The \texttt{frontier} package

Riccardo (Jack) Lucchetti

July 18, 2018

Contents

1 Introduction 1

2 Syntax and examples 2

\hspace{1em} 2.1 Estimating a basic model . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2

\hspace{1em} 2.2 Reparametrisations . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

\hspace{1em} 2.3 Modelling inefficiency . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

\hspace{1em} 2.4 An example . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

3 Public functions 5

1 Introduction

This package can be used to estimate stochastic frontier models; these models, introduced independently by Aigner \textit{et al.} (1977) and Meeusen and van den Broeck (1977),\footnote{For an introductory yet fairly comprehensive treatment, see section 19.2.4 in Greene (2012).} are an extension of the usual linear model where the equation is normally a cost or a production function (but not always) and the error term is the sum of two components:

\[ y_i = x_i' \beta + \varepsilon_i \]  \hspace{1em} (1)

\[ \varepsilon_i = v_i + \text{sgn} \cdot u_i \]

where \( v_i \) and \( u_i \) are two independent unobservable random variables. The former is almost invariably assumed to be a Gaussian r.v. with zero mean and variance \( \sigma^2_v \). The latter, instead, has support over the positive reals only; therefore, it is quite natural to interpret \( u_i \) as an “unobserved inefficiency” component. Clearly, for this interpretation to make sense, the scalar \( \text{sgn} \) should equal 1 for cost functions and \(-1\) for production functions. The
The frontier package supports the three mostly widely used specifications for the density of $u_i$:

1. The half-normal density, where $u_i \sim |N(0, \sigma_u^2)|$;
2. The exponential density, where $f(u_i) = \frac{1}{\sigma_u} \exp(-u_i/\sigma_u)$;
3. The truncated normal density, where $u_i \sim N(\mu_i, \sigma_u^2)$ with truncation at 0.

Note that the half-normal model is a special case of the truncated normal with $\mu_i = 0$.

The density of the composite error term $\varepsilon_i$ can be derived analytically, so estimation of (1) is a rather straightforward case of ML estimation. The frontier package uses the standard mle gretl command with analytical derivatives.

In several settings, it may be desirable to make the two variances $\sigma_v^2$ and $\sigma_u^2$ functions of observable covariates. This is accomplished by assuming a form of multiplicative heteroskedasticity, as in

$$
\sigma^2_v = \exp (z_{v,i}' \gamma_v), \quad \sigma^2_u = \exp (z_{u,i}' \gamma_u).
$$

It is worth noting that, given the functional forms for the densities that are generally adopted, higher variance for $u_i$ translates into higher mean, and therefore the $z_{u,i}$ covariates may be interpreted as explanatory variables for the inefficiency component (see also Caudill et al. (1995)).

Apart from the model parameters, interest is often centred on estimating efficiency scores for the units in the sample. The quantities that are almost universally used for this purpose are the unbiased estimator of $u_i$ given $\varepsilon_i$, or of its exponential:

$$
\hat{u}_i = E(u_i | \varepsilon_i), \quad \hat{e}_i = E[\exp(-\text{sgn} \cdot u_i) | \varepsilon_i]
$$

2 Syntax and examples

2.1 Estimating a basic model

The simplest invocations is

```r
bundle m = frontier(rC, X, 1, 1)
```

When calling the frontier function, you can append an optional fifth parameter to control for the degree of verbosity of the function. 0 means “print nothing”, 1 (the default) prints the estimates, while 2 shows you the BFGS iterations, which may be useful when things go wrong.

The resulting bundle m contains all the quantities that may be of interest, such as a coeff element, a vcov element, etc.
If you choose to run the function with 0 verbosity, but want to print out the bundle later on, you can use the `frontier_printout` function, that takes as its only argument the pointer to the bundle you want to print.

After estimation, the bundle will contain two retrievable series called `uhat` and `ehat` for $\hat{u}_i$ and $\hat{e}_i$ in equation (3), respectively.

### 2.2 Reparametrisations

In order to achieve best numerical performance, some parameters are re-expressed in an equivalent way, and that’s the way they appear in the printout.

For the half-normal and the exponential models, the two variances $\sigma^2_u$ and $\sigma^2_v$ are turned into their natural logarithm. The reparametrisation that takes place in the truncated normal model, instead, is more complex: the log density of $\varepsilon_i$ is written as

$$
\ell_i = -\frac{\ln(2\pi)}{2} - \lambda - \frac{e_i - sgn \cdot \mu_i}{2} + \ln \Phi(w_i) - \ln \Phi\left(\frac{\mu_i}{\sigma\sqrt{\gamma}}\right)
$$

where $\Phi(\cdot)$ is the standard Normal cdf,

$$
\lambda = \ln \sigma^2, \quad \sigma = \sqrt{\sigma^2_u + \sigma^2_v} \quad \gamma = \frac{\sigma^2_u}{\sigma^2} \quad R = \frac{\gamma}{1 - \gamma} \quad G = \ln R
$$

and

$$
w_i = \frac{1}{\sigma} \left[ \mu_i / R + sgn \cdot R \cdot e_i \right].
$$

The parameters that you actually see printed in the output are what we just called $\lambda$ and $G$.

### 2.3 Modelling inefficiency

The `frontier` function can be given two additional list parameters. The first one, `Zu` is used for modelling inefficiency directly, and its interpretation changes with the type of model: for the half-normal and exponential models, it affects the variance of $u_i$ in the multiplicative form shown in equation (2); for the truncated normal model, instead, the $z_i$ variables enter linearly the term $\mu_i$. In other words, if the parameter `Zu` is not `null` and the model type is 3, the assumed distribution for $u_i$ is a zero-truncated normal with mean $z'_{u,i} \cdot \gamma_u$ and variance $\sigma^2_v$.

The second one is for heteroskedasticity of $v_i$ (not used in type 3).

### 2.4 An example

The following code snippet reproduces the results for 1991 in Lucchetti et al. (2001).
set verbose off
include frontier.gfn

open banks91.gdt
rC = log(VC/P1)
q1 = log(Q1)
q2 = log(Q2)
rP2 = log(P2/P1)
rP3 = log(P3/P1)

list X = const q1 q2 rP2 rP3
model = frontier(rC, X, 1, 1)
# table 2
frontier_printout(model)
# table 3
series u = model.uhat
summary u --simple

yields

Stochastic frontier model (cost): half-normal

Main equation

<table>
<thead>
<tr>
<th>coefficient</th>
<th>std. error</th>
<th>z</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>const</td>
<td>-1.42516</td>
<td>0.370801</td>
<td>-3.843</td>
</tr>
<tr>
<td>q1</td>
<td>0.989612</td>
<td>0.0289675</td>
<td>34.16</td>
</tr>
<tr>
<td>q2</td>
<td>0.0203770</td>
<td>0.0277650</td>
<td>0.7339</td>
</tr>
<tr>
<td>rP2</td>
<td>0.172992</td>
<td>0.0321305</td>
<td>5.384</td>
</tr>
<tr>
<td>rP3</td>
<td>0.581563</td>
<td>0.0498758</td>
<td>11.66</td>
</tr>
</tbody>
</table>

Inefficiency component

<table>
<thead>
<tr>
<th>coefficient</th>
<th>std. error</th>
<th>z</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>log V(u)</td>
<td>-4.17182</td>
<td>0.229327</td>
<td>-18.19</td>
</tr>
</tbody>
</table>

Idiosyncratic variance

<table>
<thead>
<tr>
<th>coefficient</th>
<th>std. error</th>
<th>z</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>log V(v)</td>
<td>-5.71351</td>
<td>0.282474</td>
<td>-20.23</td>
</tr>
</tbody>
</table>

N. of observations = 198, Log-likelihood = 190.825

Summary statistics, using the observations 1 - 198
for the variable 'u' (198 valid observations)

Mean                    0.098029
3 Public functions

function bundle frontier(series y, list X, int type, bool cost, int verbose, list Zu, list Zv)

Fits the model and returns a bundle. The arguments are

1. y, the dependent variable
2. X, the explanatory variables for the production/cost function
3. type, distribution for the inefficiency term: Half-normal, Exponential or Truncated normal, respectively
4. cost, Boolean, for a cost function (default: false)
5. verbose verbosity level (0 to 2, default 1)
6. Zv List of explanatory variables for the inefficiency term (default: constant)
7. Zu List of explanatory variables for heteroskedasticity (default: constant)

function void frontier_printout(bundle *b)

Given the bundle whose pointer is the function argument, prints the model that it contains, and returns nothing
References


