Package ‘refweekreg’

May 21, 2024

Title Generate reference week regressors

Version 1.1

Description Generates reference week regressors to include in time series models.
These refernce week regressors are indicator variables that have a 1 if the week
a given US holiday occurs in is a reference week, 0 otherwise.
Routines also generate the day that starts the reference week
(usually the week that contains the 12th of the month).
Another routine generates a 4 or 5 week regressor used to calendar adjust
selected CES series.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Imports lubridate,
timeDate

Suggests seasonal

Depends R (>= 2.10)

Contents

calendar_mean_adj *Calendar mean adjust regressor*

Description

Remove the calendar month mean of a given regressor expressed as a time series object

Usage

```
calendar_mean_adj(this_reg = NULL)
```

Arguments

this_reg	double precision time series array; a regressor from which the calendar month mean will be removed. This is a required entry.

Value

double precision time series array of the regressor with the calendar month mean removed

Examples

```
gf_years <- c(2001, 2006, 2017, 2022)
this_gf <-
     gen_rw_regressors(gf_years, 4, 2000, 2027, 'gf', remove_cal_means = FALSE)
this_gf_mean_adj <- calendar_mean_adj(this_gf)
```

gen_4to5_week_regressors *generate 4/5 week regressors*

Description

Generate 4 to 5 week effect regressors from paper by Cano, Scott, Kropf, Scott and Stamas (1996)

Usage

```
gen_4to5_week_regressors(
  start_year = NULL,
  end_year = NULL,
  omit_march = TRUE,
  remove_cal_means = FALSE
)
```

Arguments

`start_year`	First year of the sequence This is a required argument.
`end_year`	Ending year of the sequence This is a required argument.
`omit_march`	Logical scalar, exclude March from the set of regressors returned Default is TRUE
`remove_cal_means`	Logical scalar; if TRUE, calendar month means are removed from the final regression matrix. Default setting is FALSE.

Value

Matrix of time series arrays of 4/5 week regressors starting in January of `start_year` and ending in December of `end_year`, with each column regresenting a different month.

Examples

```
four2five_reg     <- gen_4to5_week_regressors(2005, 2024)
four2five_reg_cal <- gen_4to5_week_regressors(2005, 2024, remove_cal_means = TRUE)
```

`gen_indirect_quarterly_holiday`
Generate indirect quarterly holiday adjustments

Description

generate indirect quarterly holiday factors and an indirect holiday adjusted series from monthly time series and monthly holiday factors

Usage

```
gen_indirect_quarterly_holiday(this_a1 = NULL, this_hol = NULL, ratio = TRUE)
```

Arguments

`this_a1`	Real array; ts object of the original series This is a required argument.
`this_hol`	Real array; ts object of the holiday factors This is a required argument.
`ratio`	Logical scalar; if TRUE, holiday factors are assumed to be ratios; otherwise, the factors are assumed to be on the same scale as the original series. Default setting is TRUE.

Value

list of two ts objects: holadj, which contains the indirect holiday adjusted quarterly series and holfac, the indirect holiday factors

Examples

```
n2033157_hol_q_list <- gen_indirect_quarterly_holiday(n2033157_a1, n2033157_hol)
```

gen_reference_day *Reference day generation*

Description

Generate the start of the reference week for a span of years, with an option to incorporate exceptions

Usage

```
gen_reference_day(
  start_year = NULL,
  end_year = NULL,
  census_adj = TRUE,
  back_dates = NULL
)
```

Arguments

start_year	First year of the sequence This is a required argument.
end_year	Ending year of the sequence This is a required argument.
census_adj	Logical scalar, apply adjustment done by US Census Bureau for November and December starting in 2005. Default is TRUE.
back_dates	Array of date objects where the reference week needs to be set back a week by interviewers

Value

Array of reference week dates starting in January of `start_year` and ending in December of `end_year`

Examples

```
replacement_dates <- c(as.Date('2013-11-10'), as.Date('2019-11-10'))
rw2005 <- gen_reference_day(2005, 2024, back_dates = replacement_dates)
```

gen_reference_week_start_regressor *generate reference week start holiday regressors*

Description

generate reference week start holiday regressors, defined as a time series matrix.

Usage

```
gen_reference_week_start_regressor(
  this_reference_day = NULL,
  collapse_col = NULL,
  reg_means = NULL,
  contrast_reg = TRUE
)
```

Arguments

`this_reference_day`
: Array of date objects of the start of the reference week for a given month This is a required argument.

`collapse_col`
: integer scalar; collapses the first `collapse_col` columns into a single column; Default setting is NULL, all columns are returned.

`reg_means`
: numeric vector; vector of means to be removed from the regressors; Default is NULL, no mean removal done.

`contrast_reg`
: Logical scalar; if TRUE, contrast regressors are generated. Default setting is TRUE.

Value

regression matrix time series object with reference week start regressors.

Examples

```
replacement_dates <- c(as.Date('2013-11-10'), as.Date('2019-11-10'))
rw2005 <- gen_reference_day(2005, 2024, back_dates = replacement_dates)
ref_week_start_reg <- gen_reference_week_start_regressor(rw2005)
```

---

gen_rw_holiday_matrix *Reference week regression matrix*

Description

Generate full regression matrix for reference week related holiday regressors, defined as a time series object

Usage

```
gen_rw_holiday_matrix(
  this_reference_week = NULL,
  add_gf = TRUE,
  add_easter = TRUE,
  add_labor = TRUE,
  add_columbus = TRUE,
  add_vet = TRUE,
  remove_cal_means = TRUE
)
```

Arguments

`this_reference_week`
: Array of date objects of the start of the reference week for a given month This is a required argument.

`add_gf`
: Logical scalar; if TRUE, a Good Friday holiday regressor will be included in the regression matrix. Default setting is TRUE.

`add_easter`
: Logical scalar; if TRUE, an Easter holiday regressor will be included in the regression matrix. Default setting is TRUE.

add_labor — Logical scalar; if TRUE, a Labor Day holiday regressor will be included in the regression matrix. Default setting is TRUE.

add_columbus — Logical scalar; if TRUE, a Columbus Day holiday regressor will be included in the regression matrix. Default setting is TRUE.

add_vet — Logical scalar; if TRUE, a Veteran's Day holiday regressor will be included in the regression matrix. Default setting is TRUE.

remove_cal_means — Logical scalar; if TRUE, calendar month means are removed from the final regression matrix. Default setting is TRUE.

Value

Array of reference week dates starting in January of `start_year` and ending in December of `end_year`

Examples

```
replacement_dates <- c(as.Date('2013-11-10'), as.Date('2019-11-10'))
rw2005 <- gen_reference_day(2005, 2024, back_dates = replacement_dates)
holiday_matrix_2005 <- gen_rw_holiday_matrix(rw2005, add_gf = FALSE, add_easter = FALSE,
                        remove_cal_means = FALSE)
```

---

gen_rw_regressors — *Generate reference week regressor*

Description

generate specific reference week related holiday regressors, defined as a time series object. The object returned is either a time series vector or a matrix of time series indicator variables.

Usage

```
gen_rw_regressors(
  holiday_years = NULL,
  holiday_month = NULL,
  start_year = NULL,
  end_year = NULL,
  col_label = "Reg",
  join_regressors = TRUE,
  remove_cal_means = TRUE
)
```

Arguments

holiday_years — Integer array; Years where the holiday appears in the reference week (or for Columbus Day or Veteran's day, years where the holiday does not occur in the reference week) This is a required argument.

holiday_month — Integer scalar; The month in which this holiday occurs. For Good Friday and Easter, this would be April (4); for Labor Day, this would be September (9), etc. This is a required argument.

`start_year` Integer scalar; First year of the generated holiday regressor. The regressor will begin on the first observation of this year. This is a required argument.

`end_year` Integer scalar; Final year of the generated holiday regressor. The regressor will end on the last observation of this year. This is a required argument.

`col_label` Character scalar; A label used to generate column names for the individual AO regressors if `join_regressors = FALSE`.

`join_regressors`
Logical scalar; if TRUE, individual indicator regressors are combined into one grouped regressor. If FALSE, a matrix of the individual AO regressors will be returned. Default setting is TRUE.

`remove_cal_means`
Logical scalar; if TRUE, calendar month means are removed from the final regression matrix. Default setting is TRUE.

Value

if `join_regressors = TRUE`, a time series object with the holiday regressor is returned; otherwise, a matrix of AO regressors for the individual holidays is returned.

Examples

```
gf_years <- c(2001, 2006, 2017, 2022)
# returns a time series object with one grouped regressor
this_gf_grouped <-
     gen_rw_regressors(gf_years, 4, 2000, 2027)
# returns a time series matrix with four columns of indicator regressors
this_gf_individual <-
     gen_rw_regressors(gf_years, 4, 2000, 2027, 'gf', join_regressors = FALSE)
```

`n2033157_a1` *At Work Series*

Description

A time series object of an at work hours series

Usage

```
n2033157_a1
```

Format

A time series object of an at work hours series from January of 2003 to May of 2020

n2033157_hol *At Work Series Monthly Holiday Factors*

Description

A time series object of monthly holiday factors from an at work hours series

Usage

```
n2033157_hol
```

Format

A time series object of monthly holiday factors from an at work hours series from January of 2003 to May of 2020

save_user_reg *save user regression matrix*

Description

Save a user-defined regression array or matrix with time series attributes to an external ASCII file in X-13ARIMA-SEATS' datevalue format

Usage

```
save_user_reg(this_reg = NULL, this_reg_file = NULL)
```

Arguments

this_reg	double precision time series array or matrix to be saved. This is a required argument.
this_reg_file	character string; name of file time series array or matrix to be saved to. This is a required argument.

Value

file with user-defined regressors will be produced

Examples

```
gf_years <- c(2001, 2006, 2017, 2022)
this_gf <-
     gen_rw_regressors(gf_years, 4, 2000, 2027, 'gf', remove_cal_means = FALSE)
## Not run: save_user_reg(this_gf, 'gf_2000_2027.txt')
```

`xt_data_list` *US Building Permits*

Description

A list object with 12 components of US Building Permits expressed as time series objects

Usage

```
xt_data_list
```

Format

A list object with 12 time series elements:

mw1u Midwest one family building permits

mwto Midwest total building permits

ne1u Northeast one family building permits

neto Northeast total building permits

so1u South one family building permits

soto South total building permits

we1u West one family building permits

weto West total building permits

us1u US one family building permits

us24 US 2-4 family building permits

us5p US 5+ family building permits

usto US total family building permits

Index