

Package ‘pgt’

September 22, 2026

Type Package

Title Data Envelopment Analysis for Pollution-Generating Technologies

Version 0.7.0

Description Nonparametric efficiency analysis for pollution-generating technologies under the materials-balance principle. Implements the weak-G-disposability model of Rodseth (2025) [<doi:10.1007/s11123-025-00768-0>](https://doi.org/10.1007/s11123-025-00768-0) and its factorially determined multi-output representation, the by-production intersection technology of Murty, Russell and Levkoff (2012) [<doi:10.1016/j.jeem.2012.02.005>](https://doi.org/10.1016/j.jeem.2012.02.005), the materials-balance cost model of Coelli, Lauwers and Van Huylenbroeck (2007) [<doi:10.1007/s11123-007-0052-8>](https://doi.org/10.1007/s11123-007-0052-8) and a weak-disposability reference model, with a pre-estimation audit of every materials-balance account, metafrontier decompositions, bad-output shadow prices, marginal abatement cost curves, a cross-axiom comparison harness, a global Malmquist-Luenberger productivity index and subsampling sensitivity intervals. Estimators are solved with 'lpSolveAPI'.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.0.0)

Imports stats, graphics, utils, lpSolveAPI

Suggests testthat (>= 3.0.0), knitr, rmarkdown, frontier

Config/testthat/edition 3

VignetteBuilder knitr

LazyData true

URL <https://github.com/iik1/pgt>

BugReports <https://github.com/iik1/pgt/issues>

NeedsCompilation no

Author Erik Enstad [aut, cre] (ORCID: [<https://orcid.org/0009-0006-1053-3849>](https://orcid.org/0009-0006-1053-3849))

Maintainer Erik Enstad <erik.enstad@nhh.no>
Repository CRAN
Date/Publication 2026-09-22 14:40:02 UTC

Contents

boot_pgt	2
boot_pgt_sensitivity	5
compare_models	6
mac_curve	8
mb_check	10
pgt	11
pgt_decompose	15
pgt_ml	17
pgt_tech	19
pigfarms	22
shadow_prices	23
steeldemo	25
uscoal	26
usfgd	28
Index	30

boot_pgt	<i>Subsampling inference for pollution-generating efficiency scores</i>
----------	---

Description

Constructs subsampling confidence intervals for the environmental efficiency scores of a pgt model and for group means, by re-solving each DMU against random subsamples of the reference set. Subsampling (m out of n, without replacement) is the resampling scheme with the firmest footing for the boundary-estimator, slow-convergence setting of nonparametric frontiers (Kneip, Simar and Wilson 2008; Simar and Wilson 2011).

Usage

```
boot_pgt(  
  tech,  
  model = c("wgd", "wgd_input_fixed", "envelope", "byprod", "mb_cost", "wd"),  
  B = 200,  
  m = NULL,  
  level = 0.95,  
  kappa = NULL,  
  returns = c("vrs", "crs"),  
  peers = c("all", "group"),  
  pollutant = 1L,  
  seed = NULL  
)
```

Arguments

tech	A <code>pgt_tech()</code> object.
model	One of the environmental-efficiency models ("wgd", "wgd_input_fixed", "envelope", "byprod", "mb_cost", "wd"); the directional model "fdmo" is not supported.
B	Number of subsampling replicates.
m	Subsample size. Defaults to $\max(2, \text{round}(L^{0.7}))$, a point in the interior of the admissible range; Simar and Wilson (2011) show that sensitivity to m is the central practical issue, so check the default with <code>boot_pgt_sensitivity()</code> .
level	Confidence level for the intervals.
kappa	Convergence-rate exponent used to rescale the subsample distribution. Defaults to the DEA rate $2/(d+1)$ under VRS and $2/d$ under CRS (Kneip, Simar and Wilson 2008) for the model's effective frontier dimension d : $M+1$ for "wgd" and "envelope", whose frontiers live in the (y, b) space with the inputs free; $N+M+1$ for "wgd_input_fixed" and "wd" (inputs, good and bad outputs); $N+M$ for "mb_cost", whose programmes live in the (x, y) space; and the number of emission-causing inputs plus one for "byprod", whose principal score comes from the residual sub-technology. A heuristic; supply your own exponent to override.
returns, peers, pollutant	Passed to the underlying model, as in <code>pgt()</code> .
seed	Optional integer seed for reproducibility. The caller's RNG state is restored on exit.

Details

The intervals follow the subsampling recipe of Politis and Romano (1994): the law of $L^\kappa(\hat{\theta} - \theta)$ is approximated by the recentred, rate-scaled subsample distribution $m^\kappa(\theta_m^* - \hat{\theta})$, giving

$$[\hat{\theta} - (m/L)^\kappa q_{1-\alpha/2}(\theta_m^* - \hat{\theta}), \hat{\theta} - (m/L)^\kappa q_{\alpha/2}(\theta_m^* - \hat{\theta})].$$

Each DMU is evaluated against the subsample frontier alone: the evaluated unit is not added to the reference set, so frontier units carry genuine resampling variation instead of a degenerate zero-width interval. Replicates in which a DMU's program is infeasible are dropped as NA and counted: `per_dmu$n_ok` reports the number of feasible replicates behind each interval, and a DMU whose count falls below half of B is returned with NA bounds and standard error, with a warning, rather than with an interval resting on the few feasible replicates. Because a subsample frontier lies weakly inside the full-sample frontier, the deviations are non-negative and the interval extends downward from the point estimate, matching the direction of the frontier bias; the point estimate itself typically lies at or above its own upper endpoint, since the interval targets the true score, which the estimate overestimates. The reported `se` is the subsample standard deviation rescaled by $(m/L)^\kappa$.

With `peers = "group"` the subsample is stratified: each group contributes in proportion to its size, with a floor of two units, so groups smaller than the floor are included whole and effectively not resampled.

Group-mean intervals are a descriptive companion, not a supported inference: the replicate statistic averages all group members against a common subsample frontier, so replicate variation reflects

frontier movement only, and the per-DMU rate exponent is reused although means of frontier estimators converge at their own rates (see Kneip, Simar and Wilson 2015 for the aggregate theory). Read them as a stability band around the group mean, not as a confidence interval with nominal coverage.

The intervals remain heuristic. Subsampling theory for data envelopment analysis backs the radial and directional-distance technologies, not the materials-balance-constrained programs implemented here, so kappa borrows the DEA convergence rate as an approximation. Use the subsample-size sensitivity of `boot_pgt_sensitivity()` to check that the intervals are stable in `m` (its implied `kappa_hat` makes the borrowed rate checkable) before relying on them.

Value

An object of class "pgt_boot": a list with `per_dmu` (a data frame of point estimate, lower and upper bound, rescaled subsampling standard error, and `n_ok`, the number of feasible replicates, per DMU), `group_means` (the same for group means, when a group is present), and the call settings.

References

- Kneip, A., Simar, L., & Wilson, P. W. (2008). Asymptotics and consistent bootstraps for DEA estimators in nonparametric frontier models. *Econometric Theory*, 24(6), 1663–1697. doi:10.1017/S0266466608080651
- Kneip, A., Simar, L., & Wilson, P. W. (2015). When bias kills the variance: Central limit theorems for DEA and FDH efficiency scores. *Econometric Theory*, 31(2), 394–422. doi:10.1017/S0266466614000413
- Politis, D. N., & Romano, J. P. (1994). Large sample confidence regions based on subsamples under minimal assumptions. *The Annals of Statistics*, 22(4), 2031–2050. doi:10.1214/aos/1176325770
- Politis, D. N., Romano, J. P., & Wolf, M. (1999). *Subsampling*. Springer, New York. doi:10.1007/9781461215547
- Simar, L., & Wilson, P. W. (2011). Inference by the `m` out of `n` bootstrap in nonparametric frontier models. *Journal of Productivity Analysis*, 36(1), 33–53. doi:10.1007/s112301002004

See Also

`pgt()`, `boot_pgt_sensitivity()`

Examples

```
data(steeldemo)
steel60 <- steeldemo[steeldemo$year == 2021, ]
tech <- pgt_tech(
  x = steel60[, c("coal_coke", "other_fuel", "raw_material", "flux",
                 "capture_energy")],
  y = steel60$production, b = steel60$emissions, a = steel60$captured,
  v = 0.01467, x_abate = "capture_energy",
  group = steel60$route, id = steel60$plant
)

bt <- boot_pgt(tech, model = "wgd", B = 40, seed = 1)
head(bt$per_dmu)
```

boot_pgt_sensitivity *Subsample-size sensitivity of subsampling intervals*

Description

Re-runs `boot_pgt()` over a grid of subsample sizes `m` and reports how the median interval width responds. Stable widths across the grid support the choice of `m`; a strong trend warns that the intervals are driven by the resampling size rather than the data.

Usage

```
boot_pgt_sensitivity(
  tech,
  model = "wgd",
  B = 100,
  m_grid = NULL,
  level = 0.95,
  kappa = NULL,
  returns = "vrs",
  peers = "all",
  pollutant = 1L,
  seed = NULL
)
```

Arguments

`tech` A `pgt_tech()` object.

`model`, `B`, `level`, `kappa`, `returns`, `peers`, `pollutant`, `seed`
 As in `boot_pgt()`.

`m_grid` Integer vector of subsample sizes. Defaults to five sizes spanning $L^{0.5}$ to $L^{0.9}$.

Details

When at least three grid points have positive widths, the attribute "kappa_hat" reports the convergence-rate exponent implied by the log-spread regression of Politis, Romano and Wolf (1999): the rescaled width scales as $m^{\kappa - \kappa_0}$ for true rate κ_0 and working rate κ , so $\hat{\kappa}_0 = \kappa - \text{slope of log width on log } m$. Agreement between `kappa_hat` and the working `kappa` (attribute "kappa_used") supports the borrowed DEA rate; disagreement suggests supplying `kappa_hat` to `boot_pgt()`. The same seed is reset before every grid point, so replicate draws are comparable across `m`.

Value

A data frame with one row per `m`: the size, the median per-DMU interval width, the median over strictly positive widths, and the mean rescaled standard error, with attributes "kappa_used" and (when estimable) "kappa_hat".

See Also[boot_pgt\(\)](#)**Examples**

```

data(steeldemo)
steel60 <- steeldemo[steeldemo$year == 2021, ]
tech <- pgt_tech(
  x = steel60[, c("coal_coke", "other_fuel", "raw_material", "flux",
                 "capture_energy")],
  y = steel60$production, b = steel60$emissions, a = steel60$captured,
  v = 0.01467, x_abate = "capture_energy",
  id = steel60$plant
)

boot_pgt_sensitivity(tech, model = "wgd", B = 25,
                    m_grid = c(15, 25, 40), seed = 1)

```

compare_models

*Compare competing axiom systems on identical data***Description**

Fits several pollution-generating technology models to the same [pgt_tech\(\)](#) object and reports how their environmental efficiency scores and rankings agree. The materials-balance debate turns on which disposability axiom is appropriate for bad outputs (Forsund 2021 and the accompanying Journal of Productivity Analysis symposium); this function puts the competing systems side by side so the sensitivity of conclusions to that choice is visible.

Usage

```

compare_models(
  tech,
  models = c("wgd", "byprod", "mb_cost", "wd"),
  returns = c("vrs", "crs"),
  peers = c("all", "group"),
  pollutant = 1L
)

```

Arguments

tech	A pgt_tech() object.
models	Character vector of models to compare. Defaults to c("wgd", "byprod", "mb_cost", "wd"): the weak-G-disposability (materials balance), by-production, materials-balance cost and weak-disposability systems. "wgd_input_fixed" and "envelope" may be added.

returns	Returns to scale passed to every model: "vrs" (default) or "crs".
peers	Reference set passed to every model: "all" (default) or "group".
pollutant	For a multi-pollutant technology, the pollutant to compare on. Defaults to the first.

Details

Each model reports its principal environmental-efficiency score, normalised so that 1 is efficient: b^*/b for "wgd", "wgd_input_fixed", "envelope", "byprod" and "wd", and the material-inflow ratio $EE = u'x^*/u'x$ for "mb_cost". The directional model "fdmo" stays excluded, since its score is a gross inefficiency on another scale. Because the scores measure different quantities, only the rank-based statistics (the Spearman matrix and the bottom-quartile overlap) are strictly comparable across models; the median column is a per-model summary. Rank agreement is measured by Spearman correlation over the DMUs solved by both members of each pair. Note that "wgd_input_fixed" scores can exceed 1 for DMUs violating another pollutant's materials-balance identity (see [pgt\(\)](#)); such DMUs enter the comparison unflagged.

Value

An object of class "pgt_compare": a list with

scores Data frame of environmental efficiency, one column per model, one row per DMU (with id and group).

spearman Matrix of pairwise Spearman rank correlations.

agreement Data frame with, per model, the number of DMUs solved, the median score, and bottom_q_overlap: the share of the DMUs this model places in its own bottom efficiency quartile that the reference model (the first entry of models) also places in its bottom quartile (top-of-agenda overlap). The bottom quartile includes all DMUs tied at the 25 per cent cutoff, so it can exceed a quarter of the sample and its size can differ across models.

models, returns, peers Call settings.

References

- Forsund, F. R. (2021). Performance measurement and joint production of intended and unintended outputs. *Journal of Productivity Analysis*, 55(3), 157–175. doi:10.1007/s11123021005999
- Murty, S., Russell, R. R., & Levkoff, S. B. (2012). On modeling pollution-generating technologies. *Journal of Environmental Economics and Management*, 64(1), 117–135. doi:10.1016/j.jeem.2012.02.005

See Also

[pgt\(\)](#)

Examples

```
data(steeldemo)
tech <- pgt_tech(
  x = steeldemo[, c("coal_coke", "other_fuel", "raw_material", "flux",
    "capture_energy")],
```

```

y = steeldemo$production,
b = steeldemo$emissions,
a = steeldemo$captured,
v = 0.01467,
x_abate = "capture_energy",
group = steeldemo$route,
id = steeldemo$plant
)
cmp <- compare_models(tech, models = c("wgd", "byprod", "mb_cost"))
cmp

```

mac_curve

Marginal abatement cost curve

Description

Builds a marginal abatement cost (MAC) curve from a [pgt\(\)](#) fit. Along the estimated frontier, reducing the bad output by one unit requires forgoing $1/\eta_l$ units of the good output, where $\eta_l = \partial b^*/\partial y$ is the output-constraint dual. The DMU's marginal abatement cost is therefore p/η_l for an output price p ; with no price supplied the curve is expressed in output units per unit of bad output ($1/\eta_l$). Each DMU's abatement potential is its distance to the frontier, $b_l - b_l^*$. DMUs are ordered by increasing marginal cost and the potential is accumulated.

Usage

```
mac_curve(fit, price = NULL)
```

Arguments

fit	A pgt() fit with model = "wgd", "wgd_input_fixed" or "envelope", the models that report output duals. The fit must have a single intended output; with several, the per-output duals are in shadow_prices() .
price	Optional scalar price of the good output. If supplied, mac is in money per unit of bad output.

Details

The curve therefore combines two margins. It orders DMUs by the shadow price at their frontier projection and plots against it the abatement available from eliminating inefficiency (the gap $b_l - b_l^*$), which the model itself prices at zero output loss; the mac value is the marginal cost of abatement beyond the frontier point. The area under the curve is consequently not a total-cost estimate.

DMUs with $\eta_l \leq 0$ are excluded. A zero output dual arises when the output constraint is slack, i.e. the DMU projects onto the flat segment of the (y, b) frontier, where abatement via output contraction has locally infinite marginal cost. A negative dual (possible under model = "wgd", whose dual nets out the retained content v) means output contraction increases emissions at the margin, so it admits no abatement-cost reading either. Unsolved LPs are excluded as well. The number of excluded DMUs and the summed abatement potential of the solved-but-excluded ones (which is NOT part of

the curve) are attached as attributes "n_excluded" and "excluded_abatement" and reported by print.

A DMU that projects onto a vertex of the piecewise-linear frontier has a range of optimal duals, and the solver reports the dual of whichever optimal basis it terminates in, so the MAC of such a DMU is one point of an interval and can differ across solver versions or platforms. Frontier-interior projections have unique duals.

Value

A data frame of class "pgt_mac", ordered by mac: id, group (if present), mac, abatement ($b - b^*$) and cum_abatement.

References

Fare, R., Grosskopf, S., Lovell, C. A. K., & Yaisawarng, S. (1993). Derivation of shadow prices for undesirable outputs: A distance function approach. *The Review of Economics and Statistics*, 75(2), 374–380. doi:[10.2307/2109448](https://doi.org/10.2307/2109448)

Rodseth, K. L. (2025). On the development of a unified, nonparametric materials balance-based efficiency analysis model and its applications. *Journal of Productivity Analysis*, 64(3), 305–319. doi:[10.1007/s11123025007680](https://doi.org/10.1007/s11123025007680)

See Also

[pgt\(\)](#), [shadow_prices\(\)](#)

Examples

```
data(steeldemo)
tech <- pgt_tech(
  x = steeldemo[, c("coal_coke", "other_fuel", "raw_material", "flux",
                    "capture_energy")],
  y = steeldemo$production,
  b = steeldemo$emissions,
  a = steeldemo$captured,
  v = 0.01467,
  x_abate = "capture_energy",
  group = steeldemo$route,
  id = steeldemo$plant
)
fit <- pgt(tech, model = "wgd")
mac <- mac_curve(fit, price = 550)
plot(mac)
```

mb_check

*Audit the materials-balance identity***Description**

Pre-estimation feasibility audit of the materials-balance condition $u'x_l - v'y_l \geq b_l$ for every DMU (and every pollutant). DMUs that violate the identity carry inconsistent material accounts (more pollutant leaves than enters). A violation makes self-reference infeasible in the input-fixed "wgd_input_fixed" programme of `pgt()`, so that DMU's LP solves only if some peer mix meets every constraint within its materials-balance cap; conversely, infeasibility is confined to DMUs with an exact violation, $\text{gap} < 0$, in at least one pollutant account (pollutant p 's LP can be infeasible because of a violation in a different pollutant q ; DMUs satisfying every account always solve via self-reference). The reduced-form "wgd" programme is always self-feasible, so its scores carry no feasibility screen; the audit remains the data-quality check behind its materials-balance interpretation. Audit before estimation.

Usage

```
mb_check(tech, tol = 1e-08)
```

Arguments

tech	A <code>pgt_tech()</code> object.
tol	Relative tolerance: a DMU-pollutant account is flagged (violated) when its closure gap is below $-\text{tol}$ times its pollutant potential $u'x_l$. The flag is a data-quality screen; LP infeasibility tracks the exact sign of the gap, so the flagged set is a subset of the accounts with $\text{gap} < 0$ (counted separately in attribute "n_negative"). An account with zero pollutant potential always has a negative gap (strictly positive b); <code>rel_gap</code> divides by the potential floored at machine epsilon, so it is reported as a large but finite negative number rather than $-\text{Inf}$, and the account is flagged whenever b exceeds tol times machine epsilon.

Value

A data frame of class "pgt_mb" with one row per DMU (per pollutant when several are present): `id`, `group` (if present), `pollutant` (when several), `potential` ($u'x_l$), `retained` ($v'y_l$), `b`, `gap` ($u'x_l - v'y_l - b_l$), `rel_gap` ($\text{gap} / \text{potential}$, the denominator floored at machine epsilon) and `violated`. When the technology records an abatement output, also `a` and `closure` ($\text{gap} - a_l$): the equality residual $u'x_l - v'y_l - b_l - a_l$, zero when the account closes exactly, which `model = "fdmo"` requires. The number of violations is attached as attribute "n_violations".

See Also

`pgt_tech()`, `pgt()`

Examples

```
data(steeldemo)
tech <- pgt_tech(
  x = steeldemo[, c("coal_coke", "other_fuel", "raw_material", "flux",
                    "capture_energy")],
  y = steeldemo$production,
  b = steeldemo$emissions,
  a = steeldemo$captured,
  v = 0.01467,
  x_abate = "capture_energy",
  group = steeldemo$route,
  id = steeldemo$plant
)
mb <- mb_check(tech)
mb
```

pgt

Estimate a pollution-generating technology model

Description

Fits a nonparametric materials-balance efficiency model by solving one linear program per DMU. The minimising models return the least bad output b_i^* attainable at the DMU's activity level, with environmental efficiency b_i^*/b_i ; the directional model returns the joint good-output expansion and bad-output contraction.

Usage

```
pgt(
  tech,
  model = c("wgd", "wgd_rodseth", "wgd_input_fixed", "envelope", "fdmo", "ddf", "byprod",
            "mb_cost", "wd"),
  returns = c("vrs", "crs"),
  peers = c("all", "group"),
  pollutant = 1L
)
```

Arguments

tech	A <code>pgt_tech()</code> object.
model	Character: "wgd" (alias "wgd_rodseth"), "wgd_input_fixed", "envelope", "fdmo" (alias "ddf"), "mb_cost", "byprod" or "wd". See Details.
returns	Returns to scale: "vrs" (default) or "crs".
peers	Reference set: "all" pools every DMU; "group" restricts each DMU's peers to its own technology group (requires group in <code>pgt_tech()</code>).

pollutant	For a multi-pollutant technology, the pollutant to estimate: a column name or index of <code>b</code> . Defaults to the first pollutant. This selects which bad output the objective acts on. For <code>model = "wgd_input_fixed"</code> the materials-balance caps of every pollutant constrain the projection; the other models use only the selected pollutant's data.
-----------	---

Details

The models fall into two axiom families. The materials-balance models ("`wgd`", "`wgd_input_fixed`", "`envelope`", "`fdmo`", "`mb_cost`") are built on the account $u'x_l - v'y_l \geq b_l$, which enters each programme differently: as a cap on the evaluated DMU's projected emission in "`wgd_input_fixed`", as an equality in "`fdmo`", as a price system in "`mb_cost`", and only as a valuation of retained output content in the objective of "`wgd`" and "`envelope`", whose programmes carry no row from the DMU's own account. The audit of the account is `mb_check()`. The reference models ("`byprod`", "`wd`") implement competing systems for cross-comparison (see `compare_models()`).

"`wgd`" The weak-G-disposability model of Rodseth (2025), Eq. 6 in reduced form. Equation 6 fixes only the intended outputs at the evaluated DMU's levels; the inputs are decision variables. Substituting the weak-G summing-up condition on the disposal slacks therefore collapses the programme to minimising the peer emission envelope plus the retained content of the good output produced beyond the DMU's own levels,

$$b_i^* = \min_{\lambda} \sum_l \lambda_l b_l + v'_i \left(\sum_l \lambda_l y_l - y_i \right),$$

over the output rows $\sum_l \lambda_l y_{ml} \geq y_{mi}$ and the returns-to-scale row, valued at the DMU's own coefficients v_i . Self-reference is always feasible, so every score lies in $(0, 1]$. When the abatement output `a` is observed, the optimal peer mix also delivers the implied uncontrolled emission $z_star = \sum_l \lambda_l z_l$ and abatement $a_star = \sum_l \lambda_l a_l$, the Table 2 quantities of Rodseth (2025). `model = "wgd_rodseth"` is an alias.

"`wgd_input_fixed`" The input-fixed companion benchmark (the model estimated as "`wgd`" before version 0.6.0): minimise the bad output subject to output feasibility, the input rows $\sum_l \lambda_l x_{nl} \leq x_{ni}$, the peer emission envelope, and the evaluated DMU's own materials-balance cap $b \leq u'_i x_i - v'_i y_i$. With several pollutants the caps of the remaining pollutants constrain the peer mix as well, so the projection respects every pollutant's materials balance. DMUs whose data violate a materials-balance identity lose the self-reference guarantee: their LP may be infeasible (returning NA), and every infeasible LP belongs to such a DMU. A DMU violating another pollutant's identity can also score above 1, since its MB-consistent projection may emit more of the selected pollutant than the DMU reports. Audit with `mb_check()` first.

"`envelope`" The "`wgd`" programme without the retained-content term (its $v = 0$ case): the convex lower envelope of the (y, b) scatter. Self-reference is always feasible, so all scores lie in $(0, 1]$.

"`fdmo`" The factorially determined multi-output (directional) representation of Rodseth (2025), Eq. 14, which his Proposition 4 derives from Eq. 13 by fixing the abatement output at each DMU's own level. It jointly maximises the expansion of the good output (θ_y) and the contraction of the bad output (θ_b) along the materials-balance frontier; gross inefficiency is $\theta_y + \theta_b$, with 0 for a frontier DMU. The gross score adds good-output units to bad-output units (direction $(1, 1)$), so it is unit-dependent and comparable only within one unit system, as in Rodseth's Table 3; rescaling `y` or `b` changes the direction's implicit weighting and can change the optimiser,

not merely the reported number. Requires an abatement output a in `pgt_tech()` and a single intended output. The model imposes the materials-balance identity as an exact equality $u'x_l - vy_l = b_l + a_l$: accounts that do not close exactly either make the LP infeasible or shift the closure gap into the scores, and `pgt()` warns when it detects open accounts (absolute closure gap $|u'x_l - vy_l - b_l - a_l|$ exceeding 10^{-6} times the pollutant potential $u'x_l$). `model = "ddf"` is an alias.

"mb_cost" The materials-balance cost model of Coelli, Lauwers and Van Huylenbroeck (2007): environmental efficiency is the ratio of minimal to observed aggregate material inflow $u'x$, decomposed as $EE = TE \times EAE$ into technical and environmental-allocative efficiency (returned as `te` and `eae`). A unit that uses none of the pollutant-bearing inputs has an undefined inflow ratio: its scores are NA with status 2, reported without a solver call.

"byprod" The by-production intersection technology of Murty, Russell and Levkoff (2012): the good output is produced by one sub-technology and the bad output generated by another over the emission-causing inputs. The implemented model is the original Murty-Russell-Levkoff intersection technology without integrated abatement; Hampf (2014) develops the abatement-integrated extension. Returns emission efficiency (b^*/b , reported as `efficiency`), output efficiency (`output_eff`) and `mean_eff`, the arithmetic mean of the two sub-efficiencies: a package summary in the spirit of, but not identical to, the Fare-Grosskopf-Lovell graph measure, and not a measure defined in the source. Set the emission-causing inputs with `polluting` in `pgt_tech()`.

"wd" The weak-disposability model (Kuosmanen 2005 correct VRS formulation; see the exchange settled in Kuosmanen and Podinovski 2009), a reference axiom system that imposes no materials balance. Returns emission efficiency b^*/b .

Several intended outputs are supported by every model except "fdmo": supply `y` as an $L \times M$ matrix in `pgt_tech()`. The output rows then hold all M outputs at the evaluated DMU's levels, and the retained content generalises to v'_iy_i with one coefficient per output and pollutant.

Value

An object of class "pgt": a list with

`results` Data frame with one row per DMU. The good outputs appear as a single column `y` when $M = 1$ and as one named column per output otherwise, with the output duals named accordingly (`dual_output`, or `dual_<name>` per output). For "wgd": `id`, `group` (if present), the output columns, `b`, `b_star`, efficiency (b^*/b), the output duals (total derivatives $\partial b^*/\partial y_{mi} = \mu_m - v_{mi}$, which can be negative when the retained-content coefficient is large), `z_star` and `a_star` (only when `a` is observed) and `status`, the `lp_solve` solver code (0 = solved; 2 = infeasible; 5 = numerically failed). For "wgd_input_fixed", "envelope" and "wd": the same score columns with non-negative output duals (NA for "wd", which reports no duals) and `mb_headroom`, the cap slack $u'_ix_i - v'_iy_i - b_i^*$ (NA except for "wgd_input_fixed"). For "fdmo": `id`, `group` (if present), `y`, `b`, `gross` ($\theta_y + \theta_b$), `good_eff` (θ_y), `bad_eff` (θ_b), `maximal_y` ($y_l + \theta_y$) and `status`.

`weights` List of named vectors of non-zero peer weights per DMU, keyed by `make.unique(id)`.

`model`, `returns`, `peers`, `pollutant` The call settings.

References

- Coelli, T., Lauwers, L., & Van Huylenbroeck, G. (2007). Environmental efficiency measurement and the materials balance condition. *Journal of Productivity Analysis*, 28(1–2), 3–12. doi:10.1007/s1112300700528
- Fare, R., Grosskopf, S., & Lovell, C. A. K. (1985). *The Measurement of Efficiency of Production*. Kluwer-Nijhoff, Boston. doi:10.1007/9789401577212
- Hampf, B. (2014). Separating environmental efficiency into production and abatement efficiency: A nonparametric model with application to US power plants. *Journal of Productivity Analysis*, 41(3), 457–473. doi:10.1007/s1112301303578
- Kuosmanen, T. (2005). Weak disposability in nonparametric production analysis with undesirable outputs. *American Journal of Agricultural Economics*, 87(4), 1077–1082. doi:10.1111/j.1467-8276.2005.00788.x
- Kuosmanen, T., & Podinovski, V. V. (2009). Weak disposability in nonparametric production analysis: Reply to Fare and Grosskopf. *American Journal of Agricultural Economics*, 91(2), 539–545. doi:10.1111/j.14678276.2008.01238.x
- Murty, S., Russell, R. R., & Levkoff, S. B. (2012). On modeling pollution-generating technologies. *Journal of Environmental Economics and Management*, 64(1), 117–135. doi:10.1016/j.jeem.2012.02.005
- Rodseth, K. L. (2025). On the development of a unified, nonparametric materials balance-based efficiency analysis model and its applications. *Journal of Productivity Analysis*, 64(3), 305–319. doi:10.1007/s11123025007680

See Also

`pgt_tech()`, `pgt_decompose()`, `shadow_prices()`, `mac_curve()`

Examples

```
data(steeldemo)
tech <- pgt_tech(
  x = steeldemo[, c("coal_coke", "other_fuel", "raw_material", "flux",
                    "capture_energy")],
  y = steeldemo$production,
  b = steeldemo$emissions,
  a = steeldemo$captured,
  v = 0.01467,
  x_abate = "capture_energy",
  group = steeldemo$route,
  id = steeldemo$plant
)
fit <- pgt(tech, model = "wgd")
summary(fit)
```

pgt_decompose *Decompose environmental efficiency*

Description

Multiplicative decompositions of the environmental efficiency score b^*/b of Rodseth's (2025) weak-G-disposability technology: the five-component source decomposition of Rodseth (2025, Eq. 11), and a metafrontier decomposition across technology groups in the tradition of Battese, Rao and O'Donnell (2004) and O'Donnell, Rao and Battese (2008).

Usage

```
pgt_decompose(
  tech,
  type = c("envelope", "rodseth"),
  returns = c("vrs", "crs"),
  pollutant = 1L
)
```

Arguments

tech	A <code>pgt_tech()</code> object. <code>type = "envelope"</code> requires a non-NULL group.
type	"envelope" (default) or "rodseth". See Details.
returns	Returns to scale: "vrs" (default) or "crs".
pollutant	For a multi-pollutant technology, the pollutant to decompose: a column name or index of <code>b</code> . Defaults to the first.

Details

"envelope" The metafrontier decomposition on the (y, b) lower envelope (the "envelope" model of `pgt()`). Stages differ by peer set only:

$$Total = \frac{b_{all}^*}{b} = WR \times TGR, \quad WR = \frac{b_{group}^*}{b}, \quad TGR = \frac{b_{all}^*}{b_{group}^*}.$$

WR is within-group reallocation efficiency; TGR is the technology-gap ratio (O'Donnell, Rao and Battese 2008). Self-reference is always feasible, so every component lies in $(0, 1]$ with no feasibility screen. Requires a group in `pgt_tech()`.

"rodseth" The five-component source decomposition of Rodseth (2025, Eq. 11), computed from stage programmes of the extended representation (Eq. 9). Stage 1 holds the evaluated DMU's production inputs, pollution-control inputs, coefficient quality and (when observed) abatement output at their observed levels; the later stages free them one at a time, so each stage relaxes the previous one. The components are the stage-to-stage ratios `te_production` (b_1^*/b), quality (b_2^*/b_1^*), `ae_production` (b_3^*/b_2^*), `te_abatement` (b_4^*/b_3^*) and `ae_abatement` (b_5^*/b_4^*), with

$$Total = \frac{b_5^*}{b} = TE_{prod} \times Quality \times AE_{prod} \times TE_{abate} \times AE_{abate},$$

equal to the "wgd" efficiency of `pgt()`. Every stage is self-feasible, so all components lie in $(0, 1]$; a failed solve yields NA components and a warning. A component collapses to 1 when the data cannot separate it: `quality` requires producer-specific coefficients (the stage prices the peer mix at the peers' own u, v instead of the evaluated DMU's, which changes nothing under homogeneous coefficients), `te_abatement` requires an observed abatement output `a`, and `ae_abatement` requires pollution-control input columns marked by `x_abate` in `pgt_tech()`. Groups are not required; when present they are carried into the results for the summaries.

Value

An object of class "pgt_decomp": a list with results (one row per DMU with the stage minima and the multiplicative components), type and returns. The component columns are `WR`, `TGR`, `total` for `type = "envelope"` and `te_production`, `quality`, `ae_production`, `te_abatement`, `ae_abatement`, `total` for `type = "rodseth"`.

References

- Battese, G. E., Rao, D. S. P., & O'Donnell, C. J. (2004). A metafrontier production function for estimation of technical efficiencies and technology gaps for firms operating under different technologies. *Journal of Productivity Analysis*, 21(1), 91–103. doi:10.1023/B:PROD.0000012454.06094.29
- O'Donnell, C. J., Rao, D. S. P., & Battese, G. E. (2008). Metafrontier frameworks for the study of firm-level efficiencies and technology ratios. *Empirical Economics*, 34(2), 231–255. doi:10.1007/s0018100701194
- Rodseth, K. L. (2025). On the development of a unified, nonparametric materials balance-based efficiency analysis model and its applications. *Journal of Productivity Analysis*, 64(3), 305–319. doi:10.1007/s11123025007680

See Also

`pgt()`, `pgt_tech()`

Examples

```
data(steeldemo)
tech <- pgt_tech(
  x = steeldemo[, c("coal_coke", "other_fuel", "raw_material", "flux",
                    "capture_energy")],
  y = steeldemo$production,
  b = steeldemo$emissions,
  a = steeldemo$captured,
  v = 0.01467,
  x_abate = "capture_energy",
  group = steeldemo$route,
  id = steeldemo$plant
)
dec <- pgt_decompose(tech, type = "envelope")
summary(dec)

# Source decomposition with observed abatement (Rodseth 2025)
data(pigfarms)
```

```

tech2 <- pgt_tech(
  x = pigfarms[, c("uncontrolled", "labor", "capital")],
  y = pigfarms$meat,
  b = pigfarms$controlled,
  u = c(1, 0, 0),
  a = pigfarms$abatement,
  id = pigfarms$farm
)
pgt_decompose(tech2, type = "rodseth")

```

pgt_ml

Global Malmquist-Luenberger productivity index

Description

Computes the global Malmquist-Luenberger (GML) productivity index of Oh (2010) for a panel of pollution-generating technologies, decomposed into efficiency change and best-practice (frontier) change. The index is built on the directional distance function of the pollution-generating technology, which expands the good output and contracts the bad output jointly, so productivity change accounts for emissions rather than ignoring them.

Usage

```

pgt_ml(
  tech,
  technology = c("wd", "input_envelope", "envelope"),
  returns = c("vrs", "crs"),
  pollutant = 1L
)

```

Arguments

tech	A <code>pgt_tech()</code> object with a non-NULL period and id identifying the same DMU across periods. The index is defined for a single good output.
technology	Reference technology: "wd" (the default) imposes weak disposability of the bad output, the technology of Oh (2010); "input_envelope" keeps the input constraints and the lower emission envelope; "envelope" frees the inputs, so the frontier becomes the (y, b) envelope. See Details.
returns	Returns to scale: "vrs" (default) or "crs".
pollutant	For a multi-pollutant technology, the pollutant to index. Defaults to the first.

Details

The directional distance to a reference technology R is

$$D_R(x, y, b) = \max\{\beta : (y + \beta g_y, b - \beta g_b) \text{ feasible in } R \text{ given } x\},$$

with direction $g = (y, b)$ (each observation is scaled by its own level). The global reference pools every period into one technology (Oh 2010), which removes the cross-period infeasibility of the adjacent-period Malmquist-Luenberger index (Aparicio, Pastor and Zofio 2013) and makes the index circular. For a DMU observed in periods t and $t + 1$,

$$GML = \frac{1 + D_G(t)}{1 + D_G(t+1)} = \underbrace{\frac{1 + D_t(t)}{1 + D_{t+1}(t+1)}}_{EC} \times \underbrace{GML/EC}_{BPC},$$

where D_G is the distance to the global frontier and D_t the distance to period t 's own frontier. $GML > 1$ is productivity growth; EC is catch-up to the frontier and BPC is frontier movement.

Three reference technologies are available. "wd" (the default) imposes weak disposability of the bad output (the Kuosmanen 2005 VRS form, with the intensity weights split into an active and an abatement part): this is the technology under which Chung, Fare and Grosskopf (1997) and Oh (2010) define the (global) Malmquist-Luenberger index, so technology = "wd" is the comparator defined by Oh (2010). "input_envelope" and "envelope" instead treat the bad output as reducible down to the peer emission envelope with no proportional output sacrifice, with ("input_envelope") and without ("envelope") the input rows; the input-keeping variant mirrors the input-fixed "wgd_input_fixed" shape of `pgt()` rather than the input-free Eq. 6 programme. Under constant returns the free-disposal technologies contain the weak-disposability set, so their distances are weakly larger than under "wd"; under variable returns the sets are not nested in general, because the Kuosmanen form lets the active intensity weights sum below one.

This estimator is experimental: the global Malmquist-Luenberger index under a materials-balance technology is not yet settled in the literature. Under "input_envelope" and "envelope" the index does not re-impose the per-DMU materials-balance cap on the projected point, and the bad output is freely disposable toward the envelope rather than weakly disposable, so these variants are exploratory companions to the "wd" index, not implementations of Oh (2010).

Value

An object of class "pgt_ml": a list with results, a data frame with one row per DMU per consecutive period transition (id, from, to, gml, ec, bpc), and the call settings. Transitions are formed only for DMUs present in both adjacent periods.

References

- Chambers, R. G., Chung, Y., & Fare, R. (1996). Benefit and distance functions. *Journal of Economic Theory*, 70(2), 407–419. doi:10.1006/jeth.1996.0096
- Chung, Y. H., Fare, R., & Grosskopf, S. (1997). Productivity and undesirable outputs: A directional distance function approach. *Journal of Environmental Management*, 51(3), 229–240. doi:10.1006/jema.1997.0146
- Oh, D.-h. (2010). A global Malmquist-Luenberger productivity index. *Journal of Productivity Analysis*, 34(3), 183–197. doi:10.1007/s111230100178y
- Aparicio, J., Pastor, J. T., & Zofio, J. L. (2013). On the inconsistency of the Malmquist-Luenberger index. *European Journal of Operational Research*, 229(3), 738–742. doi:10.1016/j.ejor.2013.03.031

See Also

`pgt()`, `pgt_tech()`

Examples

```
# Two-period panel: period 2's frontier emits less for the same output.
set.seed(1)
n <- 12
d1 <- data.frame(id = 1:n, y = runif(n, 5, 10), x = runif(n, 8, 12),
                 b = runif(n, 2, 6))
d2 <- data.frame(id = 1:n, y = d1$y * 1.05, x = d1$x, b = d1$b * 0.85)
d <- rbind(cbind(d1, period = 1), cbind(d2, period = 2))
tech <- pgt_tech(x = d[, "x", drop = FALSE], y = d$y, b = d$b,
               period = d$period, id = d$id)
m1 <- pgt_ml(tech) # weak-disposability reference, the Oh (2010) form
summary(m1)
```

pgt_tech

Construct a pollution-generating technology

Description

Bundles the data of a pollution-generating technology, one or more good outputs, one or more bad outputs, and material inputs, together with the material flow coefficients that tie them to the materials-balance identity. The returned object is the input to [pgt\(\)](#), [pgt_decompose\(\)](#) and [mb_check\(\)](#).

Usage

```
pgt_tech(
  x,
  y,
  b,
  u = NULL,
  v = 0,
  a = NULL,
  x_abate = NULL,
  polluting = NULL,
  group = NULL,
  period = NULL,
  id = NULL
)
```

Arguments

- | | |
|---|--|
| x | Numeric matrix or data frame of material inputs, one row per decision-making unit (DMU), one column per input. |
| y | Good outputs, strictly positive: a numeric vector for a single intended output, or an $L \times M$ matrix or data frame with one named column per output. |
| b | Bad outputs (controlled emissions), strictly positive: a numeric vector for a single pollutant, or an $L \times P$ matrix or data frame with one column per pollutant. |

<code>u</code>	Material flow coefficients of the inputs. A length- N vector (one coefficient per input column, shared by all DMUs), an $L \times N$ matrix (DMU-specific coefficients), or, with several pollutants, a named list with one such element per pollutant. Defaults to 1 for every input and pollutant.
<code>v</code>	Material flow coefficients of the good outputs (pollutant embodied in the products). A scalar, a length- M vector (one coefficient per output), a length- L vector (single output, DMU-specific), or an $L \times M$ matrix; with several pollutants, a named list with one such element per pollutant, an $M \times P$ matrix, an $L \times M \times P$ array, or the single-output shapes of earlier versions (length- P vector, $L \times P$ matrix). Defaults to 0.
<code>a</code>	Optional abatement output (pollutant removed by end-of-pipe control), non-negative: a numeric vector for a single pollutant or an $L \times P$ matrix. Required by the models that treat pollution control explicitly; see <code>pgt()</code> .
<code>x_abate</code>	Optional marker of the pollution-control input columns of <code>x</code> (Rodseth 2025, Eq. 7): a character vector of column names or an integer vector of column positions. The remaining columns are production inputs. Used by the Rodseth decomposition of <code>pgt_decompose()</code> , whose final stage frees the pollution-control inputs to separate abatement-input allocative efficiency; the <code>pgt()</code> estimators treat all inputs alike.
<code>polluting</code>	Optional marker of the emission-generating input columns of <code>x</code> (the residual-generating sub-technology of the by-production model; Murty, Russell and Levkoff 2012): a character vector of column names or an integer vector of column positions. Defaults, when the by-production model is fitted, to the inputs with a positive material flow coefficient.
<code>group</code>	Optional factor (or vector coercible to factor) assigning each DMU to a technology group, for example a production route. Required by <code>pgt_decompose()</code> and by group-referenced estimation.
<code>period</code>	Optional factor (or vector coercible to factor) giving the time period of each row, for panel data. Required by <code>pgt_ml()</code> . The pair (<code>id</code> , <code>period</code>) identifies an observation.
<code>id</code>	Optional vector of DMU identifiers. Defaults to the row names of <code>x</code> , or a running index.

Details

The materials-balance principle requires that the pollutant content of the inputs is either embodied in the good output, removed by end-of-pipe abatement, or emitted:

$$u'x_l - v'y_l \geq b_l,$$

where u holds the material flow coefficients of the inputs (for example tonnes of CO2 potential per unit of input) and v the coefficients of the good outputs (pollutant retained in the products). Data already expressed in pollutant-potential units use the default $u = 1$ for every input. When the abatement output a is observed, the identity closes as an equality, $u'x_l - v'y_l = b_l + a_l$. Use `mb_check()` to audit the identity before estimation.

Material flow coefficients may vary across DMUs (heterogeneous input or output quality; Eder 2022, Rodseth 2025). Supply u as an $L \times N$ matrix and v as a length- L vector (single output)

or $L \times M$ matrix to attach DMU-specific coefficients. With homogeneous coefficients a length- N vector u and scalar v suffice, and all estimators reduce to the homogeneous-quality case.

With several intended outputs, supply y as an $L \times M$ matrix (one named column per output). Every estimator supports $M > 1$ except `model = "fdmo"` in `pgt()` and `pgt_ml()`; the retained content of pollutant p generalises to $v'_p y_l$ with one coefficient per output.

With several pollutants, supply b as an $L \times P$ matrix (one named column per pollutant), u as a named list with one element per pollutant (each a length- N vector or $L \times N$ matrix), and v as a named list with one element per pollutant (each a scalar, length- M vector or $L \times M$ matrix), an $M \times P$ matrix, or an $L \times M \times P$ array; the single-output shapes of earlier versions (length- P vector, $L \times P$ matrix) remain accepted. Each pollutant carries its own materials-balance identity.

Value

An object of class "pgt_tech": a list with elements x ($L \times N$ matrix), y ($L \times M$ matrix), b ($L \times P$ matrix), u ($L \times N \times P$ array), v ($L \times M \times P$ array), a ($L \times P$ matrix or NULL), x_abate (integer vector of pollution-control input columns, possibly empty), `group`, `id`, `L`, `N`, `M`, `P`, `outputs` and `pollutants` (character vectors of output and pollutant names). Single-output, single-pollutant input is stored in the same canonical shapes with $M = P = 1$.

References

- Rodseth, K. L. (2025). On the development of a unified, nonparametric materials balance-based efficiency analysis model and its applications. *Journal of Productivity Analysis*, 64(3), 305–319. doi:10.1007/s11123025007680
- Coelli, T., Lauwers, L., & Van Huylenbroeck, G. (2007). Environmental efficiency measurement and the materials balance condition. *Journal of Productivity Analysis*, 28(1–2), 3–12. doi:10.1007/s1112300700528
- Eder, A. (2022). Environmental efficiency measurement when producers control pollutants under heterogeneous conditions: a generalization of the materials balance approach. *Journal of Productivity Analysis*, 57(2), 157–176. doi:10.1007/s1112302100623y

See Also

`mb_check()`, `pgt()`, `pgt_decompose()`

Examples

```
data(steeldemo)
tech <- pgt_tech(
  x = steeldemo[, c("coal_coke", "other_fuel", "raw_material", "flux",
                    "capture_energy")],
  y = steeldemo$production,
  b = steeldemo$emissions,
  a = steeldemo$captured,
  v = 0.01467,
  x_abate = "capture_energy",
  group = steeldemo$route,
  id = steeldemo$plant
)
```

```

tech

# Explicit abatement: Rodseth (2025) pig-finishing example, with the
# uncontrolled-emission aggregate as the material carrier
data(pigfarms)
tech2 <- pgt_tech(
  x = pigfarms[, c("uncontrolled", "labor", "capital")],
  y = pigfarms$meat,
  b = pigfarms$controlled,
  u = c(1, 0, 0),
  a = pigfarms$abatement,
  id = pigfarms$farm
)

```

pigfarms

Pig-finishing farms from Rodseth (2025), Table 1

Description

The synthetic manure-transport example printed as Table 1 of Rodseth (2025): five pig-finishing farms with piglets and feed as material inputs, labour and capital as non-material inputs, saleable meat as the good output, and nitrogen emissions as the bad output, extended with end-of-pipe abatement. The example descends from Coelli, Lauwers and Van Huylbroeck (2007) via Rodseth (2016). The columns satisfy $\text{uncontrolled} - \text{controlled} = \text{abatement}$ exactly.

Usage

pigfarms

Format

A data frame with 5 rows and 9 columns:

farm Farm identifier, "A" to "E".

feed Feed input.

piglet Piglet input.

labor Labour input.

capital Capital input.

meat Saleable meat (good output).

controlled Controlled nitrogen emissions (bad output).

uncontrolled Uncontrolled (ex ante) nitrogen emissions.

abatement End-of-pipe abatement, $\text{uncontrolled} - \text{controlled}$.

Details

The paper does not print the nitrogen coefficients of feed and piglets, so the technology is constructed in pollutant-potential units: the uncontrolled-emission aggregate carries the material with $u = 1$ on that column and $v = 0$, which reproduces the paper's materials-balance cap exactly. With this mapping, `pgt(tech, model = "wgd")` reproduces the minimal controlled emissions of the paper's Table 2 (16, 16, 16, 20, 16), and `mb_check()`'s closure gap recovers the abatement column.

Source

Rodseth, K. L. (2025). On the development of a unified, nonparametric materials balance-based efficiency analysis model and its applications. *Journal of Productivity Analysis*, 64(3), 305–319, Table 1. doi:10.1007/s11123025007680

References

Rodseth, K. L. (2016). Environmental efficiency measurement and the materials balance condition reconsidered. *European Journal of Operational Research*, 250(1), 342–346. doi:10.1016/j.ejor.2015.10.061

Examples

```
data(pigfarms)
tech <- pgt_tech(
  x = pigfarms[, c("feed", "piglet", "labor", "capital", "uncontrolled")],
  y = pigfarms$meat,
  b = pigfarms$controlled,
  u = c(0, 0, 0, 0, 1),
  v = 0,
  id = pigfarms$farm
)
# closure gap recovers the paper's abatement column
mb_check(tech)$gap
# minimal controlled emissions of Rodseth (2025), Table 2
pgt(tech, model = "wgd")$results$b_star
```

shadow_prices

Extract shadow values from a pgt fit

Description

Returns the informative constraint diagnostics of the per-DMU linear programs. `dual_output` (one column per intended output when several are present) is the shadow value of the output constraint: the marginal bad-output content of the good output along the frontier (for CO₂, the marginal emission intensity in tonnes of CO₂ per tonne of output). For `model = "wgd"` the reported dual is the total derivative $\partial b^* / \partial y_m = \mu_m - v_m$, which can be negative when the retained-content coefficient v is large (producing more output binds more pollutant into the product); for `"wgd_input_fixed"` and `"envelope"` the dual is non-negative. `mb_headroom` (`"wgd_input_fixed"` only) is the slack $u'x_l - v'y_l - b_l^*$ between the projection and the DMU's materials-balance ceiling.

Usage

```
shadow_prices(fit)
```

Arguments

`fit` A `pgt()` fit with `model = "wgd", "wgd_input_fixed" or "envelope"`, the models that report output duals.

Details

The dual of the peer emission envelope $\sum_l \lambda_l b_l \leq b$ in the "wgd_input_fixed" programme is not reported: it equals -1 whenever the LP solves (dual feasibility on the emission variable, with the materials-balance row slack at any optimum), so it carries no cross-DMU information. In the boundary case $b^* = u'x_l - v'y_l$ exactly, the split of this dual between the envelope and cap rows is basis-dependent. A materials-balance cap that would bind manifests as an infeasible LP, not as a capped projection.

Values are reported in the units of the linear program (quantities, not money). To monetise the output dual, combine it with an output price via `mac_curve()`.

Value

A data frame with one row per DMU: `id`, `group` (if present), `b`, `b_star`, the output-dual columns and `mb_headroom` where the fit reports it ("wgd_input_fixed"; the column is NA for "envelope" and absent for "wgd").

References

Fare, R., Grosskopf, S., Lovell, C. A. K., & Yaisawarng, S. (1993). Derivation of shadow prices for undesirable outputs: A distance function approach. *The Review of Economics and Statistics*, 75(2), 374–380. doi:10.2307/2109448

Rodseth, K. L. (2025). On the development of a unified, nonparametric materials balance-based efficiency analysis model and its applications. *Journal of Productivity Analysis*, 64(3), 305–319. doi:10.1007/s11123025007680

See Also

`pgt()`, `mac_curve()`

Examples

```
data(steeldemo)
tech <- pgt_tech(
  x = steeldemo[, c("coal_coke", "other_fuel", "raw_material", "flux",
    "capture_energy")],
  y = steeldemo$production,
  b = steeldemo$emissions,
  a = steeldemo$captured,
  v = 0.01467,
  x_abate = "capture_energy",
```

```

    group = steeldemo$route,
    id = steeldemo$plant
  )
  fit <- pgt(tech, model = "wgd")
  head(shadow_prices(fit))

```

steeldemo

Synthetic steel plant panel

Description

A simulated panel of steel plants loosely calibrated to the structure of global plant-level data: two production routes with different carbon intensities, four material inputs expressed in CO₂-potential units (so the material flow coefficients are $u = 1$), crude steel output, Scope 1 CO₂ emissions, and an observed abatement output with its pollution-control input. The materials-balance identity closes exactly in every row, $u'x - vy = b + a$, with $v = 0.01467$ (carbon retained in the product), b the emissions and a the captured CO₂; the five input columns, `capture_energy` included, make up x . The synthetic generator (see `data-raw/steeldemo.R`) assumes roughly 0.4 per cent retained carbon by mass, converted to CO₂ units: $0.004 \times 44/12 = 0.01467$ tonnes of CO₂ per tonne of steel. Abatement is assigned at plant level: eight integrated plants adopt post-combustion carbon capture and storage (CCS) in 2022 or 2023 and capture 60 to 90 per cent of their gross emissions from the adoption year on, six plants of either route capture 5 to 15 per cent for utilisation (CCU), and ten plants carbonate slag (mineralisation, 1 to 3 per cent). Capturing costs energy: `capture_energy` is the CO₂ potential of that energy (0.10 to 0.20 tonnes per tonne captured for CCS and CCU, 0.01 to 0.03 for mineralisation), and its CO₂ is part of the gross emissions the capture process treats. The data are synthetic; they mimic magnitudes, not any real plant. Versions before 0.7.0 shipped this panel without abatement and with a small unexplained closure gap in every row; the values of all columns changed with the regeneration.

Usage

```
steeldemo
```

Format

A data frame with 180 rows (60 plants over 3 years) and 12 columns:

plant Plant identifier.

year Observation year.

route Production route: "Integrated" or "Minimill".

production Crude steel output (tonnes).

coal_coke Coal and coke input (tonnes CO₂ potential).

other_fuel Other fuel input (tonnes CO₂ potential).

raw_material Raw material input (tonnes CO₂ potential).

flux Flux and alloy input (tonnes CO₂ potential).

emissions Scope 1 CO2 emissions (tonnes).

captured CO2 leaving as an abatement output (tonnes): the a slot of `pgt_tech`; zero where no abatement is active.

capture_energy CO2 potential of the energy consumed by the capture process (tonnes): a pollution-control input, the natural `x_abate` marker.

abatement_tech Abatement active in the plant-year: "none", "CCS", "CCU" or "mineralisation".

Source

Simulated; see `data-raw/steeldemo.R` in the package sources.

Examples

```
data(steeldemo)
tech <- pgt_tech(
  x = steeldemo[, c("coal_coke", "other_fuel", "raw_material", "flux",
    "capture_energy")],
  y = steeldemo$production, b = steeldemo$emissions,
  a = steeldemo$captured, v = 0.01467, x_abate = "capture_energy",
  group = steeldemo$route, id = steeldemo$plant)
attr(mb_check(tech), "n_violations")
table(steeldemo$abatement_tech, steeldemo$year)
```

uscoal

US coal-fired power plants with a measured SO2 account, 2022

Description

A cross-section of 212 US coal-fired power plants for 2022 with a measured sulfur-dioxide materials-balance account, built from two public sources matched on the ORIS plant code: EIA-923 (fuel consumption, coal receipts with sulfur content, and Schedule 8C air-emissions-control information) and EPA eGRID2022 (measured SO2 emissions, net generation, nameplate capacity). The SO2 potential of a plant's coal is $2 * \text{sulfur} / 100$ short tons per short ton of coal (molar mass ratio, full conversion), giving a producer-specific material flow coefficient; electricity retains no sulfur, so $v = 0$. Unlike the synthetic `steeldemo` panel, the two sides of the account here come from separate data collections, and it does not close by construction: 8 of the 212 accounts violate the materials-balance condition (measured SO2 exceeds the sulfur-implied potential, a data inconsistency the audit exists to catch), and the remaining accounts are open by the sulfur retained in ash and, at the 180 plants with flue-gas desulphurisation (FGD), by the sulfur removed by the scrubbers. The FGD sorbent quantity is a dedicated pollution-control input for `x_abate`. Note that sorbent is 0 both at plants without SO2 controls and at the 36 FGD plants that list a control but report no sorbent quantity, so a zero does not by itself mark an unscrubbed plant; `fgd` is the scrubbing indicator. `coal` is the EIA-923 total fuel consumption, which at combined-heat-and-power plants includes coal burned for useful thermal output, while `gen` is net electric generation only and eGRID allocates a CHP plant's SO2 to electricity with a factor built from EIA heat-input data; at such plants the sulfur-implied potential is overstated relative to the emissions side, and their accounts lean toward looser closure (fewer violations, larger relative gaps).

Usage

uscoal

Format

A data frame with 212 rows and 11 columns:

plant ORIS plant code (character).

name Plant name.

state State abbreviation.

fgd TRUE if the plant reports at least one SO2 control in EIA-923 Schedule 8C.

coal Coal consumption (short tons).

other_heat Non-coal fuel consumption (MMBtu).

capacity Nameplate capacity (MW).

sorbent FGD sorbent quantity (short tons; 0 both without SO2 controls and where an FGD plant reports no quantity).

gen Net generation (MWh).

so2 Annual SO2 emissions from eGRID (short tons; see Details for measurement provenance).

sulfur Receipt-tonnage-weighted sulfur content of the coal (percent by weight).

Details

eGRID's SO2 value is a monitored stack measurement only for units reporting to the EPA Clean Air Markets Division's (CAMD) Power Sector Emissions Data, which generally covers fossil units serving generators above 25 MW; for other units eGRID estimates SO2 from EIA heat input and fuel-specific emission factors, that is, from the same EIA-923 fuel data that build the potential (see the eGRID2022 Technical Guide). No filter on this measurement provenance is applied, so the two sides of the account are fully independent only at CAMD-monitored plants.

Construction conventions, filters and download URLs are documented in `data-raw/uscoal.R` in the package sources (sources accessed 2026-08-04). Plants burning only stockpiled coal drop out because their 2022 receipts carry no sulfur observation.

Source

US Energy Information Administration, Form EIA-923 (2022 final revision), <https://www.eia.gov/electricity/data/eia923/>; US Environmental Protection Agency, eGRID2022, <https://www.epa.gov/egrid>. Both public; matched and filtered as documented in `data-raw/uscoal.R`.

Examples

```
data(uscoal)
tech <- pgt_tech(
  x = uscoal[, c("coal", "other_heat", "capacity", "sorbent")],
  y = uscoal$gen,
  b = uscoal$so2,
  u = cbind(coal = 2 * uscoal$sulfur / 100, other_heat = 0,
```

```

        capacity = 0, sorbent = 0),
  v = 0,
  x_abate = "sorbent",
  group = factor(ifelse(uscoal$fgd, "FGD", "No FGD")),
  id = uscoal$plant
)
mb <- mb_check(tech)
attr(mb, "n_violations")

```

usfgd

US coal-fired power plants with flue-gas desulphurisation, 2023

Description

A cross-section of 154 US coal-fired power plants that operated flue-gas desulphurisation (FGD) in 2023, with a sulfur-dioxide materials-balance account whose abatement side is observed through the SO₂ removed by the scrubbers and whose control inputs, the FGD sorbent and the FGD electricity, are reported. The account closes by construction, $2 \cdot \text{sulfur} / 100 \cdot \text{coal} = \text{so2} + \text{so2_removed}$, so the directional model (`model = "fmo"`) and the abatement stages of the five-component decomposition can be run on real data.

Usage

usfgd

Format

A data frame with 154 rows and 17 columns:

plant ORIS plant code (character).

name Plant name (CAMD).

state State abbreviation.

coal Coal consumption (short tons).

sulfur Receipt-tonnage-weighted sulfur content of the coal (percent by weight).

other_heat Non-coal fuel consumption (MMBtu).

capacity Nameplate capacity of the generators associated with the plant's monitored units (MW).

gen Net generation (MWh).

heat_input Heat input of the monitored units (MMBtu).

coal_units Number of monitored units whose primary fuel is coal.

scrubbed_units Number of those units listing SO₂ control equipment.

so2 Measured SO₂ emissions (short tons).

so2_removed Derived SO₂ abatement, potential minus so2 (short tons): the a slot of [pgt_tech](#).

efficiency Reported SO₂ removal efficiency at the annual operating factor, mean over the plant's operating SO₂ control units (fraction; NA for 14 plants).

sorbent FGD sorbent quantity (short tons).

fgd_mwh FGD electricity consumption (MWh).

n_fgd Number of operating SO₂ control units in Schedule 8C.

Details

The abatement is derived, not metered: `so2_removed` is the sulfur-implied SO₂ potential minus the SO₂ measured at the stack, and therefore also absorbs sulfur retained in ash, sulfur emitted by unscrubbed units at partially scrubbed plants, and any error in the receipt-weighted sulfur share. The reported removal efficiency from EIA-923 Schedule 8C is an independent check (EIA instructs respondents to base it on monitoring data where available and on design specifications otherwise): the derived removal share $\text{so2_removed} / (\text{so2} + \text{so2_removed})$ has a median of 0.953 against a median reported efficiency of 0.950, the efficiency-implied removal lies within 1.2 per cent of the derived one for half the plants, and the two rank with a Spearman correlation of 0.82 across the 140 plants reporting an efficiency. The largest departures are low-sulfur plants, where a small error in the sulfur share moves the derived share a lot, and the 12 plants where `scrubbed_units < coal_units`. Emissions come from EPA's Clean Air Markets Division continuous emission monitoring, summed over all monitored units of the plant; fuel, sulfur, generation and FGD data from EIA-923. Plants enter when at least one monitored coal unit lists SO₂ controls, at least one Schedule 8C SO₂ control unit is operating, the quantities are positive and the derived abatement is positive (one plant was dropped on the last criterion). Coal is on the total-fuel basis, so at combined-heat-and-power plants the potential includes coal burned for thermal output while gen is electric only; non-coal fuels carry no sulfur in the account; sorbent and `fgd_mwh` are zero where a plant reports no quantity (31 and 54 plants). The construction, filters and download provenance are in `data-raw/usfgd.R`.

Source

US Energy Information Administration, Form EIA-923 (2023 final revision), and US Environmental Protection Agency, Clean Air Markets Division Power Sector Emissions Data (annual apportioned emissions and facility attributes, 2023); see `data-raw/usfgd.R` in the package sources.

Examples

```
data(usfgd)
tech <- pgt_tech(
  x = usfgd[, c("coal", "other_heat", "capacity", "sorbent", "fgd_mwh")],
  y = usfgd$gen, b = usfgd$so2, a = usfgd$so2_removed,
  u = cbind(coal = 2 * usfgd$sulfur / 100, other_heat = 0,
            capacity = 0, sorbent = 0, fgd_mwh = 0),
  v = 0, x_abate = c("sorbent", "fgd_mwh"), id = usfgd$plant)
attr(mb_check(tech), "n_violations")
# derived removal share against the reported efficiency
share <- usfgd$so2_removed / (usfgd$so2 + usfgd$so2_removed)
cor(share, usfgd$efficiency, method = "spearman", use = "complete.obs")
```

Index

* datasets

- pigfarms, [22](#)
- steeldemo, [25](#)
- uscoal, [26](#)
- usfgd, [28](#)

boot_pgt, [2](#)
boot_pgt(), [5](#), [6](#)
boot_pgt_sensitivity, [5](#)
boot_pgt_sensitivity(), [3](#), [4](#)

compare_models, [6](#)
compare_models(), [12](#)

mac_curve, [8](#)
mac_curve(), [14](#), [24](#)
mb_check, [10](#)
mb_check(), [12](#), [19–21](#), [23](#)

pgt, [11](#)
pgt(), [3](#), [4](#), [7–10](#), [15](#), [16](#), [18–21](#), [24](#)
pgt_decompose, [15](#)
pgt_decompose(), [14](#), [19–21](#)
pgt_ml, [17](#)
pgt_ml(), [20](#), [21](#)
pgt_tech, [19](#), [26](#), [28](#)
pgt_tech(), [3](#), [5](#), [6](#), [10](#), [11](#), [13–18](#)
pigfarms, [22](#)

shadow_prices, [23](#)
shadow_prices(), [8](#), [9](#), [14](#)
steeldemo, [25](#)

uscoal, [26](#)
usfgd, [28](#)