## Stochastic Frontier Analysis

### SFA Overview

Output-oriented technical efficiency can be written as:

$$\ln y_i = \ln y_i^* - u_i, \;\; u_i \geq 0$$

$$\ln y_i^* = f(\mathbf{x_i}; \beta) + v_i$$

Where $u_i$ is production inefficiency and $v_i$ is a zero-mean random error term.

If we rearrange this equation, we cal see that $u_i$ is the difference between the frontier production and the observed production, i.e. $u_i = \ln y_i^* - \ln y_i$. As $u_i$ approaches zero, the producer is becoming *more efficienct*. We can also construct a measure of *efficiency*: $e^{-u_i} = \frac{y_i}{y_i^*}$. Where $e^{-u_i} \cdot 100$ gives the percentage of the maximum output that firm $i$ produces. And $0 < e^{-u_i} \leq 1$

### Distribution-Free Approaches

Today we will cover two (of the three) distribution-free approaches for measuring $u_i$ and (later) will use this to introduce the maximum likelihood estimation methods.

These are called distribution-free approaches because they do not impose any structure on the error term ($v_i$). These models are deterministic (like DEA), meaning that they exclude the error term ($v_i$).

### Corrected OLS (COLS)

Winsten (1957)

The deterministic model can be written:

$$\ln y_i = \ln y_i^* - u_i, \quad u_i \geq 0$$

$$\ln y_i^* = f(\mathbf{x_i}; \beta)$$

If we specify a Cobb-Douglass production function, we can write this as:

$$\ln y_i = \beta_0 + \sum_j \beta_j \ln x_j - u_i$$

Or in the translog case:

$$\ln y_i = \beta_0 + \sum_j \beta_j \ln x_j + \frac{1}{2} \sum_j \sum_k \beta_{jk} \ln x_j \ln x_k - u_i, \ \beta_{jk} = \beta_{kj}$$

We need to estimate a frontier function that bounds the observations ($\ln y_i$) from above. COLS does this in a two-stage procedure where slope coefficients are estimated and the resulting production function is shifted upward until it bounds all observations in the data.

Formally:

**Stage 1:** OLS:

$$\ln y_i = \hat{\beta}_0 + \mathbf{x_i'} \hat{\tilde{\beta}} + \hat{e}_i$$

Because $E[u_i] \neq 0$, $\hat{\beta}_0$ is a biased estimate of $\beta_0$, but $\hat{\tilde{\beta}}$ is a consistent estimate of $\tilde{\beta}$

**Stage 2:** Adjust the OLS slope intercept upward by the amount of $\max\{\hat{e}_i\}$, so that the adjusted function bounds all observations from above. The residuals of this new estimating equation can now be written as:

$$\hat{e}_i - \max\{\hat{e}_i\} = \ln y_i - \{\left[\hat{\beta}_0 + \max\{\hat{e}_i\}\right] + \mathbf{x_i'}\hat{\tilde{\beta}}\} \leq 0$$

With

$$\hat{u}_i = -(\hat{e}_i - \max\{\hat{e}_i\}) \geq 0$$

Where $\hat{u}_i$ is our measure of technical inefficiency, and we can write technical efficiency as: $e^{-\hat{u}_i}$. Recall that $e^{-u_i} \cdot 100$ gives the percentage of the maximum output that firm $i$ produces. And $0 < e^{-u_i} \leq 1$.

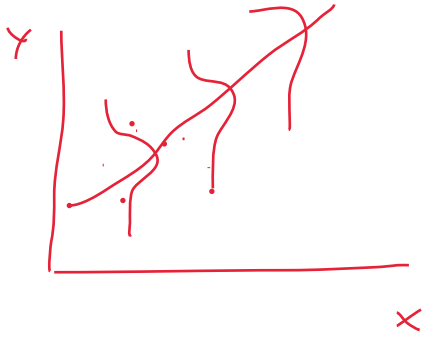
**Corrected Mean Absolute Deviation (CMAD)**

Follow the same two step procedure, but use a regression through the median, rather than mean in the first stage.

This is equivalent to the quantile regressions we discussed earlier, but rather than running multiple quantile regressions, you will just run one through the middle (medians).

$$Q_{y_i}(\tau|x_i) = \alpha(\tau) + \beta(\tau)x_i$$

$$Q_{y_i}(\frac{1}{2}|x_i) = \alpha_{\frac{1}{2}} + \beta_{\frac{1}{2}}x_i$$

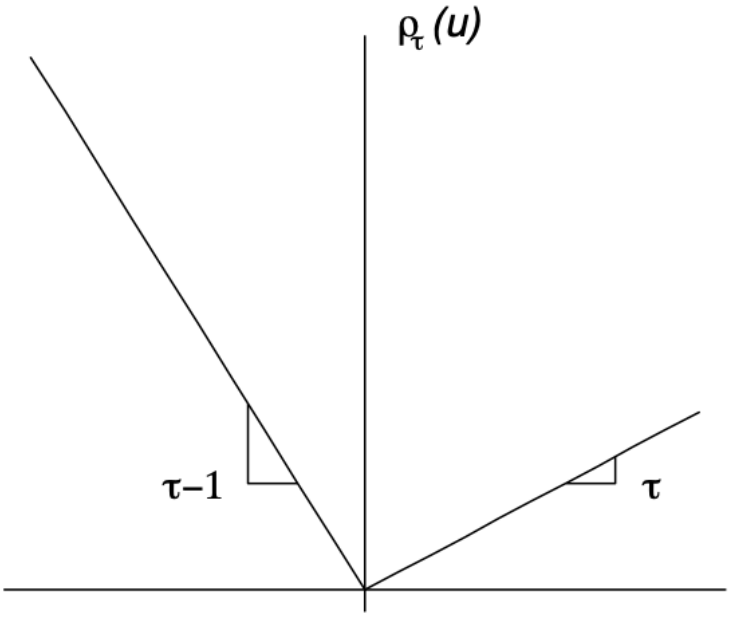



Where

$$\hat{\beta(\tau)} = argmin_{\beta \in \mathbb{R}^p} \sum \rho_{1/2}(y_i - x'\beta)$$

And

$$\rho_\tau(u) = u(\tau - I(u < 0))$$




Evaluated at the median, the quantile regression estimator is equivalent to:

$$\hat{\beta(\tau)} = \min_{\beta \in \mathbb{R}^p} \sum |y_i - x'\beta|$$

**Input Oriented TE**

Recall that the OO production function is:

$$y = f(\mathbf{x}e^{-\eta})$$

And the IO production function is:

$$y = f(\mathbf{x})e^{-\eta}$$

Then the Cobb-Douglas IO model is:

$$\ln y_i = \beta_0 + \sum_j \beta_j \ln x_j - (\sum_j \beta_j)\eta$$

Which is equivalent to the OO model with $u_i = \eta \sum_j \beta_j$

So we can get IO TE from: $\eta_i = \frac{u_i}{\sum \beta_j}$

**Distribution-free approaches in the dairy data:**

We have already used the `dairy` dataset, consisting of cross-sectional observations on 196 dairy farms, for our DEA analysis. We are now going to extend this to the SFA (parametric) framework.

For the SFA analysis we will be using some user-written STATA programs. To install these:

```
* install data and ado files from Kumbhakar, S.C. and Wang, H-J a
net install sfbook_install, from(https://sites.google.com/site/sf

sfbook_install

* Load dataset
use dairy, clear
```

Note that this will install (user written) SFA ado files to your computer in your `PLUS` directory and will install the accompanying datasets in your `c:\sfbook_demo` for Windows and `/users/c(username)/sfbook_demo` for Mac.

#### COLS IN STATA

```
* Let's start with single input, single output
global xvar llabor

* Stage 1: OLS regression
        regress ly $xvar

* recall that the OLS coefficient on llabor is consistent, but th
```

```
* Stage 2: adjust intercept and estimate efficiency
        * store residuals
        predict e, resid

        * get max(resid)
        sum e

        * generate inefficiency and efficiency
        gen double u = -(e -r(max))
        gen double eff = exp(-u)

        sum u eff

        * can plot OLS and COLS:
        regress ly $xvar

        predict y_hat, xb

        qui sum e
        gen double y_hat_cols = y_hat + r(max)

        twoway (scatter ly llabor) (line y_hat llabor) (line y_ha
```

#### CMAD IN STATA

```
* Stage 1: quantile regression
        qreg ly $xvar, quant(0.5) /* note that quant(0.5) is defa

* Stage 2: adjust intercept and estimate efficiency
        * store residuals
        predict e_cmad, resid

        * get max(resid)
        sum e_cmad

        * generate inefficiency and efficiency
        gen double u_cmad = -(e_cmad -r(max))
        gen double eff_cmad = exp(-u_cmad)
```

```
sum u_cmad eff_cmad

* can plot OLS and COLS:
reg ly $xvar, quant(0.5)

predict y_hat_mad, xb

qui sum e_cmad
gen double y_hat_cmad = y_hat_mad + r(max)

twoway (scatter ly llabor) (line y_hat_mad llabor) (line
```

**COMPARE:**

```
* compare COLS and CMAD frontiers
        twoway (scatter ly llabor) (line y_hat_mad llabor

* compare with DEA frontier
        * create DEA efficiency score for one-input one-o
        gen ratio = ly/llabor
        qui sum ratio
        gen eff_dea = ratio/r(max)
        sum eff_dea

        * scatter plot with labels of efficiency score (r
        gen eff_dea_round = round(eff_dea, 0.01)
        twoway scatter ly llabor , legend(off) mlabel(eff

        * create a line that goes through the origin and
        sum ly if eff_dea==1
        local delta_y = r(max)
        sum llabor if eff_dea==1
        local delta_x = r(max)

        local slope = `delta_y'/`delta_x'

        gen y_hat_dea = `slope'*llabor
```

```
* compare all
        twoway (scatter ly llabor) (line y_hat_mad llabor


* compare measures of efficiency:
        order eff_dea eff_cmad eff farmid
        gsort -eff_dea
```

| eff_dea | eff_cmad | eff_cols | farmid |
|---|---|---|---|
| 1 | 1 | 1 | 57 |
| .97081 | .81200465 | .81200465 | 171 |
| .96567729 | .66734779 | .66734779 | 129 |
| .96566893 | .87113102 | .87113102 | 194 |
| .9638675 | .95473928 | .95473928 | 1 |
| .96335548 | .58838447 | .58838447 | 97 |
| .96195741 | .75124638 | .75124638 | 159 |
| .95838363 | .71491256 | .71491256 | 144 |
| .95799417 | .70243602 | .70243602 | 83 |
| .95345017 | .57336746 | .57336746 | 65 |
| .95306717 | .76619812 | .76619812 | 116 |
| .95181347 | .7193202 | .7193202 | 192 |
| .95175939 | .79975178 | .79975178 | 41 |
| .95168365 | .69475982 | .69475982 | 121 |
| .95163918 | .45847222 | .45847222 | 68 |
| .9514526 | .70254981 | .70254981 | 180 |
| .94982993 | .53316075 | .53316075 | 73 |
| .94966259 | .6898161 | .6898161 | 38 |
| .94904323 | .70213407 | .70213407 | 100 |

```
* compare ranks:
        foreach var of varlist eff eff_cmad eff_dea {
                gsort -`var'
```

```
        gen rank_`var' = _n
}

order eff_dea eff_cmad eff rank_* farmid
gsort -eff_dea
```

| rank_eff_cols | rank_eff_cmad | rank_eff_dea | farmid |
|---|---|---|---|
| 1 | 1 | 1 | 57 |
| 6 | 6 | 2 | 171 |
| 27 | 27 | 3 | 129 |
| 3 | 3 | 4 | 194 |
| 2 | 2 | 5 | 1 |
| 40 | 40 | 6 | 97 |
| 13 | 13 | 7 | 159 |
| 16 | 16 | 8 | 144 |
| 20 | 20 | 9 | 83 |
| 46 | 46 | 10 | 65 |
| 10 | 10 | 11 | 116 |
| 15 | 15 | 12 | 192 |
| 7 | 7 | 13 | 41 |
| 24 | 24 | 14 | 121 |
| 98 | 98 | 15 | 68 |
| 19 | 19 | 16 | 180 |
| 63 | 63 | 17 | 73 |
| 25 | 25 | 18 | 38 |
| 21 | 21 | 19 | 100 |
| 4 | 4 | 20 | 84 |