Package ‘modTempEff’

April 13, 2012

Type Package

Title Modelling temperature effects using time series data

Version 1.5.1

Date 2012-04-12

Author Vito M. R. Muggeo

Maintainer Vito M.R. Muggeo <vito.muggeo@unipa.it>

Description Fits a Constrained Segmented Distributed Lag regression model to epidemiological time series of mortality, temperature,and other confounders.

Depends mgcv, splines

Repository CRAN

Date/Publication 2012-04-13 20:41:08

R topics documented:

`modTempEff-package` *Modelling temperature effects on mortality via the constrained segmented distributed lag parameterization*

Description

Estimation of a Poisson log linear regression to model the temperature effect on health using the 'constrained segmented distributed lag parameterization' which allows to account simultaneously for typical features of temperature effects: nonlinear effect delayed over several days.

Details

Package:	modTempEff
Type:	Package
Version:	1.5.1
Date:	2012-04-12
License:	GPL

Package `modTempEff` fits a Poisson log linear regression to model the temperature effect on health using the 'constrained segmented distributed lag parameterization' which allows to account simultaneously for typical features of temperature effects: nonlinear effect delayed over several days. Fitting is performed by the function `tempeff` and 'print', 'summary', 'plot', 'anova' and 'coef' methods are included to summarize and to display results.

Author(s)

Vito M.R. Muggeo `<vito.muggeo@unipa.it>`

References

Muggeo, V.M.R. (2010) Analyzing temperature effects on mortality within the R environment: the constrained segmented distributed lag parameterization *Journal of Statistical Software* **32**, 12, 1–17.

Muggeo, V.M.R. (2008) Modeling temperature effects on mortality: multiple segmented relationships with common break points *Biostatistics* **9**, 613–620.

Muggeo, V.M.R., Hajat, S. (2009) Modelling the nonlinear multiple-lag effects of ambient temperature on mortality in Santiago and Palermo : a constrained segmented distributed lag approach. *Occupational Environmental Medicine* **66**, 584–591.

Muggeo, V.M.R. (2003) Estimating regression models with unknown break-points. *Statistics in Medicine* **22**, 3055–3071.

Some background references on temperature effect on mortality via time series data (epidemiological papers)

Armstrong, B. (2006) Models for the relationship between ambient temperature and daily mortality *Epidemiology* **17**, 624–631.

Basu, R., Samet, J. (2002). Relation between elevated ambient temperature and mortality: a review of the epidemiologic evidence *Epidemiological Reviews* **24**, 190–202.

See Also

mgcv

Examples

```
## Not run:
data(dataDeathTemp)
o1<-tempeff(dec1~day+factor(dweek)+factor(year)+factor(month)+
      csdl(mtemp,L=c(60,60),psi=20, ridge=list(cold="l^2",heat="l^2")),
      data=dataDeathTemp, fcontrol = fit.control(display=TRUE))

o2<-tempeff(dec1~seas(day,30)+
      csdl(mtemp,L=c(60,60),psi=20, ridge=list(cold="l^2",heat="l^2")),
      data=dataDeathTemp, fcontrol = fit.control(display=FALSE))


## End(Not run)
```

---

anova.modTempEff *The anova method for a 'modTempEff' object*

Description

Comparing "modTempEff" objects returned by tempeff() using an analysis of deviance table.

Usage

```
## S3 method for class 'modTempEff'
anova(object, ..., dispersion = NULL, test = NULL)
```

Arguments

object,...	fitted model objects of class "modTempEff" returned by tempeff().
dispersion	currently ignored.
test	what sort of likelihood based criterion has to be used for model comparisons. One of "Chisq", "F", "Cp" or "BIC".

Details

anova.modTempEff performs model comparisons in terms of likelihood-based criteria depending on its argument test. In anova.modTempEff, test="BIC" is also allowed. The BIC appears to be the best choice to select the number of breakpoints. When test="Chisq", likelihood ratio tests are carried out; however note that the p-values for smooth terms (included via seas() or csdl()) are approximate only. The function does *not* work for a single "modTempEff" fit.

Author(s)

Vito Muggeo

See Also

print.modTempEff

Examples

```
  ## Not run:
  #continues from ?tempeff
  anova(o3,o1,test="Chisq") #approximate p-value..

  anova(o3,o2,o1,test="Cp")


## End(Not run)
```

---

coef.modTempEff — *Extract DL coefficients from a 'modTempEff' object*

Description

The function extracts the estimated coefficients of the DL curves for cold and/or heat effect.

Usage

```
## S3 method for class 'modTempEff'
coef(object, which = c("cold", "heat"), L, ...)
```

Arguments

object	the "modTempEff" object returned by tempeff.
which	which DL curve should be returned?
L	the number of DL coefficients required. 'L+1' coefficients for lags 0 to L are returned.
...	additional arguments (ignored).

Details

The resulting estimates, returned as a matrix, are DL coefficients for the cold and/or the heat effect. Each coefficient at specific lag represents the log relative risk (of mortality) for one-unit increase in cold (or heat) values.

Value

A matrix with DL coefficients. Each row corresponds to a specific lag.

Author(s)

Vito Muggeo

See Also

tempeff

Examples

```
## Not run:
 #continues from ?tempeff
 coef(o1,"heat",L=7) #log RR for heat corresponding to lag 0 to 7

## End(Not run)
```

---

csdl *Defining the constrained segmented distributed lag term*

Description

Function employed within the tempeff formula to specify the variable with a csdl relationship. The function does not fit the model, it simply returns information exploited by tempeff to fit a "modTempEff" model.

Usage

```
csdl(z, psi, L, ridge = NULL, ndx = round(L/3),
    DL = FALSE, diff.varying = FALSE)
```

Arguments

z	the variable, typically the temperature, having the constrained segmented distributed lag parameterization with the response.
psi	numeric to provide the starting value for one or two breakpoints of the constrained segmented distributed lag relationship.
L	a numerical two-length vector to specify how many lags have to be considered to assess the effect of cold and heat.
ndx	*apparent* dimension (i.e. the rank) of the two B-spline bases for the DL curves of cold and heat. Default to `ndx=round(L/3)`.
ridge	a two-length named list of characters to specify the possible ridge penalty to be applied to DL coefficients. This list has to be named ('cold' and 'heat') and each component has to be in terms of `"l"`. Use `ridge=NULL` whether no ridge penalty has to be employed. See examples below.
DL	logical indicating if the difference penalty should be applied to the DL coefficients or to spline coefficients. Default to spline coefficients (`DL=FALSE`).
diff.varying	logical indicating if the difference penalty should be global or depending on the lag value to penalize differences mainly at larger rather than early lags.

Details

This function has to be used within the formula in tempeff. It returns information to fit a constrained segmented distributed lag parameterization within the Poisson regression model via the function tempeff.

Value

A list with the temperature variable and the arguments as attributes.

Note

All the arguments of csdl may be passed via ... in the call of tempeff. This feature may be useful when the same model has to be re-fitted (via update) by modifying only one argument of csdl(). See tempeff.

Author(s)

Vito Muggeo

References

Eilers, P., Marx, B. D. (1996). Flexible Smoothing with B-splines and Penalties *Statistical Science* **11**, 89–121.

Wood, S. N. (2006) *Generalized Additive Models: An Introduction with R.* Chapman and Hall/CRC Press.

Muggeo, V. M. R. (2008) Modeling temperature effects on mortality: multiple segmented relationships with common break points *Biostatistics* **9**, 613–620.

See Also

tempeff

Examples

```
## Not run:
# Evaluate temperature effects up to 45 lags for cold and heat and
# a single breakpoint; use 20 as starting value, a global difference
# penalty on spline coefficients and no ridge penalty
csdl(my.temperature,psi=20,L=c(45,45),ridge=NULL)

# Evaluate temperature effects up to 45 lags for cold and 15 lag for
# heat, via P-splines with a global difference penalty on DL coefficients
# and an additional quadratic ridge penalty
csdl(my.temperature,psi=20,L=c(45,15),DL=TRUE,ridge=list(cold="l^2",
  heat="l^2"))

## End(Not run)
```

dataDeathTemp *Simulated dataset*

Description

Simulated dataset of daily time series of mortality, temperature, pollutant, and seasonal variables.

Usage

```
data(dataDeathTemp)
```

Format

A data frame with 1825 observations on the following 8 variables.

`dec1` daily counts of mortality

`mtemp` daily mean temperatute

`month` month

`year` year

`day` day

`dweek` day of week

`decNS` another daily counts of mortality

`dec2` yet another daily counts of mortality

Details

These data represent a typical dataset employed to investigate the short term effect of temperature (and pollutant) on health via daily time series. `decNS` is not associated with `mtemp`, `dec1` and `dec2` are associated with `mtemp` via *csdl* parameterizations with one and two breakpoints respectively.

Examples

```
## Not run: data(dataDeathTemp)
```

`fit.control` *Auxiliary function for controlling model fitting*

Description

Auxiliary function as user interface for fitting. Typically only used when calling 'tempeff' or 'tempeff.fit'.

Usage

```
fit.control(tol = 1e-06, display = FALSE, it.max = 20,
  GLM = FALSE, maxit.inner = 3)
```

Arguments

`tol`	positive convergence tolerance.
`display`	logical indicating if deviance should be printed for each iteration. This argument is ignored (actually it is `FALSE`) when `it.max=0` or when the model is fitted without the temperature effect.
`it.max`	integer giving the maximal number of iterations.
`GLM`	logical indicating if at each iteration a GLM (rather than a GAM) has to be fitted.
`maxit.inner`	integer giving the maximal number of inner iterations.

Details

Fitting of Constrained Distributed Lag Model is performed via iterative estimate of proper Generalized Additive (or Linear when `GLM=TRUE`) models. The algorithm stops when the relative increase in deviance is smaller than `tol` or when the maximum number of iterations `it.max` is attained. The maximum number of (inner) iterations to estimate the working GLM at each (outer) iteration is controlled by `maxit.inner`.

Value

A list with the arguments as components to be used by 'tempeff' or 'tempeff.fit'.

Author(s)

Vito Muggeo

See Also

`tempeff`

`logLik.modTempEff` *Extract the log likelihood for a modTempEff fit*

Description

Function to extract the log-likelihood for a fitted `modTempEff` model.

Usage

```
## S3 method for class 'modTempEff'
logLik(object,...)
```

Arguments

`object`	fitted model objects of class `modTempEff` as produced by `tempeff()`.
`...`	ignored

Details

This function extracts the *penalized* log likelihood for a fit produced by `tempeff()`, namely a Poisson GAM fit as returned by `mgcv::gam()`.

Notice that the model degrees of freedom are the effective degrees of freedom and not the number of coefficients, as the model is estimated by penalized likelihood.

Value

Standard `logLik` object: see `logLik`.

Author(s)

Vito M. R. Muggeo `<vito.muggeo@unipa.it>` based directly on `logLik.gam` in `mgcv` package by S. Wood.

See Also

`AIC`

plot.modTempEff *Plot method for the class 'modTempEff'*

Description

Plots distributed lags curves from the `modTempEff` fit.

Usage

```
## S3 method for class 'modTempEff'
plot(x, which = c("cold", "heat"), add=FALSE, new=TRUE,
  var.bayes = FALSE, delta.rr = TRUE, level = 0.95, ...)
```

Arguments

`x`	object of class `"modTempEff"`.
`which`	Which DL curve should be plotted? for cold, heat or both of them (default).
`add`	logical; if `TRUE` the fitted DL curve for cold *or* heat is added to an existing plot.
`new`	logical indicating if a new device should be opened. If `add=TRUE`, `new` is set to `FALSE`.
`var.bayes`	logical indicating if the 'Bayesian' rather than the frequentist standard errors should be employed to compute the pointwise confidence intervals to be plotted
`delta.rr`	logical indicating if the DL curves should be plotted on the log scale or as per cent change in relative risk, i.e. 100*(exp(.)-1).
`level`	the selected confidence level of the pointwise confidence intervals to be plotted
`...`	additional arguments..

Details

Takes a fitted `"modTempEff"` object produced by `tempeff()` and plots the DL curves for cold and heat effect with relevant pointwise confidence intervals. `plot.modTempEff` also works with objects with fixed (not estimated) breakpoint, namely fits returned by

`tempeff(.., fcontrol=fit.control(it.max=0))`.

Note `add=TRUE` makes sense (and works) only for a single (cold *or* heat) DL curve to be superimposed to an existing plot.

Value

The function simply plots the required estimated DL curve. If the fitted model includes only a smooth term for the long term trend, `plot.modTempEff` draws it.

Author(s)

Vito Muggeo

See Also

tempeff

Examples

```
  ## Not run:
  #obj is an object returned by tempeff()
  #plots DL curves for cold and heat with 95% pointwise CI
  # using frequentist standard errors
  plot(obj)

  #plots the estimated DL curve only for heat with 90% pointwise CI
  # using bayesian standard errors
  plot(obj, "heat", var.bayes=TRUE, level=.90)

## End(Not run)
```

`print.modTempEff` *The print method for a 'modTempEff' object*

Description

Prints a Constrained Segmented Distributed Lag Model object

Usage

```
## S3 method for class 'modTempEff'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

`x`	object of class `modTempEff`
`digits`	number of digits to be printed
`...`	additional arguments..

Details

Prints the minimal features of a `modTempEff` object.

Author(s)

Vito Muggeo

See Also

`summary.modTempEff`

`seas` *Specifying a smooth long term trend within a 'modTempEff' model*

Description

Function employed within the `tempeff` formula to specify the variable with a smooth effect. The function does not fit the model, it simply returns information exploited by `tempeff`.

Usage

```
seas(x, ndx = stop("please, provide 'ndx' in seas()"))
```

Arguments

`x`	the long term trend variable
`ndx`	the apparent dimension (i.e. the rank) of the B-spline employed to model the long trend. For instance it could be `min(40,n/4)` where `n` is the time series length.

Details

The function is used within the formula of `tempeff()` when the long term trend of the daily time series is fitted via a smooth term. A standard P-spline with a cubic B-spline basis and a second-order difference penalty is employed.

Value

A list with relevant information.

Author(s)

Vito Muggeo

References

Eilers, P., Marx, B. (1996). Flexible Smoothing with B-splines and Penalties *Statistical Science* **11**, 89–121.

Wood, S.N. (2006) *Generalized Additive Models: An Introduction with R.* Chapman and Hall/CRC Press.

See Also

`csdl`, `tempeff`

summary.modTempEff — *Summary method for the class 'modTempEff'*

Description

Summarizes fit for the constrained segmented distributed effect model.

Usage

```
## S3 method for class 'modTempEff'
summary(object, spar = TRUE,
  digits = max(3, getOption("digits") - 3), ...)
```

Arguments

`object`	object of class `"modTempEff"`
`spar`	logical indicating if values of smoothing parameters should be printed.
`digits`	number of digits to be printed.
`...`	additional arguments.

Details

Prints the most important features of a `modTempEff` object including fit summary (AIC, BIC, Ubre, residual deviance) and point estimates along with standard errors of the net effect of cold and heat, and the breakpoints where mortality reaches its minimum. Smoothing parameters (selected via `gam.fit()` of the mgcv package) are also printed when `spar=TRUE`. The method also works when the model has been fitted with fixed breakpoints, but it does *not* work if a `csdl()` is not included in the model formula.

Author(s)

Vito Muggeo

See Also

`print.modTempEff`

`tempeff` *Modelling temperature effects on mortality*

Description

Fits the constrained segmented distributed lag log-linear regression model to daily time series data of mortality and temperature and additional confounding factors.

Usage

```
tempeff(formula, data, fcontrol = fit.control(), etastart = NULL,
  drop.L, ...)
```

Arguments

`formula`	the model formula such as 'response ~ parametric terms + csdl(temperature) + seas(day)', see details.
`data`	the dataset where the variables are stored.
`fcontrol`	a list with components returned by `fit.control()`.
`etastart`	possible starting values on the scale of the linear predictor.
`drop.L`	integer, specifying whether the first '`drop.L`' observations should be removed before fitting. This is useful for model comparison purposes, see notes.
`...`	additional arguments to be passed to `csdl()` in the formula, see details.

Details

The function fits a log-linear regression model to assess the effects of temperature on mortality using a 'constrained segmented distributed lag parameterization' (csdl). It is assumed that the data are daily time series of mortality (or perhaps morbidity) and temperature. The left hand side of the formula includes the response (daily counts), and the right hand side may include one or more of the following

- linear confounders (such as influenza epidemics or day-of-week);
- nonparametric long term trend, via the function `seas`;
- the csdl effect of temperature via the function `csdl`.

All the arguments of `csdl()` may be passed via `...` directly in the call of `tempeff`. This may facilitate the user when different models have to be fitted by changing only some of (and not all) the arguments of `csdl()`. See the example below.

Value

The function returns an object of class `"modTempEff"`. It is the list returned by `gam.fit` of package `mgcv` with the additional components

`psi`	The estimated breakpoint with corresponding standard error (bayesian and frequentist).
`betaCold`	The estimated DL coefficients for the cold effect.
`SE.c`	The frequentist standard errors of the cold DL estimates.
`SE.c.bayes`	The bayesian standard errors of the cold DL estimates.
`ToTcold`	Estimate and frequentist standard error of the total (net) effect of cold.
`ToTcold.bayes`	Estimate and bayesian standard error of the total (net) effect of cold.
`edf.cold`	The df associated at each spline coefficient of the DL curve of cold.
`rank.cold`	The apparent dimension of the B-spline basis of the DL for cold.
`betaHeat`	The estimated DL coefficients for the heat effect.
`SE.h`	The frequentist standard errors of the heat DL estimates.
`SE.h.bayes`	The bayesian standard errors of the heat DL estimates.
`ToTheat`	Estimate and frequentist standard error of the total (net) effect of heat.
`ToTheat.bayes`	Estimate and bayesian standard error of the total (net) effect of heat.
`edf.heat`	The df associated at each spline coefficient of the DL curve of heat.
`rank.heat`	The apparent dimension of the B-spline basis of the DL for heat.
`rank.seas`	When `ndx.seas>0`, the apparent dimension of the B-spline basis for seasonality.
`edf.seas`	When `ndx.seas>0`, the df associated at spline coefficients of seasonality.
`fit.seas`	When `ndx.seas>0`, the fitted long-term trend (on the log scale).

Note

When a `csdl` term is included in the formula, the first `max(L)` observations are discarded before model fitting. When a csdl term is *not* included, the argument `drop.L` may be used to discard the first `drop.L` observations anyway. Fitting models with the same number of observations may be useful to compare them via likelihood-based criteria (via `anova.modTempEff`, say).
`tempeff()` returns objects of class `"modTempEff"`, so proper methods may be employed. The returned object has class `"modTempEff"` even if `tempeff()` is called without `csdl()` in the formula, or even if the model is fitted with fixed (not estimated) breakpoints (via
`tempeff(..,fcontrol=fit.control(it.max=0))`).

Author(s)

Vito Muggeo, <vito.muggeo@unipa.it>

References

Muggeo, V.M.R. (2008) Modeling temperature effects on mortality: multiple segmented relationships with common break points *Biostatistics* **9**, 613–620.

Muggeo, V.M.R. (2009) Analyzing Temperature Effects on Mortality Within the R Environment: The Constrained Segmented Distributed Lag Parameterization *Journal of Statistical Software*, **32** 12, 1–17.

See Also

modTempEff-package, plot.modTempEff, summary.modTempEff, gam.fit in package mgcv

Examples

```
## Not run:
library(modTempEff)
data(dataDeathTemp)
o1<-tempeff(dec1~day+factor(dweek)+factor(year)+factor(month)+
      csdl(mtemp,L=c(60,60),psi=20),
      data=dataDeathTemp, fcontrol = fit.control(display=TRUE))

#add a ridge penalty: note how you *can* specify ridge!
#you do NOT need to use csdl(..,ridge=..)
o2<-update(o1, ridge=list(cold="l^2", heat="l^2"))

#a model without temperature effects (the first drop.L obs are dropped)
o3<-tempeff(dec1~day+factor(dweek)+factor(year)+factor(month),
      data=dataset,drop.L=60)

#see ?anova.modTempEff for model comparisons

## End(Not run)
```

`tempeff.fit` *Estimation of constrained segmented distributed lag model*

Description

This is an internal function of package modTempEff and it should be not called by the user.

Usage

```
tempeff.fit(y, X, Af = NULL, Ac = NULL, Xf = NULL, Xc = NULL, V=NULL,
  ndx.seas=0, only.seas = FALSE, penalty = list(DL=FALSE,
  diff.varying=FALSE,ridge.formulas=NULL), gam.fit.it = NULL,
  etastart = NULL, spstart = NULL, fit.method = "magic")
```

Arguments

y	See tempeff
X	See tempeff
Af	See tempeff
Ac	See tempeff
Xf	See tempeff
Xc	See tempeff

V	See tempeff
ndx.seas	See tempeff
penalty	See tempeff
only.seas	See tempeff
gam.fit.it	See tempeff
etastart	See tempeff
spstart	See tempeff
fit.method	See tempeff

Details

This function is called by tempeff to fit the constrained segmented distributed lag model. It is based on the function gam.fit of the mgcv package by S. Wood.

Value

A list of fit information.

Author(s)

Vito Muggeo

See Also

tempeff

Index