

# Package ‘DLL’

February 18, 2022

**Type** Package

**Title** Decorrelated Local Linear Estimator

**Version** 1.0.0

**Author** Wei Yuan, Zhenyu Wang, Zijian Guo, Cun-Hui Zhang

**Maintainer** Zijian Guo <zijguo@stat.rutgers.edu>

**Description** Implementation of the Decorrelated Local Linear estimator proposed in <[arxiv:1907.12732](https://arxiv.org/abs/1907.12732)>. It constructs the confidence interval for the derivative of the function of interest under the high-dimensional sparse additive model.

**URL** <https://github.com/zijguo/HighDim-Additive-Inference>

**License** GPL-3

**Encoding** UTF-8

**Imports** MASS, stats, splines, glmnet, SAM, locpol

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2022-02-18 08:30:12 UTC

## R topics documented:

DLL . . . . .	1
<b>Index</b>	<b>4</b>

---

DLL	<i>DLL</i>
-----	------------

---

## Description

It constructs the Decorrelated Local Linear estimator and estimates its standard error. It further constructs the confidence interval for the derivative of the function of interest.

**Usage**

```

DLL(
  X,
  y,
  D.ind,
  d0,
  h = NULL,
  lam.seq = NULL,
  treatment.SAM = FALSE,
  data.swap = FALSE,
  quant.trans = FALSE,
  alpha = 0.05
)

```

**Arguments**

X	the covariates matrix, of dimension $n \times p$
y	the outcome vector, of length $n$
D.ind	the column index(es) of X, indicating the index(es) of the variable(s) of interest. It can be a scalar or a vector. If vector, then do inference for each index of the sequence.
d0	evaluation points for derivative estimation. It can be scalar or vector.
h	bandwidth, computed by Rule of Thumb from the package “locpol” if not provided.
lam.seq	a sequence of tuning parameters considered in fitting the sparse additive model. Cross validation is used to choose the best one. If not provided(default), the default sequence ranges from 5e-3 to 1 with the length of 100. If provided, the sequence needs to be in a decreasing order for the reason of computation efficiency.
treatment.SAM	Whether a sparse additive model is used for fitting the treatment model? If ‘False’(default), Lasso with cross validation is used to fit the treatment model. Default is ‘FALSE’
data.swap	Whether data swapping is conducted or not? Default is ‘FALSE’
quant.trans	Whether quantile transformation is conducted or not? Default is ‘FALSE’
alpha	the significance level. Default is 0.05

**Value**

est	point estimates of the function derivative
est.se	estimated standard errors of est
CI	list of lower and upper bounds of confidence intervals
d0	evaluation points
bw.save	selected bandwidth at each element of d0
sigma1.sq	estimated variance of the error term in the outcome model

**Examples**

```
# evaluation points
d0 = c(-0.5,0.25)
f = function(x) 1.5*sin(x)
f.deriv = function(x) 1.5*cos(x)
g1 = function(x) 2*exp(-x/2)
g2 = function(x) (x-1)^2 - 25/12
g3 = function(x) x - 1/3
g4 = function(x) 0.75*x
g5 = function(x) 0.5*x
# sample size and dimension of X
n = 200
p = 100
# covariance structure of D and X
Cov_Matrix = toeplitz(c(1, 0.7, 0.5, 0.3, seq(0.1, 0, length.out = p-3)))
set.seed(123)
# X represents the (D,X) here
X = MASS::mvrnorm(n,rep(-0.25,p+1),Sigma = Cov_Matrix)
e = rnorm(n,sd=1)
# generating response
y = f(X[,1]) + g1(X[,2]) + g2(X[,3]) + g3(X[,4]) + g4(X[,5]) + g5(X[,6]) + e
### DLL inference
DLL.model = DLL(X=X, y=y, D.ind = 1, d0 = d0)
# true values
f.deriv(d0)
# point estimates
DLL.model$est
# standard errors
DLL.model$est.se
# confidence interval
DLL.model$CI
```

# Index

DLL, [1](#)