Package ‘InvariantCausalPrediction’

November 10, 2019

Type Package

Title Invariant Causal Prediction

Version 0.8

Date 2019-10-10

Author Nicolai Meinshausen

Depends glmnet, mboost

Maintainer Nicolai Meinshausen <meinshausen@stat.math.ethz.ch>

Description Confidence intervals for causal effects, using data collected in different experimental or environmental conditions. Hidden variables can be included in the model with a more experimental version.

License GPL

NeedsCompilation no

Repository CRAN

Date/Publication 2019-11-10 11:40:02 UTC

R topics documented:

InvariantCausalPrediction-package *Invariant causal prediction*

Description

Confidence intervals for causal prediction in a regression setting. An experimental version is also available for classification.

Details

Package:	InvariantCausalPrediction
Type:	Package
Version:	0.6-1
Date:	2016-05-02
License:	GPL

Confidence intervals can be computed with function `ICP`, both for regression and binary classification. `print`, `plot` and `summary` methods are available.

Author(s)

Nicolai Meinshausen <meinshausen@stat.math.ethz.ch>

Maintainer: Nicolai Meinshausen <meinshausen@stat.math.ethz.ch>

hiddenICP *Invariant Causal Prediction with hidden variables*

Description

Confidence intervals for causal effects in a regression setting with possible confounders.

Usage

```
hiddenICP(X, Y, ExpInd, alpha = 0.1, mode = "asymptotic", intercept=FALSE)
```

Arguments

`X`	A matrix (or data frame) with the predictor variables for all experimental settings
`Y`	The response or target variable of interest. Can be numeric for regression or a factor with two levels for binary classification.

`ExpInd` Indicator of the experiment or the intervention type an observation belongs to. Can be a numeric vector of the same length as `Y` with `K` unique entries if there are `K` different experiments (for example entry 1 for all observational data and entry 2 for intervention data). Can also be a list, where each element of the list contains the indices of all observations that belong to the corresponding grouping in the data (for example two elements: first element is a vector that contains indices of observations that are observational data and second element is a vector that contains indices of all observations that are of interventional type).

`alpha` The level of the test procedure. Use the default `alpha`=0.1 to obtain 90% confidence intervals.

`mode` Currently only mode "asymptotic" is implemented; the argument is thus in the current version without effect.

`intercept` Boolean variable; if `TRUE`, an intercept is added to the design matrix (but coefficients are returned without intercept term).

Value

A list with elements

`betahat` The point estimator for the causal effects

`maximinCoefficients`
The value in the confidence interval for each variable effects that is closest to 0. Is hence non-zero for variables with significant effects.

`ConfInt` The matrix with confidence intervals for the causal coefficient of all variables. First row is the upper bound and second row the lower bound.

`pvalues` The p-values of all variables.

`colnames` The column-names of the predictor variables.

`alpha` The chosen level.

Author(s)

Nicolai Meinshausen <meinshausen@stat.math.ethz.ch>

References

none yet.

See Also

`ICP` for reconstructing the parents of a variable under arbitrary interventions on all other variables (but no hidden variables). See package "backShift" for constructing point estimates of causal cyclic models in the presence of hidden variables (again under shift interventions) .

Examples

```
 ##########################################
 ####### 1st example:
 ####### Simulate data with interventions
      set.seed(1)
    ## sample size n
      n <- 2000
    ## 4 predictor variables
      p  <- 4
    ## simulate as independent Gaussian variables
      X <- matrix(rnorm(n*p),nrow=n)
    ## divide data into observational (ExpInd=1) and interventional (ExpInd=2)
      ExpInd <- c(rep(1,n/2),rep(2,n/2))
    ## for interventional data (ExpInd==2): change distribution
      nI <- sum(ExpInd==2)
      X[ExpInd==2,] <- X[ExpInd==2,] + matrix( 5*rt( nI*p,df=3),ncol=p)
      ## add hidden variables
      W <- rnorm(n) * 5
      X <- X + outer(W, rep(1,4))

      ## first two variables are the causal predictors of Y
      beta <- c(1,1,0,0)
    ## response variable Y
      Y <- as.numeric(X%*%beta - 2*W + rnorm(n))


####### Compute "hidden Invariant Causal Prediction” Confidence Intervals
      icp <- hiddenICP(X,Y,ExpInd,alpha=0.01)
      print(icp)

 ###### Print point estimates and points in the confidence interval closest to 0
      print(icp$betahat)
      print(icp$maximinCoefficients)
      cat(”true coefficients are:”, beta)

 #### compare with coefficients from a linear model
      cat(”coefficients from linear model:”)
      print(summary(lm(Y ~ X-1)))




##########################################
####### 2nd example:
####### Simulate model X -> Y -> Z with hidden variables, trying to
######  estimate causal effects from (X,Z) on Y
      set.seed(1)
    ## sample size n
      n <- 10000
    ## simulate as independent Gaussian variables
      W <- rnorm(n)
      noiseX <- rnorm(n)
```

```
     noiseY <- rnorm(n)
     noiseZ <- rnorm(n)
   ## divide data into observational (ExpInd=1) and interventional (ExpInd=2)
     ExpInd <- c(rep(1,n/2),rep(2,n/2))
     noiseX[ which(ExpInd==2)] <- noiseX[ which(ExpInd==2)] * 5
     noiseZ[ which(ExpInd==2)] <- noiseZ[ which(ExpInd==2)] * 3

   ## simulate equilibrium data
     beta <- -0.5
     alpha <- 0.9
     X <- noiseX + 3*W
     Y <- beta* X + noiseY + 3*W
     Z <- alpha*Y + noiseZ


####### Compute "Invariant Causal Prediction" Confidence Intervals
     icp <- hiddenICP(cbind(X,Z),Y,ExpInd,alpha=0.1)
     print(icp)

###### Print/plot/show summary of output (truth here is (beta,0))
     print(signif(icp$betahat,3))
     print(signif(icp$maximinCoefficients,3))
     cat("true coefficients are:", beta,0)

#### compare with coefficients from a linear model
     cat("coefficients from linear model:")
     print(summary(lm(Y ~ X + Z -1)))
```

ICP *Invariant Causal Prediction*

Description

Confidence intervals for causal effects in a regression setting.

Usage

```
ICP(X, Y, ExpInd, alpha=0.01, test = "normal",
              selection = c("lasso", "all", "stability", "boosting")
                                      [if (ncol(X) <=8) 2 else 4],
              maxNoVariables = 8, maxNoVariablesSimult = 8,
              maxNoObs = 200,
              showAcceptedSets = TRUE,
              showCompletion = TRUE,
              stopIfEmpty = FALSE,
              gof = max(0.01,alpha))
```

Arguments

`X` A matrix (or data frame) with the predictor variables for all experimental settings

`Y` The response or target variable of interest. Can be numeric for regression or a factor with two levels for binary classification.

`ExpInd` Indicator of the experiment or the intervention type an observation belongs to. Can be a numeric vector of the same length as `Y` with `K` unique entries if there are `K` different experiments (for example entry 1 for all observational data and entry 2 for intervention data). Can also be a list, where each element of the list contains the indices of all observations that belong to the corresponding grouping in the data (for example two elements: first element is a vector that contains indices of observations that are observational data and second element is a vector that contains indices of all observations that are of interventional type).

`alpha` Determines coverage of the confidence regions. `alpha` is the probability of non-coverage, so the default of 0.01 produces 99% confidence intervals. If no model has a p-value of at least `alpha`, the model assumptions seem violated and `modelReject` will return `TRUE`.

`test` Use "exact" for an exact test in a regression setting, especially if sample size is small. However, this test is computationally demanding if sample size is high. The default "normal" tests for a shift in mean and variance of the residuals between different populations/environments. Using "correlation" tests additionally for vanishing correlation between predictor variables and residuals in each environment. Other options are "ranks" that uses a rank-based alternative and "ks" for a Kolmogorov-Smirnov test to detect differences in the distributions of the residuals between different environments. It is also possible to supply a function of the form `function(x,z)` that takes samples `x` and `z` of two populations as arguments and whose return value is the p-value for the null hypothesis that the two underlying distribution are identical.

`selection` The method for pre-selection of variables to save computational resources. Can use "all" for no pre-selection (which guarantees coverage but might take longer to compute), "boosting" for a boosting-type, "lasso" for Lasso cross-validated or "stability" for a stability-selection-type pre-selection. Default is "all" if `p` does not exceed 10 and "boosting" otherwise.

`maxNoVariables` The maximal number of variables to pre-select (choosing smaller values saves computational resources but increases approximation error).

`maxNoVariablesSimult`
The maximal size of sets of variables considered in the procedure (same comment as for `maxNoVariables`).

`maxNoObs` The maximal number of observations used for the "exact" test (same comment as for `maxNoVariables`).

`showAcceptedSets`
If `TRUE`, print out information about accepted sets of variables.

`showCompletion` If `TRUE`, print out information about progress of computation.

`stopIfEmpty` If `TRUE`, the procedure will stop computing confidence intervals if the empty set has been accepted (and hence no variable can have a signicificant causal effect).

Setting to `TRUE` will save computational time in these cases, but means that the confidence intervals lose their coverage properties for values different to 0.

`gof` If no set of variables (including the empty set) leads to a p-value larger than the goodness-of-fit cutoff `gof`, the whole model will be rejected. If the model is correct, this will happen with a probability of `gof` and this option protects again making statements when the model is obviously not suitable for the data.

Value

A list with elements

`ConfInt` The matrix with confidence intervals for the causal coefficient of all variables. First row is the upper bound and second row the lower bound.

`maximinCoefficients` The value in the confidence interval closest to 0. Is hence non-zero for variables with significant effects.

`colnames` The names of the variables (replaced with generic "Variable 1" etc. if not available).

`factor` Logical indicating whether the response is a factor or not.

`dimX` The dimensions of the matrix with predictor variables.

`Coeff` A list which contains for all variables the vector with point-estimates among all accepted sets where the variable was part of the set.

`CoeffVar` Same as `Coeff` but with the standard deviation of the point-estimate.

`modelReject` Logical indicating if the whole model was rejected (the p-value of the best fitting model is too low).

`acceptedSets` A list with one element per accepted set. For each accepted model the list entry is a vector that contains the indices of the variables in the accepted set.

`usedvariables` The pre-selected variables if not all variables are used for the analysis; otherwise all variables.

`pvalues` The p-values of all variables.

`stopIfEmpty` A boolean value indicating whether computations stop as soon as intersection of accepted sets is empty.

`noEnv` The number of distinct environments.

`gof` The goodness-of-fit cutoff `gof` used.

`bestModel` The largest p-value across all tested sets of variables.

Author(s)

Nicolai Meinshausen <meinshausen@stat.math.ethz.ch>

References

Jonas Peters, Peter Buhlmann, Nicolai Meinshausen (2015):

Causal inference using invariant prediction: identification and confidence intervals

arxiv preprint http://arxiv.org/abs/1501.01332

See Also

hiddenICP for reconstructing the parents of a variable in the presence of hidden variables (but assuming shift/additive interventions), which also allows construction of confidence intervals for the causal coefficients. See package "backShift" for constructing point estimates of causal cyclic models in the presence of hidden variables (again under shift interventions).

Examples

```
 ##########################################
 ####### 1st example:
 ####### Simulate data with interventions
set.seed(1)
    ## sample size n
      n <- 4000
    ## 5 predictor variables
      p  <- 5
    ## simulate as independent Gaussian variables
      X <- matrix(rnorm(n*p),nrow=n)
    ## divide data into observational (ExpInd=1) and interventional (ExpInd=2)
      ExpInd <- c(rep(1,n/2),rep(2,n/2))
    ## for interventional data (ExpInd==2): change distribution
      X[ExpInd==2,] <- sweep(X[ExpInd==2,],2, 5*rnorm(p) ,FUN=”*”)
    ## first two variables are the causal predictors of Y
      beta <- c(1,1,rep(0,p-2))
    ## response variable Y
      Y <- as.numeric(X%*%beta + rnorm(n))
   ## optinal: make last variable a child of Y (so last variable is non-causal for Y)
      X[,p] <- 0.3*Y + rnorm(n)

 ####### Compute ”Invariant Causal Prediction” Confidence Intervals
      icp <- ICP(X,Y,ExpInd)

 ###### Print/plot/show summary of output
      print(icp)
      plot(icp)

#### compare with linear model
      cat(”\n compare with linear model  \n”)
      print(summary(lm(Y~X)))


##########################################
 ####### 2nd example:
 ####### Simulate a DAG where X1 -> Y, Y -> X2 and Y -> X3
 ####### noise interventions on second half of data on X1
 ####### structure of DAG (at Y -> X2) is changing under interventions
     n1 <- 400
     n2 <- 500
     ExpInd <- c(rep(1,n1), rep(2,n2))
   ## index for observational (ExpInd=1) and intervention data (ExpInd=2)
     X1 <- c(rnorm(n1), 2 * rnorm(n2) + 1)
     Y <- 0.5 * X1 + 0.2 * rnorm(n1 + n2)
     X2 <- c(1.5 * Y[1:n1]      + 0.4 * rnorm(n1),
            -0.3 * Y[(n1+1):n2] + 0.4 * rnorm(n2))
     X3 <-  -0.4 * Y            + 0.2 * rnorm(n1 + n2)
     X <- cbind(X1, X2, X3)

 ### Compute ”Invariant Causal Prediction” Confidence Intervals
  ## use a rank-based test to detect shift in distribution of residuals
     icp <- ICP(X, Y, ExpInd,test=”ranks”)
  ## use a Kolmogorov-Smirnov test to detect shift in distribution of residuals
     icp <- ICP(X, Y, ExpInd,test=”ks”)
  ## can also supply test as a function
  ## here chosen to be equivalent to option ”ks” above
     icp <- ICP(X, Y, ExpInd,test=function(x,z) ks.test(x,z)$p.val)
 ## use a test based on normal approximations
     icp <- ICP(X, Y, ExpInd, test=”normal”)


 ### Print/plot/show summary of output
     print(icp)
     plot(icp)

#### compare with linear model
     cat(”\n compare with linear model \n”)
     print(summary(lm(Y~X)))




## Not run:
 ##########################################
 ####### 3rd example:
 ####### College Distance data
     library(AER)
     data(”CollegeDistance”)
     CD <- CollegeDistance

  ##  define two experimental settings by
  ##  distance to closest 4-year college
     ExpInd <- list()
     ExpInd[[1]] <- which(CD$distance < quantile(CD$distance,0.5))
     ExpInd[[2]] <- which(CD$distance >= quantile(CD$distance,0.5))

  ## target variable is binary (did education lead at least to BA degree?)
     Y <- as.factor(CD$education>=16)
  ## use these predictors
     X <- CD[,c(”gender”,”ethnicity”,”score”,”fcollege”,”mcollege”,”home”,
        ”urban”,”unemp”,”wage”,”tuition”,”income”,”region”)]

  ## searching all subsets (use selection=”lasso” or selection=”stability”
  ##     to select a subset of subsets to search)
  ## with selection=”all” the function will take several minutes
    icp <- ICP(X,Y,ExpInd,selection=”all”,alpha=0.1)
  ## Print/plot/show summary of output
     print(icp)
     summary(icp)
     plot(icp)


## End(Not run)
```

plot.InvariantCausalPrediction

Plots of "InvariantCausalPrediction" objects

Description

Plots confidence intervals for invariant causal prediction (output of `ICP` function)

Usage

```
## S3 method for class 'InvariantCausalPrediction'
plot(x,
                   maxshow = 50, col1 = NULL, col2 = NULL, col3 = NULL,
                   mar = c(10, 4, 3, 1), lwd = 1, ...)
```

Arguments

`x`	Object of class "InvariantCausalPrediction", as generated by function `ICP`.
`maxshow`	Maximal number of variables to show confidence intervals for
`col1`	Colour of confidence intervals generated by accepted sets of variables (defaults to light red).
`col2`	Colour of point-estimates generated by accepted sets of variables (defaults to light red).
`col3`	Colour of confidence intervals generated by procedure (defaults to blue).
`mar`	Margins for the figure (might need to be adjusted to allow for long variable names.
`lwd`	Scaling of the width of the lines used for individual confidence intervals.
`...`	Additional inputs to generic `plot` function (not used).

Value

Does not return an object.

Author(s)

Nicolai Meinshausen <meinshausen@stat.math.ethz.ch>

References

Jonas Peters, Peter Buhlmann, Nicolai Meinshausen (2015):

Causal inference using invariant prediction: identification and confidence intervals

arxiv preprint http://arxiv.org/abs/1501.01332

See Also

ICP

---

summary.InvariantCausalPrediction

Summary of "InvariantCausalPrediction" objects

Description

Prints confidence intervals for invariant causal prediction (output of ICP function)

Usage

```
## S3 method for class 'InvariantCausalPrediction'
summary(object, maxshow = 50, ...)
```

Arguments

object	Object of class "InvariantCausalPrediction", as generated by function ICP.
maxshow	Maximal number of variables to show in the summary.
...	Additional inputs to generic summary function (not used).

Value

Does not return an object.

Author(s)

Nicolai Meinshausen <meinshausen@stat.math.ethz.ch>

References

Jonas Peters, Peter Buhlmann, Nicolai Meinshausen (2015):

Causal inference using invariant prediction: identification and confidence intervals

arxiv preprint http://arxiv.org/abs/1501.01332

See Also

ICP

Index