Package ‘dfvad’

March 5, 2020

Type Package

Title Diewert and Fox’s Method of Value Added Growth Decomposition

Version 0.3.0

Author Shipei Zeng

Maintainer Shipei Zeng <shipei.zeng@unsw.edu.au>


Depends R (>= 2.10)

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 7.0.0

Suggests testthat, knitr, rmarkdown

VignetteBuilder knitr

Imports stats

URL https://github.com/shipei-zeng/dfvad

BugReports https://github.com/shipei-zeng/dfvad/issues

NeedsCompilation no

Repository CRAN

Date/Publication 2020-03-05 10:50:05 UTC
R topics documented:

- mining
- roll_div
- roll_prod
- sector
- t_weight
- value_decom

Index

---

mining	Sample Data for Value Added Decomposition

Description

Mining inputs and outputs adopted to demonstrate the decomposition of value added growth.

Usage

mining

Format

A data frame with the following columns:

- **year**: A time period column.
- **p2**: Output prices.
- **w2**: Wages of labour inputs.
- **u2**: Prices of capital services.
- **y2**: Output quantities.
- **h2**: Labour input quantities.
- **x2**: Capital services quantities.

References

**roll_div**

*Converting Level Values to Growth Values*

**Description**

`roll_div()` converts level values to growth values for a vector.

**Usage**

`roll_div(x)`

**Arguments**

- `x`: A vector with level values.

**Value**

A vector of growth values.

**Examples**

```r
table2 <- value_decom(c("h2","x2"), c("w2","u2"), "y2", "p2", "year", mining)[[2]]
roll_div(table2[, "TFP"])
```

---

**roll_prod**

*Converting Growth Values to Level Values*

**Description**

`roll_prod()` converts growth values to level values for a vector.

**Usage**

`roll_prod(x)`

**Arguments**

- `x`: A vector with growth values.

**Value**

A vector of level values.

**Examples**

```r
table1 <- value_decom(c("h2","x2"), c("w2","u2"), "y2", "p2", "year", mining)[[1]]
roll_prod(table1[, "TFPG"])
```
**t_weight**

**Description**

Explanatory factors of value added decomposition adopted to demonstrate the aggregation over industries.

**Usage**

sector

**Format**

A data frame with the following columns:

- **year** A time period column.
- **p** Output prices.
- **y** Output quantities.
- **alpha** Net output price indexes.
- **beta** Input quantity indexes
- **gamma** Input mix indexes.
- **epsilon** Value added efficiency indexes.
- **tau** Technical progress indexes.
- **industry** Industry codes.

**References**


---

**t_weight**

**Aggregation over Sectors with a Weighted Average Approach**

**Description**

This "bottom up" approach uses weighted averages of the sectoral decompositions to provide an approximate decomposition into explanatory components at the aggregate level. Specifically, the Tornqvist index is adopted in the aggregation.

**Usage**

```
t_weight(y, p, id, t, alpha, beta, gamma, epsilon, tau, data)
```
Arguments

y A string (or a vector of strings) indicating the output quantity columns.
p A string (or a vector of strings) indicating the output price columns.
id A string indicating the industry column.
t A string indicating the time period column.
alpha A string indicating net output price indexes.
beta A string indicating input quantity indexes.
gamma A string indicating input mixe indexes.
epsilon A string indicating value added efficiency indexes.
tau A string indicating technical progress indexes.
data A data frame containing input prices, input quantities, industry identities, the time period, and explanatory factors of value added growth.

Value

A list containing a growth-value table and a level-value table of explanatory factors for value added growth decomposition. It is sorted by the time period.

References


Examples

# Use the built-in dataset "sector"
table1 <- t_weight("y", "p", "industry", "year", "alpha", "beta", "gamma", "epsilon", "tau", sector)[[1]]
table2 <- t_weight("y", "p", "industry", "year", "alpha", "beta", "gamma", "epsilon", "tau", sector)[[2]]

value_decom Decomposing Value Added Growth into Explanatory Factors

Description

This method for decomposing nominal value added growth is proposed by Diewert and Fox (2018), which identifies the contributions from efficiency change, growth of primary inputs, changes in output and input prices, technical progress and returns to scale.

Usage

value_decom(x, w, y, p, t, data)
Arguments

- **x**: A string (or a vector of strings) indicating the quantity columns.
- **w**: A string (or a vector of strings) indicating the input price columns.
- **y**: A string (or a vector of strings) indicating the output quantity columns.
- **p**: A string (or a vector of strings) indicating the output price columns.
- **t**: A string indicating the time period column.
- **data**: A data frame containing input prices, input quantities, output prices, output quantities, and the time period.

Value

A list containing a growth-value table and a level-value table of explanatory factors for value added growth decomposition. It is sorted by the time period.

References


Examples

```r
# Use the built-in dataset "mining"

table1 <- value_decom(c("h2", "x2"), c("w2", "u2"), "y2", "p2", "year", mining)[[1]]
table2 <- value_decom(c("h2", "x2"), c("w2", "u2"), "y2", "p2", "year", mining)[[2]]
```
Index

Topic **datasets**
- mining, 2
- sector, 4

mining, 2
roll_div, 3
roll_prod, 3
sector, 4
t_weight, 4
value_decom, 5