

```

MODEL:
! Data Envelopment Analysis of Decision Maker Efficiency ;
! Keywords: Benchmarking, Data Envelopment Analysis, DEA, Efficiency,
      LINGO, Pareto optimal;
SETS:
DMU: !The decisionmaking units;
      SLK, ! Slack variable on SCORE(k) <= 1;
      SCORE; ! Each decision making unit has a
              score to be computed;
FACTOR: TW;
! There is a set of factors, input & output;
DXF( DMU, FACTOR): F, ! F( I, J) = Jth factor of DMU I;
W; ! Weights used to compute DMU I's score;
ENDSETS

DATA:
!Case01; DMU = BL HW NT OP YK EL; ! The schools;
! Inputs are spending/pupil, % not low income;
! Outputs are Writing score and Science score;
!Case01; NINPUTS = 2; ! The first NINPUTS factors are inputs;
!Case01; FACTOR= COST RICH WRIT SCIN;
! The inputs, the outputs;
!Case01;F =
!Case01;      89.39  64.3    25.2  223 ! BL;
!Case01;      86.25  99      28.2  287 ! HW;
!Case01;      108.13 99.6    29.4  317 ! NT;
!Case01;      106.38 96      26.4  291 ! OP;
!Case01;      62.40  96.2    27.2  295 ! YK;
!Case01;      47.19  79.9    25.5  222; ! EL;
!Case01; WGTMIN = .0005; ! Min weight applied to every factor;
      BIGM = 999999; ! Biggest a weight can be;
      EPSILON = .000001; ! Zero tolerance;

!Case02 WGTMIN = 0.0005; ! Min weight applied to every factor;
!Case02 DMU = F1 F2 F3 F4 F5 F6 F7 F8 F9;
!Case02 FACTOR = X1 X2 Y1;
!Case02 NINPUTS = 2; ! The first NINPUTS factors are inputs;
! The inputs, the outputs;
!Case02 F =
144 360 90
460 600 200
110 660 110
360 300 120
780 130 130
150 240 60
760 266 190
231 140 70
110 665 110;

ENDDATA
!-----;

SUBMODEL DEA:
! The Model;
! IU = DMU we are currently considering;
! Try to make the score of DMU IU as high as possible;
MAX = TSCORE;
      TSCORE = @SUM( FACTOR(J) | J #GT# NINPUTS:
                  F(INOW, J) * TW( J) );

! Sum of inputs(denominator) = 1;
[SUM21] @SUM( FACTOR( J) | J #LE# NINPUTS: F( INOW, J) * TW( J) ) = 1;

! Using DMU IU's weights, no DMU can score better than 1
Note, Numer/Denom <= 1 implies Numer <= Denom;
@FOR( DMU( K) :
[LE1] @SUM( FACTOR( J) | J #GT# NINPUTS: F( K, J) * TW( J) ) +
      SLK(K) = @SUM( FACTOR( J) | J #LE# NINPUTS: F( K, J) * TW( J) )

```

```

);
! The weights, TW( ), must be greater than zero;
@FOR( FACTOR( j): @BND( WGTMIN, TW(j), BIGM));
ENDSUBMODEL

CALC:
@SET( 'TERSEO', 2); !Minimal output;
!Write out a solution report;
@WRITE( @NEWLINE( 2),
32*' ', 'Factor Weight:', @NEWLINE( 1),
'      DMU      Score'
);
@FOR( FACTOR( I): @WRITE( ' ', @FORMAT( FACTOR( I), '8s')));
@WRITE( ' Dominated by some combination of...');
@WRITE( @NEWLINE( 1));

! Solve the DEA model for each DMU D;
@FOR( DMU( D):
INOW = D;
@SOLVE( DEA); ! Solve the model above;
@WRITE( ' ', DMU( D), ' ', @FORMAT( TSCORE, '6.3f'), ' ');
@FOR( FACTOR( I):
@WRITE( @FORMAT( TW(I), '9.5f'), ' ');
);
@IFC( TSCORE #LT# 1 - EPSILON: ! Does D get a score < 1? ;
@FOR( DMU(k) | k #NE# D: ! Yes, find those k that get a score of 1, ;
@IFC( SLK(k) #LE# EPSILON: ! using D's weights. These k dominate D;
@WRITE( ' ', DMU(k));
);
);
);
@WRITE( @NEWLINE( 1));
);
ENDCALC

END

```