Data Envelopment Analysis

SAPIENZA UNIVERSITÀ DI ROMA

Outline

Managerial motivation

Efficiency and the CCR model

The envelopment form: the dual

BCC, scale and slacks

Worked example by hand

Case study and results

Sensitivity analysis

Exercises

Learning objectives (1/2)

At the end of this chapter you will be able to:

- define relative efficiency as a weighted ratio of outputs to inputs;
- turn the fractional model into an LP and write its dual;

Learning objectives (2/2)

At the end of this chapter you will be able to:

- read in the dual the reference units and the improvement targets;
- separate technical inefficiency from scale inefficiency.

Prerequisites

- From the module *Linear Programming*, Chapter 9 «Duality theory»: the dual of an LP and strong duality, which give the envelopment form
- From this module, Chapter 2 «Background: linear programming and sensitivity analysis»: shadow prices and their checking by perturbation

Managerial motivation

Introduction: Managerial motivation

- In this chapter we compare the **efficiency** of similar units
- Problem class: **LP**, one per unit
- The dual says whom to learn from and by how much to reduce resources

Twelve branches to compare

- A bank has twelve branches: they use staff and operating costs, and produce loans and deposits
- Which ones perform well? By how much should the others reduce their resources?
- The outputs have no common price, and a large branch is not inefficient just because it spends more
- We need a comparison with the **best observed practices**

Data Envelopment Analysis

- **DEA** (*Data Envelopment Analysis*) builds an efficiency index by solving one LP per unit
- No price fixed in advance: each unit chooses the most favourable weights
- Used for hospitals, schools, public offices, plants, departments

Summary: Managerial motivation

- Similar units with several inputs and several outputs
- No common price for the outputs
- One LP per unit: efficiency, peers, targets

Efficiency and the CCR model

Introduction: Efficiency and the CCR model

- We define the units, the data and relative efficiency
- We write the fractional model of Charnes, Cooper and Rhodes (CCR)
- We turn it into an LP: the multiplier form

Data (input of the model)

The units are called DMUs (*Decision Making Units*)

Symbol	Type	Meaning
n	$\in \mathbb{Z}_{\geq 1}$	number of DMUs, indexed by $j \in \{1, 2, \ldots, n\}$
m	$\in \mathbb{Z}_{\geq 1}$	number of inputs, indexed by $i \in \{1, 2, \ldots, m\}$
p	$\in \mathbb{Z}_{\geq 1}$	number of outputs, indexed by $r \in \{1, 2, \ldots, p\}$
x_{ij}	$\in \mathbb{Q}_{>0}$	amount of input i used by DMU j
y_{rj}	$\in \mathbb{Q}_{>0}$	amount of output r produced by DMU j
o	$\in \{1, 2, \ldots, n\}$	the DMU being evaluated

Relative efficiency

Definition 1: Relative efficiency

Given weights $u_r \geq 0$ *and* $v_i \geq 0$, *not all zero, the efficiency ratio of DMU* j *is*

$$E_j(u, v) = \frac{\sum_{r=1}^{p} u_r \, y_{rj}}{\sum_{i=1}^{m} v_i \, x_{ij}}.$$

The relative efficiency of DMU o *is the maximum of* $E_o(u, v)$ *over the weights for which no DMU has a ratio greater than* 1. *DMU* o *is efficient if it equals* 1.

The problem in words

- *We decide* the weights of the inputs and of the outputs
- *The objective* is to maximise the efficiency ratio of DMU o
- *The constraints*: with the same weights no DMU has a ratio greater than 1
- We introduce the following $p + m$ non-negative variables:

$$\begin{cases} u_r = \text{weight of output } r, & \forall r \in \{1, 2, \ldots, p\}, \\ v_i = \text{weight of input } i, & \forall i \in \{1, 2, \ldots, m\}. \end{cases}$$

The fractional model

Fractional CCR model for DMU o

$$\max \quad \frac{\sum_{r=1}^{p} u_r\, y_{ro}}{\sum_{i=1}^{m} v_i\, x_{io}} \tag{6.1a}$$

$$\text{subject to} \quad \frac{\sum_{r=1}^{p} u_r\, y_{rj}}{\sum_{i=1}^{m} v_i\, x_{ij}} \leq 1, \quad \forall j \in \{1, 2, \ldots, n\}, \tag{6.1b}$$

$$u_r \geq 0, \quad \forall r \in \{1, 2, \ldots, p\}, \tag{6.1c}$$

$$v_i \geq 0, \quad \forall i \in \{1, 2, \ldots, m\}. \tag{6.1d}$$

- (6.1a) maximises the ratio of DMU o
- (6.1b): no DMU exceeds the ratio 1 (n fractional constraints)
- (6.1c) and (6.1d) define the variables

The Charnes–Cooper transformation

- If all the weights are multiplied by the same $t > 0$, no ratio changes
- We choose t so that the denominator of the objective equals 1:

$$\sum_{i=1}^{m} v_i \, x_{io} = 1$$

- The objective becomes the numerator alone
- Each constraint (6.1b), multiplied by its positive denominator, becomes linear

The multiplier form

CCR model, multiplier form, for DMU o

$$\max \quad \sum_{r=1}^{p} u_r \, y_{ro} \tag{6.2a}$$

$$\text{subject to} \quad \sum_{i=1}^{m} v_i \, x_{io} = 1, \tag{6.2b}$$

$$\sum_{r=1}^{p} u_r \, y_{rj} - \sum_{i=1}^{m} v_i \, x_{ij} \leq 0, \quad \forall j \in \{1, 2, \ldots, n\}, \tag{6.2c}$$

$$u_r \geq 0, \quad \forall r \in \{1, 2, \ldots, p\}, \tag{6.2d}$$

$$v_i \geq 0, \quad \forall i \in \{1, 2, \ldots, m\}. \tag{6.2e}$$

Objective function and constraints

- (6.2a) maximises the weighted output of DMU o: it is its ratio, because the weighted input equals 1
- (6.2b) normalises the weighted input of DMU o to 1 (1 linear constraint)
- (6.2c): for every DMU the weighted output does not exceed the weighted input (n linear constraints)
- (6.2d) and (6.2e) define the variables
- The optimal value $\tilde{h}_o \in (0, 1]$ is the relative efficiency; for n DMUs we solve n LPs

Summary: Efficiency and the CCR model

- Efficiency = weighted output/input ratio, with the most favourable weights
- The fractional model is invariant to scaling the weights
- Fixing the weighted input to 1 gives an LP

The envelopment form: the dual

Introduction: The envelopment form

- We write the dual of the multiplier form
- We read it as a virtual DMU built from the observed DMUs
- We obtain peers, targets and shadow prices

The dual variables

- The primal is a maximisation: one equality constraint and $n \leq$ constraints
- To follow the DEA literature the dual variables are not called π:

$$\begin{cases} \theta = \text{dual variable of constraint (6.2b)}, \\ \lambda_j = \text{dual variable of constraint (6.2c) of DMU } j. \end{cases}$$

- θ is free (equality), $\lambda_j \geq 0$ ($\leq$ constraints of a maximisation)
- Full theory: Duality theory (Linear Programming course)

The envelopment form

CCR model, envelopment form, for DMU o

$$\min \quad \theta \tag{6.3a}$$

$$\text{subject to} \quad \theta\, x_{io} - \sum_{j=1}^{n} \lambda_j\, x_{ij} \geq 0, \qquad \forall i \in \{1, 2, \ldots, m\}, \tag{6.3b}$$

$$\sum_{j=1}^{n} \lambda_j\, y_{rj} \geq y_{ro}, \quad \forall r \in \{1, 2, \ldots, p\}, \tag{6.3c}$$

$$\theta \gtreqless 0, \tag{6.3d}$$

$$\lambda_j \geq 0, \qquad \forall j \in \{1, 2, \ldots, n\}. \tag{6.3e}$$

Objective function and constraints

- (6.3a) minimises the factor θ by which all the inputs of DMU o are contracted
- (6.3b): the combination weighted by the λ_j uses at most θ times each input of o (m constraints, from the v_i)
- (6.3c): the same combination produces at least each output of o (p constraints, from the u_r)
- (6.3d) and (6.3e) define the variables

What the envelopment form says

- $\sum_{j=1}^{n} \lambda_j \left(x_{\cdot j}, y_{\cdot j}\right)$ is a **virtual** DMU
- With $\lambda_o = 1$ and the other $\lambda_j = 0$ we get $\theta = 1$: hence $\tilde{\theta} \leq 1$
- Strong duality: $\tilde{\theta} = \tilde{h}_o$
- The optimal radial contraction of the inputs **is** the relative efficiency

Three pieces of managerial information

- **Peers**: the reference set $\{j : \tilde{\lambda}_j > 0\}$, the efficient DMUs to learn from
- **Targets**: the projection of o onto the efficient frontier
- **Shadow prices** of constraints (6.3b) and (6.3c): they are the optimal weights $\tilde{v}_i$ and $\tilde{u}_r$ of the primal (one of the optimal vectors, if it is not unique)

Summary: The envelopment form

- The dual looks for the virtual DMU that does at least as much with less
- $\tilde{\theta} = \tilde{h}_o$ by strong duality
- The $\tilde{\lambda}_j > 0$ identify the peers

BCC, scale and slacks

Introduction: BCC, scale and slacks

- Variable returns to scale: the BCC model
- Scale efficiency as the CCR/BCC ratio
- Slacks, targets, and weak and strong efficiency

Returns to scale

- The CCR model assumes **constant** returns to scale: double inputs, double outputs
- For branches of very different sizes the assumption is strong
- The model of Banker, Charnes and Cooper (BCC) requires $\sum_{j=1}^{n} \lambda_j = 1$
- The virtual DMU becomes a **convex** combination, of comparable size

The BCC model

BCC model, envelopment form, for DMU o

$$\min \quad \theta \tag{6.4a}$$

$$\text{subject to} \quad \theta\, x_{io} - \sum_{j=1}^{n} \lambda_j\, x_{ij} \geq 0, \qquad \forall i \in \{1, 2, \ldots, m\}, \tag{6.4b}$$

$$\sum_{j=1}^{n} \lambda_j\, y_{rj} \geq y_{ro}, \quad \forall r \in \{1, 2, \ldots, p\}, \tag{6.4c}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \tag{6.4d}$$

$$\theta \gtreqless 0, \tag{6.4e}$$

$$\lambda_j \geq 0, \qquad \forall j \in \{1, 2, \ldots, n\}. \tag{6.4f}$$

Scale efficiency

- (6.4d) shrinks the region: in a minimisation $\tilde{\theta}^{\mathrm{BCC}} \geq \tilde{\theta}^{\mathrm{CCR}}$

Definition 2: Scale efficiency

$$\mathrm{SE}_o = \frac{\tilde{\theta}_o^{\mathrm{CCR}}}{\tilde{\theta}_o^{\mathrm{BCC}}} \in (0, 1].$$

It equals 1 *at the most productive scale; it is smaller than* 1 *when part of the inefficiency depends only on size.*

- In the CCR: $\sum_{j=1}^{n} \tilde{\lambda}_j < 1$ too small, > 1 too large (if the solution is unique)

Slacks: the second phase

Second phase: maximum slacks for DMU o

$$\max \quad \sum_{i=1}^{m} s_i^- + \sum_{r=1}^{p} s_r^+ \tag{6.5a}$$

$$\text{subject to} \quad \sum_{j=1}^{n} \lambda_j \, x_{ij} + s_i^- = \tilde{\theta} \, x_{io}, \quad \forall i \in \{1, 2, \ldots, m\}, \tag{6.5b}$$

$$\sum_{j=1}^{n} \lambda_j \, y_{rj} - s_r^+ = y_{ro}, \quad \forall r \in \{1, 2, \ldots, p\}, \tag{6.5c}$$

$$\lambda_j \geq 0, \quad \forall j \in \{1, 2, \ldots, n\}, \tag{6.5d}$$

$$s_i^- \geq 0, \quad \forall i \in \{1, 2, \ldots, m\}, \tag{6.5e}$$

$$s_r^+ \geq 0, \quad \forall r \in \{1, 2, \ldots, p\}. \tag{6.5f}$$

Targets and weak and strong efficiency

- Targets of DMU o:

$$\hat{x}_{io} = \tilde{\theta}\, x_{io} - \tilde{s}_i^-, \qquad \hat{y}_{ro} = y_{ro} + \tilde{s}_r^+$$

Definition 3: Weak and strong efficiency

DMU o is strongly efficient if $\tilde{\theta} = 1$ and all the optimal slacks are zero; it is weakly efficient if $\tilde{\theta} = 1$ but at least one slack is positive.

Summary: BCC, scale and slacks

- BCC: $\sum_{j=1}^{n} \lambda_j = 1$, variable returns
- $\mathrm{SE} = \tilde{\theta}^{\mathrm{CCR}} / \tilde{\theta}^{\mathrm{BCC}}$
- The second phase measures the slacks and gives the targets

Worked example by hand

Introduction: Worked example by hand

- Five DMUs with one input and one output
- The solver gives the values, then we check them
- CCR, duality, BCC and weak efficiency on the same example

The instance

DMU	1	2	3	4	5
x_{1j}	2	4	6	5	2
y_{1j}	2	6	7	3	1

Multiplier form for DMU 1

$$
\begin{aligned}
\max \quad & 2\,u_1 & & & & \\
\text{subject to} \quad & & 2\,v_1 & = & 1 \\
& 2\,u_1 & - 2\,v_1 & \leq & 0 \\
& 6\,u_1 & - 4\,v_1 & \leq & 0 \\
& 7\,u_1 & - 6\,v_1 & \leq & 0 \\
& 3\,u_1 & - 5\,v_1 & \leq & 0 \\
& u_1 & - 2\,v_1 & \leq & 0 \\
& u_1, & v_1 & \geq & 0
\end{aligned}
$$

Example (1/3)

- **Step 1 — the solver values.**

$$\tilde{\theta}^{\mathrm{CCR}} = (0.6667,\ 1,\ 0.7778,\ 0.4,\ 0.3333)$$

$$\tilde{\theta}^{\mathrm{BCC}} = (1,\ 1,\ 1,\ 0.5,\ 1)$$

DMU 1: $\tilde{u}_1 = 0.3333$, $\tilde{v}_1 = 0.5$, only peer DMU 2 with $\tilde{\lambda}_2 = 0.3333$; DMU 5 (BCC): slack $\tilde{s}_1^+ = 1$

- **Step 2 — check of the CCR.** The ratio y/x divided by the best one, 1.5:

$$\tilde{h}_1 = \frac{2/2}{1.5} = 0.6667, \quad \tilde{h}_3 = \frac{7/6}{1.5} = 0.7778, \quad \tilde{h}_4 = \frac{3/5}{1.5} = 0.4$$

Example (2/3)

- **Step 3 — duality for DMU 1.** Normalisation, efficiency, active constraint of DMU 2:

$$2 \cdot 0.5 = 1, \qquad 2 \cdot 0.3333 = 0.6667, \qquad 6 \cdot 0.3333 - 4 \cdot 0.5 = 0$$

The virtual DMU $\tilde{\lambda}_2 \cdot (4, 6)$:

$$0.3333 \cdot 4 = 1.3333 = 0.6667 \cdot 2, \qquad 0.3333 \cdot 6 = 2$$

Example (3/3)

- **Step 4 — BCC for DMU 4.** $\tilde{\lambda}_1 = 0.75$, $\tilde{\lambda}_2 = 0.25$:

  $$0.75 \cdot 2 + 0.25 \cdot 4 = 2.5 = 0.5 \cdot 5, \qquad 0.75 \cdot 2 + 0.25 \cdot 6 = 3$$

  Scale efficiency: DMU 4 0.8, DMU 1 0.6667, DMU 3 0.7778

- **Step 5 — weak efficiency.** DMU 5 has $\tilde{\theta}^{\mathrm{BCC}} = 1$, but DMU 1 uses the same input and produces 1 more: with $\tilde{s}_1^+ = 1$ DMU 5 is weakly efficient

Summary: Worked example by hand

- With one input and one output the CCR efficiency is (y/x) divided by the best one
- The active constraint of the primal identifies the peer of the dual
- The BCC compares with convex combinations

Case study and results

Introduction: Case study and results

- Twelve branches, two inputs and two outputs
- CCR, BCC and scale efficiencies, peers and targets
- The frontier in a one-input, one-output view

The branch data

branch	1	2	3	4	5	6	7	8	9	10	11	12
staff (FTE)	10	25	23	18	17	27	9	24	12	10	20	30
costs (k€)	720	1420	1130	940	990	1620	640	1300	690	500	1020	1370
loans (M€)	19.6	60.7	43.5	31.7	39.6	39.7	21.9	34.5	29.2	23.7	42.8	63.5
deposits (M€)	18.5	58.7	63.0	50.5	43.2	73.1	38.0	72.7	28.7	26.3	45.7	62.9

- Realistic synthetic data generated by the script with a fixed seed (`dea_filiali.csv`)
- FTE: *Full-Time Equivalent*

The model in Gurobi

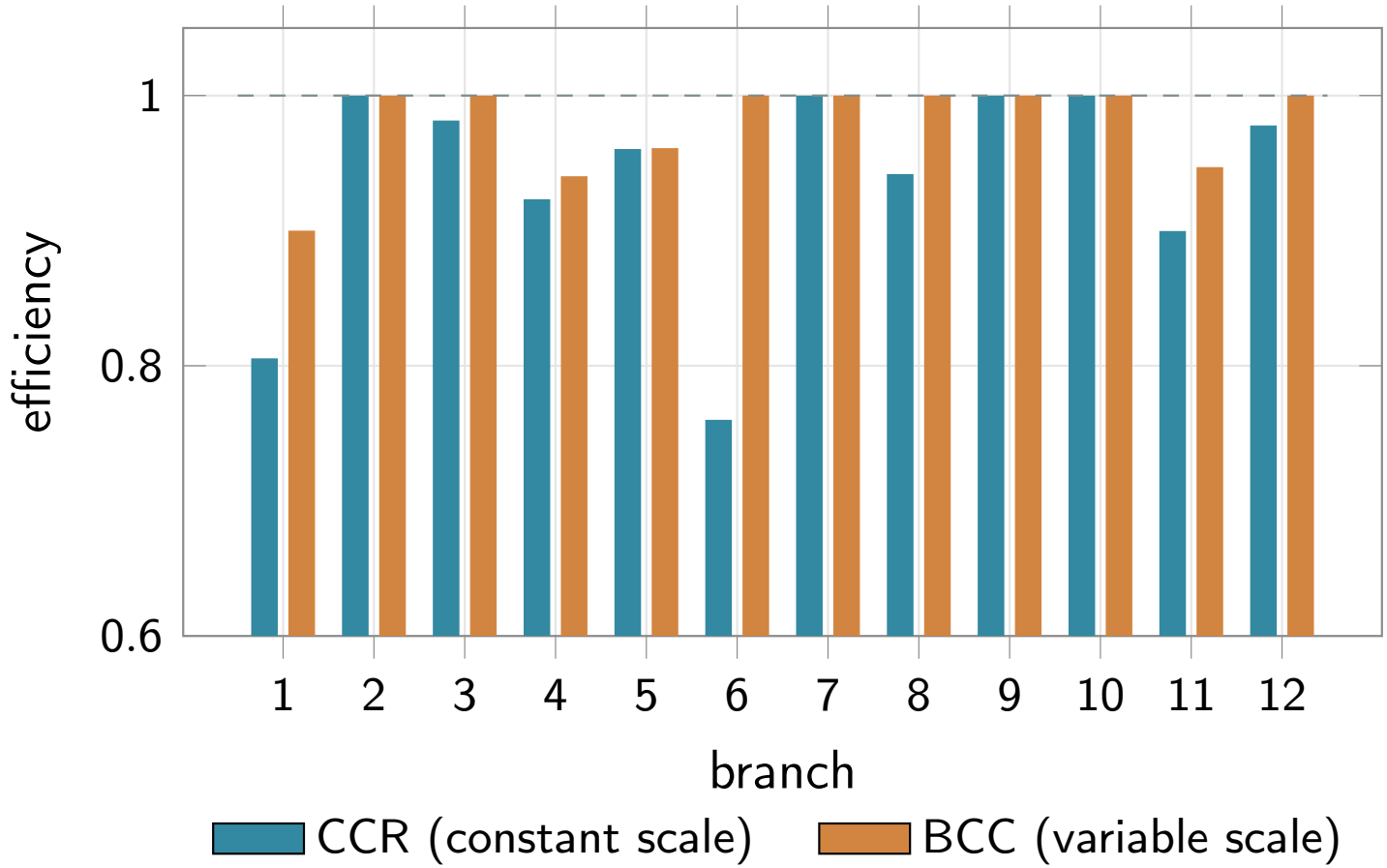
```
def inviluppo(X, Y, o, vrs=False):
    m_, n_ = X.shape           # inputs x DMUs
    p_ = Y.shape[0]            # outputs
    md = gp.Model("dea_envelopment")
    theta = md.addVar(lb=-GRB.INFINITY, name="theta")
    lam = md.addVars(n_, name="lambda")
    c_in = md.addConstrs(
        (theta * X[i, o] - gp.quicksum(lam[j] * X[i, j]
         for j in range(n_)) >= 0 for i in range(m_)), name="input")
    c_out = md.addConstrs(
        (gp.quicksum(lam[j] * Y[r, j] for j in range(n_))
         >= Y[r, o] for r in range(p_)), name="output")
    if vrs:                    # BCC: convex combination
        md.addConstr(lam.sum() == 1, name="convexity")
    md.setObjective(theta, GRB.MINIMIZE)
    md.optimize()
    return md, theta, lam, c_in, c_out
```

Results

branch	CCR	BCC	scale	sum lambda (CCR)	returns
1	0.8055	0.9000	0.8950	0.671	increasing
2	1.0000	1.0000	1.0000	1.000	constant
3	0.9815	1.0000	0.9815	1.917	decreasing
4	0.9233	0.9403	0.9819	1.423	decreasing
5	0.9604	0.9611	0.9993	1.450	decreasing
6	0.7600	1.0000	0.7600	1.924	decreasing
7	1.0000	1.0000	1.0000	1.000	constant
8	0.9419	1.0000	0.9419	1.913	decreasing
9	1.0000	1.0000	1.0000	1.000	constant
10	1.0000	1.0000	1.0000	1.000	constant
11	0.8997	0.9470	0.9501	1.641	decreasing
12	0.9779	1.0000	0.9779	2.679	decreasing

- For every branch $\tilde{\theta}$ (dual) $= \tilde{h}_o$ (primal)

CCR and BCC efficiencies



- CCR-efficient: 2, 7, 9, 10; BCC-efficient also 3, 6, 8, 12

Peers and input targets

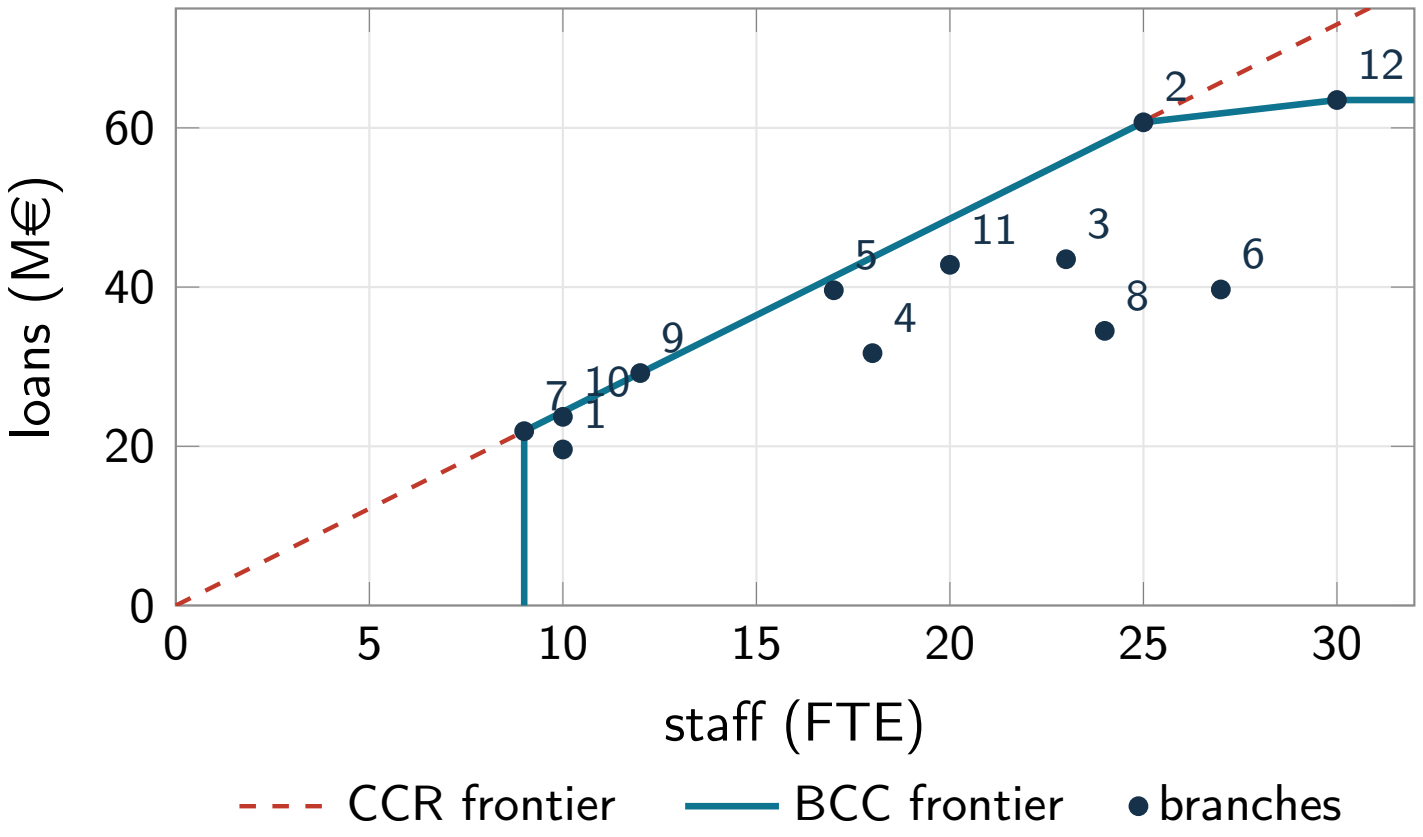
branch	$\tilde{\theta}$	peers (j: $\tilde{\lambda}_j$)	staff	target	costs	target
1	0.8055	9: 0.671	10	8.05	720	463.2
3	0.9815	7: 1.075; 10: 0.842	23	18.10	1130	1109.1
4	0.9233	7: 1.119; 10: 0.304	18	13.11	940	867.9
5	0.9604	7: 0.223; 9: 1.024; 10: 0.203	17	16.33	990	950.8
6	0.7600	7: 1.924	27	17.31	1620	1231.2
8	0.9419	7: 1.913	24	17.22	1300	1224.4
11	0.8997	2: 0.106; 10: 1.535	20	17.99	1020	917.7
12	0.9779	10: 2.679	30	26.79	1370	1339.7

- Targets after the second phase (`dea_target.csv`)

Managerial insight

- Branch 6 is the least efficient (0.76) but it is BCC-efficient: it is **too large**
- Only peer branch 7 taken 1.924 times: decreasing returns; consider splitting it into two offices
- Branch 1 is too small ($\sum \tilde{\lambda}_j = 0.671$) and inefficient also in the BCC (0.90)
- Branch 1 keeps a slack of 116.8 k€ on costs: target 463.2 against 720

Efficient frontiers: staff and loans



Reading the chart

- Reduced one-input, one-output view: the efficiencies differ from the full model
- CCR frontier: ray from the origin with slope 2.4333 M€ per FTE (branch 9)
- BCC frontier: piecewise line through branches 7, 9, 2 and 12
- Below the ray a branch pays technical and scale inefficiency; below the piecewise line only technical
- Branch 12: BCC 1, CCR 0.870 in the view

Summary: Case study and results

- Four CCR-efficient branches, eight BCC-efficient
- Branches 3, 6, 8, 12 have only scale inefficiency
- Branches 7 and 10 are the most cited peers (five times each)

Sensitivity analysis

Introduction: Sensitivity analysis

- Shadow prices of the envelopment form, checked by perturbation
- Stability of the frontier: what happens without a peer

Shadow prices: branch 6

```
branch 6: theta = 0.7600
 loans   : y = 39.7 | Pi = 0.00000 | valid in [-inf, 42.13]
 deposits: y = 73.1 | Pi = 0.01040 | valid in [68.89, inf]
Check: deposits from 73.1 to 74.1
  -> theta from 0.7600 to 0.7704 (change +0.01040)
Primal weight u = (0.00000, 0.01040)
```

- $\geq$ constraints in a minimisation: raising the right-hand side shrinks the region, $\tilde{\theta}$ does not decrease, shadow prices ≥ 0
- The shadow price of deposits is the weight $\tilde{u}_2$ of the primal

Reading the shadow prices

- One more million of deposits raises $\tilde{\theta}$ by 0.0104, for any right-hand side above 68.89
- The shadow price of loans is zero up to 42.13
- The virtual DMU already produces 2.43 million of loans more (the slack of the second phase)

Stability: without branch 7

```
  branch  3: CCR 0.9815 -> 1.0000  <- changes
  branch  4: CCR 0.9233 -> 0.9795  <- changes
  branch  5: CCR 0.9604 -> 0.9791  <- changes
  branch  6: CCR 0.7600 -> 0.9156  <- changes
  branch  8: CCR 0.9419 -> 1.0000  <- changes
Without branch 7: 5 branches change efficiency;
newly efficient: [3, 8]
```

- Efficiency is **relative**: a single DMU on the frontier changes the verdict on many
- Before acting: check the data of the most cited peers and repeat the analysis without them

Summary: Sensitivity analysis

- The shadow prices of the output constraints are the weights u_r
- Checked by perturbation, with their ranges
- The frontier depends on the observed DMUs

Exercises

Introduction: Exercises

- Five exercises to consolidate the concepts of the chapter
- Check, output orientation, weak efficiency, weight restrictions, super-efficiency
- They are solved with the laboratory script

Exercises (1/2)

- **Check.** With the optimal weights of branch 4 compute the ratio of all branches: which ones have ratio 1? How do they relate to the peers?
- **Output orientation.** Write the output-oriented CCR and check $\tilde{\varphi} = 1/\tilde{\theta}$; in the BCC the relation does not hold in general
- **A weakly efficient branch.** Branch 13 equal to branch 7 but with costs 700 k€: $\tilde{\theta}$ and slacks?

Exercises (2/2)

- **Weight restrictions.** Add $u_2 \geq u_1$ to the multiplier form: which branches remain efficient?
- **Super-efficiency.** Remove DMU o from the reference set ($\lambda_o = 0$) to rank the efficient branches: which one is the most robust?

Summary (1/2)

Model:

- Relative efficiency = weighted output/input ratio with the most favourable weights
- Charnes–Cooper: multiplier form (LP)
- Dual: envelopment form, $\tilde{\theta} = \tilde{h}_o$, peers and targets

Scale:

- BCC with $\sum_{j=1}^{n} \lambda_j = 1$; $\mathrm{SE} = \tilde{\theta}^{\mathrm{CCR}} / \tilde{\theta}^{\mathrm{BCC}}$

Summary (2/2)

Managerial message:

- Tell apart those who work badly from those who have the wrong size
- The targets say by how much to reduce each input

Caution:

- Efficiency is relative to the observed DMUs: check the most cited peers