Package 'curvir'

July 22, 2025

Type Package

Title Specify Reserve Demand Curves

Version 0.1.1

Maintainer Nikolaos Kourentzes <nikolaos@kourentzes.com>

Description Automatic specification and estimation of reserve demand curves for central bank operations. The package can help to choose the best demand curve and identify additional explanatory variables. Various plot and predict options are included. For more details, see Chen et al. (2023) <https://www.imf.org/en/Publications/WP/Issues/2023/09/01/Modeling-the-Reserve-Demand-to-Facilitate-Central-Bank-Operations-538754>.

Imports RColorBrewer, abind, cvTools, pbapply, pso, mgcv, qgam, quantregForest, randomForest

Depends R (>= 3.5.0)

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

NeedsCompilation no

Author Nikolaos Kourentzes [aut, cre], Zhuohui Chen [aut], Romain Veyrune [aut]

Repository CRAN

Date/Publication 2023-11-24 17:00:05 UTC

Contents

curve 2
curveopt 4
curvepred 6
curvir 7
cvfit 8
cvfitplot 10

cvnpcurve 11
ecb 12
invcurve 13
npcurve 14
plot.curvir 15
predict.curvir 17
varselect 19

Index **21**

`curve` *Reserve demand curve*

Description

Fits the reserve demand curve between excess reserves and normalised rates

Usage

```
curve(x, y, type = "logistic", dummy = NULL, q = NULL, ...)
```

Arguments

`x`	A matrix of explanatory variables. Excess reserve must be the first input.Additional regressor follow (optional).
`y`	A vector of normalised interest rates.
`type`	The type of the reserve demand curve. This can be any of `logistic`, `redLogistic`, `fixLogistic`, `doubleExp`, `exponential`, `fixExponential`, `arctan`, `linear`. See details in `curve`
`dummy`	Optional input to signify a regime change (vertical shifts in the curve). Must be a vector of equal length to the rows of `x`. If not needed use `NULL`.
`q`	Target interval. This is a scalar below 1, for example 0.9 is the 90% interval. If `NULL` then no quantiles are estimated.
`...`	Additional arguments passed to optimiser `curveopt`.

Details

For a description of the parametric curves, see the provided reference. Below we list their functions:

- `logisitc` (Logistic)

$$r_i = \alpha + \kappa/(1 - \beta e^{g(\boldsymbol{C}_i)}) + \varepsilon_i$$

- `redLogistic` (Reduced logistic)

$$r_i = \alpha + 1/(1 - \beta e^{g(\boldsymbol{C}_i)}) + \varepsilon_i$$

- `fixLogistic` (Fixed logistic)

$$r_i = \alpha + 1/(1 - e^{g(\boldsymbol{C}_i)}) + \varepsilon_i$$

- `doubleExp` (Double exponential)

$$r_i = \alpha + \beta e^{\rho e^{g(\boldsymbol{C}_i)}} + \varepsilon_i$$

- `exponential` (Exponential)

$$r_i = \alpha + \beta e^{g(\boldsymbol{C}_i)} + \varepsilon_i$$

- `fixExponential` (Fixed exponential)

$$r_i = \beta e^{g(\boldsymbol{C}_i)} + \varepsilon_i$$

- `arctan` (Arctangent)

$$r_i = \alpha + \beta \arctan(g(\boldsymbol{C}_i)) + \varepsilon_i$$

- `linear` (Linear)

$$r_i = g(\boldsymbol{C}_i) + \varepsilon_i$$

And $g(\boldsymbol{C}) = c + \boldsymbol{C} w_g$, where α, β, κ, ρ are curve parameters, c is a constant togglable by `constant`, $\boldsymbol{C}$ are the regressors including the excess reserves. w_g their coefficients, and finally ε_i is the error term of the curve.

Value

Returns a model of class `curvir`. This includes

- `type` the type of the curve.
- `constant` a logical indicating the use of a constant.
- `w` a list including: `mean` the curve parameters for the mean of the curve, `upper` and `lower` the parameters for the curve at the upper and lower intervals.
- `data` a list including the `y`, `x`, and `dummy` used for the fitting of the curve.
- `mse` the MSE from the fitting of the curve (the mean only).
- `q` the interval used in the fitting of the curve.

Note

An additional column for the constant is automatically generated, unless requested otherwise.

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

`predict.curvir`, `plot.curvir`, and `curveopt`.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1,drop=FALSE]
curve(x,rate)

# An arctangent curve
curve(x,rate,type="arctan")
```

---

curveopt *Optimise curve parameters*

Description

Finds optimal curve parameters.

Usage

```
curveopt(
  x,
  y,
  type = "logistic",
  constant = c(TRUE, FALSE),
  reps = 3,
  sign = NULL,
  q = NULL,
  winit = NULL,
  yhat = NULL,
  wsel = c("select", "combine"),
  dummy = NULL,
  sameSign = c(TRUE, FALSE)
)
```

Arguments

x	A matrix of explanatory variables. Excess reserve must be the first input.Additional regressor follow (optional).
y	A vector of normalised interest rates.
type	The type of the reserve demand curve. This can be any of `logistic`, `redLogistic`, `fixLogistic`, `doubleExp`, `exponential`, `fixExponential`, `arctan`, `linear`. See details in `curve`
constant	A logical (`TRUE` or `FALSE`) whether to include a constant or not.
reps	Number of repetitions for the particle swarm optimisation.

sign	A vector of equal length to the number of additional regressors in x (excluding the constant (if used) and the excess reserves) of positive and negative values (any) that will be used to obtain signs to restrict the values of the estimated parameters. Use NULL for no restrictions.
q	The desired quantile to optimise for. Use NULL to get the conditional expectation.
winit	A vector of initial values for the optimisation. This will also carry over to sign restrictions if sameSign==TRUE.
yhat	Useful when estimating quantiles. Supply here the predicted values for the conditional expectation to add restrictions for the quantiles to not cross the conditional expectation. Use NULL to not add any restrictions.
wsel	Use the minimum error set of parameters (select) or the combination of the pool parameters using the heuristic in Kourentzes et al., (2019) (combine).
dummy	Optional input to signify a regime change (vertical shifts in the curve). Must be a vector of equal length to the rows of x. If not needed use NULL.
sameSign	Used if winit != NULL to take any sign restrictions from winit.

Value

Returns a list of

- w The optimal parameters
- mse The Mean Squared Error of the fitted curve.

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

- Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).
- Kourentzes, N., Barrow, D., & Petropoulos, F. (2019). Another look at forecast selection and combination: Evidence from forecast pooling. International Journal of Production Economics, 209, 226-235.

See Also

curve, and curvepred.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1,drop=FALSE]
curveopt(x,rate)
```

curvepred *Reserve demand curve predicted values*

Description

Provides the predicted values for the reserve demand curve of choice. For general use prefer the predict() function, which handles the constant internally.

Usage

```
curvepred(x, w, type = "logistic", dummy = NULL)
```

Arguments

x	A matrix with the inputs. If there is a constant in the estimated curve, then the first column in x should be the constant. Next column should be the excess reserves. Additional regressor follow (optional).
w	The vector of weights for the desired curve. Estimated using the curveopt function.
type	The type of the reserve demand curve. This can be any of logistic, redLogistic, fixLogistic, doubleExp, exponential, fixExponential, arctan, linear. See details in curve
dummy	Optional input to signify a regime change (vertical shifts in the curve). Must be a vector of equal length to the rows of x. If not needed use NULL.

Details

For a description of the parametric curves, see the provided reference. Below we list their functions:

- logisitc (Logistic)
$$r_i = \alpha + \kappa/(1 - \beta e^{g(\boldsymbol{C}_i)}) + \varepsilon_i$$
- redLogistic (Reduced logistic)
$$r_i = \alpha + 1/(1 - \beta e^{g(\boldsymbol{C}_i)}) + \varepsilon_i$$
- fixLogistic (Fixed logistic)
$$r_i = \alpha + 1/(1 - e^{g(\boldsymbol{C}_i)}) + \varepsilon_i$$
- doubleExp (Double exponential)
$$r_i = \alpha + \beta e^{\rho e^{g(\boldsymbol{C}_i)}} + \varepsilon_i$$
- exponential (Exponential)
$$r_i = \alpha + \beta e^{g(\boldsymbol{C}_i)} + \varepsilon_i$$

- `fixExponential` (Fixed exponential)

$$r_i = \beta e^{g(\boldsymbol{C}_i)} + \varepsilon_i$$

- `arctan` (Arctangent)

$$r_i = \alpha + \beta \arctan(g(\boldsymbol{C}_i)) + \varepsilon_i$$

- `linear` (Linear)

$$r_i = g(\boldsymbol{C}_i) + \varepsilon_i$$

And $g(\boldsymbol{C}) = c + \boldsymbol{C}w_g$, where α, β, κ, ρ are curve parameters, c is a constant togglable by `constant`, $\boldsymbol{C}$ are the regressors including the excess reserves. w_g their coefficients, and finally ε_i is the error term of the curve.

Value

Returns a vector of the predicted values.

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

`curve`, and `curveopt`.

`curvir` *curvir:Reserve demand curve modelling toolbox*

Description

The **curvir** package provides tools for building reserve demand curves for central bank operations.

Parametric curve modelling

- `curve` - fit a parametric curve
- `curveopt` - optimise curve parameters (called through `curve`)
- `curvepred` - predict curve values (called through `predict`)
- `invcurve` - produce predictions from the inverse curve

Non-parametric curve modelling

- `npcurve` - fit a non-parametric curve

Curve specification

- `cvfit` - Joint automatic specification of curve type and regressors
- `varselect` - Automatic specification of regressors
- `cvnpcurve` - Provide cross-validated errors for non-paramteric curves
- `cvfitplot` - Summary plot for `cvfit` and `cvnpcurve`

General

- `predict` Predict parametric and non-parametric curves
- `plot` Plot parametric and non-parametric curves

Data

- `ecb` ECB dataset from Chen et al. (2023)

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>, Zhuohui Chen, <zchen4@imf.org>, Romain R. Veyrune.

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

`cvfit` *Automatically select curve type and explanatory variables*

Description

Using cross-validation automatically select explanatory variables jointly with curve type. When running `cvfit` there is no need to use `varselect` separately.

Usage

```
cvfit(
  x,
  y,
  folds = 10,
  constant = c(TRUE, FALSE),
  sign = NULL,
  reps = 3,
  parallel = c(FALSE, TRUE),
  usepbapply = c(FALSE, TRUE),
 alltype = c("logistic", "redLogistic", "doubleExp", "exponential", "arctan", "linear"),
  search = c("backward", "forward"),
```

```
  wsel = c("select", "combine"),
  dummy = NULL
)
```

Arguments

x	A matrix of explanatory variables. Excess reserve must be the first input.Additional regressor follow (optional).
y	A vector of normalised interest rates.
folds	Folds to use for cross-validation.
constant	A logical (TRUE or FALSE) whether to include a constant or not.
sign	A vector of equal length to the number of additional regressors in x (excluding the constant (if used) and the excess reserves) of positive and negative values (any) that will be used to obtain signs to restrict the values of the estimated parameters. Use NULL for no restrictions.
reps	Number of repetitions for the particle swarm optimisation.
parallel	Initialise and use parallel processing for the cross-validation. Note that the maximum number of cores that will be used is equal to the alternative types of curves that are evaluated (alltype).
usepbapply	A logical to indicate whether to use pbapply to report on the parallel calculation progress. Note that pbapply does not support load balancing.
alltype	A vector of the curve types to consider in the selection process.
search	Search strategy for variable inclusion. With forward the model starts from no explanatory variables, and evaluates potential additions. With backward the model starts with all variables and evaluates for potential exclusions.
wsel	Use the minimum error set of parameters (select) or the combination of the pool parameters using the heuristic in Kourentzes et al., (2019) (combine).
dummy	Optional input to signify a regime change (vertical shifts in the curve). Must be a vector of equal length to the rows of x. If not needed use NULL.

Value

Returns a list with the recommended variable selection choice:

- type the type of selected curve.
- keep a logical vector with which variables to keep.
- varRes the result from varselect for each evaluated curve type.
- cvIndex a matrix detailing how the sample was split for the cross-validation. First column is the fold number and second column is the index of the observation.

Use cvfitplot to visualise the output.

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

cvfitplot, curve, and varselect.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1:3,drop=FALSE]
cvKeep <- cvfit(x,rate,folds=5,alltype=c("logistic","arctan"),parallel=TRUE)
# Print result
print(cvKeep)
# Fit curve with the selected variables
curve(x[,cvKeep$keep,drop=FALSE],rate,type=cvKeep$type)
```

---

cvfitplot *Plot output of cvfit to facilitate comparison*

Description

Plot summarised error of different curves specification from cvfit. Assuming normal errors, plot the mean cross-validated error and the 95

Usage

```
cvfitplot(cvKeep, xlock = c("mean", "median"), cvRF = NULL, cvSP = NULL)
```

Arguments

cvKeep	The output of cvfit.
xlock	Focus horizontal axis on mean or median, The latter provides a narrower range, facilitating comparison when there are outliers.
cvRF	Include cross-validation results from random forecast non-parametric curves. Obtain these from cvnpcurve with type="rforest". Use NULL to ignore.
cvSP	Include cross-validation results from spline regression non-parametric curves. Obtain these from cvnpcurve with type="spline". Use NULL to ignore.

Value

No returned value. Produces a summary plot of cross-validated errors.

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

cvfit.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1:3,drop=FALSE]
cvKeep <- cvfit(x,rate,folds=5,alltype=c("logistic","arctan"),parallel=TRUE)
cvfitplot(cvKeep)
# Add results from non-parameteric curves
cvRF <- cvnpcurve(x,rate,cvKeep$cvIndex)
cvSP <- cvnpcurve(x,rate,cvKeep$cvIndex,type="spline")
cvfitplot(cvKeep,cvRF=cvRF,cvSP=cvSP)
```

---

cvnpcurve *Cross-validated errors for non-parametric curve*

Description

Obtain cross-validated errors for a non-parametric curve with given sample splits.

Usage

```
cvnpcurve(x, y, cvIndex, type = "rforest", dummy = NULL)
```

Arguments

x	A matrix of explanatory variables. Excess reserve must be the first input.Additional regressor follow (optional).
y	A vector of normalised interest rates.
cvIndex	A matrix detailing how the sample was split for the cross-validation. Output from cvfit.
type	The type of the reserve demand curve. This can be any of rforecast for random forecast or spline for spline regression.
dummy	Optional input to signify a regime change (vertical shifts in the curve). Must be a vector of equal length to the rows of x. If not needed use NULL.

Value

Returns summary cross-validated errors, comparable with the output from `cvfit`.

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

`cvfit`, and `cvfitplot`.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1:3,drop=FALSE]
cvKeep <- cvfit(x,rate,folds=5,alltype=c("logistic","arctan"),parallel=TRUE)
# Get non-parametric curve cross-validated errors
cvRF <- cvnpcurve(x,rate,cvKeep$cvIndex)
cvSP <- cvnpcurve(x,rate,cvKeep$cvIndex,type="spline")
```

---

ecb *ECB excess reserve data*

Description

The short-term interest rate used for the European Central Bank (ECB) is the volume-weighted Euro Overnight Index Average (EONIA) rate. Various explanatory variables are provided, as listed in the Chen et al. (2023). Data are collected from 1999 to 2019, resulting in 239 maintenance periods where all data is available.

Usage

```
ecb
```

Format

`ecb` a list containing

- `rate` the normalised EONIA rate.
- `x` a matrix including the ECB excess reserves and various regressors.

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

Examples

```
plot(ecb$x[,1],ecb$rate,xlab="Excess Reserves",ylab="Rate")
```

`invcurve` *Calculate the inverse curve prediction*

Description

Calculate the predicted reserves given some rate, i.e., calculate the prediction of the inverse curve.

Usage

```
invcurve(
  object,
  ynew = NULL,
  xnew = NULL,
  dummynew = NULL,
  warn = c(TRUE, FALSE)
)
```

Arguments

`object`	A model fit with `curve`.
`ynew`	The input rate. If `NULL` this corresponds to the values from `predict(object)`.
`xnew`	The values for the additional regressors that were used in the curve fit. Must be a matrix, ordered (columns) as they were input in the fitting of the curve. The constant is dealt with automatically. Do not input the excess reserves. If `NULL` this is picked up from the data used to fit the curve.
`dummynew`	The values for the indicator, if one was used in the fitting of the curve. If `NULL` then the data used in the fitting of the model are used.
`warn`	A logical (`TRUE` or `FALSE`) to issue a warning if the resulting values are more than 10% away from the min-max of the excess reserves used to estimate the curve.

Value

Returns a vector of values of the predicted reserves

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

curve, and predict.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1,drop=FALSE]
fit <- curve(x,rate,type="logistic")
invcurve(fit)

# Use a different input rate
invcurve(fit,ynew=0.1)
```

---

npcurve *Reserve demand non-parametric curve*

Description

Fits a non-parametric reserve demand curve between excess reserves and normalised rates

Usage

```
npcurve(x, y, type = c("rforest", "spline"), dummy = NULL, q = NULL, ...)
```

Arguments

x	A matrix of explanatory variables. Excess reserve must be the first input.Additional regressor follow (optional).
y	A vector of normalised interest rates.
type	The type of the reserve demand curve. This can be any of rforecast for random forecast or spline for spline regression.
dummy	Optional input to signify a regime change (vertical shifts in the curve). Must be a vector of equal length to the rows of x. If not needed use NULL.
q	Target interval. This is a scalar below 1, for example 0.9 is the 90% interval. If NULL then no quantiles are estimated.
...	Additional arguments (unused).

Value

Returns a model of class npcurvir. This includes

- type the type of the curve.
- fit the non-parametric model for the mean.
- fitQ The non-parametric model for the quantiles.
- data a list including the y, x, and dummy used for the fitting of the curve.
- q the interval used in the fitting of the curve.

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

predict.npcurvir, and plot.npcurvir.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1,drop=FALSE]
npcurve(x,rate)
```

---

plot.curvir *Plot method for curvir reserve demand models*

Description

Plot a reserve demand curve estimated using curve.

Plot a non-parametric reserve demand curve estimated using npcurve.

Usage

```
## S3 method for class 'curvir'
plot(
  x,
  ploty = c(FALSE, TRUE),
  plotq = c(TRUE, FALSE),
  usemean = c(FALSE, TRUE),
```

```
  prcmp = c(FALSE, TRUE),
  useline = c(FALSE, TRUE),
  main = NULL,
  pch = 20,
  ...
)

## S3 method for class 'npcurvir'
plot(
  x,
  ploty = c(FALSE, TRUE),
  plotq = c(TRUE, FALSE),
  usemean = c(FALSE, TRUE),
  prcmp = c(FALSE, TRUE),
  useline = c(FALSE, TRUE),
  main = NULL,
  pch = 20,
  ...
)
```

Arguments

x	A model fit with curve.
ploty	Logical (TRUE or FALSE) to plot data used for fitting or not.
plotq	Logical (TRUE or FALSE) to plot intervals.
usemean	Logical (TRUE or FALSE) to replace all explanatory variables (except from excess reserves) with their mean. It cannot be used if a dummy is used in the fitting of the curve.
prcmp	Logical (TRUE or FALSE) to use principal components to project all variables into two dimensions. Requires usemean == FALSE. This may be useful when multiple explanatory variables are used.
useline	Logical (TRUE or FALSE) to force line plots when usemean == FALSE or prcomp == FALSE and there are many explanatory variables.
main	Use to overide the default plot title.
pch	Use to overide the dafault marker type for points.
...	Additional inputs (currently unused).

Value

No returned value. Produces a plot of the estimated curve.

Methods (by class)

- plot(curvir): Plot parametric curves
- plot(npcurvir): Plot non-parametric curves

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

curve.

npcurve.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1,drop=FALSE]
fit <- curve(x,rate)
plot(fit)
```

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1,drop=FALSE]
fit <- npcurve(x,rate)
plot(fit)
```

---

predict.curvir *Predict method for curvir reserve demand models*

Description

Predicted values based on curvir model object

Predicted values based on npcurvir model object

Usage

```
## S3 method for class 'curvir'
predict(object, newdata = NULL, newdummy = NULL, ...)

## S3 method for class 'npcurvir'
predict(object, newdata = NULL, newdummy = NULL, ...)
```

Arguments

object	A model fit with npcurve.
newdata	New input data organised as the x matrix in curve. If NULL then the data used to fit the model is re-used.
newdummy	New input dummy organised as the dummy vector in curve. If NULL then the dummy used to fit the model is re-used.
...	Further arguments (unused)

Value

Returns a matrix of predicted values. If the model has estimates for intervals then it will provide upper and lower intervals.

Returns a matrix of predicted values. If the model has estimates for intervals then it will provide upper and lower intervals.

Methods (by class)

- predict(curvir): Predicted values for parametric curves
- predict(npcurvir): Predicted values for non-parametric curves

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

curve.

npcurve.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1,drop=FALSE]
fit <- curve(x,rate)
predict(fit)

# An example with new data
predict(fit,newdata=tail(x))


# Use ECB example data
```

```
rate <- ecb$rate
x <- ecb$x[,1,drop=FALSE]
fit <- npcurve(x,rate)
predict(fit)
```

---

varselect *Automatically select explanatory variables for a curve type*

Description

Using cross-validation automatically select explanatory variables for a reserve demand curve type.

Usage

```
varselect(
  x,
  y,
  type = "logistic",
  folds = 10,
  constant = c(TRUE, FALSE),
  sign = NULL,
  reps = 3,
  search = c("backward", "forward"),
  wsel = c("select", "combine"),
  dummy = NULL
)
```

Arguments

x	A matrix of explanatory variables. Excess reserve must be the first input.Additional regressor follow (optional).
y	A vector of normalised interest rates.
type	The type of the reserve demand curve. This can be any of logistic, redLogistic, fixLogistic, doubleExp, exponential, fixExponential, arctan, linear. See details in curve
folds	Folds to use for cross-validation.
constant	A logical (TRUE or FALSE) whether to include a constant or not.
sign	A vector of equal length to the number of additional regressors in x (excluding the constant (if used) and the excess reserves) of positive and negative values (any) that will be used to obtain signs to restrict the values of the estimated parameters. Use NULL for no restrictions.
reps	Number of repetitions for the particle swarm optimisation.
search	Search strategy for variable inclusion. With forward the model starts from no explanatory variables, and evaluates potential additions. With backward the model starts with all variables and evaluates for potential exclusions.

wsel	Use the minimum error set of parameters (select) or the combination of the pool parameters using the heuristic in Kourentzes et al., (2019) (combine).
dummy	Optional input to signify a regime change (vertical shifts in the curve). Must be a vector of equal length to the rows of x. If not needed use NULL.

Value

Returns a list with the recommended variable selection choice:

- keep a logical vector with which variables to keep.
- errors statistics of the cross-validated MSE error.

Author(s)

Nikolaos Kourentzes, <nikolaos@kourentzes.com>

References

Chen, Z., Kourentzes, N., & Veyrune, R. (2023). Modeling the Reserve Demand to Facilitate Central Bank Operations. IMF Working Papers, 2023(179).

See Also

curve, and cvfit.

Examples

```
# Use ECB example data
rate <- ecb$rate
x <- ecb$x[,1:3,drop=FALSE]
varKeep <- varselect(x,rate,folds=5)
# Print result
print(varKeep)
# Fit curve with the selected variables
curve(x[,varKeep$keep,drop=FALSE],rate)
```

Index

∗ **datasets**
ecb, 12

curve, *2*, 2, *4–7*, *10*, *13–20*
curveopt, *2, 3*, 4, *6, 7*
curvepred, *5*, 6, *7*
curvir, 7
cvfit, *8*, 8, *10–12*, *20*
cvfitplot, *8–10*, 10, *12*
cvnpcurve, *8*, *10*, 11

ecb, *8*, 12

invcurve, *7*, 13

npcurve, *7*, 14, *15*, *17, 18*

plot, *8*
plot.curvir, *3*, 15
plot.npcurvir, *15*
plot.npcurvir *(*plot.curvir*)*, 15
predict, *7, 8*, *14*
predict.curvir, *3*, 17
predict.npcurvir, *15*
predict.npcurvir *(*predict.curvir*)*, 17

varselect, *8–10*, 19