Misspecification in Stochastic Frontier Models. Backpropagation Neural Network Versus Translog Model: A Monte Carlo Comparison[1]

by

C. Guermat
University of Exeter, Exeter EX4 4RJ, UK
C.Guermat@exeter.ac.uk
and
K. Hadri
City University, Northampton Square, London EC1V 0HB, UK
K.Hadri@city.ac.uk

First version: April 1999

Corresponding author
Kaddour Hadri
Department of Economics
City University
London, EC1 0HB, UK
E-mail: K.Hadri@City.ac.uk

JEL classification: C15; C21; C24; D24; Q12.
Keywords: Stochastic Frontier Production; Backpropagation Neural Network; Technical Efficiency; Monte Carlo; Maximum likelihood Estimation.

[1] We gratefully acknowledge the hospitality of the School of Business and Economics at Exeter University and my thanks go to the Head of the School, Mr. Martin Timbrell.

Summary:

Little attention has been given to the effects of misspecification on the estimation of stochastic frontier models and to the possibility of using backpropagation neural network as a flexible functional form to approximate the production and cost functions. This paper has two aims. First, it uses Monte Carlo experimentation to investigate the effect of misspecification on the finite sample properties of the maximum likelihood (ML) estimators of the half-normal stochastic frontier production functions. Second, it compares the performance of backpropagation neural network with that of translog. It is found that when the neural network is a serious alternative to the translog formulation, but that both options produce poor efficiency rankings when the data is generated by a Leon Tief or CES functions. Hence, when estimating efficiency, one should be aware of the possibilities of misspecification and alternative models should be considered.

1. Introduction

Stochastic frontier production for cross-section data was independently suggested by Aigner, Lovell and Schmidt (1977), Battese and Corra (1977) and Meeusen and van den Broeck (1977). Although there has been considerable research to extend and apply the initial model, stochastic frontier modeling has been centered around the Cobb-Douglas and the Translog formulations. A recent survey of this research is provided by Green (1993). The first step in modeling a production function is to assume a structure of technology. This has usually meant specifying a production function with few parameters. However, recent advances in computer power and statistical techniques have made it easier to hanlde more complicated, but more flexbile, functional forms. To the best of our knowledge, there has been no attempt to apply neural network methods as an alternative to the translog model. There is also a lack of research on the effect of misspecification on the finite sample properties of efficiency estimation.

Caudill (1995) conducted an extensive Monte Carlo study on the finite sample properties of stochastic frontier production functions assuming only a constant in the production function. The effects of heteroscedasticity in stochastic frontier models were investigated by Caudill and Ford (1993) and Guermat and Hadri (1999) using Monte Carlo experiments. The effects of misspecification have not been investigated yet and we intend to fill this gap. At the same time we attempt to see whether backpropagation neural network can be a useful addition to the modeling of efficiency using stochastic production frontiers.

The remainder of this paper is organized as follows. Section 2 outlines the models and notation while in section 3 we describe the Monte Carlo experiment. In section 4 we present and discuss the results. The final section presents the main findings and remarks.

2. Models and Notation

The basic model used in the literature to describe a frontier production function can be written as follows:

$$y_i = g(X_i, \beta) + w_i - v_i, \tag{1}$$

where y_i denotes the logarithm of the production for the ith sample farm ($i = 1, \ldots, N$) ; $g(.)$ is a function such as the Cobb-Douglas or the Translog function; X_i is a $(1 \times k)$ vector of the logarithm of the inputs associated with the ith sample farm (the first element would be one when an intercept term is included); β is a $(k \times 1)$ vector of unknown parameters to be estimated; w_i is a two-sided error term with $E[w_i] = 0$, $E[w_i w_j] = 0$ for all i and j, $i \neq j$; $\text{var}(w_i) = \sigma_w^2$; v_i is a non-negative one-sided error term with $E[v_i] > 0$, $E[v_i v_j] = 0$ for all i and j, $i \neq j$; and $\text{var}(v_i) = \sigma_v^2$. Furthermore, it is assumed that w and v are uncorrelated. The one-sided disturbance v reflects the fact that each firm's production must lie on or below its frontier. Such a term represents factors under the firm's control. The two-sided error term represents factors outside the firm's control.

If we assume that v_i is half-normal and w_i is normal, then the density function of their sum, derived by Weinstein (1964), takes the form:

$$f(\epsilon_i) = (2/\sigma) f^*(\epsilon_i/\sigma)\left(1 - F^*(\lambda\epsilon_i/\sigma)\right), \quad -\infty < \epsilon_i < +\infty, \tag{2}$$

where $\epsilon_i = w_i + v_i, \sigma^2 = \sigma_w^2 + \sigma_v^2$, $\lambda = \sigma_v/\sigma_w$ and $f^*(.)$ and $F^*(.)$ are, respectively, the standard normal density and distribution functions. The advantage of stochastic frontier estimation is that it permits the estimation of firm-specific inefficiency. The most widely used measure of firm-specific inefficiency suggested by Jondrow, Lovell, Materov and Schmidt (1982) based on the conditional expected value of v_i given ϵ_i is given by

$$E\left[v_i | \epsilon_i\right] = \sigma_* \left[-\epsilon_i\lambda/\sigma + f^*(\epsilon_i\lambda/\sigma)/F^*(-\epsilon_i\lambda/\sigma)\right], \tag{3}$$

where $\sigma_* = \sigma_v \sigma_w / \sigma$.

The log-likelihood function is

$$\log L\left(\beta, \gamma, \theta\right) = \sum_{i=1}^{N} \log\left(f_i\left(\epsilon_i\right)\right). \tag{6}$$

In this paper we consider the following functional forms for $g(.)$:

1. Cobb-Douglass:

$$g(x) = \beta_0 \prod_{i=1}^{k} x_i^{\beta_i}$$

2. Translog:

$$g(x) = \beta_0 + \sum_{j=1}^{k} \beta_j \ln x_j + \frac{1}{2} \sum_i \sum_j \beta_{ij} \ln x_i \ln x_j$$

$(\beta_{ij} = \beta_{ii})$.

3. Backpropagation Neural Network:
We use the three layer (neurons) backpropagation function:

$$g(x) = \sum_{j=1}^{3} \beta_j / (1 + \exp(\sum_{i=1}^{k} \beta_{ij} x_i))$$

(in the simulation we take $\ln x$ instead of x for comparison purpose).

4. CES:

$$g(x) = \beta_0 (\sum_{i=1}^{k} \beta_i x_i^p)^{1/p}$$

5. Generalised Leontief:

$$g(x) = \sum_i \sum_j \beta_{ij} (x_i x_j)^{1/2}$$

$(\beta_{ij} = \beta_{ii})$.

3. The Monte Carlo Design

In order to investigate the performance of the neural network specification against the translog and in order to examine the effects of misspecification on the finite sample properties of the maximum likelihood (ML) estimators of the half-normal stochastic frontier production functions we use a Monte Carlo experiment. We use two inputs (k=2) to simulate frontier data using a simple Cobb-Douglass function, the CES function (with p=0.5), and the Generalised Leontief function. We also use the translog function to compare the performance of the neural network model with the simple Cobb-Douglass model. The general form of the models is as follows:

$$lnQ_i = g(L_i, K_i, \beta) + W_i - V_i,$$

where W and V are a normal error variable and a half normal error respectively. We generate data using the following procedure:

$\{L_i, K_j\}$ are pairs (i, j), $i = 1, \ldots, \sqrt{N}$, and $j = 1, \ldots, \sqrt{N}$ recursively. The two error terms were generated as $W \sim N(0, \sigma_w^2)$ and $V \sim |N(0, \sigma_v^2)|$.

The variances were modeled as follows:

$\sigma_v = \alpha_0$

$\sigma_w = \gamma_0$

The parameters were set as follows:

1. For the Cobb-Douglass and Translog models:
$\beta_0 = 1$,
$\beta_1 = \beta_2 = 0.5$,
$\alpha_0 = 0.3$,
$\gamma_0 = 0.01$.
We also used $\alpha_0 = 3, \gamma_0 = 0.1$ for the Cobb-Douglass model.

2. For the CES and the Generalised Leontief models:
$\beta_0 = 2$,
$\beta_1 = \beta_2 = 1$,
$\alpha_0 = 0.3$,
$\gamma_0 = 0.01$.

We also considered different sample sizes: 100, 200, 300 and 400 observations.

To find out if the number of replication of 1000 (see Hadri and Garry (1998) on the importance of the number of replications) was sufficient, we carried out some experiments with 20000 replications. We found that the results were not significantly different.

For each replication, and for both models, we compute the mean estimated efficiency, its standard deviation, the rank correlation coefficient between the estimated efficiencies and the true efficiencies, and the maximum and minimum estimated efficiencies. At the end of the replications, these statistics are averaged and compared to the true mean, maximum and minimum efficiencies.

4. The Results

The main results of the Monte Carlo experiment are presented in Table 1-5.

The mean efficiency, the standard deviations, the mean rank correlation coefficients between the true rank and the estimated one, and the mean maximum and minimum are shown.

In Table 1 the data was generated by a Cobb-Douglass function with two different sets of variances. This shows the case of high mean efficiency as in Table

1 as well as the case of low mean efficiency as in Table 2. Both models performed well. The mean estimated efficiency is very close to the true one and the standard deviation is very small for all N. The maxima and minima are also very close to the true ones. The mean rank correlations are all over 93%. However, the tables reveal that the translog model is slightly more accurate than the neural network model. This is expected since the translog model is nested with the Cobb-Douglass model.

Table 3 shows the result for the CES case. The estimators are slightly less accurate compared to the previous case, but the mean efficiency, standard deviation, maximum and minimum efficiencies are close enough to the true ones. We find no significant difference between the translog and the neural network models in terms of these statistics. However, the ranking of efficiencies deteriorates significantly, varying between 68.7% and 26.8%. We note that the neural network produces a higher ranking than the translog.

Table 4 displays the results for the generalised Leontief case. The accuracy is similar to the CES case, except the ranking which has worsened. In this case the highest mean rank correlation is only 42.1%, with the translog model being marginally better for N=100 and 200 while the neural network has higher ranking for N=300 and 400. In general there is little difference between the two models when the true data are generated by a generalised Leontief model.

Finally, Table 5 shows the results when the data were gerated by the translog function. The comparison is now made between the neural network model and the Cobb-Douglass model. The superiority of the neural network is clear in every aspect.

5. Conclusion

The results of the present Monte Carlo experiment show that misspecification can have a serious effect on the ranking of firms. However, if the researcher is only interested in the mean efficiency, the simulation shows that misspecification does not affect the mean, maximum or mimimum significantly. The estimation remains acceptable for both the translog and the neural network models. On the other hand, if the aim is to rank the firms in terms of efficiency, we note that when using the wrong specification, the ranking could be misleading. The rank correlation between the true and estimated efficiencies can drop to less than 30% when we use the wrong model. The neural network specification is a good alternative to the most commonly used translog function. In the Cobb-Douglass case the translog is only marginally superior. In the generalised Leontief case, the two models are similar. But in the CES case the neural network produces higher mean rank correlations. Finally, when we compared the neural network with the Cobb-Douglass models on data generated by a translog function, the neural network model was well ahead.

This paper shows that there is a need for considering various specifications when modeling production and cost functions. The general practice of relying solely on a Cobb-Douglass or Translog functions may lead to serious misrepresentation of the ranking of firms, although the mean efficiency remains accurate. We have also shown that the neural network specification should also be considered among other specifications.

References

Aigner, D.J., Lovell, C.A.K., and Schmidt, P. (1977), "Formulation and Estimation of Stochastic Frontier Production Function Models," *Journal of Econometrics,* 6, 21-37.

Battese, G.E., and Corra, G.S. (1977), "Estimation of a Production Frontier Model: With Application to the Pastoral Zone of Eastern Australia," *Australian Journal of Agricultural Economics*, 21, 169-179.

Caudill, S.B. and Ford, J.M. (1993) "Biases in frontier estimation due to heteroscedasticity," *Economics Letters,* 41, 17-20.

Caudill, S.B, Ford, J.M., and Gropper, D.M. (1995), "Frontier Estimation and Firm-Specific Inefficiency Measures in the Presence of Heteroscedasticity," *Journal of Business & Economic Statistics*, 13, 105-111.

Coelli, T.J. (1995) Estimation and hypothesis tests for a stochastic frontier function: a Monte Carlo analysis. *J. Prod. Anal.,* 6, 247-268.

Green, W.H. (1993), "The econometric Approach to Efficiency Analysis," in *The measurement of Productive Efficiency*, eds. H.O. Fried, C.A.K. Lovell, and S.S. Schmidt, New York: Oxford University Press, pp. 68-119.

Hadri, K., and J. Whittaker. (1995) "efficiency, environmental contaminants and farm size: testing for links using stochastic production frontiers" discussion paper in economics 95/05, Exeter University.

Hadri, K. (1997). "A frontier Approach to Disequilibrium Models", *Applied Economic Letters,* No. 4, pp. 699-701.

Hadri, K., C. Guermat, and J. Whittaker. (1999) "Doubly heteroscedastic stochastic production frontiers with application to English cereals farms" Discussion paper in economics , Exeter University.

Hadri, K. and G.D.A. Phillips. (1999) "The accuracy of the higher order bias approximation for the 2SLS estimator," *Economics Letters,* 62, 167-174.

Hadri, K. (1999) "Estimation of a doubly heteroscedastic stochastic frontier cost function", forthcoming in the *Journal of Business & Economic Statistic.*

Jondraw, J., Lovell, C.A.K., Materov, I., and Schmidt, P. (1982), "On the Estimation of Technical Inefficiency in Stochastic Production Function Model," *Journal of Econometrics,* 19, 233-238.

Meeusen, W., and van den Broeck, J. (1977), "Efficiency Estimation from Cobb-Douglas Production Function with Composed Error", *International Economic Review,* 18, 435-444.

Prais, S.J. and H.S. Houthaker. (1955), *The analysis of family budgets,* Cambridge University Press, Cambridge.

Weistein, M.A. (1964), "The Sum of Values From a Normal and a Truncated Normal Distribution," *Technometrics*, 6, 104-105 (with some additional material, 469-470).

White, H. (1982), "Maximum Likelihood Estimation of Misspecified Models," *Econometrica,* 50, 1-25.

Table 1. Simulation Results (DGP=Cobb Douglass) (σ_v=0.3, σ_w =0.01).

	Translog	Neural Net.	True
N=100			
Eff. mean	0.796641	0.797461	0.790652
Eff. SD	0.000200	0.000264	
Mean Rank	0.940294	0.934821	
Eff. Max	0.991886	0.991840	0.996310
Eff. Min	0.454515	0.455328	0.447107
N=200			
Eff. mean	0.784962	0.787442	0.780099
Eff. SD	0.000183	0.000238	
Mean Rank	0.941050	0.935395	
Eff. Max	0.993648	0.992635	0.998476
Eff. Min	0.377532	0.400251	0.387052
N=300			
Eff. mean	0.791277	0.791880	0.789338
Eff. SD	0.000103	0.000124	
Mean Rank	0.952605	0.947841	
Eff. Max	0.994400	0.994140	0.998476
Eff. Min	0.387995	0.383656	0.387052
N=400			
Eff. mean	0.793538	0.793798	0.791558
Eff. SD	0.000092	0.000102	
Mean Rank	0.954144	0.951636	
Eff. Max	0.994875	0.994781	0.998476
Eff. Min	0.389131	0.386802	0.387052

Table 2. Simulation Results (DGP=Cobb Douglass) (σ_v=3, σ_w =0.1).

	Translog	Neural Net.	True
N=100			
Eff. mean	0.257712	0.267214	0.230708
Eff. SD	0.005487	0.009389	
Mean Rank	0.934592	0.916695	
Eff. Max	0.983434	0.982717	0.963709
Eff. Min	0.000378	0.000373	0.000319
N=200			
Eff. mean	0.227999	0.230520	0.215391
Eff. SD	0.003115	0.003859	
Mean Rank	0.944472	0.938829	
Eff. Max	0.987240	0.984453	0.984866
Eff. Min	0.000068	0.000085	0.000075
N=300			
Eff. mean	0.225337	0.228553	0.225204
Eff. SD	0.002088	0.002151	
Mean Rank	0.949353	0.948893	
Eff. Max	0.981174	0.979451	0.984866
Eff. Min	0.000075	0.000078	0.000075
N=400			
Eff. mean	0.230968	0.232154	0.230575
Eff. SD	0.001757	0.001793	
Mean Rank	0.951837	0.951606	
Eff. Max	0.972730	0.973659	0.984866
Eff. Min	0.000077	0.000076	0.000075

Table 3. Simulation Results (DGP=CES) (σ_v=0.3, σ_w =0.01).

	Translog	Neural Net.	True
N=100			
Eff. mean	0.974336	0.978044	0.980236
Eff. SD	0.000197	0.000081	
Mean Rank	0.549103	0.689246	
Eff. Max	0.995745	0.995982	0.999827
Eff. Min	0.823909	0.828176	0.796639
N=200			
Eff. mean	0.965881	0.975387	0.981159
Eff. SD	0.001381	0.000210	
Mean Rank	0.307336	0.558957	
Eff. Max	0.997586	0.996601	0.999940
Eff. Min	0.747190	0.782932	0.743487
N=300			
Eff. mean	0.970621	0.981424	0.985739
Eff. SD	0.000660	0.000184	
Mean Rank	0.332620	0.502229	
Eff. Max	0.996251	0.997187	0.999929
Eff. Min	0.809134	0.828903	0.771402
N=400			
Eff. mean	0.986188	0.986988	0.992107
Eff. SD	0.000143	0.000129	
Mean Rank	0.268291	0.301877	
Eff. Max	0.997218	0.997287	0.999965
Eff. Min	0.871615	0.875337	0.810093

Table 4. Simulation Results (DGP=Leontief) (σ_v=0.3, σ_w =0.01).

	Translog	Neural Net.	True
N=100			
Eff. Mean	0.977440	0.978329	0.987949
Eff. SD	0.000245	0.000216	
Mean Rank	0.421267	0.409931	
Eff. Max	0.995939	0.995681	0.999881
Eff. Min	0.903712	0.904911	0.885448
N=200			
Eff. mean	0.981095	0.981842	0.988621
Eff. SD	0.000205	0.000183	
Mean Rank	0.351678	0.347385	
Eff. Max	0.996553	0.997091	0.999966
Eff. Min	0.895192	0.896057	0.854368
N=300			
Eff. mean	0.986886	0.988267	0.991234
Eff. SD	0.000133	0.000117	
Mean Rank	0.339024	0.345104	
Eff. Max	0.997049	0.997953	0.999959
Eff. Min	0.925520	0.930232	0.871182
N=400			
Eff. mean	0.986188	0.986988	0.992107
Eff. SD	0.000143	0.000129	
Mean Rank	0.268291	0.301877	
Eff. Max	0.997218	0.997287	0.999965
Eff. Min	0.871615	0.875337	0.810093

Table 5. Simulation Results (DGP=Translog) (σ_v=0.3, σ_w =0.01).

	Cobb Douglass	Neural Net.	True
N=100			
Eff. mean	0.731847	0.812962	0.790652
Eff. SD	0.010491	0.002628	
Mean Rank	0.634684	0.789943	
Eff. Max	0.992474	0.990079	0.996310
Eff. Min	0.419832	0.440518	0.447107
N=200			
Eff. mean	0.758908	0.807021	0.809038
Eff. SD	0.011463	0.000363	
Mean Rank	0.623901	0.913693	
Eff. Max	0.999858	0.993724	0.999117
Eff. Min	0.144988	0.486185	0.498365
N=300			
Eff. mean	0.685419	0.804701	0.802925
Eff. SD	0.018940	0.000405	
Mean Rank	0.695494	0.897739	
Eff. Max	0.996351	0.994653	0.999963
Eff. Min	0.351580	0.434296	0.430894
N=400			
Eff. mean	0.695379	0.802147	0.798988
Eff. SD	0.016046	0.000305	
Mean Rank	0.721267	0.915956	
Eff. Max	0.998189	0.995860	0.999342
Eff. Min	0.254976	0.421143	0.428512