Package ‘earth’

February 19, 2015

Version 4.2.0

Title Multivariate Adaptive Regression Splines

Author Stephen Milborrow. Derived from mda:mars by Trevor Hastie and Rob Tibshirani. Uses Alan Miller’s Fortran utilities with Thomas Lumley’s leaps wrapper.

Maintainer Stephen Milborrow <milbo@sonic.net>

Depends plotmo (>= 2.2.1), plotrix

Suggests gam, mgcv, mda, MASS

Description Build regression models using the techniques in Friedman's papers `Fast MARS' and `Multivariate Adaptive Regression Splines'. (The term `MARS' is trademarked and thus not used in the name of the package.)

License GPL-3

URL http://www.milbo.users.sonic.net/earth

Repository CRAN

Date/Publication 2015-01-12 23:01:12

NeedsCompilation yes

R topics documented:

contr.earth.response ........................................... 2
earth ........................................................................ 3
etitanic ..................................................................... 15
evimp ......................................................................... 17
format.earth ............................................................. 18
mars.to.earth ............................................................. 21
model.matrix.earth ..................................................... 22
ozone1 ....................................................................... 23
plot.earth ................................................................. 24
plot.earth.models ....................................................... 29
plot.evimp ................................................................. 32
plot.varmod ............................................................. 33
Contrasts for the earth response

Description

Contrasts function for factors in the earth response. For internal use by earth.

Usage

```r
contr.earth.response(x, base, contrasts)
```

Arguments

- `x` 
a factor
- `base` 
unused
- `contrasts` 
unused

Value

Returns a diagonal matrix. An example for a 3 level factor with levels A, B, and C:

```
A B C
A 1 0 0
B 0 1 0
C 0 0 1
```

Note

Earth uses this function internally. You shouldn’t need it. It is made publicly available only because it seems that is necessary for model.matrix.

See Also

- `contrasts`
Multivariate Adaptive Regression Splines

Description

Build a regression model using the techniques in Friedman’s papers "Multivariate Adaptive Regression Splines" and "Fast MARS".

See the package vignette “Notes on the earth package”.

Usage

## S3 method for class 'formula'
earth(formula = stop("no 'formula' arg"), data = NULL,
      weights = NULL, wp = NULL, subset = NULL,
      na.action = na.fail, keepxy = FALSE, trace = 0, glm = NULL,
      ncross=1, nfold=0, stratify=TRUE,
      varmod.method = "none", varmod.exponent = 1,
      varmod.conv = 1, varmod.clamp = .1, varmod.minspan = -3,
      Scale.y = (NCOL(y)==1), ...)

## Default S3 method:
earth(x = stop("no 'x' arg"), y = stop("no 'y' arg"),
      weights = NULL, wp = NULL, subset = NULL,
      na.action = na.fail, keepxy = FALSE, trace = 0, glm = NULL,
      degree = 1, penalty = if(degree > 1) 3 else 2,
      nk = min(200, max(20, 2 * ncol(x))) + 1,
      thresh = 0.001, minspan = 0, endspan = 0,
      newvar.penalty = 0, fast.k = 20, fast.beta = 1,
      linpreds = FALSE, allowed = NULL,
      pmethod = c("backward", "none", "exhaustive", "forward", "seqrep"),
      nprune = NULL, Object = NULL,
      Eval.model.subsets = eval.model.subsets, Adjust.endspan = 2,
      Scale.y = (NCOL(y)==1), Force.xtx.prune = FALSE, Force.weights = FALSE,
      Use.beta.cache = TRUE, Exhaustive.tol = 1e-10, ...)
Arguments

To start off, look at the arguments `formula`, `data`, `x`, `y`, `nk`, `degree`, and `trace`. Most users will find that those arguments are all they need, plus in some cases `keepxy` and `nprune`.

For Generalized Linear Models, use the `glm` argument (useful for example when the response is binary).

For cross validation, use the `nfold` argument.

For prediction intervals, use the `varmod.method` argument.

Model formula.

- `formula`: Data frame for `formula`.
- `x`: Matrix or dataframe containing the independent variables.
- `y`: Vector containing the response variable, or, in the case of multiple responses, a matrix or dataframe whose columns are the values for each response.
- `subset`: Index vector specifying which cases to use, i.e., which rows in `x` to use. Default is NULL, meaning all.
- `weights`: Default is NULL, meaning no case weights. If specified, `weights` must have length equal to `nrow(x)` before applying `subset`. The current implementation of weights is provisional and not fully tested. Thus if weights are used a warning will be issued. Supress just this warning with `trace=1` or `force.weights=TRUE`.
- `wp`: Response weights. Default is NULL, meaning no response weights. If specified, `wp` must have an element for each column of `y` (after `factors` in `y`, if any, have been expanded). Zero values are converted to a small nonzero value.
- `na.action`: NA action. Default is `na.fail`, and only `na.fail` is supported.
- `keepxy`: Default is FALSE. Set to TRUE to retain the following in the returned value: `x` and `y` (or `data`), `subset`, and `weights`. The function `update.earth` and friends will use these if present instead of searching for them in the environment at the time `update.earth` is invoked.

When the `nfold` argument is used with `keepxy=TRUE`, `earth` keeps more data (see `cv.list` in the “Value” section below), and generates `cv.oof.rsq.tab` and `cv.infold.rsq.tab`. It therefore makes cross-validation significantly slower.

- `trace`: Trace `earth`’s execution. Default is 0. Values:
  0: no tracing
  0.3: variance model (the `varmod.method` arg)
  0.5: cross validation (the `nfold` arg)
  1: overview
  2: forward pass
  3: pruning
  4: model mats, pruning details
  5: internal details of operation

- `glm`: NULL (default) or a list of arguments to pass on to `glm`. See the documentation of `glm` for a description of these arguments. See “Generalized linear models” in the package vignette. Example:
earth(survived~., data=etitanic, degree=2, glm=list(family=binomial))

The following arguments are for the forward pass.

**degree**
Maximum degree of interaction (Friedman’s *mi*). Default is 1, meaning build an additive model (i.e., no interaction terms).

**penalty**
Generalized Cross Validation (GCV) penalty per knot. Default is \( \text{if}(\text{degree}>1)\ 3\ \text{else}\ 2 \). A value of 0 penalizes only terms, not knots. The value -1 is treated specially to mean no penalty, so GCV=RSS/n. Simulation studies have suggested values in the range of about 2 to 4. The FAQ section in the package vignette has some information on GCVs.

**nk**
Maximum number of model terms before pruning, i.e., the maximum number of terms created by the forward pass. Includes the intercept. The actual number of terms created by the forward pass will often be less than nk because of other stopping conditions. See “Termination conditions for the forward pass” in the package vignette. The default is semi-automatically calculated from the number of predictors but may need adjusting.

**thresh**
Forward stepping threshold. Default is 0.001. This is one of the arguments used to decide when forward stepping should terminate: the forward pass terminates if adding a term changes RSq by less than thresh. See “Termination conditions for the forward pass” in the package vignette.

**minspan**
Minimum number of observations between knots. (This increases resistance to runs of correlated noise in the input data.) The default minspan=0 is treated specially and means calculate the minspan internally as per Friedman’s MARS paper section 3.8 with \( \alpha = 0.05 \). Set trace>=2 to see the calculated value. Use minspan=1 and endspan=1 to consider all x values. Negative values of minspan specify the maximum number of knots per predictor. These will be equally spaced. Knots that fall in the endzones specified by endspan will be ignored as usual.

**endspan**
Minimum number of observations before the first and after the final knot. The default endspan=0 is treated specially and means calculate the minspan internally as per the MARS paper equation 45 with \( \alpha = 0.05 \). Set trace>=2 to see the calculated value. See also the Adjust.endspan argument.

**newvar.penalty**
Penalty for adding a new variable in the forward pass (Friedman’s *gamma*, equation 74 in the MARS paper). Default is 0, meaning no penalty for adding a new variable. Useful non-zero values typically range from about 0.01 to 0.2 and sometimes higher — you will need to experiment. A word of explanation. With the default newvar.penal ty=0, if two variables have nearly the same effect (e.g. they are collinear), at any step in the forward pass earth will arbitrarily select one or the other (depending on noise in the sample). Both variables can appear in the final model, complicating model interpretation. On the other hand with a non-zero newvar.penalty, the forward pass will be reluctant to add a new variable — it will rather try to use a variable already in the model, if that does not affect RSq too much. The resulting final model may be easier to interpret, if you are lucky. There will often be a small performance hit (a worse GCV).
**fast.k** Maximum number of parent terms considered at each step of the forward pass. (This speeds up the forward pass. See the Fast MARS paper section 3.0.) Default is \(20\). A value of \(0\) is treated specially (as being equivalent to infinity), meaning no Fast MARS. Typical values, apart from \(0\), are \(20, 10, \) or \(5\). In general, with a lower `fast.k` (say \(5\)), earth is faster; with a higher `fast.k`, or with `fast.k` disabled (set to \(0\)), earth builds a better model. However, because of random variation this general rule often doesn’t apply.

**fast.beta** Fast MARS ageing coefficient, as described in the Fast MARS paper section 3.1. Default is \(1\). A value of \(P\) sometimes gives better results.

**linpreds** Index vector specifying which predictors should enter linearly, as in `lm`. The default is \(FALSE\), meaning all predictors enter in the standard MARS fashion, i.e., in hinge functions.

This does not say that a predictor *must* enter the model; only that if it enters, it enters linearly. See “The linpreds argument” in the package vignette.

A predictor’s index in `linpreds` is the column number in the input matrix `x` (after factors have been expanded).

- `linpreds=TRUE` makes all predictors enter linearly (the TRUE gets recycled).
- `linpreds` may also be a character vector e.g. `linpreds=c("wind","vis")`.

Note: `grep` is used for matching. Thus "wind" will match all variables that have "wind" in their names. Use "wind$" to match only the variable named "wind".

**allowed** Function specifying which predictors can interact and how. Default is `NULL`, meaning all standard MARS terms are allowed.

During the forward pass, earth calls the `allowed` function before considering a term for inclusion; the term can go into the model only if the `allowed` function returns `TRUE`. See “The allowed argument” in the package vignette.

---

**pmethod** Pruning method. One of: backward none exhaustive forward seqrep. Default is "backward".

Use "none" to retain all the terms created by the forward pass.

If `y` has multiple columns, then only "backward" or "none" is allowed.

Pruning can take a while if "exhaustive" is chosen and the model is big (more than about 30 terms). The current version of the `leaps` package used during pruning does not allow user interrupts (i.e., you have to kill your R session to interrupt; in Windows use the Task Manager or from the command line use `taskkill`).

**nprune** Maximum number of terms (including intercept) in the pruned model. Default is `NULL`, meaning all terms created by the forward pass (but typically not all terms will remain after pruning). Use this to enforce an upper bound on the model size (that is less than `nk`), or to reduce exhaustive search time with `pmethod="exhaustive"`.

---

**ncross** Only applies if `nfold>1`. Number of cross-validations. Each cross-validation has `nfold` folds. Default 1.
**nfold**

Number of cross-validation folds. Default is 0, no cross validation. If greater than 1, earth first builds a standard model as usual with all the data. It then builds nfold cross-validated models, measuring R-Squared on the out-of-fold (left out) data each time. The final cross validation R-Squared (cvrsq) is the mean of these out-of-fold R-Squares. The above process of building nfold models is repeated ncross times (by default, once). Use trace=.5 to trace cross-validation.

Further statistics are calculated if keepxy=TRUE or if a binomial or poisson model (specified with the glm argument). See "Cross validation" in the package vignette.

**stratify**

Only applies if nfold>1. Default is TRUE. Stratify the cross-validation samples so that an approximately equal number of cases with a non-zero response occur in each cross validation subset. So if the response y is logical, the TRUEs will be spread evenly across folds. And if the response is a multilevel factor, there will be an approximately equal number of each factor level in each fold (because a multilevel factor response gets expanded to columns of zeros and ones, see "Factors" in the package vignette). We say "approximately equal" because the number of occurrences of a factor level may not be exactly divisible by the number of folds.

**varmod.method**

Construct a variance model. For details, see varmod and the vignette "Variance models in earth". Use trace=.3 to trace construction of the variance model. This argument requires nfold and ncross. (We suggest at least ncross=3 here to properly calculate the variance of the errors — although you can use a smaller value, say 3, for debugging.) The varmod.method argument should be one of

- "none" Default. Don’t build a variance model.
- "const" Assume homoscedastic errors.
- "lm" Use lm to estimate standard deviation as a function of the predicted response.
- "rlm" Use rlm.
- "earth" Use earth.
- "gam" Use gam. This will use either gam or the mgcv package, whichever is loaded.
- "power" Estimate standard deviation as intercept + coef * predicted response^exponent, where intercept, coef, and exponent will be estimated by nls. This is equivalent to varmod.method="lm" except that exponent is automatically estimated instead of being held at the value set by the varmod.exponent argument.
- "power0" Same as "power" but no intercept (offset) term.
- "x.lm", "x.rlm", "x.earth", "x.gam" Like the similarly named options above, but estimate standard deviation by regressing on the predictors x (instead of the predicted response). A current implementation restriction is that "x.gam" allows only models with one predictor (x must have only one column).

**varmod.exponent**

Power transform applied to the rhs before regressing the absolute residuals with the specified varmod.method. Default is 1. For example, with varmod.method="lm", if you expect the standard deviance to
increase linearly with the mean response, use \texttt{varmod.exponent=1}. If you expect the standard deviance to increase with the square root of the mean response, use \texttt{varmod.exponent=.5} (where negative response values will be treated as 0, and you will get an error message if more than 20\% of them are negative).

\textbf{\texttt{varmod.conv}}  
\textit{Convergence criterion for the Iteratively Reweighted Least Squares used when creating the variance model.}  
Iterations stop when the mean value of the coefficients of the residual model change by less than \texttt{varmod.conv percent}. Default is 1 percent. Negative values force the specified number of iterations, e.g. \texttt{varmod.conv=-2} means iterate twice. Positive values are ignored for \texttt{varmod=\textchar`"const"} and also currently ignored for \texttt{varmod=\textchar`"earth"} (these are iterated just once, the same as using \texttt{varmod.conv=-1}).

\textbf{\texttt{varmod.clamp}}  
\textit{The estimated standard deviation of the main model errors is forced to be at least a small positive value, which we call min.sd. This prevents negative or absurdly small estimated standard deviations. Clamping takes place in predict.varmod, which is called by predict.earth when estimating prediction intervals. The value of min.sd is determined when building the variance model as min.sd = varmod.clamp \times \text{mean(sd training residuals)}. The default varmod.clamp is 0.1.}

\textbf{\texttt{varmod.minspan}}  
Only applies when \texttt{varmod.method=\textchar`"earth"} or \texttt{\textchar`"x.earth"}. This is the minspan used in the internal call to earth when creating the variance model (not the main earth model). Default is -3, i.e., three evenly spaced knots per predictor. Residuals tend to be very noisy, and allowing only this small number of knots helps prevent overfitting.

\textbf{The following arguments are for internal or advanced use.}

\textbf{Object}  
Earth object to be updated, for use by \texttt{update.earth}.

\textbf{Eval.model.subsets}  
Function to evaluate model subsets — see notes in source code.

\textbf{Adjust.endspan}  
\textbf{New in version 4.2.0.} In interaction terms, endspan gets multiplied by this value. This reduces the possibility of an overfitted interaction term supported by just a few cases on the boundary of the predictor space (as verified in our simulation studies). The default is 2. Use \texttt{Adjust.endspan=1} for compatibility with previous version of earth.

\textbf{Scale.y}  
\textit{Scale y in the forward pass for better numeric stability. Scaling here means subtract the mean and divide by the standard deviation. Default is \texttt{NCOL(y)==1}, i.e., scale y unless y has multiple columns.}

\textbf{Force.xtx.prune}  
Default is \texttt{FALSE}. This argument pertains to subset evaluation in the pruning pass. By default, if y has a single column then earth calls the \texttt{leaps} routines; if y has multiple columns then earth calls \texttt{EvalSubsetsUsingXtx}. The \texttt{leaps} routines are numerically more stable but do not support multiple responses (\texttt{leaps} is based on the QR decomposition and \texttt{EvalSubsetsUsingXtx} is based on the inverse of \texttt{X'X}). Setting \texttt{Force.xtx.prune=TRUE} forces use of \texttt{EvalSubsetsUsingXtx}, even if y has a single column.
**Force.weights**
Default is FALSE. For testing the weights argument. Force use of the code for handling weights in the `earth` code, even if weights=NULL or all the weights are the same. This will not necessarily generate an identical model, primarily because the non-weighted code requires some tests for numerical stability that can sometimes affect knot selection.

**Use.beta.cache**
Default is TRUE. Using the “beta cache” takes more memory but is faster (by 20% and often much more for large models). The beta cache uses nk * nk * ncol(x) * sizeof(double) bytes. Set `Use.beta.cache=FALSE` to save memory. (The beta cache is an innovation in this implementation of MARS and does not appear in Friedman’s papers. It is not related to the fast.beta argument. Certain regression coefficients in the forward pass can be saved and re-used, thus saving recalculation time.)

**Exhaustive.tol**
Default 1e-10. Applies only when `pmethod="exhaustive"`. If the reciprocal of the condition number of `bx` is less than Exhaustive.tol, earth forces `pmethod="backward"`. See “XHAUST returned error code -999” in the package vignette.

... Dots are passed on to `earth.fit`.

### Value

An object of class "earth" which is a list with the components listed below. Term refers to a term created during the forward pass (each line of the output from `format.earth` is a term). Term number 1 is always the intercept.

- **rss** Residual sum-of-squares (RSS) of the model (summed over all responses, if `y` has multiple columns).
- **rsq** \(1 - \text{rss/tss}\). R-Squared of the model (calculated over all responses, and calculated using the weights argument if it was supplied). A measure of how well the model fits the training data. Note that tss is the total sum-of-squares, \(\sum((y - \text{mean}(y))^2)\).
- **gcv** Generalized Cross Validation (GCV) of the model (summed over all responses). The GCV is calculated using the penalty argument. For details of the GCV calculation, see equation 30 in Friedman’s MARS paper and `earth::get.gcv`.
- **grsq** \(1 - \text{gcv/gcv.null}\). An estimate of the predictive power of the model (calculated over all responses, and calculated using the weights argument if it was supplied). `gcv.null` is the GCV of an intercept-only model. See “Can GRSq be negative?” in the package vignette.
- **bx** Matrix of basis functions applied to `x`. Each column corresponds to a selected term. Each row corresponds to a row in in the input matrix `x`, after taking subset. See `model.matrix.earth` for an example of `bx` handling. Example `bx`:

```r
(Intercept) h(Girth-12.9) h(12.9-Girth) h(Girth-12.9)*h(...
[1,] 1 0.0 4.6 0
[2,] 1 0.0 4.3 0
[3,] 1 0.0 4.1 0
...
```
Matrix with one row per MARS term, and with ij-th element equal to

0 if predictor j is not in term i
−1 if an expression of the form $h(\text{const} - x_j)$ is in term i
1 if an expression of the form $h(x_j - \text{const})$ is in term i
2 if predictor j should enter term i linearly (either because specified by the linpreds argument or because earth discovered that a knot was unnecessary).

This matrix includes all terms generated by the forward pass, including those not in selected.terms. Note that here the terms may not all be in pairs, because although the forward pass add terms as hinged pairs (so both sides of the hinge are available as building blocks for further terms), it also deletes linearly dependent terms before handing control to the pruning pass. Example dirs:

```
Girth Height
(Intercept) 0 0 #intercept
h(12.9-Girth) -1 0 #2nd term uses Girth
h(Girth-12.9) 1 0 #3rd term uses Girth
h(Girth-12.9)*h(Height-76) 1 1 #4th term uses Girth and Height...
```

cuts

Matrix with ij-th element equal to the cut point for predictor j in term i. This matrix includes all terms generated by the forward pass, including those not in selected.terms. Note for programmers: the precedent is to use dirs for term names etc. and to only use cuts where cut information needed. Example cuts:

```
Girth Height
(Intercept) 0 0 #intercept, no cuts
h(12.9-Girth) 12.9 0 #2nd term has cut at 12.9
h(Girth-12.9) 12.9 0 #3rd term has cut at 12.9
h(Girth-12.9)*h(Height-76) 12.9 76 #4th term has two cuts...
```

selected.terms

Vector of term numbers in the selected model. Can be used as a row index vector into cuts and dirs. The first element selected.terms[1] is always 1, the intercept.

prune.terms

A matrix specifying which terms appear in which pruning pass subsets. The row index of prune.terms is the model size. (The model size is the number of terms in the model. The intercept is counted as a term.) Each row is a vector of term numbers for the best model of that size. An element is 0 if the term is not in the model, thus prune.terms is a lower triangular matrix, with dimensions nprune x nprune. The model selected by the pruning pass is at row number length(selected.terms). Example prune.terms:

```
[1,] 1 0 0 0 0 0 0 #intercept-only model
[2,] 1 2 0 0 0 0 0 #best 2 term model uses terms 1,2
[3,] 1 2 4 0 0 0 0 #best 3 term model uses terms 1,2,4
[4,] 1 2 6 9 0 0 0 #and so on...
```
rss.per.response
A vector of the RSS for each response. Length is the number of responses, i.e., ncol(y) after factors in y have been expanded. The rss component above is equal to sum(rss.per.response).

rsq.per.response
A vector of the R-Squared for each response (where R-Squared is calculated using the weights argument if it was supplied). Length is the number of responses.

gcv.per.response
A vector of the GCV for each response. Length is the number of responses. The gcv component above is equal to sum(gcv.per.response).

grsq.per.response
A vector of the GRSq for each response (calculated using the weights argument if it was supplied). Length is the number of responses.

rss.per.subset
A vector of the RSS for each model subset generated by the pruning pass. Length is nprune. For multiple responses, the RSS is summed over all responses for each subset. The rss above is rss.per.subset[length(selected.terms)]. The RSS of an intercept-only model is rss.per.subset[1].

gcv.per.subset
A vector of the GCV for each model in prune.terms. Length is nprune. For multiple responses, the GCV is summed over all responses for each subset. The gcv above is gcv.per.subset[length(selected.terms)]. The GCV of an intercept-only model is gcv.per.subset[1].

fitted.values
Fitted values. A matrix with dimensions nrow(y) x ncol(y) after factors in y have been expanded.

residuals
Residuals. A matrix with dimensions nrow(y) x ncol(y) after factors in y have been expanded.

coefficients
Regression coefficients. A matrix with dimensions length(selected.terms) x ncol(y) after factors in y have been expanded. Each column holds the least squares coefficients from regressing that column of y on bx. The first row holds the intercept coefficient(s).

leverages
Diagonal of the hat matrix (from the linear regression of the response on bx).

penalty,nk,thresh
Copies of corresponding arguments to earth.

weights,wp
Copies of corresponding arguments to earth.

termcond
Reason the forward pass terminated (an integer).

call
The call used to invoke earth.

terms
Model frame terms. This component exists only if the model was built using earth.formula.

namesx
Column names of x, generated internally by earth when necessary so each column of x has a name. Used, for example, by predict.earth to name columns if necessary.

namesx.org
Original column names of x.
levels

Levels of \( y \) if \( y \) is a \texttt{factor}
c(\texttt{FALSE}, \texttt{TRUE}) if \( y \) is \texttt{logical}
Else \texttt{NULL}

The following fields appear only if \texttt{earth}'s argument \texttt{keepxy} is \texttt{TRUE}.

\texttt{x,y,data,subset}

Copies of the corresponding arguments to \texttt{earth}. Only exist if \texttt{keepxy=TRUE}.

The following fields appear only if \texttt{earth}'s \texttt{glm} argument is used.

\texttt{glm.list}

List of GLM models. Each element is the value returned by \texttt{earth}'s internal call to \texttt{glm} for each response.
Thus if there is a single response (or a single binomial pair, see "Binomial pairs" in the package vignette) this will be a one element list and you access the GLM model with \texttt{earth.mod$glm.list[[1]]}.

\texttt{glm.coefficients}

GLM regression coefficients. Analogous to the \texttt{coefficients} field described above but for the GLM model(s). A matrix with dimensions \texttt{length(selected.terms)} x \texttt{ncol(y)} after factors in \( y \) have been expanded. Each column holds the coefficients from the GLM regression of that column of \( y \) on \( bx \). This duplicates, for convenience, information buried in \texttt{glm.list}.

\texttt{glm.bpairs}

NULL unless there are paired binomial columns. A logical vector, derived internally by \texttt{earth}, or a copy the \texttt{bpairs} specified by the user in the \texttt{glm} list. See "Binomial pairs" in the package vignette.

The following fields appear only if the \texttt{nfold} argument is greater than 1.

\texttt{cv.list}

List of \texttt{earth} models, one model for each fold (\texttt{ncross} * \texttt{nfold} models).
The fold models have two extra fields, \texttt{icross} (an integer from 1 to \texttt{ncross}) and \texttt{ifold} (an integer from 1 to \texttt{nfold}).
To save memory, lengthy fields in the fold models are removed unless you use \texttt{keepxy=TRUE}. The "lengthy fields" are \$bx, \$fitted.values, and \$residuals.

\texttt{cv.nterms}

Vector of length \texttt{ncross} * \texttt{nfold} + 1. Number of MARS terms in the model generated at each cross-validation fold, with the final element being the mean of these.

\texttt{cv.nvars}

Vector of length \texttt{ncross} * \texttt{nfold} + 1. Number of predictors in the model generated at each cross-validation fold, with the final element being the mean of these.

\texttt{cv.groups}

Specifies which cases went into which folds. Matrix with two columns and number of rows equal to the the number of cases \texttt{nrow(x)} Elements of the first column specify the cross-validation number, \( 1: \texttt{ncross} \). Elements of the second column specify the fold number, \( 1: \texttt{nfold} \).

\texttt{cv.rsq.tab}

Matrix with \texttt{ncross} * \texttt{nfold} + 1 rows and \texttt{nresponse+1} columns, where \texttt{nresponse} is the number of responses, i.e., \texttt{ncol(y)} after factors in \( y \) have been expanded. The first \texttt{nresponse} elements of a row are the \texttt{cv.rsq}'s on the out-of-fold data for each response of the model generated at that row’s fold. (A \texttt{cv.rsq} is calculated from predictions on the out-of-fold data using the best model built from the in-fold data; where “best” means the model was selected
using the in-fold GCV. The R-Squareds are calculated using the weights argument if it was supplied. The final column holds the row mean (a weighted mean if wp if specified)). The final row holds the column means. The values in this final row is the mean cv.rsq printed by summary.earth.

Example for a single response model (where the mean column is redundant but included for uniformity with multiple response models):

<table>
<thead>
<tr>
<th>y</th>
<th>mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>fold1</td>
<td>0.909 0.909</td>
</tr>
<tr>
<td>fold2</td>
<td>0.869 0.869</td>
</tr>
<tr>
<td>fold3</td>
<td>0.952 0.952</td>
</tr>
<tr>
<td>fold4</td>
<td>0.157 0.157</td>
</tr>
<tr>
<td>fold5</td>
<td>0.961 0.961</td>
</tr>
<tr>
<td>mean</td>
<td>0.769 0.769</td>
</tr>
</tbody>
</table>

Example for a multiple response model:

<table>
<thead>
<tr>
<th>y1</th>
<th>y2</th>
<th>y3</th>
<th>mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>fold1</td>
<td>0.915 0.951 0.944 0.937</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fold2</td>
<td>0.962 0.970 0.970 0.968</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fold3</td>
<td>0.914 0.940 0.942 0.932</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fold4</td>
<td>0.907 0.929 0.925 0.920</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fold5</td>
<td>0.947 0.987 0.979 0.971</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mean</td>
<td>0.929 0.955 0.952 0.946</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

cv.oof.rsq.tab

Generated only if keepxy=TRUE. A matrix with ncross * nfold + 1 rows and max.nterms columns. Each element holds an out-of-fold RSq (oof.rsq), calculated from predictions from the out-of-fold observations using the model built with the in-fold data. The final row is the mean over all folds. The R-Squareds are calculated using the weights argument if it was supplied.

cv.infold.rsq.tab

Generated only if keepxy=TRUE. Like cv.oof.rsq.tab but from predictions made on the in-fold observations.

cv.class.rate.tab

Like cv.rsq.tab but is the classification rate at each fold i.e. the fraction of classes correctly predicted. Models with discrete response only. Calculated with thresh=.5 for binary responses. For responses with more than two levels, the final row is the overall classification rate. The other rows are the classification rates for each level (the level versus versus not-the-level), which are usually higher than the overall classification rate (predicting the level versus not-the-level is easier than correctly predicting one of many levels). The weights argument is ignored for all cross-validation stats except R-Squareds.

cv.maxerr.tab

Like cv.rsq.tab but is the MaxErr at each fold. This is the signed max absolute value at each fold. Results are aggregated for the final column and final row using the signed max absolute value. The signed max absolute value is defined as the maximum of the absolute difference between the predicted and observed response values, multiplied by -1 if the sign of that difference is negative.

cv.auc.tab

Like cv.rsq.tab but is the AUC at each fold. Binomial models only.
\text{cv.cor.tab} \quad \text{Like } \text{cv.rsq.tab} \text{ but is the cor at each fold. Poisson models only.}

\text{cv.deviance.tab} \quad \text{Like } \text{cv.rsq.tab} \text{ but is the MeanDev at each fold. Binomial models only.}

\text{cv.calib.int.tab} \quad \text{Like } \text{cv.rsq.tab} \text{ but is the CalibInt at each fold. Binomial models only.}

\text{cv.calib.slope.tab} \quad \text{Like } \text{cv.rsq.tab} \text{ but is the CalibSlope at each fold. Binomial models only.}

\text{cv.oof.fit.tab} \quad \text{Only appears if the varmod.method argument is used. Predicted values on the out-of-fold data. Dataframe with } \text{nrow(data)} \text{ rows and } \text{ncross} \text{ columns.}

\text{The following field appears only if the varmod.method is specified.}

\text{varmod} \quad \text{An object of class "varmod". See the varmod help page for a description. Only appears if the varmod.method argument is used.}

\textbf{Author(s)}

Stephen Milborrow, derived from \texttt{mda::mars} by Trevor Hastie and Robert Tibshirani.

The approach used for GLMs was motivated by work done by Jane Elith and John Leathwick (a representative paper is given below).

The \texttt{evimp} function uses ideas from Max Kuhn’s \texttt{caret} package \url{http://cran.r-project.org/web/packages/caret/index.html}.

Parts of Thomas Lumley’s \texttt{leaps} package have been incorporated into \texttt{earth}, so \texttt{earth} can directly access Alan Miller’s Fortran functions without going through hidden functions in the \texttt{leaps} package.

\textbf{References}

The primary references are the Friedman papers. Readers may find the MARS section in Hastie, Tibshirani, and Friedman a more accessible introduction. The Wikipedia article is recommended for an elementary introduction. Faraway takes a hands-on approach, using the \texttt{ozone} data to compare \texttt{mda::mars} with other techniques. (If you use Faraway’s examples with \texttt{earth} instead of \texttt{mars}, use \texttt{Dbx} instead of \texttt{Dx}, and check out the book’s errata.) Friedman and Silverman is recommended background reading for the MARS paper. Earth’s pruning pass uses code from the \texttt{leaps} package which is based on techniques in Miller.

Faraway (2005) \textit{Extending the Linear Model with R} \url{http://www.maths.bath.ac.uk/~jjf23}


Hastie, Tibshirani, and Friedman (2009) \textit{The Elements of Statistical Learning} (2nd ed.) \url{http://www-stat.stanford.edu/~hastie/pub.htm}


See Also

Start with `summary.earth`, `plot.earth`, `evimp`, and `plotmo`.

Please see the main package vignette “Notes on the earth package”. The vignette can also be downloaded from [http://www.milbo.org/doc/earth-notes.pdf](http://www.milbo.org/doc/earth-notes.pdf).

The vignette “Variance models in earth” is also included with the package. It describes how to build variance models and generate prediction intervals for `earth` models.

Examples

```r
earth.mod <- earth(Volume ~ ., data = trees)
plotmo(earth.mod)
summary(earth.mod, digits = 2, style = "pmax")
```

---

**etitanic**

*Titanic data with incomplete cases removed*

---

**Description**

Titanic data with incomplete cases, passenger names, and other details removed.

**Format**

A data frame with 1046 observations on 6 variables.

- **pclass**: passenger class, unordered factor: 1st 2nd 3rd
- **survived**: integer: 0 or 1
- **sex**: unordered factor: male female
- **age**: age in years, min 0.167 max 80.0
- **sibsp**: number of siblings or spouses aboard, integer: 0...8
- **parch**: number of parents or children aboard, integer: 0...6

**Source**

This dataset is included in the earth package because it is a convenient vehicle for illustrating earth’s GLM and factor handling.
The dataset was compiled by Frank Harrell and Robert Dawson: [http://biostat.mc.vanderbilt.edu/twiki/pub/Main/DataSets/titanic.html](http://biostat.mc.vanderbilt.edu/twiki/pub/Main/DataSets/titanic.html)

See also: [http://biostat.mc.vanderbilt.edu/twiki/pub/Main/DataSets/titanic3info.txt](http://biostat.mc.vanderbilt.edu/twiki/pub/Main/DataSets/titanic3info.txt).

For this version of the Titanic data, passenger details and incomplete cases were deleted and the name changed to `etitanic` to minimize confusion with other versions ("e" because it is part of the `earth` package).

Note that `survived` is an integer (it should arguably be a logical).

In this data the crew are conspicuous by their absence.

Contents of `etitanic`:

```
  pclass survived sex  age sibsp  parch
1    1st      1 female 29.000   0   0
2    1st      1  male  0.917    1   2
3    1st      0 female 2.000   1   2
4    1st      0  male 30.000   1   2
5    1st      0 female 25.000  1   2
...```

How `etitanic` was built:

```r
load("titanic3") # from Harrell's web site
# discard name, ticket, fare, cabin, embarked, body, home.dest
etitanic <- titanic[,c(1,2,4,5,6,7)]
etitanic <- etitanic[!is.na(etitanic$age),]
save(etitanic, file="etitanic.rda")
```

References

Further details and analyses of the Titanic data may be found in:

F. Harrell (2001) *Regression Modeling Strategies with Applications to Linear Models, Logistic Regression, and Survival Analysis* [http://biostat.mc.vanderbilt.edu/twiki/bin/view/Main/RmS](http://biostat.mc.vanderbilt.edu/twiki/bin/view/Main/RmS)

See Also

`earth`
**Description**

Estimate variable importances in an earth object

**Usage**

```
evimp(obj, trim=TRUE, sqrt.=TRUE)
```

**Arguments**

- **obj**: An earth object.
- **trim**: If TRUE (default), delete rows in the returned matrix for variables that don’t appear in any subsets.
- **sqrt.**: Default is TRUE, meaning take the sqrt of the GCV and RSS importances before normalizing to 0 to 100. Taking the square root gives a better indication of relative importances because the raw importances are calculated using a sum of squares. Use FALSE to not take the square root.

**Value**

This function returns a matrix showing the relative importances of the variables in the model. There is a row for each variable. The row name is the variable name, but with -unused appended if the variable does not appear in the final model.

The columns of the matrix are (not all of these are printed by `print.evimp`):

- **col**: Column index of the variable in the x argument to earth.
- **used**: 1 if the variable is used in the final model, else 0. Equivalently, 0 if the row name has an -unused suffix.
- **nsubsets**: Variable importance using the "number of subsets" criterion. Is the number of subsets that include the variable (see "Three Criteria" in the chapter on evimp in the earth vignette "Notes on the earth package").
- **gcv**: Variable importance using the GCV criterion (see "Three Criteria").
- **gcv.match**: 1, except is 0 where the rank using the gcv criterion differs from that using the nsubsets criterion. In other words, there is a 0 for values that increase as you go down the gcv column.
- **rss**: Variable importance using the RSS criterion (see "Three Criteria").
- **rss.match**: Like gcv.match but for the rss.

The rows are sorted on the nsubsets criterion. This means that values in the nsubsets column decrease as you go down the column (more accurately, they are non-increasing). The values in the gcv and rss columns are also non-increasing, except where the gcv or rss rank differs from the nsubsets ranking.
Note
There is a chapter on evimp in the earth package vignette “Notes on the earth package”.

Acknowledgment
Thanks to Max Kuhn for the original evimp code and for helpful discussions.

See Also
earth, plot.evimp

Examples
```r
data(ozone1)
earth.mod <- earth(O3 ~ ., data=ozone1, degree=2)
ev <- evimp(earth.mod, trim=FALSE)
plot(ev)
print(ev)
```

format.earth

Format earth objects

Description
Return a string representing an earth expression.

Usage
```r
## S3 method for class 'earth'
format(x = stop("no 'x' arg"),
   style = "h", decomp = "anova", digits = getOption("digits"),
   use.names = TRUE, colon.char = "\n", ...)
```

Arguments
- **x**: An earth object. This is the only required argument.
- **style**: Formatting style. One of "h" (default) more compact "pmax" for those who prefer it and for compatibility with old versions of earth "max" is the same as "pmax" but prints max rather than pmax "C" C style expression with zero based indexing "bf" basis function format
- **decomp**: One of "anova" (default) order the terms using the "anova decomposition", i.e., in increasing order of interaction "none" order the terms as created during the earth forward pass.
- **digits**: Number of significant digits. The default is getOption(digits).
use.names One of
TRUE (default), use variable names if available.
FALSE use names of the form x[1].
colon.char Change colons in the returned string to colon.char. Default is ";" (no change).
Specifying colon.char="*" can be useful in some contexts to change names of the form x1:x2 to x1*x2.

... Unused, but provided for generic/method consistency.

Value
A character representation of the earth model.
If there are multiple responses, format.earth will return multiple strings.
If there are embedded GLM model(s), the strings for the GLM model(s) come after the strings for the standard earth model(s).

Note
The FAQ section in the package vignette gives precise details of the "anova" ordering.
Using format.earth, perhaps after hand editing the returned string, you can create an alternative to predict.earth. For example:

```r
as.func <- function(object, digits = 8, use.names = FALSE, ...)
eval(parse(text=paste(
  "function(x)\n",
  "if(is.vector(x))\n",
  "  x <- matrix(x, nrow = 1, ncol = length(x))\n",
  "  with(as.data.frame(x),\n",
  "    format(object, digits = digits, use.names = use.names, style = "pmax", ...),
  "      })\n",
  "    })\n",
  "    sep = "\"")
))
earth.mod <- earth(Volume ~ ., data = trees)
my.func <- as.func(earth.mod, use.names = FALSE)
my.func(c(10,80)) # returns 16.84
predict(earth.mod, c(10,80)) # returns 16.84
```

Note that with pmax the R expression generated by format.earth can handle multiple cases. Thus the expression is consistent with the way predict functions usually work in R — we can give predict multiple cases (i.e., multiple rows in the input matrix) and it will return a vector of predicted values.

The earth package also provides a function format.lm. It has arguments as follows
format.lm(x, digits=getOption("digits"), use.names=TRUE, colon.char=": ")
(Strictly speaking, format.lm doesn't belong in the earth package.) Example:
format.earth

```r
lm.mod <- lm(Volume ~ Height*Girth, data = trees)
cat(format(lm.mod, coln.char="\n"))

# yields:
#  69.4
#  -  1.30 * Height
#  -  5.86 * Girth
#  +  0.135 * Height*Girth

See Also

earth, pmax

Examples

earth.mod <- earth(Volume ~ ., data = trees)
cat(format(earth.mod))

# yields:
#  37.9
#  -  3.92  * h(16-Girth)
#  +  7.4   * h(Girth-16)
#  +  0.484 * h(Height-75)
cat(format(earth.mod, style="pmax")) # default formatting style prior to earth version 1.4

# yields:  
#  37.9
#  -  3.92  * pmax(0, 16 - Girth)
#  +  7.4   * pmax(0, Girth - 16)
#  +  0.484 * pmax(0, Height - 75)
cat(format(earth.mod, style="C"))

# yields (note zero based indexing):
#  37.927
#  -  3.9187 * max(0, 16 - x[0])
#  +  7.4011 * max(0, x[0] - 16)
#  +  0.48411 * max(0, x[1] - 75)
cat(format(earth.mod, style="bf"))

# yields:
#  37.9
#  -  3.92  * bf1
#  +  7.4   * bf2
#  +  0.484 * bf3
#  
#  bf1  h(16-Girth)
#  bf2  h(Girth-16)
#  bf3  h(Height-75)
```
Convert a mars object from the mda package to an earth object

Description

Convert a mars object from the mda package to an earth object

Usage

mars.to.earth(object, trace=TRUE)

Arguments

object A mars object, created using mars in the mda package.
trace If TRUE (default) print a summary of the conversion.

Value

The value is the same format as that returned by earth but with skeletal versions of rss.per.subset, gcv.per.subset, and prune.terms.

You can fully initialize these components by calling update.earth after mars.to.earth, but if you do this selected.terms may change. However with pmethod="backward" a change is unlikely — selected.terms would change only if GCVs are so close that numerical errors have an effect.

Note

Differences between mars and earth objects

Perhaps the most notable difference between mars and earth objects is that mars returns the MARS basis matrix in a field called "x" whereas earth returns "bx" with only the selected terms. Also, earth returns "dirs" rather than "factors", and in earth this matrix can have entries of value 2 for linear predictors.

For details of other differences between mars and earth objects, see the comments in the source code of mars.to.earth.

Weights

The w argument is silently ignored by mars.
mars normalizes wp to (euclidean) length 1; earth normalizes wp to length equal to the number of responses, i.e., the number of columns in y. This change was made so an all ones wp (or in fact any all constant wp) is equivalent to using no wp.

If the original call to mars used the wp argument, mars.to.earth will run update.earth to force consistency. This could modify the model, so a warning is issued.

See Also

earth, mars
Examples

```r
if(require(mda)) {
  mars.mod <- mars(trees[,3], trees[,3])
  earth.mod <- mars.to.earth(mars.mod)
  # the standard earth functions can now be used
  # note the reconstructed call in the summary
  summary(earth.mod, digits = 2)
}
```

---

### model.matrix.earth

Get the earth basis matrix

**Description**

Get the basis matrix of an `earth` object.

**Usage**

```r
## S3 method for class 'earth'
model.matrix(object = stop("no 'object' arg"),
             x = NULL, subset = NULL, which.terms = NULL,
             ..., 
             env = parent.frame(),
             trace = 0,
             Callers.name = "model.matrix.earth")
```

**Arguments**

- **object**: An `earth` object. This is the only required argument.
- **x**: An input matrix with the same number of columns as the `x` matrix used to construct the original `earth` object. Default is `NULL`, meaning use the original `x` matrix after taking the original subset, if any.
- **subset**: Which rows to use in `x`. Default is `NULL`, meaning use all of `x`.
- **which.terms**: Which terms to use. Default is `NULL`, meaning use `object$selected.terms`.
- **...**: Unused, but provided for generic/method consistency.
- **env**: For internal use.
- **trace**: Default 0. Set to non-zero to see which data `model.matrix.earth` is using.
- **Callers.name**: For internal use (used by `earth` in trace messages).
Value

A basis matrix bx of the same form returned by earth.

If x, subset, and which.terms are all NULL, this function returns the object’s bx. In this case, it is perhaps easier to simply use object$bx.

The format of bx is described in earth. The matrix bx can be used as the input matrix to lm or glm, as shown below in the example. In fact, that is what earth does internally after the pruning pass — it calls lm.fit, and additionally glm if earth’s glm argument is used.

See Also

earth

Examples

data(trees)
earth.mod <- earth(Volume ~ ., data = trees)
summary(earth.mod, decomp = "none") # "none" to print terms in same seq as lm.mod below

bx <- model.matrix(earth.mod) # equivalent to bx <- earth.mod$bx
lm.mod <- lm(trees$Volume ~ bx[-1]) # -1 to drop intercept
summary(lm.mod) # yields same coefds as above summary

# displayed t values are not meaningful

---

ozone1 Ozone readings in Los Angeles with incomplete cases removed

Description

Ozone readings in Los Angeles, with incomplete cases removed.

Format

A data frame with 330 observations on 10 variables.

- **03**: daily maximum of the hourly average ozone concentrations in Upland, CA
- **vh**: 500 millibar pressure height, measured at the Vandenberg air force base
- **wind**: wind speed in mph at LAX airport
- **humidity**: humidity in percent at LAX
- **temp**: Sandburg Air Force Base temperature in degrees Fahrenheit
- **ibh**: temperature inversion base height in feet
- **dpg**: pressure gradient from LAX to Daggert in mm Hg
- **ibt**: inversion base temperature at LAX in degrees Fahrenheit
- **vis**: visibility at LAX in miles
- **doy**: day of the year
plot.earth

Source

This data was copied from library(faraway) and the name changed to ozone1 to prevent a name clash. The data were originally made available by Leo Breiman who was a consultant on a project where the data were generated. Example analyses using these data may be found in Faraway and in Hastie and Tibshirani.

```r
> ozone1
     O3  vh wind humidity temp ibh dpg ibt vis doy
1  3 5710  4  28  40 2693 -25  87  250  33
2  5 5700  3  37  45 590  -24 128  100  34
3  5 5760  3  51  54 1450  25  139  60  35
   ...  
330 1 5550  4  85  39 5000  8  44 100 390
```

References

Faraway (2005) *Extending the Linear Model with R* [http://www.maths.bath.ac.uk/~jjf23](http://www.maths.bath.ac.uk/~jjf23)


See Also

earth

---

**plot.earth**  
*Plot an earth object*

---

**Description**

Plot an earth object. By default the plot shows model selection, cumulative distribution of the residuals, residuals versus fitted values, and the residual QQ plot.

**Usage**

```r
## S3 method for class 'earth'
plot(x = stop("no \'x\' arg"), which = 1:4,
     info = FALSE, student = FALSE, delever = FALSE, level = 0, versus = 1,
     nresponse = 1, npoints = 1000, id.n = 3,
     labels.id = rownames(residuals(object, warn=FALSE)),
     center = TRUE, loess.f = .5,
     do.par = length(which) > 1, xlim = NULL, ylim = NULL,
     main = NULL, cex.main = 1.1, caption = if(do.par) NULL else "",
     xlab = NULL, ylab = NULL,
     pch = 20, col.rsq = "red",
     col.loess = "red", lwd.loess = 1, col.cv = "lightblue",
     col.qq = "gray", col.grid = "lightgray", col.points = 1, cex.points = NULL,
```
Arguments

There are numerous arguments, but most are merely for embellishing the plot.
The important arguments are listed first.

An `earth` object. This is the only required argument. (The argument is called "x" for consistency with the generic `plot`.)

`which` Which plots to plot. Default is 1:4.
1 Model selection (GRSq vs number of terms)
2 Cumulative distribution of abs residuals
3 Residuals vs fitted
4 QQ plot of residuals
5 Abs residuals vs fitted
6 Sqrt abs residuals vs fitted (Scale-Location plot)
7 Abs residuals vs log fitted
8 Cube root of the squared residuals vs log fitted
9 Log abs residuals vs log fitted

`info` Print some additional information in the residuals plots. Default is FALSE.
i) Plot the distribution of the residuals along the bottom of the plot.
ii) Show the Spearman Rank Correlation of the absolute residuals with the fitted values. Actually, correlation is measured against the absolute values of whatever is on the horizontal axis — by default this is the fitted response, but may be something else if the versus argument is used.
iii) Only for which=5 or 9. Regress the absolute residuals against the fitted values and show the regression slope. Robust linear regression is used via `rlm` in the MASS package.

`student` Default is FALSE. Use TRUE to divide each residual by $se \times \sqrt{1 - h_{ii}}$, where $se$ is the standard error of the prediction and $h_{ii}$ is described below.
Applies to all plots where the residuals are used (including the cumulative distribution and QQ plots, and to statistics calculated for the info argument). Requires that the model was built with the varmod.method argument, because we need the variance model to get the standard errors.
When the variance model holds, the studentized residuals are homoscedastic with unity variance.
delever

Default is FALSE. Use TRUE to divide the residuals by \( \sqrt{1 - h_{ii}} \), where \( h_{ii} \) are the diagonal entries of the hat matrix. The hat matrix here is from the \texttt{lm} fit on the earth basis matrix \( bx \).

Applies to all plots where the residuals are used (including the cumulative distribution and QQ plots, and to statistics calculated for the \texttt{info} argument).

level

Plot prediction intervals at the given confidence level. Default is 0, meaning do not plot the intervals. A typical value is level=.95.

Requires that the model was built with the \texttt{varmod.method} argument.

The color of the bands is determined by the arguments \texttt{shade.pints} and \texttt{shade.cints}.

versus

What do we plot the residuals against? One of:

1 Default. Plot the residuals versus the fitted values (or the log values when \( \texttt{which} = \texttt{7 to 9} \)).

2 Residuals versus observation number, after observations have been sorted on the fitted value. Same as \texttt{versus=1}, except that the residuals are spaced uniformly along the horizontal axis.

3 Residuals versus the response.

4 Residuals versus the hat matrix leverages (from the linear regression of the response on \( bx \)).

"*" Residuals versus the earth terms. A optional regex can follow the "*" to specify a subset of the terms, e.g. \texttt{versus="*wind"} will plot terms with "wind" in their name.

Else a character vector specifying which predictors to plot against.

Example 1: \texttt{versus=""} plots against all predictors (since the regex \texttt{versus=""} matches anything).

Example 2: \texttt{versus=c("wind", "vis")}.

Note: \texttt{grep} is used for matching. Thus \texttt{versus="wind"} will match all variables that have "wind" in their names. Use ""\$\" to match only the variable named "wind".

nresponse

Specify which column of the response to plot if the model has multiple responses. Default is 1. This argument does not affect the Model Selection plot which is always across all responses.

npoints

Maximum number of points to display in residual and other plots. Use -1 for all. Default is 10000 (not all, to reduce over-plotting). A systematic sample of size \texttt{nresiduals} is taken but the biggest few residuals are always included.

id.n

The largest \texttt{id.n} residuals will be labeled in the plot. Default is 3.

labels.id

Residual names. Default is \texttt{rownames(residuals(\texttt{xx}))}. Only used if \texttt{id.n > 0}.

center

Default is TRUE, meaning center the horizontal axis in the residuals plot, so asymmetry in the residual distribution is more obvious.
loess.f  Smoothing parameter for the red smooth line in the residuals plot. Default .5. Lower values make the line bumpier. This argument is passed as f in the internal call to \texttt{lowess}.

do.par  If \texttt{TRUE}, start a new page and call \texttt{par} as appropriate. Default is \texttt{length(which) > 1}, i.e., call \texttt{par} if drawing more than one plot. Use \texttt{FALSE} to use the current graphics settings, for example if you want to add plots to an existing multiframe page. The value \texttt{2} means act like \texttt{TRUE} but do not restore the \texttt{par} settings to their original state when \texttt{plot.earth} exits. This is useful if you want to add a few more plots on the same page after running \texttt{plot.earth}.

xlim  Horizontal axis limits. The default is \texttt{NULL}, meaning automatic.

ylim  Vertical axis limits. The default is \texttt{NULL}, meaning automatic.

If \texttt{which} has more than one element, this argument applies only to the Model Selection plot. In the model selection plot, the special value \texttt{min}=-1 means the minimum y axis value is the smallest GRSq or RSq value excluding the intercept values. The special value \texttt{max}=-1 means the maximum y axis value is the largest GRSq or RSq value.

main  Heading of each plot. Default is \texttt{NULL}, meaning generate the headings automatically.

cex.main  Size of plot titles. Default is 1.1. Used only if \texttt{do.par} is \texttt{TRUE} (default).

caption  Overall caption. The default value is \texttt{if(do.par) NULL else ""}. One of: \texttt{"string" string} \texttt{"" no caption} \texttt{NULL} generate a caption automatically.

xlab  Default is \texttt{NULL}, meaning label the x axis automatically.

ylab  Default is \texttt{NULL}, meaning label the y axis automatically.

pch  Default is 20.

col.rsq  Color of the RSq line in the Model Selection plot. Default is "red" Use \texttt{} for no RSq line.

col.loess  Default "red". Color of \texttt{loess} line in the residuals plot. (Actually \texttt{lowess} is used rather than \texttt{loess}, to avoid internal warnings issued by \texttt{loess}. The arguments is so named for backward compatibility.)

lwd.loess  Default 1. Line width of the \texttt{loess} line.

col.cv  Default is "lightblue". Color of cross validation line in the residuals plot. This is the residual of the mean out-fold-predicted value.

col.qq  Color of QQ line in the QQ plot. Default is "gray". Use \texttt{} for no QQ line.

col.grid  Color of grid lines in the various plots. Default is "lightgray". Use \texttt{} for no grid.

col.points  Color of plotted points. Default is 1

cex.points  Size of the plotted points. Default is \texttt{NULL}, meaning automatic.

shade.pints  Only used if the \texttt{level} argument was used. Color of the prediction intervals. Default is "mistyrose2"
shade.cints Only used if the level argument was used. Color of the confidence intervals. Default is "mistyrose4"

cum.grid Specify grid type in the Cumulative Distribution plot. One of:
"none" no grid on Cumulative Distribution plot
"grid" add grid
"percentages" (default) add grid and percentage labels to quantile lines.

col.line Deprecated, please use col.rsq instead. Default is NA.
col.residuals Deprecated, please use col.points instead. Default is NA.
nresiduals Deprecated, please use npoints instead. Default is NA.

The following are for the Model Selection plot.

legend.pos Legend position. Default is NULL, meaning automatic. Else specify c(x,y) in user coordinates, or use "topleft" etc. as explained in legend.
cex.legend Legend cex. Default is NULL, meaning automatic.
col.grsq Color of GRSq line in the Model Selection plot. Default is 1. Use 0 for no GRSq line.
col.infold.rsq Color of in-fold RSq lines for each fold in the Model Selection plot. Applies only if nfold and keepxy were used in the original call to earth. Default is 0, lines not plotted.
col.mean.infold.rsq Color of mean in-fold RSq for each number of terms in the Model Selection plot. Default is 0, line not plotted. Applies only if nfold and keepxy were used in the original call to earth.
col.mean.oof.rsq Color of mean out-of-fold RSq for each number of terms in the Model Selection plot. Applies only if nfold and keepxy were used in the original call to earth. Default is "palevioletred". Use 0 to not plot this line.
col.npreds Color of the "number of predictors" plot in the Model Selection plot. The default displays the number of predictors unless the oof.rsq's are displayed. Use 0 for no "number of predictors" plot.
col.oof.labs Color of fold number labels on the oof.rsq lines. Default is 0, no labels.
col.oof.rsq Color of out-of-fold RSq lines for each fold in the Model Selection plot. Applies only if nfold and keepxy were used in the original call to earth. Default is "mistyrose2", a pale pink. Use 0 to not plot these lines. May be a vector of colors, which will be recycled if necessary.
col.oof.vline Color of vertical line at the maximum oof.rsq in the Model Selection plot. Default is col.mean.oof.rsq.
col.pch.cv.rsq Color of point plotted on the oof.rsq line to indicate the cv.rsq for that fold (i.e., it is plotted at the number of terms selected by the in-fold GCV). Default is 0, point not plotted.
col.pch.max.oof.rsq Color of point plotted on the oof.rsq line to indicate the maximum oof.rsq for that fold. Default is 0, point not plotted.
### plot.earth.models

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>col.sel.grid</td>
<td>Color of grid lines in the Model Selection graph. Default is 0, no grid. Try something like &quot;lightgray&quot;, &quot;linen&quot;, or &quot;seashell&quot;. See also col.cum.grid, for the grid in the Cumulative Distribution plot. (TODO Is this argument now subsumed by col.grid?)</td>
</tr>
<tr>
<td>col.vline</td>
<td>Color of the vertical line at selected model in the Model Selection plot. Default is col.grsq. This will be at the maximum GRSq unless pmethod=&quot;none&quot;. Use 0 for no vertical line.</td>
</tr>
<tr>
<td>col.vseg</td>
<td>Default is 0. Color of triangular marker at top of vertical line for best GRSq.</td>
</tr>
<tr>
<td>lty.grsq</td>
<td>Line type of GRSq line in the Model Selection plot. Default is 1</td>
</tr>
<tr>
<td>lty.npreds</td>
<td>Line type of the &quot;number of predictors&quot; plot in the Model Selection plot. Default is 2.</td>
</tr>
<tr>
<td>lty.rsq</td>
<td>Line type of RSq line in the Model Selection plot. Default is 5.</td>
</tr>
<tr>
<td>lty.vline</td>
<td>Line type of vertical line at selected model in the Model Selection plot. Default is 3.</td>
</tr>
<tr>
<td>col.legend</td>
<td>Legend color. Default is 1. Use 0 for no legend.</td>
</tr>
</tbody>
</table>

**Note**

For details on interpreting the graphs, please see the earth package vignettes “Notes on the earth package” and “Variance models in earth”.

Note that cross-validation data will not be displayed unless both nfold and keepxy were used in the original call to earth.

**See Also**

earth, plot.earth.models, plotd, plotmo

**Examples**

```r
data(ozone1)
earth.mod <- earth(O3 ~ ., data = ozone1, degree = 2)
plot(earth.mod)
```

**plot.earth.models**  
*Compare earth models by plotting them.*

**Description**

Compare earth models by plotting them.
plot.earth.models

Usage

## S3 method for class 'earth.models'
plot(x = stop("no 'x' arg"), which = c(1:2),
caption = "", jitter = 0,
col.grsq = discrete.plot.cols(length(x)), lty.grsq = 1,
col.rsq = 0, lty.rsq = 5,
col.vline = col.grsq, lty.vline = 3,
col.npreds = 0, lty.npreds = 2, col.sel.grid = 0,
ylim = c(0,1),
col.legend = 1, cex.legend = NULL, legend.pos = NULL, legend.text = NULL,
col.cum = NULL, do.par = TRUE,
main = "Model Comparison", cex.main = 1.1, ...)

Arguments

x A list of one or more earth objects, or a single earth object. This is the only required argument. (This argument is called 'x' for consistency with the generic plot.)

which Which plots to plot: 1 model, 2 cumulative distribution of residuals. Default is 1:2, meaning both.
caption Overall caption. Values:
"string" string
"" (default) no caption
NULL generate a caption from the $call component of the earth objects.
jitter Jitter applied to GRSq and RSq values to minimize over-plotting. Default is 0, meaning no jitter. A typical useful value is 0.01.

txt For the col arguments below, 0 means do not plot the corresponding graph element. You can use vectors of colors.

col.grsq Vector of colors for the GRSq plot. The default is discrete.plot.cols(length(x)) which is vector of distinguishable colors, the first three of which are also distinguishable on a monochrome printer. You can examine the colors using earth:::discrete.plot.cols().
lty.grsq Line type for the GRSq plot. Default is 1.
col.rsq Vector of colors for the RSq plot. Default is 0, meaning no RSq plot.
lty.rsq Line type for the RSq plot. Default is 5.
col.vline A vertical line is drawn for each object to show which model size was chosen for that object. The color of the line is col.vline. Default is col.grsq.
lty.vline Line type of vertical lines (a vertical line is drawn to show the selected model for each object). Can be a vector. Default is 3.
col.npreds Vector of colors for the "number of predictors" plot within the model selection plot. Default is 0, meaning no "number of predictors" plot. The special value NULL means borrow col.grsq (or col.line if col.grsq is NULL).
lty.npreds Line type of the "number of predictors" plot (in the Model Selection plot). Default is 2.
plot.earth.models

col.sel.grid  Color of grid lines in the Model Selection graph. Default is 0, no grid. Try something like "lightgray", "linen", or "seashell".
ylim  Two element vector \( c(\min, \max) \) specifying min and max values on the y axis of the RSq/GRSq plot. Default is \( c(0, 1) \).
   The special value \( \min=-1 \) means the minimum y axis value is the smallest GRSq or RSq, excluding the intercept.
   The special value \( \max=-1 \) means the maximum y axis value is the largest GRSq or RSq.
col.legend  Deprecated. Use \( \text{legend.pos=NA} \) for no legend.
cex.legend  Legend cex. Default is NULL, meaning choose automatically.
legend.pos  Legend position. Default NULL, meaning position the legend automatically. Use NA for no legend. Specify \( c(x, y) \) in user coordinates, or use "topleft" etc. as explained in \text{legend}.
legend.text  Vector of strings to use as legend text. The special value NULL (default) means generate the legend text automatically from \text{callDformula}.
col.cum  Vector of colors for the cumulative distribution plot. The special value NULL (default) means borrow \text{col.grsq} (or \text{col.line} if \text{col.grsq} is NULL).

The following settings are related to \text{par()} and are included so you can override the defaults.
do.par  Call \text{par()} for global settings as appropriate. Default is TRUE, which sets \text{mfrow, mar=c(4,4,2,3)}, \text{mgp=c(1,0,0)}.
   Set to FALSE if you want to append figures to an existing plot.
main  Title of each plot. Default is NULL, meaning generate figure headings automatically.
cex.main  Size of plot titles. Default is 1.1. Used only if \text{do.par} is TRUE (default).
...

Unused, but provided for generic/method consistency.

Note

This function ignores GLM and cross-validation components of the earth model, if any.

See Also

\text{earth, plot.earth, plot.earth.models, plotd, plotmo}

Examples

data(ozone1)
a1 <- earth(O3 ~ ., data = ozone1, degree = 2)
a2 <- earth(O3 ~ .-wind, data = ozone1, degree = 2)
a3 <- earth(O3 ~ .-humidity, data = ozone1, degree = 2)
plot.earth.models(list(a1,a2,a3), ylim=c(.65,.85))
Description

Plot an evimp object.

Usage

```r
## S3 method for class 'evimp'
plot(x = stop("no 'x' arg"),
    cex.var = 1,
    type.nsubsets = "l", col.nsubsets = "black", lty.nsubsets = 1,
    type.gcv = "l", col.gcv = "red", lty.gcv = 1,
    type.rss = "l", col.rss = "gray60", lty.rss = 1,
    cex.legend = 1, x.legend = nrow(x), y.legend = x[1,"nsubsets"],
    main = "Variable importance",
    rh.col = 1, do.par = TRUE, ...)```

Arguments

- `x`: An evimp object.
- `cex.var`: cex for variable names. Default is 1. Make smaller (say 0.8) if you have lots of variables.
- `type.nsubsets`: Plot type for nsubsets graph. Default is "l". Use "n" for none, "b" looks good too.
- `col.nsubsets`: Color of nsubsets line. Default is "black".
- `lty.nsubsets`: Line type of nsubsets line. Default is 1.
- `type.gcv`, `col.gcv`, `lty.gcv`: As above but for the gcv plot
- `type.rss`, `col.rss`, `lty.rss`: As above but for the rss plot
- `cex.legend`: cex for legend strings. Default is 1. Make smaller (say 0.8) if you want a smaller legend.
- `x.legend`: x position of legend. Use 0 for no legend.
- `y.legend`: y position of legend.
- `main`: Main title. Default is "Variable importance".
- `rh.col`: Color of right hand axis label. Use rh.col=0 for no label, a workaround for when the label is mispositioned.
- `do.par`: Call par() for global settings as appropriate. Default is TRUE, which sets oma=c(bottom.marg,0,0,3), cex=cex.var. Set to FALSE if you want to append figures to an existing plot.
- ...: Extra arguments passed to plotting functions.
plot.varmod

See Also
earth, evimp, plot.earth.models.plotmo

Examples

data(ozone1)
earth.mod <- earth(O3 ~ ., data=ozone1, degree=2)
ev <- evimp(earth.mod)
plot(ev)
print(ev)

plot.varmod
Plot a varmod object

Description
Plot a variance model (a varmod object).
Typically you call this function for a variance model embedded in an earth model.

Usage
## S3 method for class 'varmod'
plot(x = stop("no 'x' arg"), which = 1:4,
do.par = length(which) > 1, info=FALSE,
cex = NULL, caption = if(do.par) NULL else "", main = NULL,
col.line = "red", col.min.sd = col.line, ...)

Arguments

x
A varmod object. Typically this is embedded in a parent earth object, and so
you invoke this function with plot(earth.mod$varmod). The varmod method
argument must have been specified when building the earth model.

which
Which plots to plot. Default is 1:4 meaning all. The term parent below refers to
the earth model in which the varmod is embedded.
1) fitted vs parent fitted
2) fitted vs parent first predictor
3) residuals vs fitted
4) model selection graph (only when varmod.method="earth" or "x.earth").

do.par
Call par() for global settings as appropriate. Default is TRUE, Set to FALSE if
you want to append figures to an existing plot.

info
Plot some additional information, including lowess fits in the first two plots.

cex
Character expansion.

caption
Default is NULL, meaning automatically generate an overall caption.

main
Default is NULL, meaning automatically generate a title for each graph.
col.line        Color of lines in the plots. Default is red.
col.min.sd     Color of the min.sd dotted horizontal line. Default is col.line. Use 0 to not plot this line.
...            Unused, but provided for generic/method consistency.

Note

The horizontal red dotted line in the first two plots shows the value of min.sd. See earth’s varmod.clamp argument.

See Also

varmod

Examples

data(ozone1)

set.seed(1) # optional, for cross validation reproducibility

# note: should really use ncross=30 below but for a quick demo we don't
earth.mod <- earth(O3~temp, data=ozone1, nfold=10, ncross=3, varmod.method="lm")

plot(earth.mod$varmod) # plot the embedded variance model (this calls plot.varmod)

---

plotd              Plot the distribution of predictions for each class

Description

Draw a plot of the distribution of the predicted values for each class. Can be used for earth models, but also for models built by lm, glm, lda, etc.

Usage

plotd(obj, hist = FALSE, type = NULL, nresponse = NULL, dichot = FALSE,
trace = FALSE, xlim = NULL, ylim = NULL, jitter = FALSE, main=NULL,
xlab = “Predicted Value”, ylab = if(hist) ”Count” else ”Density”,
lty = 1, col = c(“gray70”, “lightblue”, “brown”, “pink”, 2, 3, 4),
fill = if(hist) col[1] else 0,
breaks = “Sturges”, labels = FALSE,
kernol = “gaussian”, adjust = 1, zero.line = FALSE,
legend = TRUE, legend.names = NULL, legend.pos = NULL,
cex.legend = .8, legend.bg = ”white”, legend.extra = FALSE,
vline.col = 0, vline.thresh = .5, vline.lty = 1, vline.lwd = 1,
err.thresh = vline.thresh, err.col = 0, err.border = 0, err.lwd = 1,
xaxt = ”s”, yaxt = ”s”, xaxis.cex = 1, sd.thresh = 0.01, ...)
Arguments

To start off, look at the arguments `obj`, `hist`, `type`.

For predict methods with multiple column responses, see the `nresponse` argument.

For factor responses with more than two levels, see the `dichot` argument.

Model object. Typically a model which predicts a class or a class discriminant.

**`bbst`**

FALSE (default) to call `density` internally.

TRUE to call `hist` internally.

**`type`**

Type parameter passed to `predict`. Default is NULL, meaning automatically choose a value for the given object. (This is "response" for all objects except `rpart` models, where "vector" is used — these choices will often be inappropriate.) See the predict method for your object for legal values; for example see `predict.earth` or `predict.glm`. Typically you would set `hist=TRUE` when `type="class"`.

**`nresponse`**

Column index for predicted responses with multiple columns. The default is NULL, meaning use all columns of the predicted response.

**`dichot`**

Dichotimise the predicted response. This argument is ignored except for models where the observed response is a factor with more than two levels and the predicted response is a numeric vector. The default FALSE separates the response into a group for each factor. With `dichot=TRUE` the response is separated into just two groups: the first level of the factor versus the remaining levels.

**`trace`**

Default FALSE. Use TRUE or 1 to trace `plotd` — useful to see how `plotd` partitions the predicted response into classes. Use 2 for a full dump of the internal matrices.

**`xlim`**

Limits of the x axis. The default NULL means determine these limits automatically, else specify `c(xmin,xmax)`.

**`ylim`**

Limits of the y axis. The default NULL means determine these limits automatically, else specify `c(ymin,ymax)`.

**`jitter`**

Jitter the histograms or densities horizontally to minimize overplotting. Default FALSE. Specify TRUE to automatically calculate the jitter, else specify a numeric jitter value.

**`main`**

Main title. Values:

"string" string

"" no title

NULL (default) generate a title from the call.

**`xlab`**

x axis label. Default is "Predicted Value".

**`ylab`**

y axis label. Default is if(hist) "Count" else "Density".

**`lty`**

Per class line types for the plotted lines. Default is 1 (which gets recycled for all lines).

**`col`**

Per class line colors. The first few colors of the default are intended to be easily distinguishable on both color displays and monochrome printers.

**`fill`**

Fill color for the plot for the first class. For `hist=FALSE`, the default is 0, i.e., no fill. For `hist=TRUE`, the default is the first element in the `col` argument.
<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>breaks</td>
<td>Passed to <code>hist</code>. Only used if <code>hist=TRUE</code>. Default is &quot;Sturges&quot;. When <code>type=&quot;class&quot;</code>, setting <code>breaks</code> to a low number can be used to widen the histogram bars.</td>
</tr>
<tr>
<td>labels</td>
<td>TRUE to draw counts on the <code>hist</code> plot. Only used if <code>hist=TRUE</code>. Default is <code>FALSE</code>.</td>
</tr>
<tr>
<td>kernel</td>
<td>Passed to <code>density</code>. Only used if <code>hist=FALSE</code>. Default is &quot;gaussian&quot;.</td>
</tr>
<tr>
<td>adjust</td>
<td>Passed to <code>density</code>. Only used if <code>hist=FALSE</code>. Default is <code>1</code>.</td>
</tr>
<tr>
<td>zero.line</td>
<td>Passed to <code>plot.density</code>. Only used if <code>hist=FALSE</code>. Default is <code>FALSE</code>.</td>
</tr>
<tr>
<td>legend</td>
<td>TRUE (default) to draw a legend, else <code>FALSE</code>.</td>
</tr>
<tr>
<td>legend.names</td>
<td>Class names in legend. The default <code>NULL</code> means determine these automatically.</td>
</tr>
<tr>
<td>legend.pos</td>
<td>Position of the legend. The default <code>NULL</code> means position the legend automatically, else specify <code>c(x,y)</code>.</td>
</tr>
<tr>
<td>cex.legend</td>
<td><code>cex</code> for <code>legend</code>. Default is <code>.8</code>.</td>
</tr>
<tr>
<td>legend.bg</td>
<td>bg color for <code>legend</code>. Default is &quot;white&quot;.</td>
</tr>
<tr>
<td>legend.extra</td>
<td>Show (in the legend) the number of occurrences of each class. Default is <code>FALSE</code>.</td>
</tr>
<tr>
<td>vline.thresh</td>
<td>Horizontal position of optional vertical line. Default is <code>0.5</code>. The vertical line is intended to indicate class separation. If you use this, don’t forget to set <code>vline.col</code>.</td>
</tr>
<tr>
<td>vline.col</td>
<td>Color of vertical line. Default is <code>0</code>, meaning no vertical line.</td>
</tr>
<tr>
<td>vline.lty</td>
<td>Line type of vertical line. Default is <code>1</code>.</td>
</tr>
<tr>
<td>vline.lwd</td>
<td>Line width of vertical line. Default is <code>1</code>.</td>
</tr>
<tr>
<td>err.thresh</td>
<td>x axis value specifying the error shading threshold. See <code>err.col</code>. Default is <code>vline.thresh</code>.</td>
</tr>
<tr>
<td>err.col</td>
<td>Specify up to three colors to shade the &quot;error areas&quot; of the density plot. The default is <code>0</code>, meaning no error shading. This argument is ignored unless <code>hist=FALSE</code>. If there are more than two classes, <code>err.col</code> uses only the first two. This argument is best explained by running an example:</td>
</tr>
</tbody>
</table>

```r
data(titanic)
earth.mod <- earth(survived ~ ., data=titanic)
plotd(earth.mod, vline.col=1, err.col=c(2,3,4))
```

The three areas are (i) the error area to the left of the threshold, (ii) the error area to the right of the threshold, and, (iii) the reducible error area. If less than three values are specified, `plotd` re-uses values in a sensible manner. Use values of `0` to skip areas. Disjoint regions are not handled well by the current implementation. |
| err.border  | Borders around the error shading. Default is `0`, meaning no borders, else specify up to three colors. |
| err.lwd     | Line widths of borders of the error shading. Default is `1`, else specify up to three line widths. |
| xaxt        | Default is "s". Use `xaxt="n"` for no x axis. |
| yaxt        | Default is "s". Use `yaxt="n"` for no y axis. |
| xaxis.cex   | Only used if `hist=TRUE` and `type="class"`. Specify size of class labels drawn on the x axis. Default is `1`. |
sd.thresh  Minimum acceptable standard deviation for a density. Default is 0.01. Densities with a standard deviation less than sd.thresh will not be plotted (a warning will be issued and the legend will say "not plotted").

Extra arguments passed to the predict method for the object.

Note

This function calls predict with the data originally used to build the model, and with the type specified above. It then separates the predicted values into classes, where the class for each predicted value is determined by the class of the observed response. Finally, it calls density (or hist if hist=TRUE) for each class-specific set of values, and plots the results.

This function estimates distributions with the density and hist functions, and also calls plot.density and plot.histogram. For an overview see Venables and Ripley MASS section 5.6.

Partitioning the response into classes

Considerable effort is made to partition the predicted response into classes in a sensible way. This is not always possible for multiple column responses and the nresponse argument should be used where necessary. The partitioning details depend on the types and numbers of columns in the observed and predicted responses. These in turn depend on the model object and the type argument.

Use the trace argument to see how plotd partitions the response for your model.

Degenerate densities

A message such as Warning: standard deviation of "male" density is 0, density is degenerate? means that the density for that class will not be plotted (the legend will say "not plotted").

Set sd.thresh=0 to get rid of this check, but be aware that histograms (and sometimes x axis labels) for degenerate densities will be misleading.

Using plotd for various models

This function is included in the earth package but can also be used with other models.

Example with glm:

```r
library(earth); data(etitanic)
glm.model <- glm(sex ~ ., data=etitanic, family=binomial)
plotd(glm.model)
```

Example with lm:

```r
library(earth); data(etitanic)
lm.model <- lm(as.numeric(sex) ~ ., data=etitanic)
plotd(lm.model)
```

Using plotd with lda or qda

The plotd function has special handling for lda (and qda) objects. For such objects, the type argument can take one of the following values:
"response" (default) linear discriminant
"ld" same as "response"
"class" predicted classes
"posterior" posterior probabilities

Example:

```
library(MASS); library(earth); data(ettitane)
lda.model <- lda(sex ~ ., data=ettitane)
plotd(lda.model) # linear discriminant by default
plotd(lda.model, type="class", hist=TRUE, labels=TRUE)
```

This handling of type is handled internally by plotd and type is not passed to predict.lda (type is used merely to select fields in the list returned by predict.lda). The type names can be abbreviated down to a single character.

For objects created with lda.matrix (as opposed to lda.formula), plotd blindly assumes that the grouping argument was the second argument.

plotd does not yet support objects created with lda.data.frame.

For lda responses with more than two factor levels, use the nresponse argument to select a column in the predicted response. Thus with the default type=NULL, (which gets automatically converted by plotd to type="response"), use nresponse=1 to select just the first linear discriminant. The default nresponse=NULL selects all columns, which is typically not what you want for lda models.

Example:

```
library(MASS); library(earth);
set.seed(1)       # optional, for reproducibility
example(lda)      # creates a model called "z"
plot(z, dimen=1)  # invokes plot.lda from the MASS package
plotd(z, nresponse=1, hist=1)  # equivalent using plotd
    # nresponse=1 selects first linear discr.
```

The dichot=TRUE argument is also useful for lda responses with more than two factor levels.

**TODO**

Handle degenerate densities in a more useful way.
Add freq argument for hist.

**See Also**

density, plot.density
hist, plot.histogram
earth, plot.earth
Examples

```r
old.par <- par(no.readonly=TRUE);
par(mfrow=c(2,2), mar=c(4, 3, 1.7, 0.5), mgp=c(1.6, 0.6, 0), par(cex = 0.8))
data(titanic)
fit <- earth(survived ~ ., data=titanic, degree=2, glm=list(family=binomial))

plotd(fit)
plotd(fit, hist=TRUE, legend.pos=c(.25,220))
plotd(fit, hist=TRUE, type="class", labels=TRUE, xlab="", xaxis.cex=.8)
par(old.par)
```

predict.earth  Predict with an earth model

Description

Predict with an `earth` model.

Usage

```r
## S3 method for class 'earth'
predict(object = stop("no 'object' arg"), newdata = NULL,
type = c("link", "response", "earth", "class", "terms"),
interval = "none", level = .95,
thresh = .5, trace = FALSE, ...)
```

Arguments

- `object`: An `earth` object. This is the only required argument.
- `newdata`: Make predictions using `newdata`, which can be a data frame, a matrix, or a vector with length equal to a multiple of the number of columns of the original input matrix `x`. Note that this is more flexible than the predict methods for most R models.
  - NAs are allowed (and the predicted value will be NA unless the NAs are in variables that are unused in the earth model).
  - Default is NULL, meaning return values predicted from the training set.
- `type`: Type of prediction. One of "link" (default), "response", "earth", "class", or "terms". See the **Note** below.
- `interval`: Return prediction or confidence levels. Default is "none". Use `interval="pint"` to get prediction intervals on new data.
  - Requires that the earth model was built with `varmod.method`.
  - This argument gets passed on as the `type` argument to `predict.varmod`. See its help page for details.
level  Confidence level for the interval argument. Default is .95, meaning construct 95% confidence bands (estimate the 2.5% and 97.5% levels).
thresh Threshold, a value between 0 and 1 when predicting a probability. Only applies when type="class". Default is .5. See the Note below.
trace Default FALSE. Set to TRUE to see which data, subset, etc. predict.earth is using.
... Unused, but provided for generic/method consistency.

Value
The predicted values (a matrix for multiple response models).
If type="terms", a matrix with each column showing the contribution of a predictor.
If interval="pint" or "cint", a matrix with three columns:
  fit: the predicted values
  lwr: the lower confidence or prediction limit
  upr: the upper confidence or prediction limit
If interval="se", the standard errors.

Note
Predicting with standard earth models
Use the default type="link", or possibly type="class".
Actually, the "link", "response", and "earth" choices all return the same value unless the glm argument was used in the original call to earth.

Predicting with earth-GLM models
This section applies to earth models with a GLM component, i.e., when the glm argument was used in the original call to earth.
The "link" and "response" options: see predict.glm for a description of these. In brief: for logistic models use type="link" to get log-odds and type="response" to get probabilities.
Use option "earth" to get the linear fit (this gives the prediction you would get if your original call to earth had no glm argument).

Predicting with "class"
Use option "class" to get the predicted class. With option "class", this function first makes predictions with type="response" and then assigns the predicted values to classes as follows:
(i) When y is a logical, predict TRUE if the predicted probability is greater than thresh.
(ii) When y is a numeric, predict TRUE if the predicted value is greater than thresh. Actually, this is identical to the above case, although thresh here may legitimately be a value outside the 0...1 range.
(iii) When y is a two level factor, predict the second level if its probability is more than thresh. In other words, with the default thresh=.5 predict the most probable level.
(iv) When y is a three or more level factor, predict the most probable level (and thresh is ignored).

Predicting with "terms"
The "terms" option returns a "link" response suitable for `termplot`. Only the additive terms and the first response (for multi-response models) are returned. Also, "terms" always returns the earth terms, and ignores the GLM component of the model, if any.

See Also

`earth`, `predict`

Examples

data(trees)
earth.mod <- earth(Volume ~ ., data = trees)
predict(earth.mod) # same as earth.mod$fitted.values
predict(earth.mod, c(10,80)) # yields 16.8

---

**predict.varmod**

Predict with a varmod model

Description

You probably won't need to call this function directly. It is called by `predict.earth` when that function's `interval` argument is used.

Usage

```r
## S3 method for class 'varmod'
predict(
  object = stop("no 'object' arg"),
  newdata = NULL,
  type = c("pint", "cint", "se", "abs.residual"),
  level = .95,
  trace = FALSE,
  ...
)
```

Arguments

- `object`: A varmod object.
- `newdata`: Make predictions using newdata. Default is NULL, meaning return values predicted from the training set.
- `type`: Type of prediction. This is the interval argument of `predict.earth`. One of "pint" Prediction intervals.
  "cint" Confidence intervals. Cannot be used with `newdata`.
  "se" Standard error of the parent model residuals.
residuals.earth

"abs.residual" The absolute residuals of the parent model on which the residual model regresses.

level
Confidence level for the interval argument. Default is .95, meaning construct 95% confidence bands (estimate the 2.5% and 97.5% levels).

trace
Currently unused.

... Unused, but provided for generic/method consistency.

Note
predict.varmod is called by predict.earth when its interval argument is used.

See Also
predict.earth varmod

Examples

data(ozone1)

set.seed(1) # optional, for cross validation reproducibility

# note: should really use ncross=30 below but for a quick demo we don't
earth.mod <- earth(O3~temp, data=ozone1, nfold=10, ncross=3, varmod.method="lm")

# call predict.earth, which calls predict.varmod

predict(earth.mod, newdata=ozone1[200:203,], interval="pint", level=.95)

residuals.earth Residuals for an earth model

Description
Residuals of an earth model.

Usage

## S3 method for class 'earth'
residuals(object = stop("no 'object' arg"),
          type = NULL, warn = TRUE, ...)

Arguments

object  An `earth` object. This is the only required argument.

type  One of

"earth" (default) Residuals (from the `lm` fit on bx).
"student" Residuals divided by \( \text{se} \times \sqrt{1 - h_{ii}} \). See the student argument of `plot.earth`.
"delever" Residuals divided by \( \sqrt{1 - h_{ii}} \). See the delever argument of `plot.earth`.
"deviance" Residuals as above, unless the object has a `glm` component, in which case return the `glm` deviance residuals.
"glm.pearson"
"glm.working"
"glm.response"
"glm.partial" Return the corresponding `glm` residuals (from the `glm` fit on bx). Can be used only if the earth model has a `glm` component.

warn  This function gives warnings when the results are not what you may expect. Use `warn=FALSE` to turn of just these warnings.

...  Unused, but provided for generic/method consistency.

Value

The residual values (will be a matrix for multiple response models).

See Also

`earth`
`residuals`
`resid` identical to `residuals`

Examples

```r
data(titanic)
earth.mod <- earth(pclass ~ ., data=titanic, glm=list(family=binomial))
head(resid(earth.mod, warn=FALSE)) # earth residuals, a column for each response
head(resid(earth.mod, type="earth")) # same
head(resid(earth.mod, type="deviance")) # GLM deviance residuals, a column for each response
```

Summary method for `earth` objects.
Usage

```r
## S3 method for class 'earth'
summary(object = stop("no 'object' arg"),
        details = FALSE, style = c("h", "pmax", "max", "C", "bf"),
        decomp = "anova", digits = getOption("digits"), fixed.point=TRUE,
        newdata = NULL, ...)
```

```r
## S3 method for class 'summary.earth'
print(x = stop("no 'x' arg"),
       details = x$details,
       decomp = x$decomp, digits = x$digits, fixed.point = x$fixed.point,
       newdata = x$newdata, ...)
```

Arguments

- **object**
  - An `earth` object. This is the only required argument for `summary.earth`.

- **x**
  - A `summary.earth` object. This is the only required argument for `print.summary.earth`.

- **details**
  - Default is `FALSE`. Use `TRUE` to print more information about `earth`-`glm` models. But note that the displayed P-values of the GLM coefficients are meaningless because of the amount of preprocessing by earth to select the regression terms.

- **style**
  - Formatting style. One of
    - "h" (default) more compact
    - "pmax" for those who prefer it and for compatibility with old versions of earth
    - "max" is the same as "pmax" but prints `max` rather than `pmax`
    - "C" C style expression with zero based indexing
    - "bf" basis function format.

- **decomp**
  - Specify how terms are ordered. Default is "anova". Use "none" to order the terms as created by the forward.pass. See `format.earth` for a full description.

- **digits**
  - The number of significant digits.
  - For `summary.earth`, the default is `getOption("digits")`.
  - For `print.summary.earth`, the default is the `$digits` component of `object`.

- **fixed.point**
  - Method of printing numbers in matrices. Default is `TRUE` which prints like this (making it easier to compare coefficients):

    ```
    (Intercept)  15.029
    h(temp-58)   0.313
    h(234-ibt)  -0.046
    ...
    ```

  - whereas `fixed.point=FALSE` prints like this (which is more usual in R):

    ```
    (Intercept) 1.5e+01
    h(temp-58)  3.1e-01
    h(234-ibt) -4.6e-02
    ...
    ```

Matrices with two or fewer rows are never printed with a fixed point.
**update.earth**

update.earth

newdata | Default NULL. Else print R-Squared for the new data, and if a variance model is present print the interval coverage table for the new data.
In the current implementation, newdata is allowed only if earth was called with a formula (not x and y).

... | Extra arguments are passed to format.earth.

**Value**

The value is the same as that returned by earth but with the following extra components.

strings | String(s) created by format.earth. For multiple response models, a vector of strings.
digits
details
decomp
fixed.point | The corresponding arguments, passed on to print.summary.earth.

**Note**

The printed Estimated importance uses evimp with the nsubsets criterion. The most important predictor is printed first, and so on.

**See Also**

earth, evimp, format.earth

**Examples**

```r
earth.mod <- earth(Volume~., data = trees)
summary(earth.mod, digits = 2)
```

---

**update.earth**

*Update an earth model*

**Description**

Update an earth model.

**Usage**

```r
## S3 method for class 'earth'
update(object = stop("no 'object' arg"),
       formula. = NULL, ponly = FALSE, ..., evaluate = TRUE)
```
Arguments

- **object** The earth object

- **formula** The formula. argument is treated like earth’s formula argument.

- **ponly** Force pruning only, no forward pass. Default is FALSE, meaning update.earth decides automatically if a forward pass is needed. See note below.

... Arguments passed on to earth.

- **evaluate** If TRUE (default) evaluate the new call, else return the call. Mostly for compatibility with the generic update.

Details

If only the following arguments are used, a forward pass is unnecessary, and update.earth will perform only the pruning pass. This is usually much faster for large models.

- **object**
- **glm**
- **trace**
- **nprune**
- **pmethod**
- **Eval.model.subsets**
- **Adjust.endspan**
- **Print.pruning.pass**
- **Force.xtx.prune**
- **Use.beta.cache**

This automatic determination to do a forward pass can be overridden with the ponly argument. If ponly=TRUE the forward pass will be skipped and only the pruning pass will be executed. This is useful for doing a pruning pass with new data. (Use earth’s data argument to specify the new data.) Typically in this scenario you would also specify penalty=-1. This is because with sufficient new data, independent of the original training data, the RSS not the GCV should be used for evaluating model subsets (The GCV approximates what the RSS would be on new data — but here we actually have new data, so why bother approximating. This "use new data for pruning" approach is useful in situations where you don’t trust the GCV approximation for your data.) By making penalty=-1, earth will calculate the RSS, not the GCV. See also the description of penalty on the earth help page.

Another (somewhat esoteric) use of ponly=TRUE is to do subset selection with a different penalty from that used to build the original model.

With trace=1, update.earth will tell you if earth’s forward pass was skipped.

If you used keepxy=TRUE in your original call to earth, then update.earth will use the saved values of x, y, etc., unless you specify otherwise by arguments to update.earth. It can be helpful to set trace=1 to see which x and y is used by update.earth.
Value

The value is the same as that returned by `earth`. If `object` is the only parameter then no changes are made — the returned value will be the same as the original object.

See Also

`earth`

Examples

data(ozone1)

(earth.mod <- earth(O3 ~ ., data = ozone1, degree = 2))

update(earth.mod, formula = O3 ~ . - temp) # requires forward pass and pruning

update(earth.mod, nprune = 8) # requires only pruning

update(earth.mod, penalty=1, ponly=TRUE) # pruning pass only with a new penalty

---

**varmod**

Variance models for estimating prediction intervals

Description

A variance model estimates the variance of predicted values. It can be used to estimate prediction intervals. See the interval argument of `predict.earth`.

A variance model is built by `earth` if `earth`'s varmod.method argument is specified. Results are stored in the `$varmod` field of the `earth` model. See the vignette “Variance models in earth” for details.

You probably won’t need to directly call `print.varmod` or `summary.varmod`. They get called internally by `summary.earth`.

Usage

```r
# S3 method for class 'varmod'
summary(
  object = stop("no "object" arg"),
  level = .95,
  style = "standard",
  digits = 2,
  newdata = NULL,
  ...
)
```
Arguments

object A varmod object. This is the only required argument.
level Same as predict.earth’s level argument.
style Determines how the coefficients of the varmod are printed by summary.varmod: "standard" (default) "unit" for easy comparison normalize the coefficients by dividing by the first coefficient.
digits Number of digits to print. Default is 2.
newdata Default NULL.

... Dots are passed on.

Note

A "varmod" object has the following fields:

- call The call used internally in the parent model to build the varmod object.
- parent The parent earth model.
- method Copy of the varmod.method argument to the parent model.
- package NULL, unless method="gam" (in which case either "gam" or "mgcv").
- exponent Copy of the varmod.exponent argument to the parent model.
- lambda Copy of the varmod.lambda parameter. (Currently always 1, absolute residuals.)
- converged Did the residual submodel IRLS converge?
- iters Number of residual model IRLS iterations (1 to 50).
- residmod The residual submodel. So for example, if varmod.method="lm", this will be an lm object.
- min.sd The predicted residual standard deviation is clamped so it will always be at least this value. This prevents prediction of negative or absurdly small variances. See earth's varmod.clamp argument. Clamping takes place in predict.varmod, which is called by predict.earth when estimating prediction intervals.
- model.var An n x 1 matrix. The model.var for an observation is the estimated model variance for that observation over all datasets, and is estimated with repeated cross validation. It is the variance of the mean out-of-fold prediction for that observation over ncross repetitions.
- abs.resids An n x 1 matrix. The absolute residuals used to build the residual model.
- parent.x An n x p matrix. Parent earth model x.
- parent.y An n x 1 matrix. Parent earth model y.
- iter.rsq Weighted R-Squared of residual submodel residmod, after IRLS iteration.
- iter.stderr Standard errors of the coefficients of the residual submodel residmod, after IRLS iteration.

See Also

plot.varmod, predict.varmod
Examples

data(ozone1)

set.seed(1) # optional, for cross validation reproducibility

# note: should really use ncross=30 below but for a quick demo we don't
earth.mod <- earth(O3~temp, data=ozone1, nfold=10, ncross=3, varmod.method="lm")

print(summary(earth.mod)) # note additional info on the variance model

old.mfrow <- par(mfrow=c(2,2), mar=c(3, 3, 3, 1), mgp=c(1.5, 0.5, 0))

plotmo(earth.mod, do.par=FALSE, col.response=1, level=.90, main="earth model: O3~temp")

plot(earth.mod, which=3, level=.90) # residual plot: note 90% pred and darker conf intervals

par(par=old.mfrow)
Index

*Topic datasets
  etiitanic, 15
  ozone1, 23
*Topic models
  contr.earth.response, 2
  earth, 3
  evimp, 17
  format.earth, 18
  mars.to.earth, 21
  model.matrix.earth, 22
  plot.earth, 24
  plot.earth.models, 29
  plot.evimp, 32
  plotd, 34
  predict.earth, 39
  residuals.earth, 42
  summary.earth, 43
  update.earth, 45
*Topic regression
  earth, 3
*Topic smooth
  earth, 3
  contr.earth.response, 2
  contrasts, 2
  density, 35–38
  earth, 2, 3, 7, 16–18, 20–25, 29–31, 33, 34, 37–47
  etiitanic, 15
  evimp, 14, 15, 17, 32, 33, 45
  factor, 12
  factors, 4
  format.earth, 9, 18, 44, 45
  gam, 7
  glm, 4, 12, 23, 34, 37, 43, 44
  grep, 6, 26
  hist, 35–38
  lda, 34, 37
  leaps, 6, 8, 14
  legend, 28, 31, 36
  lm, 6, 7, 23, 26, 34, 37, 43
  lm.fit, 23
  loess, 27
  logical, 12
  lowess, 27
  mars, 14, 21
  mars.to.earth, 21
  model.matrix.earth, 9, 22
  nls, 7
  ozone, 14
  ozone1, 23
  par, 27
  plot, 25, 30
  plot.density, 36–38
  plot.earth, 15, 24, 31, 38, 43
  plot.earth.models, 29, 29, 31, 33
  plot.evimp, 18, 32
  plot.histogram, 37, 38
  plot.varmod, 33, 48
  plotd, 29, 31, 34
  plotmo, 15, 29, 31, 33
  pmax, 20
  predict, 35, 37, 41
  predict.earth, 11, 35, 39, 41, 42, 47
  predict.glm, 35, 40
  predict.varmod, 39, 41, 48
  print.summary.earth (summary.earth), 43
  qda, 37
  resid, 43
  residuals, 43

50
INDEX

residuals.earth, 42
rlm, 7, 25

Scale, 8
sqrt, 17
summary.earth, 13, 15, 43, 44, 47
summary.varmod(varmod), 47
termplot, 41
update, 46
update.earth, 4, 8, 21, 45

varmod, 7, 14, 34, 42, 47