# Package 'gtop'

March 5, 2015

**Type** Package

**Title** Game-Theoretically OPtimal (GTOP) Reconciliation Method

**Version** 0.2.0

**Date** 2015-01-23

**Author** Jairo Cugliari, Tim van Erven

**Maintainer** Jairo Cugliari <Jairo.Cugliari@univ-lyon2.fr>

**Description** In hierarchical time series (HTS) forecasting, the hierarchical relation between multiple time series is exploited to make better forecasts. This hierarchical relation implies one or more aggregate consistency constraints that the series are known to satisfy. Many existing approaches, like for example bottom-up or top-down forecasting, therefore attempt to achieve this goal in a way that guarantees that the forecasts will also be aggregate consistent. This package provides with an implementation of the Game-Theoretically OPtimal (GTOP) reconciliation method proposed in van Erven and Cugliari (2015), which is guaranteed to only improve any given set of forecasts. This opens up new possibilities for constructing the forecasts. For example, it is not necessary to assume that bottom-level forecasts are unbiased, and aggregate forecasts may be constructed by regressing both on bottom-level forecasts and on other covariates that may only be available at the aggregate level.

**License** GPL-2 | GPL-3

**Depends** hts, quadprog, lassoshooting

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2015-03-05 07:40:24

## R topics documented:

gtop *Reconciliate individual predictions using GTOP*

## Description

Uses a Game Theory approach to reconciliate hierarchical time series predicitons

## Usage

```
gtop(preds_indiv, pred_total, weights_indiv, weight_total, bounds_indiv,
  solver = "quad")
```

## Arguments

| | |
|---|---|
| preds_indiv | vector contains the individual predictions |
| pred_total | prediction for the sum of individuals |
| weights_indiv | vector, contains the weights of the individuals |
| weight_total | weight of the total |
| bounds_indiv | vector, contains the bounds of the individuals |
| solver | string, use quadratic programming (quad) or Lasso-like solvers (lasso) |

## Details

In hierarchical time series forecasts, one predicts individuals quantities and a global quantity. There exists a contraint that matches the sum of the individual quantities to the global quantity. However, forecasting models don't take into account this constraint. With GTOP you can reconciliate the individual and global quantities in order to match the aggregate consistency contraint.

## Value

A list with

- pred_indivs the reconciliated predictions for the individuals and the total,
- solution the solution to the associate minimisation problem.

## Examples

```
K <- 5
indiv <- rep(0, K)
total <- 1
gtop(preds_indiv   = indiv,
     pred_total = total,
     weights_indiv = rep(1, K),
     weight_total = 2,
     bounds_indiv  = rep(1 / K, K))
```

## hts *Prediction conciliation by ...n.*

### Description

Uses a simple L2 projection to reconciliate hierarchical time series forecasts.

### Usage

```
hts(preds_indiv, pred_total)
```

### Arguments

| | |
|---|---|
| preds_indiv | : K-length vector with predictions ybar_1,...,ybar_K for individual regions |
| pred_total | : number with prediction ybar_* for the total consumption |

### Value

A vector with the reconciliated predictions for the individuals and the total.

### References

Hydman et al. (2011)

### Examples

```
K <- 5
hts(preds_indiv = rep(0, K), 1)
```

## proj *Prediction conciliation by projection.*

### Description

Uses a simple L2 projection to reconciliate hierarchical time series forecasts.

### Usage

```
proj(preds_indiv, pred_total, weights_indiv, weight_total)
```

### Arguments

| | |
|---|---|
| preds_indiv | : K-length vector with predictions ybar_1,...,ybar_K for individual regions |
| pred_total | : number with prediction ybar_* for the total consumption |
| weights_indiv | : K-length vector with weights a_1,...,a_K for individual regions |
| weight_total | : number with weight a_* for the total consumption |

## Value

A vector with the reconciliated predictions for the individuals and the total.

## Examples

```
K <- 5
proj(preds_indiv = rep(0, K), 1,
     weights_indiv = rep(1, K),
     weight_total = 2)
```

# Index