estimating a simple linear model by least squares. We do this just to test the reliability of `cesEst` and for the sake of curiosity, but generally, we recommend using simple linear regression tools to estimate this model.

As all of our above replication exercises were successful, we conclude that the **micEconCES** package has no problems with replicating the estimations of the two-input CES functions presented in Sun *et al.* (2011).

5.2. Kemfert (1998)

Our second replication study aims to replicate the estimation results published in Kemfert (1998). She estimates nested CES production functions for the German industrial sector in order to analyse the substitutability between the three aggregate inputs *capital*, *energy*, and *labour*. As nested CES functions are not invariant to the nesting structure, Kemfert (1998) estimates nested CES functions with all three possible nesting structures. Kemfert's CES functions basically have the same specification as our three-input nested CES function defined in equation 4 and allow for Hicks-neutral technological change as in equation 13. However, Kemfert (1998) does not allow for increasing or decreasing returns to scale and the naming of the parameters is different with: $\gamma = A_s$, $\lambda = m_s$, $\delta_1 = b_s$, $\delta = a_s$, $\rho_1 = \alpha_s$, $\rho = \beta_s$, and $\nu = 1$, where the subscript $s = 1, 2, 3$ of the parameters in Kemfert (1998) indicates the nesting structure. In the first nesting structure ($s = 1$), the three inputs x_1, x_2, and x_3 are capital, energy, and labour, respectively; in the second nesting structure ($s = 2$), they are capital, labour, and energy; and in the third nesting structure ($s = 3$), they are energy, labour, and capital, where—according to the specification of the three-input nested CES function (see equation 4)—the first and second input (x_1 and x_2) are nested together so that the (constant) Allen-Uzawa elasticities of substitution between x_1 and x_3 (σ_{13}) and between x_2 and x_3 (σ_{23}) are equal.

The data used by Kemfert (1998) are available in the R package **micEconCES**. Indeed, these data were taken from the appendix of Kemfert (1998) to ensure that we used exactly the same data. The data are annual aggregated time series data of the entire German industry for the period 1960 to 1993, which were originally published by the German statistical office. Output (`Y`) is given by gross value added of the West German industrial sector (in billion Deutsche Mark at 1991 prices); capital input (`K`) is given by gross stock of fixed assets of the West German industrial sector (in billion Deutsche Mark at 1991 prices); labour input (`A`) is the total number of persons employed in the West German industrial sector (in million); and energy input (`E`) is determined by the final energy consumption of the West German industrial sector (in GWh).

The following commands load the data set, add a linear time trend (starting at zero in 1960), and remove the years of economic disruption during the oil crisis (1973 to 1975) in order to obtain the same subset as used by Kemfert (1998).

```
> data("GermanIndustry")
> GermanIndustry$time <- GermanIndustry$year - 1960
```

```
> GermanIndustry <- subset(GermanIndustry, year < 1973 | year >
+     1975, )
```

First, we try to estimate the first model specification using the standard function for non-linear least-squares estimations in R (`nls`).

```
> cesNls1 <- try(nls(Y ~ gamma * exp(lambda * time) * (delta2 *
+     (delta1 * K^(-rho1) + (1 - delta1) * E^(-rho1))^(rho/rho1) +
+     (1 - delta2) * A^(-rho))^(-1/rho), start = c(gamma = 1, lambda = 0.015,
+     delta1 = 0.5, delta2 = 0.5, rho1 = 0.2, rho = 0.2), data = GermanIndustry))
> cat(cesNls1)

Error in numericDeriv(form[[3L]], names(ind), env) :
  Missing value or an infinity produced when evaluating the model
```

However, as in many estimations of nested CES functions with real-world data, `nls` terminates with an error message. In contrast, the estimation with `cesEst` using, e.g. the Levenberg-Marquardt algorithm, usually returns parameter estimates.

```
> cesLm1 <- cesEst("Y", c("K", "E", "A"), tName = "time", data = GermanIndustry,
+     control = nls.lm.control(maxiter = 1024, maxfev = 2000))
> summary(cesLm1)

Estimated CES function with constant returns to scale

Call:
cesEst(yName = "Y", xNames = c("K", "E", "A"), data = GermanIndustry,
    tName = "time", control = nls.lm.control(maxiter = 1024,
        maxfev = 2000))

Estimation by non-linear least-squares using the 'LM' optimizer
assuming an additive error term
Convergence NOT achieved after 1024 iterations
Message: Number of iterations has reached `maxiter' == 1024.

Coefficients:
          Estimate Std. Error t value Pr(>|t|)
gamma    2.900e+01  5.793e+00   5.006 5.55e-07 ***
lambda   2.100e-02  4.581e-04  45.842  < 2e-16 ***
delta_1  1.966e-68  5.452e-66   0.004   0.9971
```

```
delta    7.635e-05  4.108e-04   0.186    0.8526
rho_1    5.303e+01  9.453e+01   0.561    0.5748
rho     -2.549e+00  1.376e+00  -1.853    0.0639 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 10.25941
Multiple R-squared: 0.9958804


Elasticities of Substitution:
               Estimate Std. Error t value Pr(>|t|)
E_1_2 (HM)      0.01851    0.03239   0.572    0.568
E_(1,2)_3 (AU)       NA         NA      NA       NA
HM = Hicks-McFadden (direct) elasticity of substitution
AU = Allen-Uzawa (partial) elasticity of substitution
```

Although we set the maximum number of iterations to the maximum possible value (1024), the default convergence criteria are not fulfilled after this number of iterations.[19] While our estimated coefficient λ is not too far away from the corresponding coefficient in Kemfert (1998) ($m_1 = 0.222$), this is not the case for the estimated coefficients ρ_1 and ρ (Kemfert 1998, $\rho_1 = \alpha_1 = 0.5300$, $\rho = \beta_1 = 0.1813$ in). Consequently, our Hicks-McFadden elasticity of substitution between capital and energy considerably deviates from the corresponding elasticity published in Kemfert (1998) ($\sigma_{\alpha_1} = 0.653$). Moreover, coefficient ρ estimated by `cesEst` is not in the economically meaningful range, i.e. contradicts economic theory, so that the elasticities of substitution between capital/energy and labour are not defined. Unfortunately, Kemfert (1998) does not report the estimates of the other coefficients.

For the two other nesting structures, `cesEst` using the Levenberg-Marquardt algorithm reaches a successful convergence, but again the estimated ρ_1 and ρ parameters are far from the estimates reported in Kemfert (1998) and are not in the economically meaningful range.

In order to avoid the problem of economically meaningless estimates, we re-estimate the model with `cesEst` using the PORT algorithm, which allows for box constraints on the parameters. If `cesEst` is called with argument `method` equal to `"PORT"`, the coefficients are constrained to be in the economically meaningful region by default.

```
> cesPort1 <- cesEst("Y", c("K", "E", "A"), tName = "time", data = GermanIndustry,
+     method = "PORT", control = list(iter.max = 1000, eval.max = 2000))
> summary(cesPort1)
```

---

[19]Of course, we can achieve "successful convergence" by increasing the tolerance levels for convergence.

```
Estimated CES function with constant returns to scale

Call:
cesEst(yName = "Y", xNames = c("K", "E", "A"), data = GermanIndustry,
    tName = "time", method = "PORT", control = list(iter.max = 1000,
        eval.max = 2000))

Estimation by non-linear least-squares using the 'PORT' optimizer
assuming an additive error term
Convergence NOT achieved after 418 iterations
Message: false convergence (8)

Coefficients:
          Estimate Std. Error t value Pr(>|t|)
gamma    1.372e+00  1.915e+00   0.717 0.473622
lambda   2.071e-02  5.503e-04  37.633  < 2e-16 ***
delta_1  1.454e-12  2.176e-11   0.067 0.946733
delta    9.728e-01  2.598e-01   3.745 0.000181 ***
rho_1    8.873e+00  5.121e+00   1.733 0.083177 .
rho      7.022e-01  2.441e+00   0.288 0.773630
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.72543
Multiple R-squared: 0.9954976

Elasticities of Substitution:
               Estimate Std. Error t value Pr(>|t|)
E_1_2 (HM)      0.10129    0.05254   1.928   0.0539 .
E_(1,2)_3 (AU)  0.58747    0.84257   0.697   0.4857
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
HM = Hicks-McFadden (direct) elasticity of substitution
AU = Allen-Uzawa (partial) elasticity of substitution
```

As expected, these estimated coefficients are in the economically meaningful region and the fit of the model is worse than for the unrestricted estimation using the Levenberg-Marquardt algorithm (larger standard deviation of the residuals). However, the optimisation algorithm

reports an unsuccessful convergence and estimates of the coefficients and elasticities of substitution are still not close to the estimates reported in Kemfert (1998).[20]

In the following, we restrict the coefficients λ, ρ_1, and ρ to the estimates reported in Kemfert (1998) and only estimate the coefficients that are not reported, i.e. γ, δ_1, and δ. While ρ_1 and ρ_2 can be restricted automatically by arguments `rho1` and `rho` of `cesEst`, we have to impose the restriction on λ manually. As the model estimated by Kemfert (1998) is restricted to have constant returns to scale (i.e. the output is linearly homogeneous in all three inputs), we can simply multiply all inputs by $e^{\lambda t}$. The following commands adjust the three input variables and re-estimate the model with coefficients λ, ρ_1, and ρ restricted to be equal to the estimates reported in Kemfert (1998).

```
> GermanIndustry$K1 <- GermanIndustry$K * exp(0.0222 * GermanIndustry$time)
> GermanIndustry$E1 <- GermanIndustry$E * exp(0.0222 * GermanIndustry$time)
> GermanIndustry$A1 <- GermanIndustry$A * exp(0.0222 * GermanIndustry$time)
> cesLmFixed1 <- cesEst("Y", c("K1", "E1", "A1"), data = GermanIndustry,
+     rho1 = 0.53, rho = 0.1813, control = nls.lm.control(maxiter = 1000,
+         maxfev = 2000))
> summary(cesLmFixed1)


Estimated CES function with constant returns to scale

Call:
cesEst(yName = "Y", xNames = c("K1", "E1", "A1"), data = GermanIndustry,
    rho1 = 0.53, rho = 0.1813, control = nls.lm.control(maxiter = 1000,
        maxfev = 2000))


Estimation by non-linear least-squares using the 'LM' optimizer
assuming an additive error term
Coefficient 'rho_1' was fixed at 0.53
Coefficient 'rho' was fixed at 0.1813
Convergence achieved after 22 iterations
Message: Relative error between `par' and the solution is at most `ptol'.


Coefficients:
        Estimate Std. Error t value Pr(>|t|)
```

---

[20]As we were unable to replicate the results of Kemfert (1998) by assuming an additive error term, we re-estimated the models assuming that the error term was *multiplicative* (i.e. by setting argument `multErr` of function `cesEst` to `TRUE`), because we thought that Kemfert (1998) could have estimated the model in logarithms without reporting this in the paper. However, even when using this method, we could not replicate the estimates reported in Kemfert (1998).

```
gamma     1.494895    6.092736    0.245     0.806
delta_1 -0.003031    0.035786   -0.085     0.933
delta     0.884490    1.680111    0.526     0.599
rho_1     0.530000    5.165851    0.103     0.918
rho       0.181300    4.075280    0.044     0.965


Residual standard error: 12.72876
Multiple R-squared: 0.9936586


Elasticities of Substitution:
              Estimate Std. Error t value Pr(>|t|)
E_1_2 (HM)      0.6536     2.2068   0.296    0.767
E_(1,2)_3 (AU)  0.8465     2.9204   0.290    0.772
HM = Hicks-McFadden (direct) elasticity of substitution
AU = Allen-Uzawa (partial) elasticity of substitution
```

The fit of this model is—of course—even worse than the fit of the two previous models. Given that ρ_1 and ρ are identical to the corresponding parameters published in Kemfert (1998), the reported elasticities of substitution are also equal to the values published in Kemfert (1998). Surprisingly, the R^2 value of our restricted estimation is not equal to the R^2 value reported in Kemfert (1998), although the coefficients λ, ρ_1, and ρ are exactly the same. Moreover, the coefficient δ_1 is not in the economically meaningful range. For the two other nesting structures, the R^2 values strongly deviate from the values reported in Kemfert (1998), as our R^2 value is much larger for one nesting structure, whilst it is considerably smaller for the other.

Given our unsuccessful attempts to reproduce the results of Kemfert (1998) and the convergence problems in our estimations above, we systematically re-estimated the models of Kemfert (1998) using `cesEst` with many different estimation methods:

- all gradient-based algorithms for unrestricted optimisation (Newton, BFGS, LM)

- all gradient-based algorithms for restricted optimisation (L-BFGS-B, PORT)

- all global algorithms (NM, SANN, DE)

- one gradient-based algorithms for unrestricted optimisation (LM) with starting values equal to the estimates from the global algorithms

- one gradient-based algorithms for restricted optimisation (PORT) with starting values equal to the estimates from the global algorithms

- two-dimensional grid search for ρ_1 and ρ using all gradient-based algorithms (Newton, BFGS, L-BFGS-B, LM, PORT)[21]

- all gradient-based algorithms (Newton, BFGS, L-BFGS-B, LM, PORT) with starting values equal to the estimates from the corresponding grid searches.

For these estimations, we changed the following control parameters of the optimisation algorithms:

- Newton: `iterlim = 500` (max. 500 iterations)

- BFGS: `maxit = 5000` (max. 5,000 iterations)

- L-BFGS-B: `maxit = 5000` (max. 5,000 iterations)

- Levenberg-Marquardt: `maxiter = 1000` (max. 1,000 iterations), `maxfev = 2000` (max. 2,000 function evaluations)

- PORT: `iter.max = 1000` (max. 1,000 iterations), `eval.max = 1000` (max. 1,000 function evaluations)

- Nelder-Nead: `maxit = 5000` (max. 5,000 iterations)

- SANN: `maxit = 2e6` (2,000,000 iterations)

- DE: `NP = 500` (500 population members), `itermax = 1e4` (max. 10,000 population generations)

The results of all these estimation approaches for the three different nesting structures are summarised in tables 1, 2, and 3, respectively.

The models with the best fit, i.e. with the smallest sums of squared residuals, are always obtained by the Levenberg-Marquardt algorithm—either with the standard starting values (third nesting structure), or with starting values taken from global algorithms (second and third nesting structure) or grid search (first and third nesting structure). However, all estimates that give the best fit are economically meaningless, because either coefficient ρ_1 or coefficient ρ is less than than minus one. Assuming that the basic assumptions of economic production theory are satisfied in reality, these results suggest that the three-input nested CES production technology is a poor approximation of the true production technology, or that the data are not reasonably constructed (e.g. problems with aggregation or deflation).

---

[21]The two-dimensional grid search included 61 different values for ρ_1 and ρ: -1.0, -0.9, -0.8, -0.7, -0.6, -0.5, -0.4, -0.3, -0.2, -0.1, 0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.2, 1.4, 1.6, 1.8, 2.0, 2.2, 2.4, 2.6, 2.8, 3.0, 3.2, 3.4, 3.6, 3.8, 4.0, 4.4, 4.8, 5.2, 5.6, 6.0, 6.4, 6.8, 7.2, 7.6, 8.0, 8.4, 8.8, 9.2, 9.6, 10.0, 10.4, 10.8, 11.2, 11.6, 12.0, 12.4, 12.8, 13.2, 13.6, and 14.0. Hence, each grid search included $61^2 = 3721$ estimations.

Table 1: Estimation results with first nesting structure

	γ	λ	δ_1	δ	ρ_1	ρ	c	RSS	R^2
Kemfert (1998)		0.0222			0.5300	0.1813			0.9996
fixed	1.4949	0.0222	-0.0030	0.8845	0.5300	0.1813	1	5023	0.9937
Newton	14.0920	0.0150	0.0210	0.9872	10.6204	3.2078	1	14969	0.9811
BFGS	2.2192	0.0206	0.0000	0.8245	7.2704	0.2045	1	3608	0.9954
L-BFGS-B	14.1386	0.0150	0.0236	0.9870	10.3934	3.2078	1	14969	0.9811
PORT	1.3720	0.0207	0.0000	0.9728	8.8728	0.7022	0	3566	0.9955
LM	28.9700	0.0210	0.0000	0.0001	51.9850	-2.5421	0	3264	0.9959
NM	3.3841	0.0203	0.0000	0.5940	3.4882	-0.1233	1	4043	0.9949
NM - PORT	3.9731	0.0207	0.0000	0.5373	8.6487	-0.1470	0	3542	0.9955
NM - LM	28.2138	0.0210	0.0000	0.0002	52.9656	-2.3677	0	3266	0.9959
SANN	16.3982	0.0126	0.8310	0.9508	0.4582	2.3205		16029	0.9798
SANN - PORT	3.6382	0.0120	0.7807	1.0000	-0.9135	9.4234	0	9162	0.9884
SANN - LM	6.3870	0.0118	0.9576	1.0000	-1.4668	7.1791	1	10794	0.9864
DE	13.8379	0.0143	0.9780	0.9955	-0.9528	3.8727		14075	0.9822
DE - PORT	3.9423	0.0119	0.8187	1.0000	-0.9901	9.2479	0	9299	0.9883
DE - LM	6.0900	0.0118	0.9517	1.0000	-1.4400	7.5447	1	10611	0.9866
Newton grid	2.5797	0.0206	0.0000	0.7565	6.8000	0.1000	1	3637	0.9954
Newton grid start	2.5797	0.0206	0.0000	0.7565	6.8000	0.1000	0	3637	0.9954
BFGS grid	11.1502	0.0208	0.0000	0.1078	11.6000	-0.7000	1	3465	0.9956
BFGS grid start	11.1502	0.0208	0.0000	0.1078	11.6000	-0.7000	1	3465	0.9956
PORT grid	4.3205	0.0208	0.0000	0.4887	11.6000	-0.2000	1	3469	0.9956
PORT grid start	4.3205	0.0208	0.0000	0.4887	11.6000	-0.2000	0	3469	0.9956
LM grid	15.1241	0.0209	0.0000	0.0370	14.0000	-1.0000	1	3416	0.9957
LM grid start	28.8340	0.0210	0.0000	0.0001	63.9347	-2.4969	0	3259	0.9959

Note: in the column titled "c", a "1" indicates that the optimisation algorithm reports a successful convergence, whereas a "0" indicates that the optimisation algorithm reports non-convergence. The row entitled "fixed" presents the results when λ, ρ_1, and ρ are restricted to have exactly the same values that are published in Kemfert (1998). The rows titled "<globAlg> - <gradAlg>" present the results when the model is first estimated by the global algorithm "<globAlg>" and then estimated by the gradient-based algorithm "<gradAlg>" using the estimates of the first step as starting values. The rows entitled "<gradAlg> grid" present the results when a two-dimensional grid search for ρ_1 and ρ was performed using the gradient-based algorithm "<gradAlg>" at each grid point. The rows entitled "<gradAlg> grid start" present the results when the gradient-based algorithm "<gradAlg>" was used with starting values equal to the results of a two-dimensional grid search for ρ_1 and ρ using the same gradient-based algorithm at each grid point. The rows highlighted with in red include estimates that are economically meaningless.

If we only consider estimates that are economically meaningful, the models with the best fit are obtained by grid-search with the Levenberg-Marquardt algorithm (first and second nesting structure), grid-search with the PORT or BFGS algorithm (second nesting structure), the PORT algorithm with default starting values (second nesting structure), or the PORT algorithm with starting values taken from global algorithms or grid search (second and third nesting structure).

Hence, we can conclude that the Levenberg-Marquardt and the PORT algorithms are—at least in this study—most likely to find the coefficients that give the best fit to the model, where

Table 2: Estimation results with second nesting structure

	γ	λ	δ_1	δ	ρ_1	ρ	c	RSS	R^2
Kemfert (1998)		0.0069			0.2155	1.1816			0.7860
fixed	1.7211	0.0069	0.6826	0.0249	0.2155	1.1816	1	14146	0.9821
Newton	6.0780	0.0207	0.9910	0.8613	5.5056	-0.7743	1	3858	0.9951
BFGS	12.3334	0.0207	0.9919	0.9984	5.6353	-2.2167	1	3778	0.9952
L-BFGS-B	6.2382	0.0208	0.9940	0.8741	5.9278	-0.8120	1	3857	0.9951
PORT	7.5015	0.0207	0.9892	0.9291	5.3101	-1.0000	1	3844	0.9951
LM	16.1034	0.0208	0.9942	1.0000	6.0167	-5.0997	1	3646	0.9954
NM	9.9466	0.0182	0.4188	0.9219	0.6891	-0.8530	1	5342	0.9933
NM - PORT	7.5015	0.0207	0.9892	0.9291	5.3101	-1.0000	1	3844	0.9951
NM - LM	16.2841	0.0208	0.9942	1.0000	6.0026	-5.3761	1	3636	0.9954
SANN	5.6749	0.0194	0.3052	0.7889	0.5208	-0.6612		4740	0.9940
SANN - PORT	7.5243	0.0207	0.9886	0.9293	5.2543	-1.0000	1	3844	0.9951
SANN - LM	16.2545	0.0208	0.9942	1.0000	6.0065	-5.3299	1	3638	0.9954
DE	5.8315	0.0206	0.9798	0.8427	4.6806	-0.7330		3875	0.9951
DE - PORT	7.5015	0.0207	0.9892	0.9291	5.3101	-1.0000	1	3844	0.9951
DE - LM	16.2132	0.0208	0.9942	1.0000	6.0133	-5.2680	1	3640	0.9954
Newton grid	2.3534	0.0203	0.9555	0.2895	3.8000	0.1000	1	3925	0.9950
Newton grid start	9.2995	0.0207	0.9885	0.9748	5.2412	-1.3350	0	3826	0.9952
BFGS grid	7.5434	0.0207	0.9880	0.9296	5.2000	-1.0000	1	3844	0.9951
BFGS grid start	7.5434	0.0207	0.9880	0.9296	5.2000	-1.0000	1	3844	0.9951
PORT grid	7.5434	0.0207	0.9880	0.9296	5.2000	-1.0000	1	3844	0.9951
PORT grid start	7.5015	0.0207	0.9892	0.9291	5.3101	-1.0000	1	3844	0.9951
LM grid	7.5434	0.0207	0.9880	0.9296	5.2000	-1.0000	1	3844	0.9951
LM grid start	16.1869	0.0208	0.9942	1.0000	6.0112	-5.2248	1	3642	0.9954

See note below figure 1.

the PORT algorithm can be used to restrict the estimates to the economically meaningful region.[22]

Given the considerable variation of the estimation results, we will examine the results of the grid searches for ρ_1 and ρ, as this will help us to understand the problems that the optimisation algorithms have when finding the minimum of the sum of squared residuals. We demonstrate this with the first nesting structure and the Levenberg-Marquardt algorithm using the same values for ρ_1 and ρ and the same control parameters as before. Hence, we get the same results as shown in table 1:

```
> rhoVec <- c(seq(-1, 1, 0.1), seq(1.2, 4, 0.2), seq(4.4, 14, 0.4))
> cesLmGridRho1 <- cesEst("Y", xNames1, tName = "time", data = GermanIndustry,
+      rho1 = rhoVec, rho = rhoVec, control = nls.lm.control(maxiter = 1000,
+          maxfev = 2000))

> print(cesLmGridRho1)
```

---

[22]Please note that the grid-search with an algorithm for unrestricted optimisation (e.g. Levenberg-Marquardt) can be used to guarantee economically meaningful values for ρ_1 and ρ, but not for γ, δ_1, δ, and ν.

Table 3: Estimation results with third nesting structure

	γ	λ	δ_1	δ	ρ_1	ρ	c	RSS	R^2
Kemfert (1998)		0.0064			1.3654	5.8327			0.9986
fixed	17.5964	0.0064	-30.8482	0.0000	1.3654	5.8327	0	72628	0.9083
Newton	9.6767	0.0196	0.1058	0.9534	-0.7944	1.2220	1	4516	0.9943
BFGS	5.5155	0.0220	0.2490	1.0068	-0.5898	-1.7305	1	4955	0.9937
L-BFGS-B	9.6353	0.0208	0.1487	0.9967	-0.6131	7.6653	1	3537	0.9955
PORT	13.8498	0.0208	0.0555	0.9437	-0.8789	7.5285	0	3518	0.9956
LM	19.7167	0.0207	0.0000	0.0085	-6.0497	6.9686	1	3357	0.9958
NM	3.5709	0.0209	0.5834	1.0000	-0.1072	8.7042	1	3582	0.9955
NM - PORT	15.5533	0.0208	0.0345	0.8632	-1.0000	7.4646	1	3510	0.9956
NM - LM	19.7167	0.0207	0.0000	0.0085	-6.0497	6.9686	1	3357	0.9958
SANN	9.3619	0.0192	0.1286	0.9440	-0.7174	1.2177		4596	0.9942
SANN - PORT	15.5533	0.0208	0.0345	0.8632	-1.0000	7.4646	1	3510	0.9956
SANN - LM	19.7198	0.0207	0.0000	0.0086	-6.0156	6.9714	0	3357	0.9958
DE	15.3492	0.0206	0.0480	0.8617	-0.8745	6.7038		3588	0.9955
DE - PORT	15.5533	0.0208	0.0345	0.8632	-1.0000	7.4646	1	3510	0.9956
DE - LM	19.7169	0.0207	0.0000	0.0085	-6.0482	6.9687	1	3357	0.9958
Newton grid	7.5286	0.0209	0.2264	0.9997	-0.5000	8.4000	1	3553	0.9955
Newton grid start	7.5303	0.0209	0.2316	0.9997	-0.4859	8.3998	1	3551	0.9955
BFGS grid	10.9258	0.0208	0.1100	0.9924	-0.7000	8.0000	1	3532	0.9955
BFGS grid start	10.9258	0.0208	0.1100	0.9924	-0.7000	8.0000	1	3532	0.9955
PORT grid	11.0898	0.0208	0.1082	0.9899	-0.7000	7.6000	1	3531	0.9955
PORT grid start	15.5533	0.0208	0.0345	0.8632	-1.0000	7.4646	1	3510	0.9956
LM grid	12.9917	0.0207	0.0724	0.9580	-0.8000	6.8000	1	3528	0.9955
LM grid start	19.7169	0.0207	0.0000	0.0085	-6.0485	6.9687	1	3357	0.9958

See note below figure 1.

```
Estimated CES function


Call:
cesEst(yName = "Y", xNames = xNames1, data = GermanIndustry,
    tName = "time", rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
    control = nls.lm.control(maxiter = 1000, maxfev = 2000))


Coefficients:
    gamma     lambda    delta_1     delta     rho_1        rho
 1.512e+01  2.087e-02  6.045e-19  3.696e-02  1.400e+01 -1.000e+00


Elasticities of Substitution:
    E_1_2 (HM) E_(1,2)_3 (AU)
       0.06667            Inf
AU = Allen-Uzawa (partial) elasticity of substitution
HM = Hicks-McFadden (direct) elasticity of substitution
```

We start our analysis of the results from the grid search by applying the standard `plot` method.

```
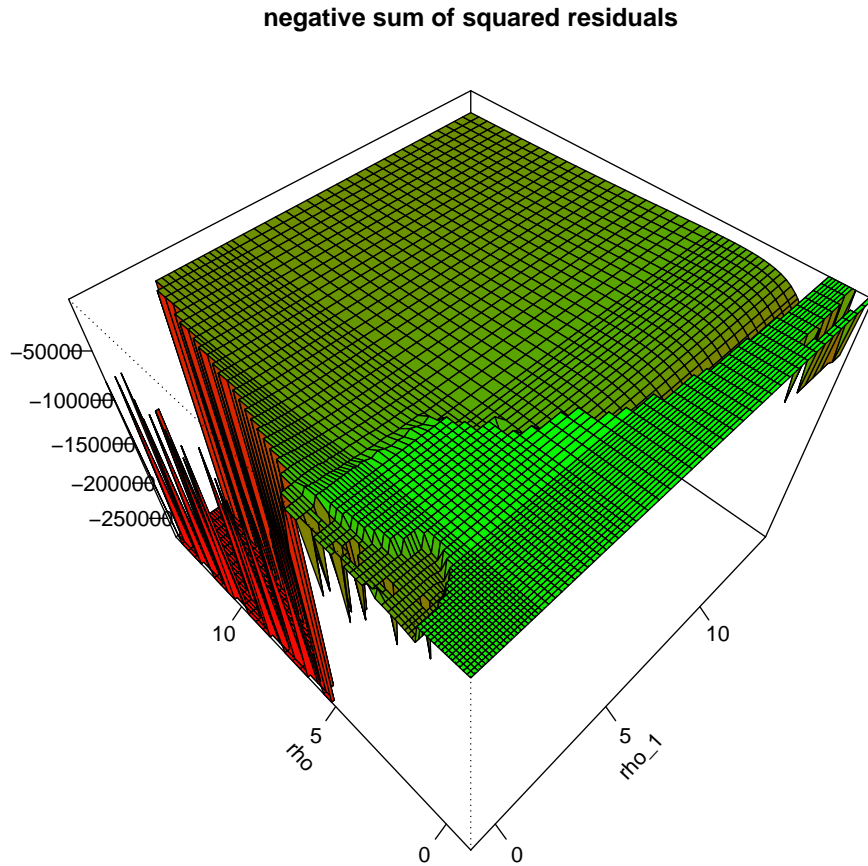> plot(cesLmGridRho1)
```

negative sum of squared residuals

Figure 6: Goodness of fit for different values of ρ_1 and ρ

As it is much easier to spot the summit of a hill than the deepest part of a valley (e.g. because the deepest part could be hidden behind a ridge), the `plot` method plots the *negative* sum of squared residuals against ρ_1 and ρ. This is shown in figure 6. This figure indicates that the surface is not always smooth. The fit of the model is best (small sum of squared residuals, light green colour) if ρ is approximately in the interval $[-1, 1]$ (no matter the value of ρ_1) or if ρ_1 is approximately in the interval $[2, 7]$ and ρ is smaller than 7. Furthermore, the fit is worst (large sum of squared residuals, red colour) if ρ is larger than 5 and ρ_1 has a low value (the upper limit is between -0.8 and 2 depending the value of ρ).

In order to get a more detailed insight into the best-fitting region, we re-plot figure 6 with only sums of squared residuals that are smaller than 5,000.

```
> cesLmGridRho1a <- cesLmGridRho1
> cesLmGridRho1a$rssArray[cesLmGridRho1a$rssArray >= 5000] <- NA
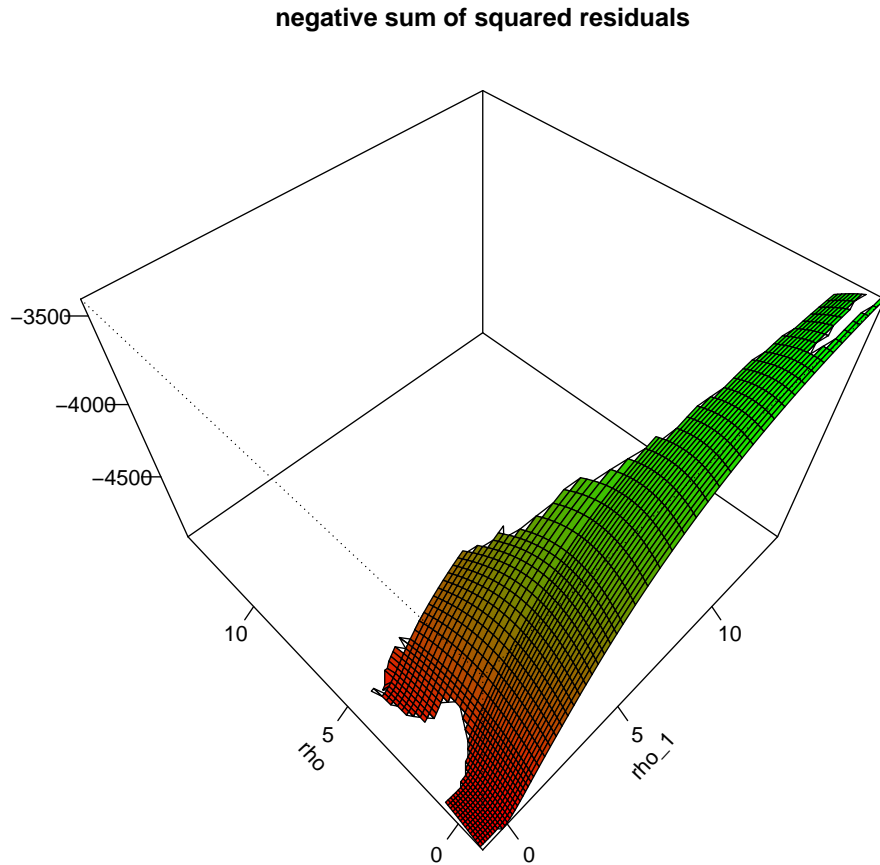> plot(cesLmGridRho1a)
```

negative sum of squared residuals

Figure 7: Goodness of fit (best-fitting region only)

The resulting figure 7 shows that the fit of the model clearly improves with increasing ρ_1 and decreasing ρ. At the estimates of Kemfert (1998), i.e. $\rho_1 = 0.53$ and $\rho = 0.1813$, the sum of the squared residuals is clearly not at its minimum. We obtained similar results for the other two nesting structures. We also replicated the estimations for seven industrial sectors, but again, we could not reproduce any of the estimates published in Kemfert (1998). We contacted the author and asked her to help us to identify the reasons for the differences between her results and ours. Unfortunately, both the Shazam scripts used for the estimations and the corresponding output files have been lost. Hence, we were unable to find the reason for the large deviations in the estimates. However, we are confident that the results obtained by cesEst are correct, i.e. correspond to the smallest sums of squared residuals.

6. Conclusion

In recent years, the CES function has gained in popularity in macroeconomics and especially growth theory, as it is clearly more flexible than the classical Cobb-Douglas function. As the CES function is not easy to estimate, given an objective function that is seldom well-behaved, a software solution to estimate the CES function may further increase its popularity.

The **micEconCES** package provides such a solution. Its function `cesEst` can not only estimate the traditional two-input CES function, but also all major extensions, i.e. technological change and three-input and four-input nested CES functions. Furthermore, the **micEconCES** package provides the user with a multitude of estimation and optimisation procedures, which include the linear approximation suggested by Kmenta (1967), gradient-based and global optimisation algorithms, and an extended grid-search procedure that returns stable results and alleviates convergence problems. Additionally, the user can impose restrictions on the parameters to enforce economically meaningful parameter estimates.

The function `cesEst` is constructed in a way that allows the user to switch easily between different estimation and optimisation procedures. Hence, the user can easily use several different methods and compare the results. The grid search procedure, in particular, increases the probability of finding a global optimum. Additionally, the grid search allows one to plot the objective function for different values of the substitution parameters (ρ_1, ρ_2, ρ). In doing so, the user can visualise the surface of the objective function to be minimised and, hence, check the extent to which the objective function is well behaved and which parameter ranges of the substitution parameters give the best fit to the model. This option is a further control instrument to ensure that the global optimum has been reached. Section 5.2 demonstrates how these control instruments support the identification of the global optimum when a simple non-linear estimation, in contrast, fails.

The **micEconCES** package is open-source software and modularly programmed, which makes it easy for the user to extend and modify (e.g. including factor augmented technological change). However, even if some readers choose not to use the **micEconCES** package for estimating a CES function, they will definitely benefit from this paper, as it provides a multitude of insights and practical hints regarding the estimation of CES functions.

A. Traditional two-input CES function

A.1. Derivation of the Kmenta approximation

This derivation of the first-order Taylor series (Kmenta) approximation of the traditional two-input CES function is based on Uebe (2000).

Traditional two-input CES function:

$$y \;=\; \gamma \, e^{\lambda t} \left(\delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \tag{29}$$

Logarithmized CES function:

$$\ln y \;=\; \ln \gamma + \lambda \, t - \frac{\nu}{\rho} \ln \left(\delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho} \right) \tag{30}$$

Define function

$$f(\rho) \;\equiv\; -\frac{\nu}{\rho} \ln \left(\delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho} \right) \tag{31}$$

so that

$$\ln y \;=\; \ln \gamma + \lambda \, t + f(\rho) . \tag{32}$$

Now we can approximate the logarithm of the CES function by a first-order Taylor series approximation around $\rho = 0$:

$$\ln y \approx \ln \gamma + \lambda \, t + f(0) + \rho f'(0) \tag{33}$$

We define function

$$g(\rho) \;\equiv\; \delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho} \tag{34}$$

so that

$$f(\rho) \;=\; -\frac{\nu}{\rho} \ln \left(g(\rho) \right) . \tag{35}$$

Now we can calculate the first partial derivative of $f(\rho)$:

$$f'(\rho) \;=\; \frac{\nu}{\rho^2} \ln \left(g(\rho) \right) - \frac{\nu}{\rho} \frac{g'(\rho)}{g(\rho)} \tag{36}$$

and the first three derivatives of $g(\rho)$

$$
\begin{align}
g'(\rho) &= -\delta\, x_1^{-\rho}\, \ln x_1 - (1-\delta)\, x_2^{-\rho}\, \ln x_2 \tag{37} \\
g''(\rho) &= \delta\, x_1^{-\rho}\, (\ln x_1)^2 + (1-\delta)\, x_2^{-\rho}\, (\ln x_2)^2 \tag{38} \\
g'''(\rho) &= -\delta\, x_1^{-\rho}\, (\ln x_1)^3 - (1-\delta)\, x_2^{-\rho}\, (\ln x_2)^3. \tag{39}
\end{align}
$$

At the point of approximation $\rho = 0$ we have

$$
\begin{align}
g(0) &= 1 \tag{40} \\
g'(0) &= -\delta\, \ln x_1 - (1-\delta)\, \ln x_2 \tag{41} \\
g''(0) &= \delta\, (\ln x_1)^2 + (1-\delta)\, (\ln x_2)^2 \tag{42} \\
g'''(0) &= -\delta\, (\ln x_1)^3 - (1-\delta)\, (\ln x_2)^3 \tag{43}
\end{align}
$$

Now we calculate the limit of $f(\rho)$ for $\rho \to 0$:

$$
\begin{align}
f(0) &= \lim_{\rho \to 0} f(\rho) \tag{44} \\
&= \lim_{\rho \to 0} \frac{-\nu \ln(g(\rho))}{\rho} \tag{45} \\
&= \lim_{\rho \to 0} \frac{-\nu \frac{g'(\rho)}{g(\rho)}}{1} \tag{46} \\
&= \nu\, (\delta\, \ln x_1 + (1-\delta)\, \ln x_2) \tag{47}
\end{align}
$$

and the limit of $f'(\rho)$ for $\rho \to 0$:

$$
\begin{align}
f'(0) &= \lim_{\rho \to 0} f'(\rho) \tag{48} \\
&= \lim_{\rho \to 0} \left(\frac{\nu}{\rho^2} \ln(g(\rho)) - \frac{\nu}{\rho} \frac{g'(\rho)}{g(\rho)} \right) \tag{49} \\
&= \lim_{\rho \to 0} \frac{\nu \ln(g(\rho)) - \nu\, \rho\, \frac{g'(\rho)}{g(\rho)}}{\rho^2} \tag{50} \\
&= \lim_{\rho \to 0} \frac{\nu \frac{g'(\rho)}{g(\rho)} - \nu \frac{g'(\rho)}{g(\rho)} - \nu\, \rho\, \frac{g''(\rho)g(\rho)-(g'(\rho))^2}{(g(\rho))^2}}{2\rho} \tag{51} \\
&= \lim_{\rho \to 0} -\frac{\nu}{2} \frac{g''(\rho)\, g(\rho) - (g'(\rho))^2}{(g(\rho))^2} \tag{52} \\
&= -\frac{\nu}{2} \frac{g''(0)\, g(0) - (g'(0))^2}{(g(0))^2} \tag{53} \\
&= -\frac{\nu}{2} \left(\delta\, (\ln x_1)^2 + (1-\delta)\, (\ln x_2)^2 - (-\delta\, \ln x_1 - (1-\delta)\, \ln x_2)^2 \right) \tag{54} \\
&= -\frac{\nu}{2} \Big(\delta\, (\ln x_1)^2 + (1-\delta)\, (\ln x_2)^2 - \delta^2\, (\ln x_1)^2 \tag{55}
\end{align}
$$

$$-2\delta\left(1-\delta\right)\ln x_1 \ln x_2 - \left(1-\delta\right)^2 \left(\ln x_2\right)^2 \Big)$$

$$= -\frac{\nu}{2}\left(\left(\delta-\delta^2\right)\left(\ln x_1\right)^2 + \left(\left(1-\delta\right)-\left(1-\delta\right)^2\right)\left(\ln x_2\right)^2 \right. \tag{56}$$

$$\left. -2\delta\left(1-\delta\right)\ln x_1 \ln x_2 \right)$$

$$= -\frac{\nu}{2}\left(\delta\left(1-\delta\right)\left(\ln x_1\right)^2 + \left(1-\delta\right)\left(1-\left(1-\delta\right)\right)\left(\ln x_2\right)^2 \right. \tag{57}$$

$$\left. -2\delta\left(1-\delta\right)\ln x_1 \ln x_2 \right)$$

$$= -\frac{\nu\delta\left(1-\delta\right)}{2}\left(\left(\ln x_1\right)^2 - 2\ln x_1 \ln x_2 + \left(\ln x_2\right)^2\right) \tag{58}$$

$$= -\frac{\nu\delta\left(1-\delta\right)}{2}\left(\ln x_1 - \ln x_2\right)^2 \tag{59}$$

so that we get following first-order Taylor series approximation around $\rho = 0$:

$$\ln y \approx \ln\gamma + \lambda t + \nu\delta\ln x_1 + \nu\left(1-\delta\right)\ln x_2 - \nu\rho\,\delta\left(1-\delta\right)\left(\ln x_1 - \ln x_2\right)^2 \tag{60}$$

A.2. Derivatives with respect to coefficients

The partial derivatives of the two-input CES function with respect to all coefficients are shown in the main paper in equations 15 to 19. The first-order Taylor series approximations of these derivatives at the point $\rho = 0$ are shown in the main paper in equations 20 to 23. The derivations of these Taylor series approximations are shown in the following. These calculations are inspired by Uebe (2000). Functions $f(\rho)$ and $g(\rho)$ are defined as in the previous section (in equations 31 and 34, respectively).

Derivatives with respect to Gamma

$$\frac{\partial y}{\partial\gamma} = e^{\lambda t}\left(\delta x_1^{-\rho} + \left(1-\delta\right)x_2^{-\rho}\right)^{-\frac{\nu}{\rho}} \tag{61}$$

$$= e^{\lambda t}\exp\left(-\frac{\nu}{\rho}\ln\left(\delta x_1^{-\rho} + \left(1-\delta\right)x_2^{-\rho}\right)\right) \tag{62}$$

$$= e^{\lambda t}\exp\left(f\left(\rho\right)\right) \tag{63}$$

$$\approx e^{\lambda t}\exp\left(f\left(0\right) + \rho f'\left(0\right)\right) \tag{64}$$

$$= e^{\lambda t}\exp\left(\nu\delta\ln x_1 + \nu\left(1-\delta\right)\ln x_2 - \frac{1}{2}\nu\rho\,\delta\left(1-\delta\right)\left(\ln x_1 - \ln x_2\right)^2\right) \tag{65}$$

$$= e^{\lambda t}x_1^{\nu\delta}x_2^{\nu(1-\delta)}\exp\left(-\frac{1}{2}\nu\rho\,\delta\left(1-\delta\right)\left(\ln x_1 - \ln x_2\right)^2\right) \tag{66}$$

Derivatives with respect to Delta

$$\frac{\partial y}{\partial \delta} = -\gamma \, e^{\lambda t} \, \frac{\nu}{\rho} \left(\delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho}\right)^{-\frac{\nu}{\rho} - 1} \left(x_1^{-\rho} - x_2^{-\rho}\right) \tag{67}$$

$$= -\gamma \, e^{\lambda t} \, \nu \, \frac{x_1^{-\rho} - x_2^{-\rho}}{\rho} \left(\delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho}\right)^{-\frac{\nu}{\rho} - 1} \tag{68}$$

Now we define the function $f_\delta(\rho)$

$$f_\delta(\rho) = \frac{x_1^{-\rho} - x_2^{-\rho}}{\rho} \left(\delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho}\right)^{-\frac{\nu}{\rho} - 1} \tag{69}$$

$$= \frac{x_1^{-\rho} - x_2^{-\rho}}{\rho} \exp\left(-\left(\frac{\nu}{\rho} + 1\right) \ln \left(\delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho}\right)\right) \tag{70}$$

so that we can approximate $\partial y / \partial \delta$ by using the first-order Taylor series approximation of $f_\delta(\rho)$:

$$\frac{\partial y}{\partial \delta} = -\gamma \, e^{\lambda t} \, \nu \, f_\delta(\rho) \tag{71}$$

$$\approx -\gamma \, e^{\lambda t} \, \nu \, \left(f_\delta(0) + \rho f_\delta'(0)\right) \tag{72}$$

Now we define the helper functions $g_\delta(\rho)$ and $h_\delta(\rho)$

$$g_\delta(\rho) = \left(\frac{\nu}{\rho} + 1\right) \ln \left(\delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho}\right) \tag{73}$$

$$= \left(\frac{\nu}{\rho} + 1\right) \ln \left(g(\rho)\right) \tag{74}$$

$$h_\delta(\rho) = \frac{x_1^{-\rho} - x_2^{-\rho}}{\rho} \tag{75}$$

with first derivatives

$$g_\delta'(\rho) = -\frac{\nu}{\rho^2} \ln \left(g(\rho)\right) + \left(\frac{\nu}{\rho} + 1\right) \frac{g'(\rho)}{g(\rho)} \tag{76}$$

$$h_\delta'(\rho) = \frac{-\rho \left(\ln x_1 \, x_1^{-\rho} - \ln x_2 \, x_2^{-\rho}\right) - x_1^{-\rho} + x_2^{-\rho}}{\rho^2} \tag{77}$$

so that

$$f_\delta(\rho) = h_\delta(\rho) \, \exp\left(-g_\delta(\rho)\right) \tag{78}$$

and

$$f_\delta'(\rho) = h_\delta'(\rho) \, \exp\left(-g_\delta(\rho)\right) - h_\delta(\rho) \, \exp\left(-g_\delta(\rho)\right) g_\delta'(\rho) \tag{79}$$

Now we can calculate the limits of $g_\delta(\rho)$, $g'_\delta(\rho)$, $h_\delta(\rho)$, and $h'_\delta(\rho)$ for $\rho \to 0$ by

$$g_\delta(0) = \lim_{\rho \to 0} g_\delta(\rho) \tag{80}$$

$$= \lim_{\rho \to 0} \left(\left(\frac{\nu}{\rho} + 1 \right) \ln (g(\rho)) \right) \tag{81}$$

$$= \lim_{\rho \to 0} \frac{(\nu + \rho) \ln (g(\rho))}{\rho} \tag{82}$$

$$= \lim_{\rho \to 0} \frac{\ln (g(\rho)) + (\nu + \rho) \frac{g'(\rho)}{g(\rho)}}{1} \tag{83}$$

$$= \ln (g(0)) + \nu \frac{g'(0)}{g(0)} \tag{84}$$

$$= -\nu \delta \ln x_1 - \nu (1 - \delta) \ln x_2 \tag{85}$$

$$g'_\delta(0) = \lim_{\rho \to 0} \frac{-\nu \ln (g(\rho)) + \rho (\nu + \rho) \frac{g'(\rho)}{g(\rho)}}{\rho^2} \tag{86}$$

$$= \lim_{\rho \to 0} \frac{-\nu \frac{g'(\rho)}{g(\rho)} + (\nu + \rho) \frac{g'(\rho)}{g(\rho)} + \rho \frac{g'(\rho)}{g(\rho)} + \rho (\nu + \rho) \frac{g''(\rho)g(\rho) - (g'(\rho))^2}{(g(\rho))^2}}{2\rho} \tag{87}$$

$$= \lim_{\rho \to 0} \left(\frac{-\nu \frac{g'(\rho)}{g(\rho)} + \nu \frac{g'(\rho)}{g(\rho)} + \rho \frac{g'(\rho)}{g(\rho)} + \rho \frac{g'(\rho)}{g(\rho)} + \rho (\nu + \rho) \frac{g''(\rho)g(\rho) - (g'(\rho))^2}{(g(\rho))^2}}{2\rho} \right) \tag{88}$$

$$= \lim_{\rho \to 0} \left(\frac{g'(\rho)}{g(\rho)} + \frac{1}{2} (\nu + \rho) \frac{g''(\rho) g(\rho) - (g'(\rho))^2}{(g(\rho))^2} \right) \tag{89}$$

$$= \frac{g'(0)}{g(0)} + \frac{1}{2} \nu \frac{g''(0) g(0) - (g'(0))^2}{(g(0))^2} \tag{90}$$

$$= -\delta \ln x_1 - (1 - \delta) \ln x_2 + \frac{\nu \delta (1 - \delta)}{2} (\ln x_1 - \ln x_2)^2 \tag{91}$$

$$h_\delta(0) = \lim_{\rho \to 0} \frac{x_1^{-\rho} - x_2^{-\rho}}{\rho} \tag{92}$$

$$= \lim_{\rho \to 0} \frac{-\ln x_1 \, x_1^{-\rho} + \ln x_2 \, x_2^{-\rho}}{1} \tag{93}$$

$$= -\ln x_1 + \ln x_2 \tag{94}$$

$$h'_\delta(0) = \lim_{\rho \to 0} \frac{-\rho \left(\ln x_1 \, x_1^{-\rho} - \ln x_2 \, x_2^{-\rho} \right) - x_1^{-\rho} + x_2^{-\rho}}{\rho^2} \tag{95}$$

$$= \lim_{\rho \to 0} \left(\frac{-\left(\ln x_1 \, x_1^{-\rho} - \ln x_2 \, x_2^{-\rho} \right) + \rho \left((\ln x_1)^2 \, x_1^{-\rho} - (\ln x_2)^2 \, x_2^{-\rho} \right)}{2\rho} \right) \tag{96}$$

$$+\frac{\ln x_1\, x_1^{-\rho} - \ln x_2\, x_2^{-\rho}}{2\rho}\Bigg)$$

$$= \lim_{\rho\to 0} \frac{1}{2}\left((\ln x_1)^2\, x_1^{-\rho} - (\ln x_2)^2\, x_2^{-\rho}\right) \tag{97}$$

$$= \frac{1}{2}\left((\ln x_1)^2 - (\ln x_2)^2\right) \tag{98}$$

so that we can calculate the limit of $f_\delta(\rho)$ and $f_\delta'(\rho)$ for $\rho \to 0$ by

$$f_\delta(0) = \lim_{\rho\to 0} f_\delta(\rho) \tag{99}$$

$$= \lim_{\rho\to 0}\left(h_\delta(\rho)\exp(-g_\delta(\rho))\right) \tag{100}$$

$$= \lim_{\rho\to 0} h_\delta(\rho)\,\lim_{\rho\to 0}\exp(-g_\delta(\rho)) \tag{101}$$

$$= \lim_{\rho\to 0} h_\delta(\rho)\,\exp\left(-\lim_{\rho\to 0} g_\delta(\rho)\right) \tag{102}$$

$$= h_\delta(0)\,\exp(-g_\delta(0)) \tag{103}$$

$$= (-\ln x_1 + \ln x_2)\exp(\nu\delta\ln x_1 + \nu(1-\delta)\ln x_2) \tag{104}$$

$$= (-\ln x_1 + \ln x_2)\, x_1^{\nu\delta} x_2^{\nu(1-\delta)} \tag{105}$$

$$f_\delta'(0) = \lim_{\rho\to 0} f_\delta'(\rho) \tag{106}$$

$$= \lim_{\rho\to 0}\left(h_\delta'(\rho)\exp(-g_\delta(\rho)) - h_\delta(\rho)\exp(-g_\delta(\rho))\,g_\delta'(\rho)\right) \tag{107}$$

$$= \lim_{\rho\to 0} h_\delta'(\rho)\,\lim_{\rho\to 0}\exp(-g_\delta(\rho))$$
$$\quad - \lim_{\rho\to 0} h_\delta(\rho)\,\lim_{\rho\to 0}\exp(-g_\delta(\rho))\,\lim_{\rho\to 0} g_\delta'(\rho) \tag{108}$$

$$= \lim_{\rho\to 0} h_\delta'(\rho)\,\exp\left(-\lim_{\rho\to 0} g_\delta(\rho)\right)$$
$$\quad - \lim_{\rho\to 0} h_\delta(\rho)\,\exp\left(-\lim_{\rho\to 0} g_\delta(\rho)\right)\,\lim_{\rho\to 0} g_\delta'(\rho) \tag{109}$$

$$= h_\delta'(0)\,\exp(-g_\delta(0)) - h_\delta(0)\,\exp(-g_\delta(0))\,g_\delta'(0) \tag{110}$$

$$= \exp(-g_\delta(0))\,(h_\delta'(0) - h_\delta(0)\,g_\delta'(0)) \tag{111}$$

$$= \exp(\nu\delta\ln x_1 + \nu(1-\delta)\ln x_2)\left(\frac{1}{2}\left((\ln x_1)^2 - (\ln x_2)^2\right)\right.$$
$$\quad - (-\ln x_1 + \ln x_2)$$
$$\quad \left.\left(-\delta\ln x_1 - (1-\delta)\ln x_2 + \frac{\nu\delta(1-\delta)}{2}(\ln x_1 - \ln x_2)^2\right)\right) \tag{112}$$

$$= x_1^{\nu\delta}\, x_2^{\nu(1-\delta)}\left(\frac{1}{2}(\ln x_1)^2 - \frac{1}{2}(\ln x_2)^2 + \ln x_1 - \ln x_2\right.$$
$$\quad \left.\left(-\delta\ln x_1 - (1-\delta)\ln x_2 + \frac{\nu\delta(1-\delta)}{2}(\ln x_1 - \ln x_2)^2\right)\right) \tag{113}$$

$$= x_1^{\nu\delta}\, x_2^{\nu(1-\delta)}\left(\frac{1}{2}(\ln x_1)^2 - \frac{1}{2}(\ln x_2)^2 - \delta(\ln x_1)^2\right. \tag{114}$$

$$- (1 - \delta) \ln x_1 \ln x_2 + \frac{\nu \delta (1 - \delta)}{2} \ln x_1 (\ln x_1 - \ln x_2)^2$$

$$+ \delta \ln x_1 \ln x_2 + (1 - \delta) (\ln x_2)^2 - \frac{\nu \delta (1 - \delta)}{2} \ln x_2 (\ln x_1 - \ln x_2)^2 \Big)$$

$$= x_1^{\nu \delta} x_2^{\nu (1 - \delta)} \left(\left(\frac{1}{2} - \delta \right) (\ln x_1)^2 + \left(\frac{1}{2} - \delta \right) (\ln x_2)^2 \right. \tag{115}$$

$$\left. -2 \left(\frac{1}{2} - \delta \right) \ln x_1 \ln x_2 + \frac{\nu \delta (1 - \delta)}{2} (\ln x_1 - \ln x_2) (\ln x_1 - \ln x_2)^2 \right)$$

$$= x_1^{\nu \delta} x_2^{\nu (1 - \delta)} \left(\left(\frac{1}{2} - \delta \right) (\ln x_1 - \ln x_2)^2 \right. \tag{116}$$

$$\left. + \frac{\nu \delta (1 - \delta)}{2} (\ln x_1 - \ln x_2) (\ln x_1 - \ln x_2)^2 \right)$$

$$= x_1^{\nu \delta} x_2^{\nu (1 - \delta)} \left(\frac{1}{2} - \delta + \frac{\nu \delta (1 - \delta)}{2} (\ln x_1 - \ln x_2) \right) (\ln x_1 - \ln x_2)^2 \tag{117}$$

$$= \frac{1 - 2\delta + \nu \delta (1 - \delta) (\ln x_1 - \ln x_2)}{2} x_1^{\nu \delta} x_2^{\nu (1 - \delta)} (\ln x_1 - \ln x_2)^2 \tag{118}$$

and approximate $\partial y / \partial \delta$ by

$$\frac{\partial y}{\partial \delta} \approx -\gamma e^{\lambda t} \nu \left(f_\delta (0) + \rho f_\delta' (0) \right) \tag{119}$$

$$= -\gamma e^{\lambda t} \nu \left((- \ln x_1 + \ln x_2) x_1^{\nu \delta} x_2^{\nu (1 - \delta)} \right. \tag{120}$$

$$\left. + \rho \frac{1 - 2\delta + \nu \delta (1 - \delta) (\ln x_1 - \ln x_2)}{2} x_1^{\nu \delta} x_2^{\nu (1 - \delta)} (\ln x_1 - \ln x_2)^2 \right)$$

$$= \gamma e^{\lambda t} \nu \left((\ln x_1 - \ln x_2) x_1^{\nu \delta} x_2^{\nu (1 - \delta)} \right. \tag{121}$$

$$\left. - \rho \frac{1 - 2\delta + \nu \delta (1 - \delta) (\ln x_1 - \ln x_2)}{2} x_1^{\nu \delta} x_2^{\nu (1 - \delta)} (\ln x_1 - \ln x_2)^2 \right)$$

$$= \gamma e^{\lambda t} \nu (\ln x_1 - \ln x_2) x_1^{\nu \delta} x_2^{\nu (1 - \delta)} \tag{122}$$

$$\left(1 - \rho \frac{1 - 2\delta + \nu \delta (1 - \delta) (\ln x_1 - \ln x_2)}{2} (\ln x_1 - \ln x_2) \right)$$

Derivatives with respect to Nu

$$\frac{\partial y}{\partial \nu} = -\gamma e^{\lambda t} \frac{1}{\rho} \ln \left(\delta x_1^{-\rho} + (1 - \delta) x_2^{-\rho} \right) \left(\delta x_1^{-\rho} + (1 - \delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \tag{123}$$

Now we define the function $f_\nu (\rho)$

$$f_\nu (\rho) = \frac{1}{\rho} \ln \left(\delta x_1^{-\rho} + (1 - \delta) x_2^{-\rho} \right) \left(\delta x_1^{-\rho} + (1 - \delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \tag{124}$$

$$= \frac{1}{\rho} \ln \left(\delta x_1^{-\rho} + (1 - \delta) x_2^{-\rho} \right) \exp \left(-\frac{\nu}{\rho} \ln \left(\delta x_1^{-\rho} + (1 - \delta) x_2^{-\rho} \right) \right) \tag{125}$$

so that we can approximate $\partial y / \partial \nu$ by using the first-order Taylor series approximation of $f_\nu(\rho)$:

$$
\frac{\partial y}{\partial \nu} = -\gamma \, e^{\lambda t} \, f_\nu(\rho) \tag{126}
$$

$$
\approx -\gamma \, e^{\lambda t} \, \left(f_\nu(0) + \rho f_\nu'(0) \right) \tag{127}
$$

Now we define the helper function $g_\nu(\rho)$

$$
g_\nu(\rho) = \frac{1}{\rho} \ln \left(\delta \, x_1^{-\rho} + (1 - \delta) \, x_2^{-\rho} \right) \tag{128}
$$

$$
= \frac{1}{\rho} \ln \left(g(\rho) \right) \tag{129}
$$

with first and second derivative

$$
g_\nu'(\rho) = \frac{\rho \frac{g'(\rho)}{g(\rho)} - \ln(g(\rho))}{\rho^2} \tag{130}
$$

$$
= \frac{1}{\rho} \frac{g'(\rho)}{g(\rho)} - \frac{\ln(g(\rho))}{\rho^2} \tag{131}
$$

$$
g_\nu''(\rho) = -\frac{1}{\rho^2} \frac{g'(\rho)}{g(\rho)} + \frac{1}{\rho} \frac{g''(\rho)}{g(\rho)} - \frac{1}{\rho} \frac{(g'(\rho))^2}{(g(\rho))^2} + 2 \frac{\ln(g(\rho))}{\rho^3} - \frac{1}{\rho^2} \frac{g'(\rho)}{g(\rho)} \tag{132}
$$

$$
= \frac{-2\rho \frac{g'(\rho)}{g(\rho)} + \rho^2 \frac{g''(\rho)}{g(\rho)} - \rho^2 \frac{(g'(\rho))^2}{(g(\rho))^2} + 2 \ln(g(\rho))}{\rho^3} \tag{133}
$$

and use the function $f(\rho)$ defined above so that

$$
f_\nu(\rho) = g_\nu(\rho) \exp(f(\rho)) \tag{134}
$$

and

$$
f_\nu'(\rho) = g_\nu'(\rho) \exp(f(\rho)) + g_\nu(\rho) \exp(f(\rho)) \, f'(\rho) \tag{135}
$$

Now we can calculate the limits of $g_\nu(\rho)$, $g_\nu'(\rho)$, and $g_\nu''(\rho)$ for $\rho \to 0$ by

$$
g_\nu(0) = \lim_{\rho \to 0} g_\nu(\rho) \tag{136}
$$

$$
= \lim_{\rho \to 0} \frac{\ln(g(\rho))}{\rho} \tag{137}
$$

$$
= \lim_{\rho \to 0} \frac{\frac{g'(\rho)}{g(\rho)}}{1} \tag{138}
$$

$$
= -\delta \ln x_1 - (1 - \delta) \ln x_2 \tag{139}
$$

$$g'_\nu(0) = \lim_{\rho \to 0} g'_\nu(\rho) \tag{140}$$

$$= \lim_{\rho \to 0} \left(\frac{1}{\rho} \frac{g'(\rho)}{g(\rho)} - \frac{\ln(g(\rho))}{\rho^2} \right) \tag{141}$$

$$= \lim_{\rho \to 0} \frac{\rho \frac{g'(\rho)}{g(\rho)} - \ln(g(\rho))}{\rho^2} \tag{142}$$

$$= \lim_{\rho \to 0} \frac{\frac{g'(\rho)}{g(\rho)} + \rho \frac{g''(\rho)g(\rho) - (g'(\rho))^2}{(g(\rho))^2} - \frac{g'(\rho)}{g(\rho)}}{2\rho} \tag{143}$$

$$= \lim_{\rho \to 0} \frac{g''(\rho)g(\rho) - (g'(\rho))^2}{2(g(\rho))^2} \tag{144}$$

$$= \frac{g''(0)g(0) - (g'(0))^2}{2(g(0))^2} \tag{145}$$

$$= \frac{1}{2} \left(\delta(\ln x_1)^2 + (1-\delta)(\ln x_2)^2 - (-\delta \ln x_1 - (1-\delta)\ln x_2)^2 \right) \tag{146}$$

$$= \frac{1}{2} \left(\delta(\ln x_1)^2 + (1-\delta)(\ln x_2)^2 - \delta^2(\ln x_1)^2 \right.$$

$$\left. -2\delta(1-\delta)\ln x_1 \ln x_2 - (1-\delta)^2(\ln x_2)^2 \right) \tag{147}$$

$$= \frac{1}{2} \left(\left(\delta - \delta^2\right)(\ln x_1)^2 + \left((1-\delta) - (1-\delta)^2\right)(\ln x_2)^2 \right.$$

$$\left. -2\delta(1-\delta)\ln x_1 \ln x_2 \right) \tag{148}$$

$$= \frac{1}{2} \left(\delta(1-\delta)(\ln x_1)^2 + (1-\delta)(1-(1-\delta))(\ln x_2)^2 \right.$$

$$\left. -2\delta(1-\delta)\ln x_1 \ln x_2 \right) \tag{149}$$

$$= \frac{\delta(1-\delta)}{2} \left((\ln x_1)^2 - 2\ln x_1 \ln x_2 + (\ln x_2)^2 \right) \tag{150}$$

$$= \frac{\delta(1-\delta)}{2} (\ln x_1 - \ln x_2)^2 \tag{151}$$

$$g''_\nu(0) = \lim_{\rho \to 0} g''_\nu(\rho) \tag{152}$$

$$= \lim_{\rho \to 0} \left(\frac{-2\rho \frac{g'(\rho)}{g(\rho)} + \rho^2 \frac{g''(\rho)}{g(\rho)} - \rho^2 \frac{(g'(\rho))^2}{(g(\rho))^2} + 2\ln(g(\rho))}{\rho^3} \right) \tag{153}$$

$$= \lim_{\rho \to 0} \left(\frac{-2\frac{g'(\rho)}{g(\rho)} - 2\rho \frac{g''(\rho)}{g(\rho)} + 2\rho \frac{(g'(\rho))^2}{(g(\rho))^2} + 2\rho \frac{g''(\rho)}{g(\rho)} + \rho^2 \frac{g'''(\rho)}{g(\rho)}}{3\rho^2} \right.$$

$$\left. + \frac{-\rho^2 \frac{g''(\rho)g'(\rho)}{(g(\rho))^2} - 2\rho \frac{(g'(\rho))^2}{(g(\rho))^2} - 2\rho^2 \frac{g'(\rho)g''(\rho)}{(g(\rho))^2} + 2\rho^2 \frac{(g'(\rho))^3}{(g(\rho))^3} + 2\frac{g'(\rho)}{g(\rho)}}{3\rho^2} \right) \tag{154}$$

$$= \lim_{\rho \to 0} \left(\frac{\rho^2 \frac{g'''(\rho)}{g(\rho)} - 3\rho^2 \frac{g''(\rho)g'(\rho)}{(g(\rho))^2} + 2\rho^2 \frac{(g'(\rho))^3}{(g(\rho))^3}}{3\rho^2} \right) \tag{155}$$

$$= \lim_{\rho \to 0} \left(\frac{1}{3} \frac{g'''(\rho)}{g(\rho)} - \frac{g''(\rho)g'(\rho)}{(g(\rho))^2} + \frac{2}{3} \frac{(g'(\rho))^3}{(g(\rho))^3} \right) \tag{156}$$

$$= \frac{1}{3} \frac{g'''(0)}{g(0)} - \frac{g''(0)g'(0)}{(g(0))^2} + \frac{2}{3} \frac{(g'(0))^3}{(g(0))^3} \tag{157}$$

$$= \frac{1}{3} \left(-\delta (\ln x_1)^3 - (1-\delta)(\ln x_2)^3 \right) \tag{158}$$
$$- \left(\delta (\ln x_1)^2 + (1-\delta)(\ln x_2)^2 \right) \left(-\delta \ln x_1 - (1-\delta)\ln x_2 \right)$$
$$+ \frac{2}{3} \left(-\delta \ln x_1 - (1-\delta)\ln x_2 \right)^3$$

$$= -\frac{1}{3}\delta (\ln x_1)^3 - \frac{1}{3}(1-\delta)(\ln x_2)^3 + \delta^2 (\ln x_1)^3 \tag{159}$$
$$+ \delta (1-\delta)(\ln x_1)^2 \ln x_2 + \delta (1-\delta) \ln x_1 (\ln x_2)^2 + (1-\delta)^2 (\ln x_2)^3$$
$$+ \frac{2}{3} \left(\delta^2 (\ln x_1)^2 + 2\delta (1-\delta) \ln x_1 \ln x_2 + (1-\delta)^2 (\ln x_2)^2 \right)$$
$$(-\delta \ln x_1 - (1-\delta)\ln x_2)$$

$$= \left(\delta^2 - \frac{1}{3}\delta \right)(\ln x_1)^3 + \delta (1-\delta)(\ln x_1)^2 \ln x_2 \tag{160}$$
$$+ \delta (1-\delta) \ln x_1 (\ln x_2)^2 + \left((1-\delta)^2 - \frac{1}{3}(1-\delta) \right)(\ln x_2)^3$$
$$+ \frac{2}{3} \left(\delta^2 (\ln x_1)^2 + 2\delta (1-\delta) \ln x_1 \ln x_2 + (1-\delta)^2 (\ln x_2)^2 \right)$$
$$(-\delta \ln x_1 - (1-\delta)\ln x_2)$$

$$= \left(\delta^2 - \frac{1}{3}\delta \right)(\ln x_1)^3 + \delta (1-\delta)(\ln x_1)^2 \ln x_2 \tag{161}$$
$$+ \delta (1-\delta) \ln x_1 (\ln x_2)^2 + \left((1-\delta)^2 - \frac{1}{3}(1-\delta) \right)(\ln x_2)^3$$
$$- \frac{2}{3} \left(\delta^3 (\ln x_1)^3 + \delta^2 (1-\delta)(\ln x_1)^2 \ln x_2 + 2\delta^2 (1-\delta)(\ln x_1)^2 \ln x_2 \right.$$
$$+ 2\delta (1-\delta)^2 \ln x_1 (\ln x_2)^2 + \delta (1-\delta)^2 \ln x_1 (\ln x_2)^2 + (1-\delta)^3 (\ln x_2)^3 \Big)$$

$$= \left(\delta^2 - \frac{1}{3}\delta \right)(\ln x_1)^3 + \delta (1-\delta)(\ln x_1)^2 \ln x_2 \tag{162}$$
$$+ \delta (1-\delta) \ln x_1 (\ln x_2)^2 + \left((1-\delta)^2 - \frac{1}{3}(1-\delta) \right)(\ln x_2)^3$$
$$- \frac{2}{3}\delta^3 (\ln x_1)^3 - 2\delta^2 (1-\delta)(\ln x_1)^2 \ln x_2 - 2\delta (1-\delta)^2 \ln x_1 (\ln x_2)^2$$
$$- \frac{2}{3}(1-\delta)^3 (\ln x_2)^3$$

$$= \left(\delta^2 - \frac{1}{3}\delta - \frac{2}{3}\delta^3 \right)(\ln x_1)^3 + \left(\delta (1-\delta) - 2\delta^2 (1-\delta) \right)(\ln x_1)^2 \ln x_2 \tag{163}$$
$$+ \left(\delta \left(1 - \delta - 2\delta (1-\delta)^2 \right) \right) \ln x_1 (\ln x_2)^2$$

$$+ \left((1-\delta)^2 - \frac{1}{3}(1-\delta) - \frac{2}{3}(1-\delta)^3 \right) (\ln x_2)^3$$

$$= \left(\delta - \frac{1}{3} - \frac{2}{3}\delta^2 \right) \delta (\ln x_1)^3 + (1 - 2\delta)\,\delta\,(1-\delta)\,(\ln x_1)^2 \ln x_2 \qquad (164)$$

$$+ (1 - 2(1-\delta))\,\delta\,(1-\delta)\ln x_1\,(\ln x_2)^2$$

$$+ \left((1-\delta) - \frac{1}{3} - \frac{2}{3}(1-\delta)^2 \right)(1-\delta)(\ln x_2)^3$$

$$= \left(-\frac{1}{3} + \frac{2}{3}\delta \right) \delta\,(1-\delta)\,(\ln x_1)^3 + (1 - 2\delta)\,\delta\,(1-\delta)\,(\ln x_1)^2 \ln x_2 \qquad (165)$$

$$+ (2\delta - 1)\,\delta\,(1-\delta)\ln x_1\,(\ln x_2)^2$$

$$+ \left(1 - \delta - \frac{1}{3} - \frac{2}{3} + \frac{4}{3}\delta - \frac{2}{3}\delta^2 \right)(1-\delta)(\ln x_2)^3$$

$$= \left(-\frac{1}{3} + \frac{2}{3}\delta \right)\delta\,(1-\delta)\,(\ln x_1)^3 + (1 - 2\delta)\,\delta\,(1-\delta)\,(\ln x_1)^2 \ln x_2 \qquad (166)$$

$$+ (2\delta - 1)\,\delta\,(1-\delta)\ln x_1\,(\ln x_2)^2 + \left(\frac{1}{3} - \frac{2}{3}\delta \right)\delta\,(1-\delta)(\ln x_2)^3$$

$$= -\frac{1}{3}(1 - 2\delta)\,\delta\,(1-\delta)\,(\ln x_1)^3 + (1 - 2\delta)\,\delta\,(1-\delta)\,(\ln x_1)^2 \ln x_2 \qquad (167)$$

$$- (1 - 2\delta)\,\delta\,(1-\delta)\ln x_1\,(\ln x_2)^2 + \frac{1}{3}(1 - 2\delta)\,\delta\,(1-\delta)(\ln x_2)^3$$

$$= -\frac{1}{3}(1 - 2\delta)\,\delta\,(1-\delta) \qquad (168)$$

$$\left((\ln x_1)^3 + 3(\ln x_1)^2 \ln x_2 + 3\ln x_1\,(\ln x_2)^2 - (\ln x_2)^3 \right)$$

$$= -\frac{1}{3}(1 - 2\delta)\,\delta\,(1-\delta)(\ln x_1 - \ln x_2)^3 \qquad (169)$$

so that we can calculate the limit of $f_\nu(\rho)$ and $f'_\nu(\rho)$ for $\rho \to 0$ by

$$f_\nu(0) = \lim_{\rho \to 0} f_\nu(\rho) \qquad (170)$$

$$= \lim_{\rho \to 0}\left(g_\nu(\rho)\exp\left(f(\rho) \right) \right) \qquad (171)$$

$$= \lim_{\rho \to 0} g_\nu(\rho) \lim_{\rho \to 0} \exp\left(f(\rho) \right) \qquad (172)$$

$$= \lim_{\rho \to 0} g_\nu(\rho) \exp\left(\lim_{\rho \to 0} f(\rho) \right) \qquad (173)$$

$$= g_\nu(0)\exp\left(f(0) \right) \qquad (174)$$

$$= \left(-\delta \ln x_1 - (1-\delta)\ln x_2 \right)\exp\left(\nu\left(\delta \ln x_1 + (1-\delta)\ln x_2 \right) \right) \qquad (175)$$

$$= -\left(\delta \ln x_1 + (1-\delta)\ln x_2 \right) x_1^{\nu\delta} x_2^{\nu(1-\delta)} \qquad (176)$$

$$f'_\nu(0) = \lim_{\rho \to 0} f'_\nu(\rho) \qquad (177)$$

$$= \lim_{\rho \to 0}\left(g'_\nu(\rho)\exp\left(f(\rho) \right) + g_\nu(\rho)\exp\left(f(\rho) \right) f'(\rho) \right) \qquad (178)$$

$$= \lim_{\rho \to 0} g'_\nu(\rho)\exp\left(\lim_{\rho \to 0} f(\rho) \right) + \lim_{\rho \to 0} g_\nu(\rho)\exp\left(\lim_{\rho \to 0} f(\rho) \right) \lim_{\rho \to 0} f'(\rho) \qquad (179)$$

$$= g'_\nu(0) \exp(f(0)) + g_\nu(0) \exp(f(0)) f'(0) \tag{180}$$

$$= \exp(f(0)) (g'_\nu(0) + g_\nu(0) f'(0)) \tag{181}$$

$$= \exp(\nu(\delta \ln x_1 + (1-\delta) \ln x_2)) \left(\frac{\delta(1-\delta)}{2} (\ln x_1 - \ln x_2)^2 \right. \tag{182}$$

$$+ (-\delta \ln x_1 - (1-\delta) \ln x_2) \left(-\frac{\nu\delta(1-\delta)}{2} (\ln x_1 - \ln x_2)^2 \right) \Big)$$

$$= x_1^{\nu\delta} x_2^{\nu(1-\delta)} \frac{\delta(1-\delta)}{2} (\ln x_1 - \ln x_2)^2 (1 + \nu(\delta \ln x_1 + (1-\delta) \ln x_2)) \tag{183}$$

and approximate $\partial y / \partial \nu$ by

$$\frac{\partial y}{\partial \nu} \approx -\gamma e^{\lambda t} (f_\nu(0) + \rho f'_\nu(0)) \tag{184}$$

$$= \gamma e^{\lambda t} (\delta \ln x_1 + (1-\delta) \ln x_2) x_1^{\nu\delta} x_2^{\nu(1-\delta)} \tag{185}$$

$$-\gamma e^{\lambda t} \rho x_1^{\nu\delta} x_2^{\nu(1-\delta)} \frac{\delta(1-\delta)}{2} (\ln x_1 - \ln x_2)^2$$

$$(1 + \nu(\delta \ln x_1 + (1-\delta) \ln x_2))$$

$$= \gamma e^{\lambda t} x_1^{\nu\delta} x_2^{\nu(1-\delta)} \Big(\delta \ln x_1 + (1-\delta) \ln x_2 \tag{186}$$

$$-\frac{\rho\delta(1-\delta)}{2} (\ln x_1 - \ln x_2)^2 (1 + \nu(\delta \ln x_1 + (1-\delta) \ln x_2)) \Big)$$

Derivatives with respect to Rho

$$\frac{\partial y}{\partial \rho} = \gamma e^{\lambda t} \frac{\nu}{\rho^2} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \ln \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right) \tag{187}$$

$$+\gamma e^{\lambda t} \frac{\nu}{\rho} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\left(\frac{\nu}{\rho}+1\right)} \left(\delta x_1^{-\rho} \ln x_1 + (1-\delta) x_2^{-\rho} \ln x_2 \right)$$

$$= \gamma e^{\lambda t} \nu \left(\frac{1}{\rho^2} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \ln \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right) \right. \tag{188}$$

$$+\frac{1}{\rho} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\left(\frac{\nu}{\rho}+1\right)} \left(\delta x_1^{-\rho} \ln x_1 + (1-\delta) x_2^{-\rho} \ln x_2 \right) \Big)$$

Now we define the function $f_\rho(\rho)$

$$f_\rho(\rho) = \frac{1}{\rho^2} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \ln \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right) \tag{189}$$

$$+\frac{1}{\rho} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\left(\frac{\nu}{\rho}+1\right)} \left(\delta x_1^{-\rho} \ln x_1 + (1-\delta) x_2^{-\rho} \ln x_2 \right)$$

so that we can approximate $\partial y/\partial \rho$ by using the first-order Taylor series approximation of $f_\rho(\rho)$:

$$\frac{\partial y}{\partial \rho} = \gamma e^{\lambda t} \nu f_\rho(\rho) \tag{190}$$

$$\approx \gamma e^{\lambda t} \nu \left(f_\rho(0) + \rho f'_\rho(0) \right) \tag{191}$$

We define the helper function $g_\rho(\rho)$

$$g_\rho(\rho) = \delta x_1^{-\rho} \ln x_1 + (1-\delta) x_2^{-\rho} \ln x_2 \tag{192}$$

with first and second derivative

$$g'_\rho(\rho) = -\delta x_1^{-\rho} (\ln x_1)^2 - (1-\delta) x_2^{-\rho} (\ln x_2)^2 \tag{193}$$

$$g''_\rho(\rho) = \delta x_1^{-\rho} (\ln x_1)^3 + (1-\delta) x_2^{-\rho} (\ln x_2)^3 \tag{194}$$

and use the functions $g(\rho)$ and $g_\nu(\rho)$ all defined above so that

$$f_\rho(\rho) = \frac{1}{\rho^2} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \ln \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right) \tag{195}$$

$$+ \frac{1}{\rho} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\left(\frac{\nu}{\rho}+1\right)} \left(\delta x_1^{-\rho} \ln x_1 + (1-\delta) x_2^{-\rho} \ln x_2 \right)$$

$$= \frac{1}{\rho^2} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \ln \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right) \tag{196}$$

$$+ \frac{1}{\rho} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-1}$$

$$\left(\delta x_1^{-\rho} \ln x_1 + (1-\delta) x_2^{-\rho} \ln x_2 \right)$$

$$= \frac{1}{\rho} \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-\frac{\nu}{\rho}} \left(\frac{1}{\rho} \ln \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right) \right. \tag{197}$$

$$\left. + \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-1} \left(\delta x_1^{-\rho} \ln x_1 + (1-\delta) x_2^{-\rho} \ln x_2 \right) \right)$$

$$= \frac{1}{\rho} \exp\left(-\frac{\nu}{\rho} \ln \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right) \right) \left(\frac{1}{\rho} \ln \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right) \right. \tag{198}$$

$$\left. + \left(\delta x_1^{-\rho} + (1-\delta) x_2^{-\rho} \right)^{-1} \left(\delta x_1^{-\rho} \ln x_1 + (1-\delta) x_2^{-\rho} \ln x_2 \right) \right)$$

$$= \frac{\exp\left(-\nu g_\nu(\rho) \right) \left(g_\nu(\rho) + g(\rho)^{-1} g_\rho(\rho) \right)}{\rho} \tag{199}$$

and we can calculate its first derivative

$$f'_\rho(\rho) = \frac{-\rho \nu \exp\left(-\nu g_\nu(\rho) \right) g'_\nu(\rho) \left(g_\nu(\rho) + g(\rho)^{-1} g_\rho(\rho) \right)}{\rho^2} \tag{200}$$

$$+ \frac{\rho \exp\left(-\nu g_\nu\left(\rho\right)\right)\left(g'_\nu\left(\rho\right) - g\left(\rho\right)^{-2} g'\left(\rho\right) g_\rho\left(\rho\right) + g\left(\rho\right)^{-1} g'_\rho\left(\rho\right)\right)}{\rho^2}$$

$$- \frac{\exp\left(-\nu g_\nu\left(\rho\right)\right)\left(g_\nu\left(\rho\right) + g\left(\rho\right)^{-1} g_\rho\left(\rho\right)\right)}{\rho^2}$$

$$= \frac{\exp\left(-\nu g_\nu\left(\rho\right)\right)\rho\left(-\nu g'_\nu\left(\rho\right) g_\nu\left(\rho\right) - \nu g'_\nu\left(\rho\right) g\left(\rho\right)^{-1} g_\rho\left(\rho\right)\right)}{\rho^2} \tag{201}$$

$$+ \frac{\exp\left(-\nu g_\nu\left(\rho\right)\right)\rho\left(g'_\nu\left(\rho\right) - g\left(\rho\right)^{-2} g'\left(\rho\right) g_\rho\left(\rho\right) + g\left(\rho\right)^{-1} g'_\rho\left(\rho\right)\right)}{\rho^2}$$

$$- \frac{\exp\left(-\nu g_\nu\left(\rho\right)\right)\left(g_\nu\left(\rho\right) + g\left(\rho\right)^{-1} g_\rho\left(\rho\right)\right)}{\rho^2}$$

$$= \frac{\exp\left(-\nu g_\nu\left(\rho\right)\right)}{\rho^2}\left(\rho\left(-\nu g'_\nu\left(\rho\right) g_\nu\left(\rho\right) - \nu g'_\nu\left(\rho\right) g\left(\rho\right)^{-1} g_\rho\left(\rho\right)\right.\right. \tag{202}$$

$$\left.+ g'_\nu\left(\rho\right) - g\left(\rho\right)^{-2} g'\left(\rho\right) g_\rho\left(\rho\right) + g\left(\rho\right)^{-1} g'_\rho\left(\rho\right)\right)$$

$$\left.- g_\nu\left(\rho\right) - g\left(\rho\right)^{-1} g_\rho\left(\rho\right)\right)$$

$$= \frac{\exp\left(-\nu g_\nu\left(\rho\right)\right)}{\rho^2}\left(-\nu \rho g'_\nu\left(\rho\right) g_\nu\left(\rho\right) - \nu \rho g'_\nu\left(\rho\right) g\left(\rho\right)^{-1} g_\rho\left(\rho\right)\right. \tag{203}$$

$$+ \rho g'_\nu\left(\rho\right) - \rho g\left(\rho\right)^{-2} g'\left(\rho\right) g_\rho\left(\rho\right) + \rho g\left(\rho\right)^{-1} g'_\rho\left(\rho\right)$$

$$\left.- g_\nu\left(\rho\right) - g\left(\rho\right)^{-1} g_\rho\left(\rho\right)\right)$$

Now we can calculate the limits of $g_\rho\left(\rho\right)$, $g'_\rho\left(\rho\right)$, and $g''_\rho\left(\rho\right)$ for $\rho \to 0$ by

$$g_\rho\left(0\right) = \lim_{\rho\to 0} g_\rho\left(\rho\right) \tag{204}$$

$$= \lim_{\rho\to 0}\left(\delta x_1^{-\rho} \ln x_1 + \left(1-\delta\right) x_2^{-\rho} \ln x_2\right) \tag{205}$$

$$= \delta \ln x_1 \lim_{\rho\to 0} x_1^{-\rho} + \left(1-\delta\right) \ln x_2 \lim_{\rho\to 0} x_2^{-\rho} \tag{206}$$

$$= \delta \ln x_1 + \left(1-\delta\right) \ln x_2 \tag{207}$$

$$g'_\rho\left(0\right) = \lim_{\rho\to 0} g'_\rho\left(\rho\right) \tag{208}$$

$$= \lim_{\rho\to 0}\left(-\delta x_1^{-\rho} \left(\ln x_1\right)^2 - \left(1-\delta\right) x_2^{-\rho} \left(\ln x_2\right)^2\right) \tag{209}$$

$$= -\delta \left(\ln x_1\right)^2 \lim_{\rho\to 0} x_1^{-\rho} - \left(1-\delta\right) \left(\ln x_2\right)^2 \lim_{\rho\to 0} x_2^{-\rho} \tag{210}$$

$$= -\delta \left(\ln x_1\right)^2 - \left(1-\delta\right) \left(\ln x_2\right)^2 \tag{211}$$

$$g''_\rho\left(0\right) = \lim_{\rho\to 0} g''_\rho\left(\rho\right) \tag{212}$$

$$= \lim_{\rho\to 0}\left(\delta x_1^{-\rho} \left(\ln x_1\right)^3 + \left(1-\delta\right) x_2^{-\rho} \left(\ln x_2\right)^3\right) \tag{213}$$

$$= \quad \delta \left(\ln x_1\right)^3 \lim_{\rho \to 0} x_1^{-\rho} + (1 - \delta) \left(\ln x_2\right)^3 \lim_{\rho \to 0} x_2^{-\rho} \tag{214}$$

$$= \quad \delta \left(\ln x_1\right)^3 + (1 - \delta) \left(\ln x_2\right)^3 \tag{215}$$

so that we can calculate the limit of $f_\rho(\rho)$ for $\rho \to 0$ by

$$f_\rho(0) \quad = \quad \lim_{\rho \to 0} f_\rho(\rho) \tag{216}$$

$$= \quad \lim_{\rho \to 0} \left(\frac{\exp\left(-\nu g_\nu(\rho)\right) \left(g_\nu(\rho) + g(\rho)^{-1} g_\rho(\rho)\right)}{\rho} \right) \tag{217}$$

$$= \quad \lim_{\rho \to 0} \Bigg(\frac{-\nu \exp\left(-\nu g_\nu(\rho)\right) g_\nu'(\rho) \left(g_\nu(\rho) + g(\rho)^{-1} g_\rho(\rho)\right)}{1} \tag{218}$$

$$+ \frac{\exp\left(-\nu g_\nu(\rho)\right) \left(g_\nu'(\rho) - g(\rho)^{-2} g'(\rho) g_\rho(\rho) + g(\rho)^{-1} g_\rho'(\rho)\right)}{1} \Bigg)$$

$$= \quad -\nu \exp\left(-\nu g_\nu(0)\right) g_\nu'(0) \left(g_\nu(0) + g(0)^{-1} g_\rho(0)\right) \tag{219}$$

$$+ \exp\left(-\nu g_\nu(0)\right) \left(g_\nu'(0) - g(0)^{-2} g'(0) g_\rho(0) + g(0)^{-1} g_\rho'(0)\right)$$

$$= \quad \exp\left(-\nu g_\nu(0)\right) \left(-\nu g_\nu'(0)\right) \left(g_\nu(0) + g(0)^{-1} g_\rho(0)\right) \tag{220}$$

$$+ \exp\left(-\nu g_\nu(0)\right) \left(g_\nu'(0) - g(0)^{-2} g'(0) g_\rho(0) + g(0)^{-1} g_\rho'(0)\right)$$

$$= \quad \exp\left(-\nu g_\nu(0)\right) \Big(\left(-\nu g_\nu'(0)\right) \left(g_\nu(0) + g(0)^{-1} g_\rho(0)\right) \tag{221}$$

$$+ g_\nu'(0) - g(0)^{-2} g'(0) g_\rho(0) + g(0)^{-1} g_\rho'(0)\Big)$$

$$= \quad \exp\left(-\nu\left(-\delta \ln x_1 - (1 - \delta) \ln x_2\right)\right) \Big(-\frac{\nu \delta (1 - \delta)}{2} \left(\ln x_1 - \ln x_2\right)^2 \tag{222}$$

$$\left(-\delta \ln x_1 - (1 - \delta) \ln x_2 + \delta \ln x_1 + (1 - \delta) \ln x_2\right)$$

$$+ \frac{\delta (1 - \delta)}{2} \left(\ln x_1 - \ln x_2\right)^2$$

$$- \left(-\delta \ln x_1 - (1 - \delta) \ln x_2\right) \left(\delta \ln x_1 + (1 - \delta) \ln x_2\right)$$

$$- \delta \left(\ln x_1\right)^2 - (1 - \delta) \left(\ln x_2\right)^2\Big)$$

$$= \quad x_1^{\nu \delta} x_2^{\nu(1-\delta)} \left(\frac{1}{2}\delta (1 - \delta) \left(\ln x_1\right)^2 - \delta (1 - \delta) \ln x_1 \ln x_2\right. \tag{223}$$

$$+ \frac{1}{2}\delta (1 - \delta) \left(\ln x_2\right)^2 + \delta^2 \left(\ln x_1\right)^2 + 2\delta (1 - \delta) \ln x_1 \ln x_2$$

$$+ \left(1 - \delta\right)^2 \left(\ln x_2\right)^2 - \delta \left(\ln x_1\right)^2 - (1 - \delta) \left(\ln x_2\right)^2\Big)$$

$$= \quad x_1^{\nu \delta} x_2^{\nu(1-\delta)} \left(\left(\frac{1}{2}\delta (1 - \delta) + \delta^2 - \delta\right) \left(\ln x_1\right)^2 \right. \tag{224}$$

$$+ \left(\frac{1}{2}\delta (1 - \delta) + (1 - \delta)^2 - (1 - \delta)\right)$$

$$\left(\ln x_2\right)^2 + \delta (1 - \delta) \ln x_1 \ln x_2\Big)$$

$$= x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(\left(\frac{1}{2}\delta - \frac{1}{2}\delta^2 + \delta^2 - \delta \right) (\ln x_1)^2 \right. \tag{225}$$

$$+ \left(\frac{1}{2}\delta - \frac{1}{2}\delta^2 + 1 - 2\delta + \delta^2 - 1 + \delta \right) (\ln x_2)^2$$

$$+ \delta (1-\delta) \ln x_1 \ln x_2)$$

$$= x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(\left(-\frac{1}{2}\delta + \frac{1}{2}\delta^2 \right) (\ln x_1)^2 \right. \tag{226}$$

$$+ \left(-\frac{1}{2}\delta + \frac{1}{2}\delta^2 \right) (\ln x_2)^2 + \delta (1-\delta) \ln x_1 \ln x_2 \Bigg)$$

$$= x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(-\frac{1}{2}\delta (1-\delta) (\ln x_1)^2 \right. \tag{227}$$

$$\left. -\frac{1}{2}\delta (1-\delta) (\ln x_2)^2 + \delta (1-\delta) \ln x_1 \ln x_2 \right)$$

$$= -\frac{1}{2}\delta (1-\delta) x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left((\ln x_1)^2 - 2 \ln x_1 \ln x_2 + (\ln x_2)^2 \right) \tag{228}$$

$$= -\frac{1}{2}\delta (1-\delta) x_1^{\nu\delta} x_2^{\nu(1-\delta)} (\ln x_1 - \ln x_2)^2 \tag{229}$$

Before we can apply de l'Hospital's rule to $\lim_{\rho \to 0} f'_\rho (\rho)$, we have to check whether also the numerator converges to zero. We do this by defining a helper function $h_\rho (\rho)$, where the numerator converges to zero if $h_\rho (\rho)$ converges to zero for $\rho \to 0$

$$h_\rho (\rho) = -g_\nu (\rho) - g(\rho)^{-1} g_\rho (\rho) \tag{230}$$

$$h_\rho (0) = \lim_{\rho \to 0} h_\rho (\rho) \tag{231}$$

$$= \lim_{\rho \to 0} \left(-g_\nu (\rho) - g(\rho)^{-1} g_\rho (\rho) \right) \tag{232}$$

$$= -g_\nu (0) - g(0)^{-1} g_\rho (0) \tag{233}$$

$$= -(-\delta \ln x_1 - (1-\delta) \ln x_2) - (\delta \ln x_1 + (1-\delta) \ln x_2) \tag{234}$$

$$= 0 \tag{235}$$

As both the numerator and the denominator converge to zero, we can calculate $\lim_{\rho \to 0} f'_\rho (\rho)$ by using de l'Hospital's rule.

$$f'_\rho (0) = \lim_{\rho \to 0} f'_\rho (\rho) \tag{236}$$

$$= \lim_{\rho \to 0} \left(\frac{\exp(-\nu g_\nu (\rho))}{\rho^2} \left(-\nu \rho g'_\nu (\rho) g_\nu (\rho) \right. \right. \tag{237}$$

$$-\nu \rho g'_\nu (\rho) g(\rho)^{-1} g_\rho (\rho) + \rho g'_\nu (\rho) - \rho g(\rho)^{-2} g'(\rho) g_\rho (\rho)$$

$$\left. +\rho g(\rho)^{-1} g'_\rho (\rho) - g_\nu (\rho) - g(\rho)^{-1} g_\rho (\rho) \right) \Bigg)$$

$$= \lim_{\rho \to 0} \left(\exp(-\nu g_\nu (\rho)) \right) \lim_{\rho \to 0} \left(\frac{1}{\rho^2} \left(-\nu \rho g'_\nu (\rho) g_\nu (\rho) \right. \right. \tag{238}$$

$$-\nu\rho g_\nu'\left(\rho\right)g\left(\rho\right)^{-1}g_\rho\left(\rho\right)+\rho g_\nu'\left(\rho\right)-\rho g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho\left(\rho\right)$$
$$+\rho g\left(\rho\right)^{-1}g_\rho'\left(\rho\right)-g_\nu\left(\rho\right)-g\left(\rho\right)^{-1}g_\rho\left(\rho\right)\bigg)\bigg)$$

$$= \lim_{\rho\to0}\left(\exp\left(-\nu g_\nu\left(\rho\right)\right)\right)\lim_{\rho\to0}\left(\frac{1}{2\rho}\left(-\nu g_\nu'\left(\rho\right)g_\nu\left(\rho\right)-\nu\rho g_\nu''\left(\rho\right)g_\nu\left(\rho\right)\right.\right. \quad (239)$$
$$-\nu\rho g_\nu'\left(\rho\right)g_\nu'\left(\rho\right)-\nu g_\nu'\left(\rho\right)g\left(\rho\right)^{-1}g_\rho\left(\rho\right)-\nu\rho g_\nu''\left(\rho\right)g\left(\rho\right)^{-1}g_\rho\left(\rho\right)$$
$$+\nu\rho g_\nu'\left(\rho\right)g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho\left(\rho\right)-\nu\rho g_\nu'\left(\rho\right)g\left(\rho\right)^{-1}g_\rho'\left(\rho\right)+g_\nu'\left(\rho\right)$$
$$+\rho g_\nu''\left(\rho\right)-g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho\left(\rho\right)+2\rho g\left(\rho\right)^{-3}\left(g'\left(\rho\right)\right)^2g_\rho\left(\rho\right)$$
$$-\rho g\left(\rho\right)^{-2}g''\left(\rho\right)g_\rho\left(\rho\right)-\rho g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho'\left(\rho\right)+g\left(\rho\right)^{-1}g_\rho'\left(\rho\right)$$
$$-\rho g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho'\left(\rho\right)+\rho g\left(\rho\right)^{-1}g_\rho''\left(\rho\right)$$
$$-g_\nu'\left(\rho\right)+g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho\left(\rho\right)-g\left(\rho\right)^{-1}g_\rho'\left(\rho\right)\bigg)\bigg)$$

$$= \frac{1}{2}\lim_{\rho\to0}\left(\exp\left(-\nu g_\nu\left(\rho\right)\right)\right)\lim_{\rho\to0}\left(\frac{1}{\rho}\left(-\nu g_\nu'\left(\rho\right)g_\nu\left(\rho\right)-\nu\rho g_\nu''\left(\rho\right)g_\nu\left(\rho\right)\right.\right. \quad (240)$$
$$-\nu\rho g_\nu'\left(\rho\right)g_\nu'\left(\rho\right)-\nu g_\nu'\left(\rho\right)g\left(\rho\right)^{-1}g_\rho\left(\rho\right)-\nu\rho g_\nu''\left(\rho\right)g\left(\rho\right)^{-1}g_\rho\left(\rho\right)$$
$$+\nu\rho g_\nu'\left(\rho\right)g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho\left(\rho\right)-\nu\rho g_\nu'\left(\rho\right)g\left(\rho\right)^{-1}g_\rho'\left(\rho\right)+\rho g_\nu''\left(\rho\right)$$
$$+2\rho g\left(\rho\right)^{-3}\left(g'\left(\rho\right)\right)^2g_\rho\left(\rho\right)-\rho g\left(\rho\right)^{-2}g''\left(\rho\right)g_\rho\left(\rho\right)$$
$$-2\rho g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho'\left(\rho\right)+\rho g\left(\rho\right)^{-1}g_\rho''\left(\rho\right)\bigg)\bigg)$$

$$= \frac{1}{2}\lim_{\rho\to0}\left(\exp\left(-\nu g_\nu\left(\rho\right)\right)\right) \quad (241)$$
$$\lim_{\rho\to0}\left(\frac{1}{\rho}\left(-\nu g_\nu'\left(\rho\right)g_\nu\left(\rho\right)-\nu g_\nu'\left(\rho\right)g\left(\rho\right)^{-1}g_\rho\left(\rho\right)\right)\right.$$
$$-\nu g_\nu''\left(\rho\right)g_\nu\left(\rho\right)-\nu g_\nu'\left(\rho\right)g_\nu'\left(\rho\right)-\nu g_\nu''\left(\rho\right)g\left(\rho\right)^{-1}g_\rho\left(\rho\right)$$
$$+\nu g_\nu'\left(\rho\right)g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho\left(\rho\right)-\nu g_\nu'\left(\rho\right)g\left(\rho\right)^{-1}g_\rho'\left(\rho\right)$$
$$+g_\nu''\left(\rho\right)+2g\left(\rho\right)^{-3}\left(g'\left(\rho\right)\right)^2g_\rho\left(\rho\right)-g\left(\rho\right)^{-2}g''\left(\rho\right)g_\rho\left(\rho\right)$$
$$-2g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho'\left(\rho\right)+g\left(\rho\right)^{-1}g_\rho''\left(\rho\right)\bigg)$$

$$= \frac{1}{2}\lim_{\rho\to0}\left(\exp\left(-\nu g_\nu\left(\rho\right)\right)\right) \quad (242)$$
$$\left(\lim_{\rho\to0}\left(\frac{1}{\rho}\left(-\nu g_\nu'\left(\rho\right)g_\nu\left(\rho\right)-\nu g_\nu'\left(\rho\right)g\left(\rho\right)^{-1}g_\rho\left(\rho\right)\right)\right)\right.$$
$$+\lim_{\rho\to0}\left(-\nu g_\nu''\left(\rho\right)g_\nu\left(\rho\right)-\nu g_\nu'\left(\rho\right)g_\nu'\left(\rho\right)-\nu g_\nu''\left(\rho\right)g\left(\rho\right)^{-1}g_\rho\left(\rho\right)\right.$$
$$+\nu g_\nu'\left(\rho\right)g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho\left(\rho\right)-\nu g_\nu'\left(\rho\right)g\left(\rho\right)^{-1}g_\rho'\left(\rho\right)+g_\nu''\left(\rho\right)$$
$$+2g\left(\rho\right)^{-3}\left(g'\left(\rho\right)\right)^2g_\rho\left(\rho\right)-g\left(\rho\right)^{-2}g''\left(\rho\right)g_\rho\left(\rho\right)$$
$$-2g\left(\rho\right)^{-2}g'\left(\rho\right)g_\rho'\left(\rho\right)+g\left(\rho\right)^{-1}g_\rho''\left(\rho\right)\bigg)\bigg)$$

Before we can apply de l'Hospital's rule again, we have to check if also the numerator converges to zero. We do this by defining a helper function $k_\rho\left(\rho\right)$, where the numerator converges to

zero if $k_\rho(\rho)$ converges to zero for $\rho \to 0$

$$
\begin{align}
k_\rho(\rho) &= -\nu g_\nu'(\rho) g_\nu(\rho) - \nu g_\nu'(\rho) g(\rho)^{-1} g_\rho(\rho) \tag{243} \\
k_\rho(0) &= \lim_{\rho \to 0} k_\rho(\rho) \tag{244} \\
&= \lim_{\rho \to 0} \left(-\nu g_\nu'(\rho) g_\nu(\rho) - \nu g_\nu'(\rho) g(\rho)^{-1} g_\rho(\rho) \right) \tag{245} \\
&= -\nu g_\nu'(0) g_\nu(0) - \nu g_\nu'(0) g(0)^{-1} g_\rho(0) \tag{246} \\
&= -\frac{\nu \delta (1-\delta)}{2} (\ln x_1 - \ln x_2)^2 (-\delta \ln x_1 - (1-\delta) \ln x_2) \tag{247} \\
&\quad -\frac{\nu \delta (1-\delta)}{2} (\ln x_1 - \ln x_2)^2 (\delta \ln x_1 + (1-\delta) \ln x_2) \\
&= 0 \tag{248}
\end{align}
$$

As both the numerator and the denominator converge to zero, we can apply de l'Hospital's rule.

$$
\begin{align}
\lim_{\rho \to 0} \frac{k_\rho(\rho)}{\rho} &= \lim_{\rho \to 0} k_\rho(\rho) \tag{249} \\
&= \lim_{\rho \to 0} \Big(-\nu g_\nu''(\rho) g_\nu(\rho) - \nu \left(g_\nu'(\rho) \right)^2 - \nu g_\nu''(\rho) g(\rho)^{-1} g_\rho(\rho) \tag{250} \\
&\qquad + \nu g_\nu'(\rho) g(\rho)^{-2} g'(\rho) g_\rho(\rho) - \nu g_\nu'(\rho) g(\rho)^{-1} g_\rho'(\rho) \Big)
\end{align}
$$

and hence,

$$
\begin{align}
f_\rho'(0) &= \frac{1}{2} \lim_{\rho \to 0} \left(\exp(-\nu g_\nu(\rho)) \right) \tag{251} \\
&\quad \left(\lim_{\rho \to 0} \left(\frac{1}{\rho} \left(-\nu g_\nu'(\rho) g_\nu(\rho) - \nu g_\nu'(\rho) g(\rho)^{-1} g_\rho(\rho) \right) \right) \right. \\
&\quad + \lim_{\rho \to 0} \Big(-\nu g_\nu''(\rho) g_\nu(\rho) - \nu g_\nu'(\rho) g_\nu'(\rho) - \nu g_\nu''(\rho) g(\rho)^{-1} g_\rho(\rho) \\
&\quad + \nu g_\nu'(\rho) g(\rho)^{-2} g'(\rho) g_\rho(\rho) - \nu g_\nu'(\rho) g(\rho)^{-1} g_\rho'(\rho) \\
&\quad + g_\nu''(\rho) + 2 g(\rho)^{-3} \left(g'(\rho) \right)^2 g_\rho(\rho) - g(\rho)^{-2} g''(\rho) g_\rho(\rho) \\
&\quad \left. -2 g(\rho)^{-2} g'(\rho) g_\rho'(\rho) + g(\rho)^{-1} g_\rho''(\rho) \Big) \right) \\
&= \frac{1}{2} \lim_{\rho \to 0} \left(\exp(-\nu g_\nu(\rho)) \right) \tag{252} \\
&\quad \left(\lim_{\rho \to 0} \left(-\nu g_\nu''(\rho) g_\nu(\rho) - \nu \left(g_\nu'(\rho) \right)^2 - \nu g_\nu''(\rho) g(\rho)^{-1} g_\rho(\rho) \right. \right. \\
&\quad \left. + \nu g_\nu'(\rho) g(\rho)^{-2} g'(\rho) g_\rho(\rho) - \nu g_\nu'(\rho) g(\rho)^{-1} g_\rho'(\rho) \right) \\
&\quad + \lim_{\rho \to 0} \Big(-\nu g_\nu''(\rho) g_\nu(\rho) - \nu g_\nu'(\rho) g_\nu'(\rho) - \nu g_\nu''(\rho) g(\rho)^{-1} g_\rho(\rho) \\
&\quad + \nu g_\nu'(\rho) g(\rho)^{-2} g'(\rho) g_\rho(\rho) - \nu g_\nu'(\rho) g(\rho)^{-1} g_\rho'(\rho) + g_\nu''(\rho)
\end{align}
$$

$$+2g\left(\rho\right)^{-3}\left(g'\left(\rho\right)\right)^{2}g_{\rho}\left(\rho\right)-g\left(\rho\right)^{-2}g''\left(\rho\right)g_{\rho}\left(\rho\right)$$

$$-2g\left(\rho\right)^{-2}g'\left(\rho\right)g'_{\rho}\left(\rho\right)+g\left(\rho\right)^{-1}g''_{\rho}\left(\rho\right)\bigg)\bigg)$$

$$=\ \frac{1}{2}\exp\left(-\nu g_{\nu}\left(0\right)\right)\left(-\nu g''_{\nu}\left(0\right)g_{\nu}\left(0\right)-\nu\left(g'_{\nu}\left(0\right)\right)^{2}\right. \tag{253}$$

$$-\nu g''_{\nu}\left(0\right)g\left(0\right)^{-1}g_{\rho}\left(0\right)+\nu g'_{\nu}\left(0\right)g\left(0\right)^{-2}g'\left(0\right)g_{\rho}\left(0\right)$$

$$-\nu g'_{\nu}\left(0\right)g\left(0\right)^{-1}g'_{\rho}\left(0\right)-\nu g''_{\nu}\left(0\right)g_{\nu}\left(0\right)-\nu g'_{\nu}\left(0\right)g'_{\nu}\left(0\right)$$

$$-\nu g''_{\nu}\left(0\right)g\left(0\right)^{-1}g_{\rho}\left(0\right)+\nu g'_{\nu}\left(0\right)g\left(0\right)^{-2}g'\left(0\right)g_{\rho}\left(0\right)$$

$$-\nu g'_{\nu}\left(0\right)g\left(0\right)^{-1}g'_{\rho}\left(0\right)+g''_{\nu}\left(0\right)+2g\left(0\right)^{-3}\left(g'\left(0\right)\right)^{2}g_{\rho}\left(0\right)$$

$$\left.-g\left(0\right)^{-2}g''\left(0\right)g_{\rho}\left(0\right)-2g\left(0\right)^{-2}g'\left(0\right)g'_{\rho}\left(0\right)+g\left(0\right)^{-1}g''_{\rho}\left(0\right)\right)$$

$$=\ \frac{1}{2}\exp\left(-\nu g_{\nu}\left(0\right)\right)\left(-\nu g''_{\nu}\left(0\right)g_{\nu}\left(0\right)-\nu\left(g'_{\nu}\left(0\right)\right)^{2}-\nu g''_{\nu}\left(0\right)g_{\rho}\left(0\right)\right. \tag{254}$$

$$+\nu g'_{\nu}\left(0\right)g'\left(0\right)g_{\rho}\left(0\right)-\nu g'_{\nu}\left(0\right)g'_{\rho}\left(0\right)$$

$$-\nu g''_{\nu}\left(0\right)g_{\nu}\left(0\right)-\nu g'_{\nu}\left(0\right)g'_{\nu}\left(0\right)-\nu g''_{\nu}\left(0\right)g_{\rho}\left(0\right)+\nu g'_{\nu}\left(0\right)g'\left(0\right)g_{\rho}\left(0\right)$$

$$-\nu g'_{\nu}\left(0\right)g'_{\rho}\left(0\right)+g''_{\nu}\left(0\right)+2\left(g'\left(0\right)\right)^{2}g_{\rho}\left(0\right)-g''\left(0\right)g_{\rho}\left(0\right)$$

$$\left.-2g'\left(0\right)g'_{\rho}\left(0\right)+g''_{\rho}\left(0\right)\right)$$

$$=\ \frac{1}{2}\exp\left(-\nu g_{\nu}\left(0\right)\right)\left(-2\nu g''_{\nu}\left(0\right)g_{\nu}\left(0\right)-2\nu\left(g'_{\nu}\left(0\right)\right)^{2}-2\nu g''_{\nu}\left(0\right)g_{\rho}\left(0\right)\right. \tag{255}$$

$$+2\nu g'_{\nu}\left(0\right)g'\left(0\right)g_{\rho}\left(0\right)-2\nu g'_{\nu}\left(0\right)g'_{\rho}\left(0\right)$$

$$+g''_{\nu}\left(0\right)+2\left(g'\left(0\right)\right)^{2}g_{\rho}\left(0\right)-g''\left(0\right)g_{\rho}\left(0\right)$$

$$\left.-2g'\left(0\right)g'_{\rho}\left(0\right)+g''_{\rho}\left(0\right)\right)$$

$$=\ \frac{1}{2}\exp\left(-\nu g_{\nu}\left(0\right)\right)\left(g''_{\nu}\left(0\right)\left(-2\nu g_{\nu}\left(0\right)-2\nu g_{\rho}\left(0\right)+1\right)\right. \tag{256}$$

$$+\nu g'_{\nu}\left(0\right)\left(-2g'_{\nu}\left(0\right)+2g'\left(0\right)g_{\rho}\left(0\right)-2g'_{\rho}\left(0\right)\right)$$

$$+2\left(g'\left(0\right)\right)^{2}g_{\rho}\left(0\right)-g''\left(0\right)g_{\rho}\left(0\right)$$

$$\left.-2g'\left(0\right)g'_{\rho}\left(0\right)+g''_{\rho}\left(0\right)\right)$$

$$=\ \frac{1}{2}\exp\left(-\nu\left(-\delta\ln x_{1}-\left(1-\delta\right)\ln x_{2}\right)\right) \tag{257}$$

$$\left(\left(-\frac{1}{3}\left(1-2\delta\right)\delta\left(1-\delta\right)\left(\ln x_{1}-\ln x_{2}\right)^{3}\right)\right.$$

$$\left(-2\nu\left(-\delta\ln x_{1}-\left(1-\delta\right)\ln x_{2}\right)-2\nu\left(\delta\ln x_{1}+\left(1-\delta\right)\ln x_{2}\right)+1\right)$$

$$+\nu\frac{\delta\left(1-\delta\right)}{2}\left(\ln x_{1}-\ln x_{2}\right)^{2}\left(-2\frac{\delta\left(1-\delta\right)}{2}\left(\ln x_{1}-\ln x_{2}\right)^{2}\right.$$

$$+2\left(-\delta\ln x_{1}-\left(1-\delta\right)\ln x_{2}\right)\left(\delta\ln x_{1}+\left(1-\delta\right)\ln x_{2}\right)$$

$$\left.-2\left(-\delta\left(\ln x_{1}\right)^{2}-\left(1-\delta\right)\left(\ln x_{2}\right)^{2}\right)\right)$$

$$+2\left(-\delta\ln x_{1}-\left(1-\delta\right)\ln x_{2}\right)^{2}\left(\delta\ln x_{1}+\left(1-\delta\right)\ln x_{2}\right)$$

$$- \left(\delta \left(\ln x_1 \right)^2 + (1 - \delta) \left(\ln x_2 \right)^2 \right) \left(\delta \ln x_1 + (1 - \delta) \ln x_2 \right)$$

$$-2 \left(-\delta \ln x_1 - (1 - \delta) \ln x_2 \right) \left(-\delta \left(\ln x_1 \right)^2 - (1 - \delta) \left(\ln x_2 \right)^2 \right)$$

$$+\delta \left(\ln x_1 \right)^3 + (1 - \delta) \left(\ln x_2 \right)^3 \bigg)$$

$$
\begin{aligned}
= \quad & \frac{1}{2} x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(-\frac{1}{3} (1 - 2\delta) \, \delta \, (1 - \delta) \left(\ln x_1 - \ln x_2 \right)^3 \right. & (258) \\
& (2\nu\delta \ln x_1 + 2\nu (1 - \delta) \ln x_2 - 2\nu\delta \ln x_1 - 2\nu (1 - \delta) \ln x_2 + 1) \\
& + \frac{1}{2} \nu\delta (1 - \delta) \left((\ln x_1)^2 - 2 \ln x_1 \ln x_2 + (\ln x_2)^2 \right) \\
& \left(-\delta (1 - \delta) (\ln x_1)^2 + 2\delta (1 - \delta) \ln x_1 \ln x_2 - \delta (1 - \delta) (\ln x_2)^2 \right. \\
& -2\delta^2 (\ln x_1)^2 - 4\delta (1 - \delta) \ln x_1 \ln x_2 - 2 (1 - \delta)^2 (\ln x_2)^2 \\
& \left. +2\delta (\ln x_1)^2 + 2 (1 - \delta) (\ln x_2)^2 \right) \\
& +2\delta^3 (\ln x_1)^3 + 6\delta^2 (1 - \delta) (\ln x_1)^2 \ln x_2 + 6\delta (1 - \delta)^2 \ln x_1 (\ln x_2)^2 \\
& +2 (1 - \delta)^3 (\ln x_2)^3 - \delta^2 (\ln x_1)^3 - \delta (1 - \delta) (\ln x_1)^2 \ln x_2 \\
& -\delta (1 - \delta) \ln x_1 (\ln x_2)^2 - (1 - \delta)^2 (\ln x_2)^3 - 2\delta^2 (\ln x_1)^3 \\
& -2\delta (1 - \delta) \ln x_1 (\ln x_2)^2 - 2\delta (1 - \delta) (\ln x_1)^2 \ln x_2 - 2 (1 - \delta)^2 (\ln x_2)^3 \\
& \left. +\delta (\ln x_1)^3 + (1 - \delta) (\ln x_2)^3 \right)
\end{aligned}
$$

$$
\begin{aligned}
= \quad & \frac{1}{2} x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(-\frac{1}{3} (1 - 2\delta) \, \delta \, (1 - \delta) \left(\ln x_1 - \ln x_2 \right)^3 \right. & (259) \\
& + \left(\frac{1}{2} \nu\delta (1 - \delta) (\ln x_1)^2 - \nu\delta (1 - \delta) \ln x_1 \ln x_2 + \frac{1}{2} \nu\delta (1 - \delta) (\ln x_2)^2 \right) \\
& \left(\left(\left(-\delta (1 - \delta) - 2\delta^2 + 2\delta \right) (\ln x_1)^2 \right. \right. \\
& + (2\delta (1 - \delta) - 4\delta (1 - \delta)) \ln x_1 \ln x_2 \\
& \left. + \left(-\delta (1 - \delta) - 2 (1 - \delta)^2 + 2 (1 - \delta) \right) (\ln x_2)^2 \right) \\
& + \left(2\delta^3 - \delta^2 - 2\delta^2 + \delta \right) (\ln x_1)^3 \\
& + \left(6\delta^2 (1 - \delta) - \delta (1 - \delta) - 2\delta (1 - \delta) \right) (\ln x_1)^2 \ln x_2 \\
& + \left(6\delta (1 - \delta)^2 - \delta (1 - \delta) - 2\delta (1 - \delta) \right) \ln x_1 (\ln x_2)^2 \\
& \left. + \left(2 (1 - \delta)^3 - (1 - \delta)^2 - 2 (1 - \delta)^2 + (1 - \delta) \right) (\ln x_2)^3 \right)
\end{aligned}
$$

$$
\begin{aligned}
= \quad & \frac{1}{2} x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(-\frac{1}{3} (1 - 2\delta) \, \delta \, (1 - \delta) \left(\ln x_1 - \ln x_2 \right)^3 \right. & (260) \\
& + \left(\frac{1}{2} \nu\delta (1 - \delta) (\ln x_1)^2 - \nu\delta (1 - \delta) \ln x_1 \ln x_2 + \frac{1}{2} \nu\delta (1 - \delta) (\ln x_2)^2 \right)
\end{aligned}
$$

$$\left(\left(-\delta + \delta^2 - 2\delta^2 + 2\delta\right)(\ln x_1)^2\right.$$
$$-2\delta\left(1 - \delta\right)\ln x_1 \ln x_2$$
$$+\left(-\delta + \delta^2 - 2 + 4\delta - 2\delta^2 + 2 - 2\delta\right)(\ln x_2)^2\right)$$
$$+\left(2\delta^3 - 3\delta^2 + \delta\right)(\ln x_1)^3$$
$$+\left(6\delta^2 - 6\delta^3 - \delta + \delta^2 - 2\delta + 2\delta^2\right)(\ln x_1)^2 \ln x_2$$
$$+\left(6\delta - 12\delta^2 + 6\delta^3 - \delta + \delta^2 - 2\delta + 2\delta^2\right)\ln x_1 (\ln x_2)^2$$
$$+\left(2 - 6\delta + 6\delta^2 - 2\delta^3 - 1 + 2\delta - \delta^2 - 2 + 4\delta - 2\delta^2 + 1 - \delta\right)(\ln x_2)^3\right)$$

$$= \quad \frac{1}{2}\, x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(-\frac{1}{3}\left(1 - 2\delta\right)\delta\left(1 - \delta\right)(\ln x_1 - \ln x_2)^3 \right. \tag{261}$$
$$+\left(\frac{1}{2}\nu\delta\left(1 - \delta\right)(\ln x_1)^2 - \nu\delta\left(1 - \delta\right)\ln x_1 \ln x_2 + \frac{1}{2}\nu\delta\left(1 - \delta\right)(\ln x_2)^2\right)$$
$$\left(\left(-\delta^2 + \delta\right)(\ln x_1)^2 - 2\delta\left(1 - \delta\right)\ln x_1 \ln x_2 + \left(-\delta^2 + \delta\right)(\ln x_2)^2\right)$$
$$+\left(2\delta^3 - 3\delta^2 + \delta\right)(\ln x_1)^3 + \left(-6\delta^3 + 9\delta^2 - 3\delta\right)(\ln x_1)^2 \ln x_2$$
$$+\left(6\delta^3 - 9\delta^2 + 3\delta\right)\ln x_1 (\ln x_2)^2 + \left(-2\delta^3 + 3\delta^2 - \delta\right)(\ln x_2)^3\right)$$

$$= \quad \frac{1}{2}\, x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(-\frac{1}{3}\left(1 - 2\delta\right)\delta\left(1 - \delta\right)(\ln x_1)^3 \right. \tag{262}$$
$$+\left(1 - 2\delta\right)\delta\left(1 - \delta\right)(\ln x_1)^2 \ln x_2 - \left(1 - 2\delta\right)\delta\left(1 - \delta\right)\ln x_1 (\ln x_2)^2$$
$$+\frac{1}{3}\left(1 - 2\delta\right)\delta\left(1 - \delta\right)(\ln x_2)^3$$
$$+\left(\frac{1}{2}\nu\delta\left(1 - \delta\right)(\ln x_1)^2 - \nu\delta\left(1 - \delta\right)\ln x_1 \ln x_2 + \frac{1}{2}\nu\delta\left(1 - \delta\right)(\ln x_2)^2\right)$$
$$\left(\delta\left(1 - \delta\right)(\ln x_1)^2 - 2\delta\left(1 - \delta\right)\ln x_1 \ln x_2 + \delta\left(1 - \delta\right)(\ln x_2)^2\right)$$
$$+\delta\left(1 - \delta\right)\left(1 - 2\delta\right)(\ln x_1)^3 - 3\delta\left(1 - \delta\right)\left(1 - 2\delta\right)(\ln x_1)^2 \ln x_2$$
$$+3\delta\left(1 - \delta\right)\left(1 - 2\delta\right)\ln x_1 (\ln x_2)^2 - \delta\left(1 - \delta\right)\left(1 - 2\delta\right)(\ln x_2)^3\right)$$

$$= \quad \frac{1}{2}\, x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(\frac{1}{2}\nu\delta^2\left(1 - \delta\right)^2 (\ln x_1)^4 - \nu\delta^2\left(1 - \delta\right)^2 (\ln x_1)^3 \ln x_2 \right. \tag{263}$$
$$+\frac{1}{2}\nu\delta^2\left(1 - \delta\right)^2 (\ln x_1)^2 (\ln x_2)^2 - \nu\delta^2\left(1 - \delta\right)^2 (\ln x_1)^3 \ln x_2$$
$$+2\nu\delta^2\left(1 - \delta\right)^2 (\ln x_1)^2 (\ln x_2)^2 - \nu\delta^2\left(1 - \delta\right)^2 \ln x_1 (\ln x_2)^3$$
$$+\frac{1}{2}\nu\delta^2\left(1 - \delta\right)^2 (\ln x_1)^2 (\ln x_2)^2 - \nu\delta^2\left(1 - \delta\right)^2 \ln x_1 (\ln x_2)^3$$
$$+\frac{1}{2}\nu\delta^2\left(1 - \delta\right)^2 (\ln x_2)^4$$

$$+ \left(-\frac{1}{3} (1 - 2\delta) \delta (1 - \delta) + \delta (1 - \delta) (1 - 2\delta) \right) (\ln x_1)^3$$
$$+ ((1 - 2\delta) \delta (1 - \delta) - 3\delta (1 - \delta) (1 - 2\delta)) (\ln x_1)^2 \ln x_2$$
$$+ (- (1 - 2\delta) \delta (1 - \delta) + 3\delta (1 - \delta) (1 - 2\delta)) \ln x_1 (\ln x_2)^2$$
$$+ \left(\frac{1}{3} (1 - 2\delta) \delta (1 - \delta) - \delta (1 - \delta) (1 - 2\delta) \right) (\ln x_2)^3 \Bigg)$$

$$
\begin{aligned}
= \ & \frac{1}{2} x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(\frac{1}{2}\nu\delta^2 (1 - \delta)^2 (\ln x_1)^4 - 2\nu\delta^2 (1 - \delta)^2 (\ln x_1)^3 \ln x_2 \right. \\
& + 3\nu\delta^2 (1 - \delta)^2 (\ln x_1)^2 (\ln x_2)^2 \\
& - 2\nu\delta^2 (1 - \delta)^2 \ln x_1 (\ln x_2)^3 + \frac{1}{2}\nu\delta^2 (1 - \delta)^2 (\ln x_2)^4 \\
& + \frac{2}{3}\delta (1 - \delta) (1 - 2\delta) (\ln x_1)^3 - 2\delta (1 - \delta) (1 - 2\delta) (\ln x_1)^2 \ln x_2 \\
& \left. + 2\delta (1 - \delta) (1 - 2\delta) \ln x_1 (\ln x_2)^2 - \frac{2}{3}\delta (1 - \delta) (1 - 2\delta) (\ln x_2)^3 \right)
\end{aligned}
\tag{264}
$$

$$
\begin{aligned}
= \ & \frac{1}{2} x_1^{\nu\delta} x_2^{\nu(1-\delta)} \left(\frac{1}{2}\nu\delta^2 (1 - \delta)^2 \left((\ln x_1)^4 - 4 (\ln x_1)^3 \ln x_2 \right. \right. \\
& + 6 (\ln x_1)^2 (\ln x_2)^2 - 4 \ln x_1 (\ln x_2)^3 + (\ln x_2)^4 \\
& + \frac{2}{3}\delta (1 - \delta) (1 - 2\delta) \\
& \left. \left. \left((\ln x_1)^3 - 3 (\ln x_1)^2 \ln x_2 + 3 \ln x_1 (\ln x_2)^2 - (\ln x_2)^3 \right) \right) \right.
\end{aligned}
\tag{265}
$$

$$
\begin{aligned}
= \ & \delta (1 - \delta) \, x_1^{\nu\delta} x_2^{\nu(1-\delta)} \\
& \left(\frac{1}{3} (1 - 2\delta) (\ln x_1 - \ln x_2)^3 + \frac{1}{4}\nu\delta (1 - \delta) (\ln x_1 - \ln x_2)^4 \right)
\end{aligned}
\tag{266}
$$

Hence, we can approximate $\partial y / \partial \rho$ by a second-order Taylor series approximation:

$$
\begin{aligned}
\frac{\partial y}{\partial \rho} \ \approx \ & \gamma \, e^{\lambda t} \, \nu \, \left(f_\rho (0) + \rho \, f_\rho' (0) \right)
\end{aligned}
\tag{267}
$$

$$
\begin{aligned}
= \ & -\frac{1}{2}\gamma \, e^{\lambda t} \, \nu \, \delta \, (1 - \delta) \, x_1^{\nu\delta} \, x_2^{\nu(1-\delta)} (\ln x_1 - \ln x_2)^2 \\
& + \gamma \, e^{\lambda t} \, \nu \, \rho \, \delta \, (1 - \delta) \, x_1^{\nu\delta} x_2^{\nu(1-\delta)} \\
& \left(\frac{2}{3} (1 - 2\delta) (\ln x_1 - \ln x_2)^3 + \frac{1}{2}\nu\delta (1 - \delta) (\ln x_1 - \ln x_2)^4 \right)
\end{aligned}
\tag{268}
$$

$$
\begin{aligned}
= \ & \gamma \, e^{\lambda t} \, \nu \, \delta \, (1 - \delta) \, x_1^{\nu\delta} \, x_2^{\nu(1-\delta)} \left(-\frac{1}{2} (\ln x_1 - \ln x_2)^2 \right. \\
\\
& \left. + \frac{1}{3}\rho (1 - 2\delta) (\ln x_1 - \ln x_2)^3 + \frac{1}{4}\rho\nu\delta (1 - \delta) (\ln x_1 - \ln x_2)^4 \right)
\end{aligned}
\tag{269}
$$

B. Three-input nested CES function

The nested CES function with three inputs is defined as

$$y = \gamma \, e^{\lambda t} \, B^{-1/\rho} \tag{270}$$

with

$$B = \delta \, B_1^{\rho/\rho_1} + (1 - \delta)x_3^{-\rho} \tag{271}$$

$$B_1 = \delta_1 x_1^{-\rho_1} + (1 - \delta_1) \, x_2^{-\rho_1} \tag{272}$$

For further simplification of the formulas in this section, we make the following definition:

$$L_1 = \delta_1 \ln(x_1) + (1 - \delta_1) \ln(x_2) \tag{273}$$

B.1.   Limits for ρ_1 and/or ρ approaching zero

The limits of the three-input nested CES function for ρ_1 and/or ρ approaching zero are:

$$\lim_{\rho_1 \to 0} y = \gamma \, e^{\lambda t} \left(\delta \, \{\exp(L_1)\}^{-\rho} + (1 - \delta)x_3^{-\rho} \right)^{-\frac{\nu}{\rho}} \tag{274}$$

$$\lim_{\rho \to 0} y = \gamma \, e^{\lambda t} \, \exp \left\{ \nu \left(\delta \left(-\frac{\ln(B_1)}{\rho_1} \right) + (1 - \delta) \ln x_3 \right) \right\} \tag{275}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} y = \gamma \, e^{\lambda t} \, \exp \left\{ \nu \left(\delta \, L_1 + (1 - \delta) \ln x_3 \right) \right\} \tag{276}$$

B.2. Derivatives with respect to coefficients

The partial derivatives of the three-input nested CES function with respect to the coefficients are:[23]

$$\frac{\partial y}{\partial \gamma} = e^{\lambda t} \, B^{-\nu/\rho} \tag{277}$$

$$\frac{\partial y}{\partial \delta_1} = -\gamma \, e^{\lambda t} \, \frac{\nu}{\rho} B^{\frac{-\nu-\rho}{\rho}} \delta \frac{\rho}{\rho_1} B_1^{\frac{\rho-\rho_1}{\rho_1}} (x_1^{-\rho_1} - x_2^{-\rho_1}) \tag{278}$$

$$\frac{\partial y}{\partial \delta} = -\gamma \, e^{\lambda t} \, \frac{\nu}{\rho} B^{\frac{-\nu-\rho}{\rho}} \left[B_1^{\frac{\rho}{\rho_1}} - x_3^{-\rho} \right] \tag{279}$$

$$\frac{\partial y}{\partial \nu} = -\gamma \, e^{\lambda t} \, \frac{1}{\rho} \ln(B) B^{-\frac{\nu}{\rho}} \tag{280}$$

$$\frac{\partial y}{\partial \rho_1} = -\gamma \, e^{\lambda t} \, \frac{\nu}{\rho} B^{\frac{-\nu-\rho}{\rho}} \delta \tag{281}$$

---

[23]The partial derivatives with respect to λ are always calculated using equation 16, where $\partial y/\partial \gamma$ is calculated according to equation 277, 283, 289, or 295 depending on the values of ρ_1 and ρ.

$$\left[\ln(B_1)B_1^{\frac{\rho}{\rho_1}}\left(-\frac{\rho}{\rho_1^2}\right)+\frac{\rho}{\rho_1}B_1\left(-\delta_1\ln x_1 x_1^{\rho_1}-(1-\delta_1)\ln x_2 x_2^{-\rho_1}\right)\right]^{\frac{\rho-\rho_1}{\rho_1}}$$

$$\frac{\partial y}{\partial \rho}=\gamma\,e^{\lambda t}\,\frac{\nu}{\rho^2}\ln(B)B^{-\frac{\nu}{\rho}}-\gamma\,e^{\lambda t}\,\frac{\nu}{\rho}B^{\frac{-\nu-\rho}{\rho}}\left\{\delta\ln(B_1)B_1^{\frac{\rho}{\rho_1}}\frac{1}{\rho_1}-(1-\delta)\ln x_3 x_3^{-\rho}\right\} \qquad (282)$$

Limits of the derivatives for ρ approaching zero

$$\lim_{\rho\to 0}\frac{\partial y}{\partial \gamma}=e^{\lambda t}\,\exp\left\{\nu\left(\delta\left(-\frac{\ln(B_1)}{\rho_1}\right)+(1-\delta)\ln x_3\right)\right\} \qquad (283)$$

$$\lim_{\rho\to 0}\frac{\partial y}{\partial \delta_1}=-\gamma\,e^{\lambda t}\,\nu\left[\left(\delta\frac{(x_1^{-\rho_1}-x_2^{-\rho_1})}{\rho_1 B_1}\right)\exp\left\{-\nu\left(-\delta\left(-\frac{\ln(B_1)}{\rho_1}\right)-(1-\delta)\ln x_3\right)\right\}\right] \qquad (284)$$

$$\lim_{\rho\to 0}\frac{\partial y}{\partial \delta}=-\gamma\,e^{\lambda t}\,\nu\left[\left(\frac{\ln(B_1)}{\rho_1}+\ln x_3\right)\exp\left\{\nu\left(-\delta\left(-\frac{\ln(B_1)}{\rho_1}\right)-(1-\delta)\ln x_3\right)\right\}\right] \qquad (285)$$

$$\lim_{\rho\to 0}\frac{\partial y}{\partial \nu}=\gamma\,e^{\lambda t}\left(\delta\left(-\frac{\ln(B_1)}{\rho_1}\right)+(1-\delta)\ln x_3\right)$$
$$\cdot\exp\left\{-\nu\left(-\delta\left(-\frac{\ln(B_1)}{\rho_1}\right)-(1-\delta)\ln x_3\right)\right\} \qquad (286)$$

$$\lim_{\rho\to 0}\frac{\partial y}{\partial \rho_1}=-\gamma\,e^{\lambda t}\,\nu\frac{\delta}{\rho_1}\left(-\frac{\ln(B_1)}{\rho_1}+\frac{-\delta_1\ln x_1 x_1^{-\rho_1}-(1-\delta_1)\ln x_2 x_2^{-\rho_1}}{B_1}\right)$$
$$\cdot\exp\left\{-\nu\left(-\delta\left(-\frac{\ln(B_1)}{\rho_1}\right)-(1-\delta)\ln x_3\right)\right\} \qquad (287)$$

$$\lim_{\rho\to 0}\frac{\partial y}{\partial \rho}=\gamma\,e^{\lambda t}\,\nu\left[-\frac{1}{2}\left(\delta\left(\frac{\ln B_1}{\rho_1}\right)^2+(1-\delta)(\ln x_3)^2\right)+\frac{1}{2}\left(\delta\frac{\ln B_1}{\rho_1}-(1-\delta)\ln x_3\right)^2\right] \qquad (288)$$
$$\cdot\exp\left(\nu\left(-\delta\frac{\ln B_1}{\rho_1}+(1-\delta)\ln x_3\right)\right)$$

Limits of the derivatives for ρ_1 approaching zero

$$\lim_{\rho_1\to 0}\frac{\partial y}{\partial \gamma}=e^{\lambda t}\left(\delta\cdot\{\exp(L_1)\}^{-\rho}+(1-\delta)x_3^{-\rho}\right)^{-\frac{\nu}{\rho}} \qquad (289)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \delta_1} = -\gamma \, e^{\lambda t} \, \nu \, \delta \left(\delta \left\{ \exp(L_1) \right\}^{-\rho} + (1 - \delta) x_3^{-\rho} \right)^{-\frac{\nu}{\rho} - 1} \left\{ \exp(L_1) \right\}^{-\rho} (\ln x_1 - \ln x_2) \quad (290)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \delta} = -\gamma \, e^{\lambda t} \, \frac{\nu}{\rho} \left(\delta \left\{ \exp(L_1) \right\}^{-\rho} + (1 - \delta) x_3^{-\rho} \right)^{-\frac{\nu}{\rho} - 1} \left(\left\{ \exp(L_1) \right\}^{-\rho} - x_3^{-\rho} \right) \quad (291)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \nu} = -\gamma \, e^{\lambda t} \, \frac{1}{\rho} \ln \left(\delta \left\{ \exp(L_1) \right\}^{-\rho} + (1 - \delta) x_3^{-\rho} \right) \left(\delta \left\{ \exp(L_1) \right\}^{-\rho} + (1 - \delta) x_3^{-\rho} \right)^{-\frac{\nu}{\rho}}$$

$$(292)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \rho_1} = -\frac{1}{2} \gamma \, e^{\lambda t} \, \nu \, \delta \, \exp\left(-\rho \, L_1 \right) \left(\left(\delta_1 \left(\ln x_1 \right)^2 + (1 - \delta_1) \left(\ln x_2 \right)^2 \right) - L_1^2 \right) \quad (293)$$

$$\left(\delta \exp\left(-\rho \, L_1 \right) + (1 - \delta) \, x_3^{-\rho} \right)^{-\frac{\nu + \rho}{\rho}}$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \rho} = \gamma \, e^{\lambda t} \, \frac{\nu}{\rho^2} \ln \left(\delta \left\{ \exp(L_1) \right\}^{-\rho} + (1 - \delta) x_3^{-\rho} \right) \quad (294)$$

$$\left(\delta \left\{ \exp(L_1) \right\}^{-\rho} + (1 - \delta) x_3^{-\rho} \right)^{-\frac{\nu}{\rho}}$$

$$- \gamma \, e^{\lambda t} \, \frac{\nu}{\rho} \left(\delta \left\{ \exp(L_1) \right\}^{-\rho} + (1 - \delta) x_3^{-\rho} \right)^{-\frac{\nu}{\rho} - 1}$$

$$\left(-\delta L_1 \left\{ \exp(L_1) \right\}^{-\rho} - (1 - \delta) \ln x_3 x_3^{-\rho} \right)$$

Limits of the derivatives for ρ_1 and ρ approaching zero

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \gamma} = e^{\lambda t} \, \exp\{\nu(\delta \, L_1 + (1 - \delta) \ln x_3)\} \quad (295)$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta_1} = -\gamma \, e^{\lambda t} \, \nu \, \delta \left(-\ln x_1 + \ln x_2 \right) \exp\{-\nu(-\delta \, L_1 - (1 - \delta) \ln x_3)\} \quad (296)$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta} = -\gamma \, e^{\lambda t} \, \nu \left(-L_1 + \ln x_3 \right) \exp\{\nu(\delta \, L_1 + (1 - \delta) \ln x_3)\} \quad (297)$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \nu} = \gamma \, e^{\lambda t} \left(\delta \, L_1 + (1 - \delta) \ln x_3 \right) \exp\left\{ -\nu(-\delta \, L_1 - (1 - \delta) \ln x_3) \right\} \quad (298)$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho_1} = \frac{1}{2} \gamma \, e^{\lambda \, t} \, \nu \, \delta \, \left(\delta_1 (\ln x_1)^2 + (1 - \delta_1)(\ln x_2)^2 - L_1^2 \right) \tag{299}$$
$$\cdot \exp\left(-\nu \left(\delta \, L_1 + (1 - \delta) \ln x_3 \right) \right)$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho} = \gamma \, e^{\lambda \, t} \, \nu \, \left[-\frac{1}{2} \left(\delta \, L_1^2 + (1 - \delta)(\ln x_3)^2 \right) + \frac{1}{2} \left(-\delta \, L_1 - (1 - \delta) \ln x_3 \right)^2 \right] \tag{300}$$
$$\cdot \exp\left\{ \nu \left(\delta \, L_1 + (1 - \delta) \ln x_3 \right) \right\}$$

B.3. Elasticities of substitution

Allen-Uzawa elasticity of substitution (see Sato 1967)

$$\sigma_{i,j} = \begin{cases} (1 - \rho)^{-1} & \text{for } i = 1, 2; \ j = 3 \\[2em] \dfrac{(1 - \rho_1)^{-1} - (1 - \rho)^{-1}}{\delta \left(\dfrac{y}{B_1^{-\frac{1}{\rho_1}}} \right)^{1 + \rho}} + (1 - \rho)^{-1} & \text{for } i = 1; \ j = 2 \end{cases} \tag{301}$$

Hicks-McFadden elasticity of substitution (see Sato 1967)

$$\sigma_{i,j} = \begin{cases} \dfrac{\dfrac{1}{\theta_i} + \dfrac{1}{\theta_j}}{(1 - \rho_1) \left(\dfrac{1}{\theta_i} - \dfrac{1}{\theta^*} \right) + (1 - \rho_2) \left(\dfrac{1}{\theta_j} - \dfrac{1}{\theta} \right) + (1 - \rho) \left(\dfrac{1}{\theta^*} - \dfrac{1}{\theta} \right)} & \text{for } i = 1, 2; \ j = 3 \\[2em] (1 - \rho_1)^{-1} & \text{text } i = 1; \ j = 2 \end{cases} \tag{302}$$

with

$$\theta^* = \delta B_1^{\frac{\rho}{\rho_1}} \cdot y^\rho \tag{303}$$

$$\theta = (1 - \delta) x_3^{-\rho} \cdot y^\rho \tag{304}$$

$$\theta_1 = \delta \delta_1 x_1^{-\rho_1} B_1^{-\frac{\rho_1 - \rho}{\rho_1}} \cdot y^\rho \tag{305}$$

$$\theta_2 = \delta (1 - \delta_1) x_2^{-\rho_1} B_1^{-\frac{\rho_1 - \rho}{\rho_1}} \cdot y^\rho \tag{306}$$

$$\theta_3 = \theta \tag{307}$$

C. Four-input nested CES function

The nested CES function with four inputs is defined as

$$y = \gamma \, e^{\lambda t} \, B^{-\nu/\rho} \tag{308}$$

with

$$B = \delta \, B_1^{\rho/\rho_1} + (1 - \delta) B_2^{\rho/\rho_2} \tag{309}$$

$$B_1 = \delta_1 x_1^{-\rho_1} + (1 - \delta_1) x_2^{-\rho_1} \tag{310}$$

$$B_2 = \delta_2 x_3^{-\rho_2} + (1 - \delta_2) x_4^{-\rho_2} \tag{311}$$

For further simplification of the formulas in this section, we make the following definitions:

$$L_1 = \delta_1 \ln(x_1) + (1 - \delta_1) \ln(x_2) \tag{312}$$

$$L_2 = \delta_2 \ln(x_3) + (1 - \delta_2) \ln(x_4) \tag{313}$$

C.1.   Limits for ρ_1, ρ_2, and/or ρ approaching zero

The limits of the four-input nested CES function for ρ_1, ρ_2, and/or ρ approaching zero are:

$$\lim_{\rho \to 0} y = \gamma \, e^{\lambda t} \, \exp\left\{-\nu \left(\frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta)\ln(B_2)}{\rho_2}\right)\right\} \tag{314}$$

$$\lim_{\rho_1 \to 0} y = \gamma \, e^{\lambda t} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta)B_2^{\frac{\rho}{\rho_2}}\right)^{-\frac{\nu}{\rho}} \tag{315}$$

$$\lim_{\rho_2 \to 0} y = \gamma \, e^{\lambda t} \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta)\exp\{-\rho \, L_2\}\right)^{-\frac{\nu}{\rho}} \tag{316}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho_1 \to 0} y = \gamma \, e^{\lambda t} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta)\exp\{-\rho \, L_2\}\right)^{-\frac{\nu}{\rho}} \tag{317}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} y = \gamma \, e^{\lambda t} \, \exp\left\{-\nu \left(-\delta \, L_1 + (1-\delta)\frac{\ln(B_2)}{\rho_2}\right)\right\} \tag{318}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho \to 0} y = \gamma \, e^{\lambda t} \, \exp\left\{-\nu \left(\delta\frac{\ln(B_1)}{\rho_1} - (1-\delta)L_2\right)\right\} \tag{319}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \lim_{\rho \to 0} y = \gamma \, e^{\lambda t} \, \exp\left\{-\nu \left(-\delta \, L_1 - (1-\delta)L_2\right)\right\} \tag{320}$$

C.2. Derivatives with respect to coefficients

The partial derivatives of the four-input nested CES function with respect to the coefficients are:[24]

$$\frac{\partial y}{\partial \gamma} = e^{\lambda t} B^{-\nu/\rho} \tag{321}$$

$$\frac{\partial y}{\partial \delta_1} = \gamma \, e^{\lambda t} \left(-\frac{\nu}{\rho}\right) B^{\frac{-\nu-\rho}{\rho}} \frac{\rho}{\rho_1} \delta B_1^{\frac{\rho-\rho_1}{\rho_1}} (x_1^{-\rho_1} - x_2^{-\rho_1}) \tag{322}$$

$$\frac{\partial y}{\partial \delta_2} = \gamma \, e^{\lambda t} \left(-\frac{\nu}{\rho}\right) B^{\frac{-\nu-\rho}{\rho}} \frac{\rho}{\rho_2} (1-\delta) B_2^{\frac{\rho-\rho_2}{\rho_2}} (x_3^{-\rho_2} - x_4^{-\rho_2}) \tag{323}$$

$$\frac{\partial y}{\partial \delta} = \gamma \, e^{\lambda t} \left(-\frac{\nu}{\rho}\right) B^{-\frac{\nu}{\rho}-1} \left[B_1^{\rho/\rho_1} - B_2^{\rho/\rho_2}\right] \tag{324}$$

$$\frac{\partial y}{\partial \nu} = \gamma \, e^{\lambda t} \ln(B) B^{-\nu/\rho} \left(-\frac{1}{\rho}\right) \tag{325}$$

$$\frac{\partial y}{\partial \rho_1} = \gamma \, e^{\lambda t} \left(-\frac{\nu}{\rho}\right) B^{\frac{-\nu-\rho}{\rho}} \tag{326}$$
$$\left[\delta \ln(B_1) B_1^{\frac{\rho}{\rho_1}} \left(-\frac{\rho}{\rho_1^2}\right) + \delta B_1^{\frac{\rho-\rho_1}{\rho_1}} \frac{\rho}{\rho_1}(-\delta_1 \ln x_1 x_1^{-\rho_1} - (1-\delta_1)\ln x_2 x_2^{-\rho_1})\right]$$

$$\frac{\partial y}{\partial \rho_2} = \gamma \, e^{\lambda t} \left(-\frac{\nu}{\rho}\right) B^{\frac{-\nu-\rho}{\rho}} \tag{327}$$
$$\left[(1-\delta) \ln(B_2) B_2^{\frac{\rho}{\rho_2}} \left(-\frac{\rho}{\rho_2^2}\right) + (1-\delta) B_2^{\frac{\rho-\rho_2}{\rho_2}} \frac{\rho}{\rho_2}(-\delta_2 \ln x_3 x_3^{-\rho_2} - (1-\delta_2)\ln x_4 x_4^{-\rho_2})\right]$$

$$\frac{\partial y}{\partial \rho} = \gamma \, e^{\lambda t} \ln(B) B^{-\nu/\rho} \frac{\nu}{\rho^2} + \gamma \, e^{\lambda t} \left(-\frac{\nu}{\rho}\right) B^{\frac{-\nu-\rho}{\rho}} \tag{328}$$
$$\left[\delta \ln(B_1) B_1^{\frac{\rho}{\rho_1}} \frac{1}{\rho_1} + (1-\delta) \ln(B_2) B_2^{\frac{\rho}{\rho_2}} \frac{1}{\rho_2}\right]$$

Limits of the derivatives for ρ approaching zero

$$\lim_{\rho \to 0} \frac{\partial y}{\partial \gamma} = e^{\lambda t} \exp\left\{-\nu \left(\frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta) \ln(B_2)}{\rho_2}\right)\right\} \tag{329}$$

$$\lim_{\rho \to 0} \frac{\partial y}{\partial \delta_1} = \gamma \, e^{\lambda t} \left(-\frac{\nu}{\rho_1} \frac{\delta(x_1^{-\rho_1} - x_2^{-\rho_1})}{B_1}\right) \cdot \exp\left\{-\nu \left(\frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta) \ln(B_2)}{\rho_2}\right)\right\} \tag{330}$$

$$\lim_{\rho \to 0} \frac{\partial y}{\partial \delta_2} = \gamma \, e^{\lambda t} \left(-\frac{\nu}{\rho_2} \frac{(1-\delta)(x_3^{-\rho_2} - x_4^{-\rho_2})}{B_2}\right) \cdot \exp\left\{-\nu \left(\frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta) \ln(B_2)}{\rho_2}\right)\right\} \tag{331}$$

---

[24]The partial derivatives with respect to λ are always calculated using equation 16, where $\partial y/\partial \gamma$ is calculated according to equation 321, 329, 337, 345, 353, 361, 369, or 377 depending on the values of ρ_1, ρ_2, and ρ.

$$\lim_{\rho \to 0} \frac{\partial y}{\partial \delta} = \gamma \, e^{\lambda t} \left(-\nu \left[\frac{\ln(B_1)}{\rho_1} - \frac{\ln(B_2)}{\rho_2} \right] \right) \cdot \exp \left\{ -\nu \left(\frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta)\ln(B_2)}{\rho_2} \right) \right\} \quad (332)$$

$$\lim_{\rho \to 0} \frac{\partial y}{\partial \nu} = -\gamma \, e^{\lambda t} \frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta)\ln(B_2)}{\rho_2} \cdot \exp \left\{ -\nu \left(\frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta)\ln(B_2)}{\rho_2} \right) \right\} \quad (333)$$

$$\lim_{\rho \to 0} \frac{\partial y}{\partial \rho_1} = \gamma \, e^{\lambda t} \left(-\nu \left(\frac{\delta \rho_1 (-\delta_1 \ln x_1 x_1^{-\rho_1} - (1-\delta_1) \ln x_2 x_2^{-\rho_1})}{\rho_1^2 \, B_1} - \frac{\delta \ln(B_1)}{\rho_1^2} \right) \right) \quad (334)$$
$$\cdot \exp \left\{ -\nu \left(\frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta)\ln(B_2)}{\rho_2} \right) \right\}$$

$$\lim_{\rho \to 0} \frac{\partial y}{\partial \rho_2} = \gamma \, e^{\lambda t} \left(-\nu \left(\frac{(1-\delta)\rho_2 (-\delta_2 \ln x_3 x_3^{-\rho_2} - (1-\delta_2) \ln x_4 x_4^{-\rho_2})}{\rho_2^2 \, B_2} - \frac{(1-\delta)\ln(B_2)}{\rho_2^2} \right) \right)$$
$$(335)$$
$$\cdot \exp \left\{ -\nu \left(\frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta)\ln(B_2)}{\rho_2} \right) \right\}$$

$$\lim_{\rho \to 0} \frac{\partial y}{\partial \rho} = \gamma \, e^{\lambda t} \, \nu \left(-\frac{1}{2} \left(\delta (\ln B_1)^2 \frac{1}{\rho_1^2} + (1-\delta)(\ln B_2)^2 \frac{1}{\rho_2^2} \right) + \frac{1}{2} \left(\delta \ln B_1 \frac{1}{\rho_1} + (1-\delta) \ln B_2 \frac{1}{\rho_2} \right)^2 \right)$$
$$(336)$$
$$\cdot \exp \left\{ -\nu \left(\frac{\delta \ln(B_1)}{\rho_1} + \frac{(1-\delta)\ln(B_2)}{\rho_2} \right) \right\}$$

Limits of the derivatives for ρ_1 approaching zero

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \gamma} = e^{\lambda t} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right)^{-\frac{\nu}{\rho}} \quad (337)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \delta_1} = -\gamma \, e^{\lambda t} \frac{\nu}{\rho} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right)^{-\frac{\nu}{\rho}-1} \quad (338)$$
$$\cdot \delta \exp\{-\rho \, L_1\} \rho (-\ln x_1 + \ln x_2)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \delta_2} = -\gamma \, e^{\lambda t} \frac{\nu}{\rho} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right)^{-\frac{\nu}{\rho}-1} \quad (339)$$
$$\cdot (1-\delta) \frac{\rho}{\rho_2} B_2^{\frac{\rho}{\rho_2}-1} \left(x_3^{-\rho_2} - x_4^{-\rho_2} \right)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \delta} = -\gamma \, e^{\lambda \, t} \, \frac{\nu}{\rho} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right)^{-\frac{\nu}{\rho}-1} \tag{340}$$
$$\cdot \left(\exp\{-\rho \, L_1\} - B_2^{\frac{\rho}{\rho_2}} \right)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \nu} = -\gamma \, e^{\lambda \, t} \, \frac{1}{\rho} \ln \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right) \tag{341}$$
$$\left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right)^{-\frac{\nu}{\rho}}$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \rho_1} = -\gamma \, e^{\lambda \, t} \, \delta \, \nu \left(\delta \exp\left\{ \rho \left(-\delta_1 \ln x_1 - (1-\delta_1) \ln x_2 \right) \right\} + (1-\delta) \, B_2^{\frac{\rho}{\rho_2}} \right)^{-\frac{\nu}{\rho}-1} \tag{342}$$
$$\cdot \exp\left\{ -\rho \left(\delta_1 \ln x_1 + (1-\delta_1) \ln x_2 \right) \right\}$$
$$\left(-\frac{1}{2} \left(-\delta_1 \ln x_1 - (1-\delta_1) \ln x_2 \right)^2 + \frac{1}{2} \left(\delta_1 \left(\ln x_1 \right)^2 + (1-\delta_1) \left(\ln x_2 \right)^2 \right) \right)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \rho_2} = -\gamma \, e^{\lambda \, t} \, \frac{\nu}{\rho} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right)^{-\frac{\nu}{\rho}-1} \tag{343}$$
$$\left(-(1-\delta) \frac{\rho}{\rho_2^2} \ln(B_2) B_2^{\frac{\rho}{\rho_2}} + (1-\delta) \frac{\rho}{\rho_2} B_2^{\frac{\rho}{\rho_2}-1} \left(-\delta_2 \ln x_3 x_3^{-\rho_2} - (1-\delta_2) \ln x_4 x_4^{-\rho_2} \right) \right)$$

$$\lim_{\rho_1 \to 0} \frac{\partial y}{\partial \rho} = \gamma \, e^{\lambda \, t} \, \frac{\nu}{\rho^2} \ln \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right) \tag{344}$$
$$\left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right)^{-\frac{\nu}{\rho}}$$
$$- \gamma \, e^{\lambda \, t} \, \frac{\nu}{\rho} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) B_2^{\frac{\rho}{\rho_2}} \right)^{-\frac{\nu}{\rho}-1}$$
$$\left(-\delta \exp\{-\rho \, L_1\} L_1 + \frac{1-\delta}{\rho_2} \ln(B_2) B_2^{\frac{\rho}{\rho_2}} \right)$$

Limits of the derivatives for ρ_2 approaching zero

$$\lim_{\rho_2 \to 0} \frac{\partial y}{\partial \gamma} = e^{\lambda \, t} \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho \, L_2\} \right)^{-\frac{\nu}{\rho}} \tag{345}$$

$$\lim_{\rho_2 \to 0} \frac{\partial y}{\partial \delta_1} = -\gamma \, e^{\lambda \, t} \, \frac{\nu}{\rho} \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho \, L_2\} \right)^{-\frac{\nu}{\rho}-1} \delta \frac{\rho}{\rho_1} B_1^{\frac{\rho}{\rho_1}-1} \left(x_1^{-\rho_1} - x_2^{-\rho_1} \right) \tag{346}$$

$$\lim_{\rho_2 \to 0} \frac{\partial y}{\partial \delta_2} = -\gamma\, e^{\lambda t}\, \nu \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho\, L_2\} \right)^{-\frac{\nu}{\rho} - 1} \tag{347}$$

$$(1-\delta) \exp\{-\rho\, L_2\} \left(-\ln x_3 + \ln x_4 \right)$$

$$\lim_{\rho_2 \to 0} \frac{\partial y}{\partial \delta} = -\gamma\, e^{\lambda t}\, \frac{\nu}{\rho} \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho\, L_2\} \right)^{-\frac{\nu}{\rho} - 1} \tag{348}$$

$$\left(B_1^{\frac{\rho}{\rho_1}} - \exp\{-\rho\, L_2\} \right)$$

$$\lim_{\rho_2 \to 0} \frac{\partial y}{\partial \nu} = -\gamma\, e^{\lambda t}\, \frac{1}{\rho} \ln \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho\, L_2\} \right) \tag{349}$$

$$\left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho\, L_2\} \right)^{-\frac{\nu}{\rho}}$$

$$\lim_{\rho_2 \to 0} \frac{\partial y}{\partial \rho_1} = -\gamma\, e^{\lambda t}\, \frac{\nu}{\rho} \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho\, L_2\} \right)^{-\frac{\nu}{\rho} - 1} \tag{350}$$

$$\left(\delta \ln(B_1) B_1^{\frac{\rho}{\rho_1}} \left(-\frac{\rho}{\rho_1^2} \right) + \delta \left(\frac{\rho}{\rho_1} \right) B_1^{\frac{\rho}{\rho_1} - 1} \left(-\delta_1 \ln x_1 x_1^{-\rho_1} - (1-\delta_1) \ln x_2 x_2^{-\rho_1} \right) \right)$$

$$\lim_{\rho_2 \to 0} \frac{\partial y}{\partial \rho_2} = -\gamma\, e^{\lambda t}\, (1-\delta)\, \nu\, \exp\left\{ -\rho\, (\delta_2 \ln x_3 + (1-\delta_2) \ln x_4) \right\} \tag{351}$$

$$\left((1-\delta) \exp\left\{ \rho\, (-\delta_2 \ln x_3 - (1-\delta_2) \ln x_4) \right\} + \delta \left(\delta_1 x_1^{-\rho_1} + (1-\delta_1) x_2^{-\rho_1} \right)^{\frac{\rho}{\rho_1}} \right)^{-\frac{\nu}{\rho} - 1}$$

$$\left(-\frac{1}{2} \left(\delta_2 \ln x_3 + (1-\delta_2) \ln x_4 \right)^2 + \frac{1}{2} \left(\delta_2 (\ln x_3)^2 + (1-\delta_2)(\ln x_4)^2 \right) \right)$$

$$\lim_{\rho_2 \to 0} \frac{\partial y}{\partial \rho} = \gamma\, e^{\lambda t}\, \frac{\nu}{\rho^2} \ln \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho\, L_2\} \right) \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho\, L_2\} \right)^{-\frac{\nu}{\rho}} \tag{352}$$

$$- \gamma\, e^{\lambda t}\, \frac{\nu}{\rho} \left(\delta B_1^{\frac{\rho}{\rho_1}} + (1-\delta) \exp\{-\rho\, L_2\} \right)^{-\frac{\nu}{\rho} - 1}$$

$$\left(-(1-\delta) \exp\{-\rho\, L_2\} L_2 + \delta\, \ln(B_1)\, B_1^{\frac{\rho}{\rho_1}} \frac{1}{\rho_1} \right)$$

Limits of the derivatives for ρ_1 and ρ_2 approaching zero

$$\lim_{\rho_2 \to 0} \lim_{\rho_1 \to 0} \frac{\partial y}{\partial \gamma} = e^{\lambda t} \exp\left\{ -\frac{\nu}{\rho} \ln \left(\delta \exp\{-\rho\, L_1\} + (1-\delta) \exp\{-\rho\, L_2\} \right) \right\} \tag{353}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho_1 \to 0} \frac{\partial y}{\partial \delta_1} = -\gamma \, e^{\lambda t} \, \nu \, \delta \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) \exp\{-\rho \, L_2\}\right)^{-\frac{\nu}{\rho}-1} \tag{354}$$
$$\exp\{-\rho \, L_1\} \left(-\ln x_1 + \ln x_2\right)$$

$$\lim_{\rho_2 \to 0} \lim_{\rho_1 \to 0} \frac{\partial y}{\partial \delta_2} = -\gamma \, e^{\lambda t} \, \nu \left((1-\delta) \exp\{-\rho L_2\} + \delta \exp\{-\rho \, L_1\}\right)^{-\frac{\nu}{\rho}-1} \tag{355}$$
$$(1-\delta) \exp\{-\rho \, L_2\}(-\ln x_3 + \ln x_4)$$

$$\lim_{\rho_2 \to 0} \lim_{\rho_1 \to 0} \frac{\partial y}{\partial \delta} = -\gamma \, e^{\lambda t} \, \frac{\nu}{\rho} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) \exp\{-\rho \, L_2\}\right)^{-\frac{\nu}{\rho}-1} \tag{356}$$
$$\left(\exp\{-\rho \, L_1\} - \exp\{-\rho \, L_2\}\right)$$

$$\lim_{\rho_2 \to 0} \lim_{\rho_1 \to 0} \frac{\partial y}{\partial \nu} = -\gamma \, e^{\lambda t} \, \frac{1}{\rho} \ln\left(\delta \exp\{-\rho \, L_1\} + (1-\delta) \exp\{-\rho \, L_2\}\right) \tag{357}$$
$$\left(\delta \exp\{-\rho \, L_1\} + (1-\delta) \exp\{-\rho L_2\}\right)^{-\frac{\nu}{\rho}}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho_1 \to 0} \frac{\partial y}{\partial \rho_1} = -\gamma \, e^{\lambda t} \, \delta \, \nu \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) \exp\{-\rho \, L_2\}\right)^{-\frac{\nu}{\rho}-1} \tag{358}$$
$$\cdot \exp\{\rho \, L_2\} \left(-\frac{1}{2} L_1^2 + \frac{1}{2} \left(\delta_1 (\ln x_1)^2 + (1-\delta_1) (\ln x_2)^2\right)\right)$$

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \frac{\partial y}{\partial \rho_2} = -\gamma \, e^{\lambda t} \, (1-\delta) \, \nu \left((1-\delta) \exp\{-\rho \, L_2\} + \delta \exp\{-\rho \, L_1\}\right)^{-\frac{\nu}{\rho}-1} \tag{359}$$
$$\cdot \exp\{-\rho \, L_2\} \left(-\frac{1}{2} L_2^2 + \frac{1}{2} \left(\delta_2 (\ln x_3)^2 + (1-\delta_2) (\ln x_4)^2\right)\right)$$

$$\lim_{\rho_2 \to 0} \lim_{\rho_1 \to 0} \frac{\partial y}{\partial \rho} = \gamma \, e^{\lambda t} \, \frac{\nu}{\rho^2} \ln\left(\delta \exp\{-\rho \, L_1\} + (1-\delta) \exp\{-\rho \, L_2\}\right) \tag{360}$$
$$\left(\delta \exp\{-\rho \, L_1\} + (1-\delta) \exp\{-\rho \, L_2\}\right)^{-\frac{\nu}{\rho}}$$
$$- \gamma \, e^{\lambda t} \, \frac{\nu}{\rho} \left(\delta \exp\{-\rho \, L_1\} + (1-\delta) \exp\{-\rho \, L_2\}\right)^{-\frac{\nu}{\rho}-1}$$
$$\left(-\delta \exp\{-\rho \, L_1\} \, L_1 - (1-\delta) \exp\{-\rho \, L_2\} \, L_2\right)$$

Limits of the derivatives for ρ_1 and ρ approaching zero

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \gamma} = e^{\lambda t} \exp\left\{-\nu\left(-\delta\, L_1 + \frac{(1-\delta)\ln(B_2)}{\rho_2}\right)\right\} \tag{361}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta_1} = \gamma\, e^{\lambda t}\left(-\nu(\delta(-\ln x_1 + \ln x_2))\right)\exp\left\{-\nu\left(-\delta\, L_1 + \frac{(1-\delta)\ln(B_2)}{\rho_2}\right)\right\} \tag{362}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta_2} = \gamma\, e^{\lambda t}\, \frac{-\nu(1-\delta)(x_3^{-\rho_2} - x_4^{-\rho_2})}{\rho_2\, B_2}\exp\left\{-\nu\left(-\delta\, L_1 + \frac{(1-\delta)\ln(B_2)}{\rho_2}\right)\right\} \tag{363}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta} = -\gamma\, e^{\lambda t}\, \nu\left(-L_1 - \frac{\ln(B_2)}{\rho_2}\right)\exp\left\{-\nu\left(-\delta\, L_1 + (1-\delta)\frac{\ln(B_2)}{\rho_2}\right)\right\} \tag{364}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \nu} = \gamma\, e^{\lambda t}\left(\delta\, L_1 - (1-\delta)\frac{\ln(B_2)}{\rho_2}\right)\exp\left\{-\nu\left(-\delta\, L_1 + (1-\delta)\frac{\ln(B_2)}{\rho_2}\right)\right\} \tag{365}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho_1} = -\gamma\, e^{\lambda t}\, \nu\, \delta\left(\frac{1}{2}(\delta_1(\ln x_1)^2 + (1-\delta_1)(\ln x_2)^2) - \frac{1}{2}L_1^2\right) \tag{366}$$
$$\cdot \exp\left\{-\nu\left(-\delta\, L_1 + (1-\delta)\frac{\ln(B_2)}{\rho_2}\right)\right\}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho_2} = -\gamma\, e^{\lambda t}\, \nu\,(1-\delta)\left(-\frac{\ln(B_2)}{\rho_2^2} + \frac{(-\delta_2 \ln x_3 x_3^{-\rho_2} - (1-\delta_2)\ln x_4 x_4^{-\rho_2})}{\rho_2\, B_2}\right) \tag{367}$$
$$\cdot \exp\left\{-\nu\left(-L_1 + (1-\delta)\frac{\ln(B_2)}{\rho_2}\right)\right\}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho} = \gamma\, e^{\lambda t}\, \nu\left(-\frac{1}{2}\left(\delta L_1^2 + (1-\delta)\left(\frac{\ln(B_2)}{\rho_2}\right)^2\right) + \frac{1}{2}\left(-\delta\, L_1 + (1-\delta)\frac{\ln(B_2)}{\rho_2}\right)^2\right) \tag{368}$$
$$\cdot \exp\left\{-\nu\left(-\delta\, L_1 + (1-\delta)\frac{\ln(B_2)}{\rho_2}\right)\right\}$$

Limits of the derivatives for ρ_2 and ρ approaching zero

$$\lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \gamma} = e^{\lambda t} \exp\left\{ -\nu \left(\frac{\delta \ln(B_1)}{\rho_1} - (1-\delta)L_2 \right) \right\} \tag{369}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta_1} = \gamma\, e^{\lambda t} \frac{-\nu\delta(x_1^{-\rho_1} - x_2^{-\rho_1})}{\rho_1\, B_1} \exp\left\{ -\nu \left(\frac{\delta \ln(B_1)}{\rho_1} - (1-\delta)L_2 \right) \right\} \tag{370}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta_2} = \gamma\, e^{\lambda t} \left(-\nu((1-\delta)(-\ln x_3 + \ln x_4)) \right) \exp\left\{ -\nu \left(\frac{\delta \ln(B_1)}{\rho_1} - (1-\delta)L_2 \right) \right\} \tag{371}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta} = -\gamma\, e^{\lambda t}\, \nu \left(\frac{\ln(B_1)}{\rho_1} + L_2 \right) \exp\left\{ -\nu \left(\delta \frac{\ln(B_1)}{\rho_1} - (1-\delta)L_2 \right) \right\} \tag{372}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \nu} = \gamma\, e^{\lambda t} \left(-\delta \frac{\ln(B_1)}{\rho_1} + (1-\delta)L_2 \right) \exp\left\{ -\nu \left(\delta \frac{\ln(B_1)}{\rho_1} - (1-\delta)L_2 \right) \right\} \tag{373}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho_1} = -\gamma\, e^{\lambda t}\, \nu\, \delta \left(-\frac{\ln(B_1)}{\rho_1^2} + \frac{(-\delta_1 \ln x_1 x_1^{-\rho_1} - (1-\delta_1)\ln x_2 x_2^{-\rho_1})}{\rho_1\, B_1} \right) \tag{374}$$
$$\cdot \exp\left\{ -\nu \left(\delta \frac{\ln(B_1)}{\rho_1} - (1-\delta)L_2 \right) \right\}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho_2} = -\gamma\, e^{\lambda t}\, \nu\, (1-\delta) \left(\frac{1}{2}(\delta_2(\ln x_3)^2 + (1-\delta_2)(\ln x_4)^2) - \frac{1}{2}L_2^2 \right) \tag{375}$$
$$\cdot \exp\left\{ -\nu \left(\delta \frac{\ln(B_1)}{\rho_1} - (1-\delta)L_2 \right) \right\}$$

$$\lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho} = \gamma\, e^{\lambda t}\, \nu \left(-\frac{1}{2} \left(\delta \left(\frac{\ln(B_1)}{\rho_1} \right)^2 + (1-\delta)L_2^2 \right) + \frac{1}{2} \left(\delta \frac{\ln(B_1)}{\rho_1} - (1-\delta)L_2 \right)^2 \right) \tag{376}$$
$$\cdot \exp\left\{ -\nu \left(\delta \frac{\ln(B_1)}{\rho_1} - (1-\delta)L_2 \right) \right\}$$

Limits of the derivatives for ρ_1, ρ_2, and ρ approaching zero

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \gamma} = e^{\lambda t} \exp\{-\nu(-\delta \, L_1 - (1-\delta)L_2)\} \tag{377}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta_1} = -\gamma \, e^{\lambda t} \, \nu \, \delta \, (-\ln x_1 + \ln x_2) \exp\{-\nu(-\delta \, L_1 - (1-\delta)L_2)\} \tag{378}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta_2} = -\gamma \, e^{\lambda t} \, \nu \, (1-\delta)(-\ln x_3 + \ln x_4) \exp\{-\nu(-\delta \, L_1 - (1-\delta)L_2)\} \tag{379}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \delta} = -\gamma \, e^{\lambda t} \, \nu \, (-L_1 + L_2) \exp\{-\nu(-\delta \, L_1 - (1-\delta)L_2)\} \tag{380}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \nu} = \gamma \, e^{\lambda t} \, (\delta \, L_1 + (1-\delta)L_2) \exp\{-\nu(-\delta \, L_1 - (1-\delta)L_2)\} \tag{381}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho_1} = -\gamma \, e^{\lambda t} \, \nu \, \delta \, \left(\frac{1}{2}(\delta_1(\ln x_1)^2 + (1-\delta_1)(\ln x_2)^2) - \frac{1}{2}L_1^2\right) \tag{382}$$
$$\cdot \exp\{-\nu(-\delta \, L_1 - (1-\delta)L_2)\}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho_2} = -\gamma \, e^{\lambda t} \, \nu \, (1-\delta) \left(\frac{1}{2}(\delta_2(\ln x_3)^2 + (1-\delta_2)(\ln x_4)^2) - \frac{1}{2}L_2^2\right) \tag{383}$$
$$\cdot \exp\{-\nu(-\delta \, L_1 - (1-\delta)L_2)\}$$

$$\lim_{\rho_1 \to 0} \lim_{\rho_2 \to 0} \lim_{\rho \to 0} \frac{\partial y}{\partial \rho} = \gamma \, e^{\lambda t} \, \nu \, \left(-\frac{1}{2}\left(\delta L_1^2 + (1-\delta)L_2^2\right) + \frac{1}{2}(-\delta \, L_1 - (1-\delta)L_2)^2\right) \tag{384}$$
$$\cdot \exp\{-\nu(-\delta \, L_1 + (1-\delta)(-\delta_2 \ln x_3 - (1-\delta_2)\ln x_4))\}$$

C.3. Elasticities of substitution

Allen-Uzawa elasticity of substitution (see Sato 1967)

$$\sigma_{i,j} = \begin{cases} (1-\rho)^{-1} & \text{for } i = 1,2; \ j = 3,4 \\[2em] \dfrac{(1-\rho_1)^{-1} - (1-\rho)^{-1}}{\delta \left(\dfrac{y}{B_1^{-\frac{1}{\rho_1}}} \right)^{1+\rho}} + (1-\rho)^{-1} & \text{for } i = 1; \ j = 2 \\[2em] \dfrac{(1-\rho_2)^{-1} - (1-\rho)^{-1}}{(1-\delta) \left(\dfrac{y}{B_2^{-\frac{1}{\rho_2}}} \right)^{1+\rho}} + (1-\rho)^{-1} & \text{for } i = 3; \ j = 4 \end{cases} \tag{385}$$

Hicks-McFadden elasticity of substitution (see Sato 1967)

$$\sigma_{i,j} = \begin{cases} \dfrac{\dfrac{1}{\theta_i} + \dfrac{1}{\theta_j}}{(1-\rho_1)\left(\dfrac{1}{\theta_i} - \dfrac{1}{\theta^*}\right) + (1-\rho_2)\left(\dfrac{1}{\theta_j} - \dfrac{1}{\theta}\right) + (1-\rho)\left(\dfrac{1}{\theta^*} - \dfrac{1}{\theta}\right)} & \text{for } i = 1,2; \ j = 3,4 \\[2em] (1-\rho_1)^{-1} & \text{for } i = 1; \ j = 2 \\[1em] (1-\rho_2)^{-1} & \text{for } i = 3, \ j = 4 \end{cases} \tag{386}$$

with

$$\theta^* = \delta B_1^{\frac{\rho}{\rho_1}} \cdot y^\rho \tag{387}$$

$$\theta = (1-\delta) B_2^{\frac{\rho}{\rho_2}} \cdot y^\rho \tag{388}$$

$$\theta_1 = \delta \delta_1 x_1^{-\rho_1} B_1^{-\frac{\rho_1-\rho}{\rho_1}} \cdot y^\rho \tag{389}$$

$$\theta_2 = \delta (1-\delta_1) x_2^{-\rho_1} B_1^{-\frac{\rho_1-\rho}{\rho_1}} \cdot y^\rho \tag{390}$$

$$\theta_3 = (1-\delta) \delta_2 x_3^{-\rho_2} B_2^{-\frac{\rho_2-\rho}{\rho_2}} \cdot y^\rho \tag{391}$$

$$\theta_4 = (1-\delta)(1-\delta_2) x_4^{-\rho_2} B_2^{-\frac{\rho_2-\rho}{\rho_2}} \cdot y^\rho \tag{392}$$

D. Scripts for replicating the analysis of Kemfert (1998)

D.1. Standard non-linear least-squares estimations

```
# load the micEconCES package
library( "micEconCES" )

# load the data set
data( "GermanIndustry" )

# remove years 1973 - 1975 because of economic disruptions (see Kemfert 1998)
GermanIndustry <- subset( GermanIndustry, year < 1973 | year > 1975, )

# add a time trend (starting with 0)
GermanIndustry$time <- GermanIndustry$year - 1960

# names of inputs
xNames1 <-  c( "K", "E", "A" )
xNames2 <-  c( "K", "A", "E" )
xNames3 <-  c( "E", "A", "K" )

################ econometric estimation with cesEst #######################

## Nelder-Mead
cesNm1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   method = "NM", control = list( maxit = 5000 ) )
summary( cesNm1 )
cesNm2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   method = "NM", control = list( maxit = 5000 ) )
summary( cesNm2 )
cesNm3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   method = "NM", control = list( maxit = 5000 ) )
summary( cesNm3 )


## Simulated Annealing
cesSann1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   method = "SANN", control = list( maxit = 2e6 ) )
summary( cesSann1 )
cesSann2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   method = "SANN", control = list( maxit = 2e6 ) )
summary( cesSann2 )
cesSann3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   method = "SANN", control = list( maxit = 2e6 ) )
summary( cesSann3 )


## BFGS
cesBfgs1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
```

```
    method = "BFGS", control = list( maxit = 5000 ) )
summary( cesBfgs1 )
cesBfgs2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    method = "BFGS", control = list( maxit = 5000 ) )
summary( cesBfgs2 )
cesBfgs3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    method = "BFGS", control = list( maxit = 5000 ) )
summary( cesBfgs3 )


## L-BFGS-B
cesBfgsCon1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    method = "L-BFGS-B", control = list( maxit = 5000 )  )
summary( cesBfgsCon1 )
cesBfgsCon2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    method = "L-BFGS-B", control = list( maxit = 5000 )  )
summary( cesBfgsCon2 )
cesBfgsCon3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    method = "L-BFGS-B", control = list( maxit = 5000 )  )
summary( cesBfgsCon3 )


## Levenberg-Marquardt
cesLm1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLm1 )
cesLm2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLm2 )
cesLm3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLm3 )


## Levenberg-Marquardt, multiplicative error term
cesLm1Me <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    multErr = TRUE, control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLm1Me )
cesLm2Me <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    multErr = TRUE, control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLm2Me )
cesLm3Me <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    multErr = TRUE, control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLm3Me )


## Newton-type
cesNewton1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    method = "Newton", iterlim = 500 )
summary( cesNewton1 )
cesNewton2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
```

```
   method = "Newton", iterlim = 500 )
summary( cesNewton2 )
cesNewton3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   method = "Newton", iterlim = 500 )
summary( cesNewton3 )


## PORT
cesPort1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   method = "PORT", control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPort1 )
cesPort2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   method = "PORT", control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPort2 )
cesPort3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   method = "PORT", control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPort3 )


## PORT, multiplicative error
cesPort1Me <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   method = "PORT", multErr = TRUE,
   control = list( eval.max = 2000, iter.max = 2000 ) )
summary( cesPort1Me )
cesPort2Me <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   method = "PORT", multErr = TRUE,
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPort2Me )
cesPort3Me <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   method = "PORT", multErr = TRUE,
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPort3Me )


## DE
cesDe1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   method = "DE", control = DEoptim.control( trace = FALSE, NP = 500,
   itermax = 1e4 ) )
summary( cesDe1 )
cesDe2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   method = "DE", control = DEoptim.control( trace = FALSE, NP = 500,
   itermax = 1e4 ) )
summary( cesDe2 )
cesDe3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   method = "DE", control = DEoptim.control( trace = FALSE, NP = 500,
   itermax = 1e4 ) )
summary( cesDe3 )


## nls
cesNls1 <- try( cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
```

```
    vrs = TRUE, method = "nls" ) )
cesNls2 <- try( cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    vrs = TRUE, method = "nls" ) )
cesNls3 <- try( cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    vrs = TRUE, method = "nls" ) )


## NM - Levenberg-Marquardt
cesNmLm1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    start = coef( cesNm1 ),
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesNmLm1 )
cesNmLm2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    start = coef( cesNm2 ),
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesNmLm2 )
cesNmLm3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    start = coef( cesNm3 ),
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesNmLm3 )


## SANN - Levenberg-Marquardt
cesSannLm1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    start = coef( cesSann1 ),
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesSannLm1 )
cesSannLm2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    start = coef( cesSann2 ),
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesSannLm2 )
cesSannLm3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    start = coef( cesSann3 ),
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesSannLm3 )


## DE - Levenberg-Marquardt
cesDeLm1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    start = coef( cesDe1 ),
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesDeLm1 )
cesDeLm2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    start = coef( cesDe2 ),
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesDeLm2 )
cesDeLm3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    start = coef( cesDe3 ),
    control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesDeLm3 )
```

```
## NM - PORT
cesNmPort1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   method = "PORT", start = coef( cesNm1 ),
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesNmPort1 )
cesNmPort2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   method = "PORT", start = coef( cesNm2 ),
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesNmPort2 )
cesNmPort3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   method = "PORT", start = coef( cesNm3 ),
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesNmPort3 )


## SANN - PORT
cesSannPort1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   method = "PORT", start = coef( cesSann1 ),
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesSannPort1 )
cesSannPort2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   method = "PORT", start = coef( cesSann2 ),
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesSannPort2 )
cesSannPort3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   method = "PORT", start = coef( cesSann3 ),
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesSannPort3 )


## DE - PORT
cesDePort1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   method = "PORT", start = coef( cesDe1 ),
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesDePort1 )
cesDePort2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   method = "PORT", start = coef( cesDe2 ),
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesDePort2 )
cesDePort3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   method = "PORT", start = coef( cesDe3 ),
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesDePort3 )



# save the workspace
save.image( "kemfert98_nls.RData" )
```

D.2. Estimations with lambda, rho1, and rho fixed

```
# load the micEconCES package
library( "micEconCES" )

# load the data set
data( "GermanIndustry" )

# remove years 1973 - 1975 because of economic disruptions (see Kemfert 1998)
GermanIndustry <- subset( GermanIndustry, year < 1973 | year > 1975, )

# time trend (starting with 0)
GermanIndustry$time <- GermanIndustry$year - 1960

# removing technological progress using the lambdas of Kemfert (1998)
# (we can do this, because the model has constant returns to scale)
GermanIndustry$K1 <- GermanIndustry$K * exp( 0.0222 * GermanIndustry$time )
GermanIndustry$E1 <- GermanIndustry$E * exp( 0.0222 * GermanIndustry$time )
GermanIndustry$A1 <- GermanIndustry$A * exp( 0.0222 * GermanIndustry$time )
GermanIndustry$K2 <- GermanIndustry$K * exp( 0.0069 * GermanIndustry$time )
GermanIndustry$E2 <- GermanIndustry$E * exp( 0.0069 * GermanIndustry$time )
GermanIndustry$A2 <- GermanIndustry$A * exp( 0.0069 * GermanIndustry$time )
GermanIndustry$K3 <- GermanIndustry$K * exp( 0.00641 * GermanIndustry$time )
GermanIndustry$E3 <- GermanIndustry$E * exp( 0.00641 * GermanIndustry$time )
GermanIndustry$A3 <- GermanIndustry$A * exp( 0.00641 * GermanIndustry$time )

# names of inputs
xNames1 <-  c( "K1", "E1", "A1" )
xNames2 <-  c( "K2", "A2", "E2" )
xNames3 <-  c( "E3", "A3", "K3" )


############# estimation with lambda, rho_1, and rho fixed ###################

## Nelder-Mead, lambda, rho_1, and rho fixed
cesNmFixed1 <- cesEst( "Y", xNames1, data = GermanIndustry,
   method = "NM", rho1 = 0.5300, rho = 0.1813,
   control = list( maxit = 5000 ) )
summary( cesNmFixed1 )
cesNmFixed2 <- cesEst( "Y", xNames2, data = GermanIndustry,
   method = "NM", rho1 = 0.2155, rho = 1.1816,
   control = list( maxit = 5000 ) )
summary( cesNmFixed2 )
cesNmFixed3 <- cesEst( "Y", xNames3, data = GermanIndustry,
   method = "NM", rho1 = 1.3654, rho = 5.8327,
   control = list( maxit = 5000 ) )
summary( cesNmFixed3 )
```

```r
## BFGS, lambda, rho_1, and rho fixed
cesBfgsFixed1 <- cesEst( "Y", xNames1, data = GermanIndustry,
   method = "BFGS", rho1 = 0.5300, rho = 0.1813,
   control = list( maxit = 5000 ) )
summary( cesBfgsFixed1 )
cesBfgsFixed2 <- cesEst( "Y", xNames2, data = GermanIndustry,
   method = "BFGS", rho1 = 0.2155, rho = 1.1816,
   control = list( maxit = 5000 ) )
summary( cesBfgsFixed2 )
cesBfgsFixed3 <- cesEst( "Y", xNames3, data = GermanIndustry,
   method = "BFGS", rho1 = 1.3654, rho = 5.8327,
   control = list( maxit = 5000 ) )
summary( cesBfgsFixed3 )


## Levenberg-Marquardt, lambda, rho_1, and rho fixed
cesLmFixed1 <- cesEst( "Y", xNames1, data = GermanIndustry,
   rho1 = 0.5300, rho = 0.1813,
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmFixed1 )
cesLmFixed2 <- cesEst( "Y", xNames2, data = GermanIndustry,
   rho1 = 0.2155, rho = 1.1816,
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmFixed2 )
cesLmFixed3 <- cesEst( "Y", xNames3, data = GermanIndustry,
   rho1 = 1.3654, rho = 5.8327,
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmFixed3 )


## Levenberg-Marquardt, lambda, rho_1, and rho fixed, multiplicative error term
cesLmFixed1Me <- cesEst( "Y", xNames1, data = GermanIndustry,
   rho1 = 0.5300, rho = 0.1813, multErr = TRUE,
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmFixed1Me )
summary( cesLmFixed1Me, rSquaredLog = FALSE )
cesLmFixed2Me <- cesEst( "Y", xNames2, data = GermanIndustry,
   rho1 = 0.2155, rho = 1.1816, multErr = TRUE,
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmFixed2Me )
summary( cesLmFixed2Me, rSquaredLog = FALSE )
cesLmFixed3Me <- cesEst( "Y", xNames3, data = GermanIndustry,
   rho1 = 1.3654, rho = 5.8327, multErr = TRUE,
   control = nls.lm.control( maxiter = 1024, maxfev = 2000 ) )
summary( cesLmFixed3Me )
summary( cesLmFixed3Me, rSquaredLog = FALSE )


## Newton-type, lambda, rho_1, and rho fixed
cesNewtonFixed1 <- cesEst( "Y", xNames1, data = GermanIndustry,
```

```
   method = "Newton", rho1 = 0.5300, rho = 0.1813, iterlim = 500 )
summary( cesNewtonFixed1 )
cesNewtonFixed2 <- cesEst( "Y", xNames2, data = GermanIndustry,
   method = "Newton", rho1 = 0.2155, rho = 1.1816, iterlim = 500 )
summary( cesNewtonFixed2 )
cesNewtonFixed3 <- cesEst( "Y", xNames3, data = GermanIndustry,
   method = "Newton", rho1 = 1.3654, rho = 5.8327, iterlim = 500 )
summary( cesNewtonFixed3 )


## PORT, lambda, rho_1, and rho fixed
cesPortFixed1 <- cesEst( "Y", xNames1, data = GermanIndustry,
   method = "PORT", rho1 = 0.5300, rho = 0.1813,
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPortFixed1 )
cesPortFixed2 <- cesEst( "Y", xNames2, data = GermanIndustry,
   method = "PORT", rho1 = 0.2155, rho = 1.1816,
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPortFixed2 )
cesPortFixed3 <- cesEst( "Y", xNames3, data = GermanIndustry,
   method = "PORT", rho1 = 1.3654, rho = 5.8327,
   control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPortFixed3 )



# compare RSSs of models with lambda, rho_1, and rho fixed
print( matrix( c( cesNmFixed1$rss, cesBfgsFixed1$rss, cesLmFixed1$rss,
   cesNewtonFixed1$rss, cesPortFixed1$rss ), ncol = 1 ), digits = 16 )
cesFixed1 <- cesLmFixed1
print( matrix( c( cesNmFixed2$rss, cesBfgsFixed2$rss, cesLmFixed2$rss,
   cesNewtonFixed2$rss, cesPortFixed2$rss ), ncol = 1 ), digits = 16 )
cesFixed2 <- cesLmFixed2
print( matrix( c( cesNmFixed3$rss, cesBfgsFixed3$rss, cesLmFixed3$rss,
   cesNewtonFixed3$rss, cesPortFixed3$rss ), ncol = 1 ), digits = 16 )
cesFixed3 <- cesLmFixed3


# save the work space
save.image( "kemfert98_fixed.RData" )


## check if removing the technical progress worked as expected
Y2Calc <- cesCalc( xNames2, data = GermanIndustry,
   coef = coef( cesFixed2 ), nested = TRUE )
all.equal( Y2Calc, fitted( cesFixed2 ) )
Y2TcCalc <- cesCalc( sub( "[123]$", "", xNames2 ), tName = "time",
   data = GermanIndustry,
   coef = c( coef( cesFixed2 )[1], lambda = 0.0069, coef( cesFixed2 )[-1] ),
   nested = TRUE )
all.equal( Y2Calc, Y2TcCalc )
```

D.3. Estimations using grid search

```
# load the micEconCES package
library( "micEconCES" )

# load the data set
data( "GermanIndustry" )

# remove years 1973 - 1975 because of economic disruptions (see Kemfert 1998)
GermanIndustry <- subset( GermanIndustry, year < 1973 | year > 1975, )

# time trend (starting with 0)
GermanIndustry$time <- GermanIndustry$year - 1960

# names of inputs
xNames1 <-  c( "K", "E", "A" )
xNames2 <-  c( "K", "A", "E" )
xNames3 <-  c( "E", "A", "K" )


########## Grid Search for Rho_1 and/or Rho #############
rhoVec <- c( seq( -1, 1, 0.1 ), seq( 1.2, 4, 0.2 ), seq( 4.4, 14, 0.4 ) )

## BFGS, grid search for rho_1 and rho
cesBfgsGridRho1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
   method = "BFGS", control = list( maxit = 5000 ) )
summary( cesBfgsGridRho1 )
plot( cesBfgsGridRho1 )
cesBfgsGridRho2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
   method = "BFGS", control = list( maxit = 5000 ) )
summary( cesBfgsGridRho2 )
plot( cesBfgsGridRho2 )
cesBfgsGridRho3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
   method = "BFGS", control = list( maxit = 5000 ) )
summary( cesBfgsGridRho3 )
plot( cesBfgsGridRho3 )

# BFGS with grid search estimates as starting values
cesBfgsGridStartRho1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   start = coef( cesBfgsGridRho1 ),
   method = "BFGS", control = list( maxit = 5000 ) )
summary( cesBfgsGridStartRho1 )
cesBfgsGridStartRho2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   start = coef( cesBfgsGridRho2 ),
   method = "BFGS", control = list( maxit = 5000 ) )
```

```
summary( cesBfgsGridStartRho2 )
cesBfgsGridStartRho3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   start = coef( cesBfgsGridRho3 ),
   method = "BFGS", control = list( maxit = 5000 ) )
summary( cesBfgsGridStartRho3 )



## Levenberg-Marquardt, grid search for rho1 and rho
cesLmGridRho1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmGridRho1 )
plot( cesLmGridRho1 )
cesLmGridRho2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmGridRho2 )
plot( cesLmGridRho2 )
cesLmGridRho3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmGridRho3 )
plot( cesLmGridRho3 )

# LM with grid search estimates as starting values
cesLmGridStartRho1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   start = coef( cesLmGridRho1 ),
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmGridStartRho1 )
cesLmGridStartRho2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   start = coef( cesLmGridRho2 ),
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmGridStartRho2 )
cesLmGridStartRho3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
   start = coef( cesLmGridRho3 ),
   control = nls.lm.control( maxiter = 1000, maxfev = 2000 ) )
summary( cesLmGridStartRho3 )



## Newton-type, grid search for rho_1 and rho
cesNewtonGridRho1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
   rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
   method = "Newton", iterlim = 500 )
summary( cesNewtonGridRho1 )
plot( cesNewtonGridRho1 )
cesNewtonGridRho2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
   rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
```

```
    method = "Newton", iterlim = 500, check.analyticals = FALSE )
summary( cesNewtonGridRho2 )
plot( cesNewtonGridRho2 )
cesNewtonGridRho3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE,
    method = "Newton", iterlim = 500 )
summary( cesNewtonGridRho3 )
plot( cesNewtonGridRho3 )


# Newton-type with grid search estimates as starting values
cesNewtonGridStartRho1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    start = coef( cesNewtonGridRho1 ),
    method = "Newton", iterlim = 500, check.analyticals = FALSE )
summary( cesNewtonGridStartRho1 )
cesNewtonGridStartRho2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    start = coef( cesNewtonGridRho2 ),
    method = "Newton", iterlim = 500, check.analyticals = FALSE )
summary( cesNewtonGridStartRho2 )
cesNewtonGridStartRho3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    start = coef( cesNewtonGridRho3 ),
    method = "Newton", iterlim = 500, check.analyticals = FALSE )
summary( cesNewtonGridStartRho3 )



## PORT, grid search for rho1 and rho
cesPortGridRho1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE, method = "PORT",
    control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPortGridRho1 )
plot( cesPortGridRho1 )
cesPortGridRho2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
    rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE, method = "PORT",
    control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPortGridRho2 )
plot( cesPortGridRho2 )
cesPortGridRho3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    rho1 = rhoVec, rho = rhoVec, returnGridAll = TRUE, method = "PORT",
    control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPortGridRho3 )
plot( cesPortGridRho3 )


# PORT with grid search estimates as starting values
cesPortGridStartRho1 <- cesEst( "Y", xNames1, tName = "time", data = GermanIndustry,
    start = coef( cesPortGridRho1 ),, method = "PORT",
    control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPortGridStartRho1 )
cesPortGridStartRho2 <- cesEst( "Y", xNames2, tName = "time", data = GermanIndustry,
```

```
    start = coef( cesPortGridRho2 ), method = "PORT",
    control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPortGridStartRho2 )
cesPortGridStartRho3 <- cesEst( "Y", xNames3, tName = "time", data = GermanIndustry,
    start = coef( cesPortGridRho3 ), method = "PORT",
    control = list( eval.max = 1000, iter.max = 1000 ) )
summary( cesPortGridStartRho3 )



save.image( "kemfert98_grid.RData" )
```

D.4. Estimations of models for the different industrial sectors

```
# load the micEconCES package
library( "micEconCES" )

# load the data set
data( "GermanIndustry" )

# remove years with missing or incomplete data
GermanIndustry <- subset( GermanIndustry, year >= 1970 & year <= 1988, )

# remove years 1973 - 1975 because of economic disruptions (see Kemfert 1998)
GermanIndustry <- subset( GermanIndustry, year < 1973 | year > 1975, )

# add a time trend (starting with 0)
GermanIndustry$time <- GermanIndustry$year - 1970

# rhos for grid search
rhoVec <- c( seq( -1, 0.6, 0.2 ), seq( 0.9, 1.5, 0.3 ),
    seq( 2, 10, 1 ), 12, 15, 20, 30, 50, 100 )

# industries (abbreviations)
indAbbr <- c( "C", "S", "N", "I", "V",  "P", "F" )

# names of inputs
xNames <- list()
xNames[[ 1 ]] <-  c( "K", "E", "A" )
xNames[[ 2 ]] <-  c( "K", "A", "E" )
xNames[[ 3 ]] <-  c( "E", "A", "K" )

# list ("folder") for results
indRes <- list()

# names of estimation methods
metNames <- c( "LM", "PORT", "PORT_Grid", "PORT_Start" )
```

```r
# table for parameter estimates
tabCoef <- array( NA, dim = c( 9, 7, length( metNames ) ),
   dimnames = list(
      paste( rep( c( "alpha_", "beta_", "m_" ), 3 ), rep( 1:3, each = 3 ),
         "␣(", rep( c( "rho_1", "rho", "lambda" ), 3 ), ")", sep = "" ),
      indAbbr, metNames ) )

# table for technological change parameters
tabLambda <- array( NA, dim = c( 7, 3, length( metNames ) ),
   dimnames = list( indAbbr, c(1:3), metNames ) )

# table for R-squared values
tabR2 <- tabLambda

# table for RSS values
tabRss <- tabLambda

# table for economic consistency of LM results
tabConsist <- tabLambda[ , , 1, drop = TRUE ]


################ econometric estimation with cesEst #######################
for( indNo in 1:length( indAbbr ) ) {

   # name of industry-specific output
   yIndName <- paste( indAbbr[ indNo ], "Y", sep = "_" )

   # sub-list ("subfolder") for all models of this industrie
   indRes[[ indNo ]] <- list()

   for( modNo in 1:3 ) {

      cat( "\n=========================================================\n" )
      cat( "Industry␣No.␣", indNo, ",␣model␣No.␣", modNo, "\n", sep = "" )
      cat( "=========================================================\n\n" )

      # names of industry-specific inputs
      xIndNames <- paste( indAbbr[ indNo ], xNames[[ modNo ]], sep = "_" )

      # sub-sub-list for all estimation results of this model/industrie
      indRes[[ indNo ]][[ modNo ]] <- list()

      ## Levenberg-Marquardt
      indRes[[ indNo ]][[ modNo ]]$lm <- cesEst( yIndName, xIndNames,
         tName = "time", data = GermanIndustry,
         control = nls.lm.control( maxiter = 1024, maxfev = 2000 ) )
      print( tmpSum <- summary( indRes[[ indNo ]][[ modNo ]]$lm ) )
```

```
tmpCoef <- coef( indRes[[ indNo ]][[ modNo ]]$lm )
tabCoef[ ( 3 * modNo - 2 ):( 3 * modNo ), indNo, "LM" ] <-
    tmpCoef[ c( "rho_1", "rho", "lambda" ) ]
tabLambda[ indNo, modNo, "LM" ] <- tmpCoef[ "lambda" ]
tabR2[ indNo, modNo, "LM" ] <- tmpSum$r.squared
tabRss[ indNo, modNo, "LM" ] <- tmpSum$rss
tabConsist[ indNo, modNo ] <- tmpCoef[ "gamma" ] >= 0 &
    tmpCoef[ "delta_1" ] >= 0 & tmpCoef[ "delta_1" ] <= 1 &
    tmpCoef[ "delta" ] >= 0 & tmpCoef[ "delta" ] <= 1 &
    tmpCoef[ "rho_1" ] >= -1 & tmpCoef[ "rho" ] >= -1

## PORT
indRes[[ indNo ]][[ modNo ]]$port <- cesEst( yIndName, xIndNames,
    tName = "time", data = GermanIndustry, method = "PORT",
    control = list( eval.max = 2000, iter.max = 2000 ) )
print( tmpSum <- summary( indRes[[ indNo ]][[ modNo ]]$port ) )
tmpCoef <- coef( indRes[[ indNo ]][[ modNo ]]$port )
tabCoef[ ( 3 * modNo - 2 ):( 3 * modNo ), indNo, "PORT" ] <-
    tmpCoef[ c( "rho_1", "rho", "lambda" ) ]
tabLambda[ indNo, modNo, "PORT" ] <- tmpCoef[ "lambda" ]
tabR2[ indNo, modNo, "PORT" ] <- tmpSum$r.squared
tabRss[ indNo, modNo, "PORT" ] <- tmpSum$rss

# PORT, grid search
indRes[[ indNo ]][[ modNo ]]$portGrid <- cesEst( yIndName, xIndNames,
    tName = "time", data = GermanIndustry, method = "PORT",
    rho = rhoVec, rho1 = rhoVec,
    control = list( eval.max = 2000, iter.max = 2000 ) )
print( tmpSum <- summary( indRes[[ indNo ]][[ modNo ]]$portGrid ) )
tmpCoef <- coef( indRes[[ indNo ]][[ modNo ]]$portGrid )
tabCoef[ ( 3 * modNo - 2 ):( 3 * modNo ), indNo, "PORT_Grid" ] <-
    tmpCoef[ c( "rho_1", "rho", "lambda" ) ]
tabLambda[ indNo, modNo, "PORT_Grid" ] <- tmpCoef[ "lambda" ]
tabR2[ indNo, modNo, "PORT_Grid" ] <- tmpSum$r.squared
tabRss[ indNo, modNo, "PORT_Grid" ] <- tmpSum$rss

# PORT, grid search for starting values
indRes[[ indNo ]][[ modNo ]]$portStart <- cesEst( yIndName, xIndNames,
    tName = "time", data = GermanIndustry, method = "PORT",
    start = coef( indRes[[ indNo ]][[ modNo ]]$portGrid ),
    control = list( eval.max = 2000, iter.max = 2000 ) )
print( tmpSum <- summary( indRes[[ indNo ]][[ modNo ]]$portStart ) )
tmpCoef <- coef( indRes[[ indNo ]][[ modNo ]]$portStart )
tabCoef[ ( 3 * modNo - 2 ):( 3 * modNo ), indNo, "PORT_Start" ] <-
    tmpCoef[ c( "rho_1", "rho", "lambda" ) ]
tabLambda[ indNo, modNo, "PORT_Start" ] <- tmpCoef[ "lambda" ]
tabR2[ indNo, modNo, "PORT_Start" ] <- tmpSum$r.squared
```

```
      tabRss[ indNo, modNo, "PORT_Start" ] <- tmpSum$rss
    }
}

print( round( tabCoef, 3 ) )
print( round( do.call( "cbind",
    lapply( 1:length( metNames ), function(x) tabR2[,,x] ) ), 3 ) )

print( round( 1e8*( tabR2[ , , "LM" ] - tabR2[ , , "PORT" ] ) ) )
print( round( 1e8*( tabR2[ , , "PORT_Grid" ] - tabR2[ , , "PORT" ] ) ) )
print( round( 1e8*( tabR2[ , , "PORT_Start" ] - tabR2[ , , "PORT" ] ) ) )
print( round( 1e8*( tabR2[ , , "PORT_Start" ] - tabR2[ , , "PORT_Grid" ] ) ) )

print( round( ( tabRss[ , , "PORT" ] - tabRss[ , , "LM" ] ) / 100 ) )
print( round( ( tabRss[ , , "PORT" ] - tabRss[ , , "PORT_Grid" ] ) / 100 ) )
print( round( ( tabRss[ , , "PORT" ] - tabRss[ , , "PORT_Start" ] ) ) )
print( round( ( tabRss[ , , "PORT_Grid" ] - tabRss[ , , "PORT_Start" ] ) ) )

print( tabConsist )
```

References

Acemoglu D (1998). "Why Do New Technologies Complement Skills? Directed Technical Change and Wage Inequality." *The Quarterly Journal of Economics*, **113**(4), 1055–1089.

Ali MM, Törn A (2004). "Population set-based global optimization algorithms: some modifications and numerical studies." *Computers & Operations Research*, **31**(10), 1703–1725.

Amras P (2004). "Is the U.S. Aggregate Production Function Cobb-Douglas? New Estimates of the Elasticity of Substitution." *Contribution in Macroeconomics*, **4**(1), Article 4.

Anderson R, Greene WH, McCullough BD, Vinod HD (2008). "The Role of Data/Code Archives in the Future of Economic Research." *Journal of Economic Methodology*, **15**(1), 99–119.

Anderson RK, Moroney JR (1994). "Substitution and Complementarity in C.E.S. Models." *Southern Economic Journal*, **60**(4), 886–895.

Arrow KJ, Chenery BH, Minhas BS, Solow RM (1961). "Capital-labor substitution and economic efficiency." *The Review of Economics and Statistics*, **43**(3), 225–250. URL http://www.jstor.org/stable/1927286.

Beale EML (1972). "A Derivation of Conjugate Gradients." In FA Lootsma (ed.), *Numerical Methods for Nonlinear Optimization*, pp. 39–43. Academic Press, London.

Bélisle CJP (1992). "Convergence Theorems for a Class of Simulated Annealing Algorithms on R^d." *Journal of Applied Probability*, **29**, 885–895.

Bentolila SJ, Gilles SP (2006). "Explaining Movements in the Labour Share." *Contributions to Macroeconomics*, **3**(1), Article 9.

Blackorby C, Russel RR (1989). "Will the Real Elasticity of Substitution please stand up? (A comparison of the Allan/Uzawa and Morishima Elasticities)." *The American Economic Review*, **79**, 882–888.

Broyden CG (1970). "The Convergence of a Class of Double-rank Minimization Algorithms." *Journal of the Institute of Mathematics and Its Applications*, **6**, 76–90.

Buckheit J, Donoho DL (1995). "Wavelab and Reproducible Research." In A Antoniadis (ed.), *Wavelets and Statistics*. Springer Verlag.

Byrd R, Lu P, Nocedal J, Zhu C (1995). "A limited memory algorithm for bound constrained optimization." *SIAM Journal for Scientific Computing*, **16**, 1190–1208.

Caselli F (2005). "Accounting for Cross-Country Income Differences." In P Aghion, SN Durlauf (eds.), *Handbook of Economic Growth*, pp. 679–742. North Holland.

Caselli F, Coleman Wilbur John I (2006). "The World Technology Frontier." *The American Economic Review*, **96**(3), 499–522. URL http://www.jstor.org/stable/30034059.

Cerny V (1985). "A thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm." *Journal of Optimization Theory and Applications*, **45**, 41–51.

Chambers JM, Hastie TJ (1992). *Statistical Models in S*. Chapman & Hall, London.

Chambers RG (1988). *Applied Production Analysis. A Dual Approach*. Cambridge University Press, Cambridge.

Davies JB (2009). "Combining Microsimulation with CGE and Macro Modelling for Distributional Analysis in Developing and Transition Countries." *International Journal of Microsimulation*, pp. 49–65. URL http://www.microsimulation.org/IJM/V2_1/IJM_2_1_4.pdf.

Dennis JE, Schnabel RB (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs (NJ, USA).

Douglas PC, Cobb CW (1928). "A theory of production." *The American Economic Review*, **18**(1), 139–165.

Durlauf SN, Johnson PA (1995). "Multiple Regimes and Cross-Country Growth Behavior." *Journal of Applied Econometrics*, **10**, 365–384.

Elzhov TV, Mullen KM (2009). **minpack.lm**: *R Interface to the Levenberg-Marquardt Nonlinear Least-Squares Algorithm Found in MINPACK*. R package version 1.1-4, URL http://CRAN.R-project.org/package=minpack.lm.

Fletcher R (1970). "A New Approach to Variable Metric Algorithms." *Computer Journal*, **13**, 317–322.

Fletcher R, Reeves C (1964). "Function minimization by conjugate gradients." *Computer Journal*, **7**, 48–154.

Fox J (2009). **car**: *Companion to Applied Regression*. R package version 1.2-16, URL http://CRAN.R-project.org/package=car.

Gay DM (1990). "Usage Summary for Selected Optimization Routines." *Computing Science Technical Report 153*, AT&T Bell Laboratories. URL http://netlib.bell-labs.com/cm/cs/cstr/153.pdf.

Goldfarb D (1970). "A Family of Variable Metric Updates Derived by Variational Means." *Mathematics of Computation*, **24**, 23–26.

Greene WH (2008). *Econometric Analysis.* 6th edition. Prentice Hall.

Griliches Z (1969). "Capital-Skill Complementarity." *The Review of Economics and Statistics*, **51**(4), 465–468. URL http://www.jstor.org/pss/1926439.

Henningsen A (2010). **micEcon**: *Tools for Microeconomic Analysis and Microeconomic Modeling.* R package version 0.6, http://CRAN.R-project.org/package=micEcon.

Henningsen A, Hamann JD (2007). "**systemfit**: A Package for Estimating Systems of Simultaneous Equations in R." *Journal of Statistical Software*, **23**(4), 1–40. URL http://www.jstatsoft.org/v23/i04/.

Henningsen A, Henningsen G (2011). **micEconCES**: *Analysis with the Constant Elasticity of Scale (CES) Function.* R package version 0.9, http://CRAN.R-project.org/package=micEconCES.

Hoff A (2004). "The Linear Approximation of the CES Function with n Input Variables." *Marine Resource Economics*, **19**, 295–306.

Kelley CT (1999). *Iterative Methods of Optimization.* SIAM Society for Industrial and Applied Mathematics, Philadelphia.

Kemfert C (1998). "Estimated Substitution Elasticities of a Nested CES Production Function Approach for Germany." *Energy Economics*, **20**(3), 249–264.

Kirkpatrick S, Gelatt CD, Vecchi MP (1983). "Optimization by Simulated Annealing." *Science*, **220**(4598), 671–680.

Kleiber C, Zeileis A (2009). **AER**: *Applied Econometrics with R.* R package version 1.1, http://CRAN.R-project.org/package=AER.

Klump R, McAdam P, Willman A (2007). "Factor Substitution and Factor-Augmenting Technical Progress in the United States: A Normalized Supply-Side System Approach." *The Review of Economics and Statistics*, **89**(1), 183–192.

Klump R, Papageorgiou C (2008). "Editorial Introduction: The CES Production Function in the Theory and Empirics of Economic Growth." *Journal of Macroeconomics*, **30**(2), 599–600.

Kmenta J (1967). "On Estimation of the CES Production Function." *International Economic Review*, **8**, 180–189.

Krusell P, Ohanian LE, Ríos-Rull JV, Violante GL (2000). "Capital-skill Complementarity and Inequality: A Macroeconomic Analysis." *Econometrica*, **68**(5), 1029–1053.

León-Ledesma MA, McAdam P, Willman A (2010). "Identifying the Elasticity of Substitution with Biased Technical Change." *American Economic Review*, **100**, 1330–1357.

Luoma A, Luoto J (2010). "The Aggregate Production Function of the Finnish Economy in the Twentieth Century." *Southern Economic Journal*, **76**(3), 723–737.

Maddala G, Kadane J (1967). "Estimation of Returns to Scale and the Elasticity of Substitution." *Econometrica*, **24**, 419–423.

Mankiw NG, Romer D, Weil DN (1992). "A Contribution to the Empirics of Economic Growth." *Quarterly Journal of Economics*, **107**, 407–437.

Marquardt DW (1963). "An Algorithm for Least-Squares Estimation of Non-linear Parameters." *Journal of the Society for Industrial and Applied Mathematics*, **11**(2), 431–441. URL http://www.jstor.org/stable/2098941.

Masanjala WH, Papageorgiou C (2004). "The Solow Model with CES Technology: Nonlinearities and Parameter Heterogeneity." *Journal of Applied Econometrics*, **19**(2), 171–201.

McCullough BD (2009). "Open Access Economics Journals and the Market for Reproducible Economic Research." *Economic Analysis and Policy*, **39**(1), 117–126. URL http://www.pages.drexel.edu/~bdm25/eap-1.pdf.

McCullough BD, McGeary KA, Harrison TD (2008). "Do Economics Journal Archives Promote Replicable Research?" *Canadian Journal of Economics*, **41**(4), 1406–1420.

McFadden D (1963). "Constant Elasticity of Substitution Production Function." *The Review of Economic Studies*, **30**, 73–83.

Mishra SK (2007). "A Note on Numerical Estimation of Sato's Two-Level CES Production Function." *MPRA Paper 1019*, North-Eastern Hill University, Shillong.

Moré JJ, Garbow BS, Hillstrom KE (1980). *MINPACK*. Argonne National Laboratory. URL http://www.netlib.org/minpack/.

Mullen KM, Ardia D, Gil DL, Windover D, Cline J (2011). "DEoptim: An R Package for Global Optimization by Differential Evolution." *Journal of Statistical Software*, **40**(6), 1–26. URL http://www.jstatsoft.org/v40/i06.

Nelder JA, Mead R (1965). "A Simplex Algorithm for Function Minimization." *Computer Journal*, **7**, 308–313.

Pandey M (2008). "Human Capital Aggregation and Relative Wages Across Countries." *Journal of Macroeconomics*, **30**(4), 1587–1601.

Papageorgiou C, Saam M (2005). "Two-Level CES Production Technology in the Solow and Diamond Growth Models." *Working paper 2005-07*, Department of Economics, Louisiana State University.

Polak E, Ribière G (1969). "Note sur la convergence de méthodes de directions conjuguées." *Revue Francaise d'Informatique et de Recherche Opérationnelle*, **16**, 35–43.

Price KV, Storn RM, Lampinen JA (2006). *Differential Evolution – A Practical Approach to Global Optimization.* Natural Computing. Springer-Verlag. ISBN 540209506.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.

Sato K (1967). "A Two-Level Constant-Elasticity-of-Substitution Production Function." *The Review of Economic Studies*, **43**, 201–218. URL http://www.jstor.org/stable/2296809.

Schnabel RB, Koontz JE, Weiss BE (1985). "A Modular System of Algorithms for Unconstrained Minimization." *ACM Transactions on Mathematical Software*, **11**(4), 419–440.

Schwab M, Karrenbach M, Claerbout J (2000). "Making Scientific Computations Reproducible." *Computing in Science & Engineering*, **2**(6), 61–67.

Shanno DF (1970). "Conditioning of Quasi-Newton Methods for Function Minimization." *Mathematics of Computation*, **24**, 647–656.

Sorenson HW (1969). "Comparison of Some Conjugate Direction Procedures for Function Minimization." *Journal of the Franklin Institute*, **288**(6), 421–441.

Storn R, Price K (1997). "Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces." *Journal of Global Optimization*, **11**, 341–359.

Sun K, Henderson DJ, Kumbhakar SC (2011). "Biases in Approximating Log Production." *Journal of Applied Econometrics*, **forthcoming**.

Thursby JG (1980). "CES Estimation Techniques." *The Review of Economics and Statistics*, **62**, 295–299.

Thursby JG, Lovell CAK (1978). "In Investigation of the Kmenta Approximation to the CES Function." *International Economic Review*, **19**(2), 363–377.

Uebe G (2000). "Kmenta Approximation of the CES Production Function." Macromoli: Goetz Uebe's notebook on macroeconometric models and literature, `http://www2.hsu-hh.de/uebe/Lexikon/K/Kmenta.pdf` [accessed 2010-02-09].

Uzawa H (1962). "Production Functions with Constant Elasticity of Substitution." *The Review of Economic Studies*, **29**, 291–299.

Affiliation:

Arne Henningsen
Institute of Food and Resource Economics
University of Copenhagen
Rolighedsvej 25
1958 Frederiksberg C, Denmark
E-mail: `arne.henningsen@gmail.com`
URL: `http://www.arne-henningsen.name/`

Géraldine Henningsen
Systems Analysis Division
Risø National Laboratory for Sustainable Energy
Technical University of Denmark
Frederiksborgvej 399
4000 Roskilde, Denmark
E-mail: `gehe@risoe.dtu.dk`