Package 'mrregression'

September 22, 2020

Type Package

Title Regression Analysis for Very Large Data Sets via Merge and Reduce

Version 1.0.0

Author Esther Denecke [aut], Leo N. Geppert [aut, cre], Steffen Maletz [ctb], R Core Team [ctb]

Maintainer Leo N. Geppert <geppert@statistik.uni-dortmund.de>

Description Frequentist and Bayesian linear regression for large data sets. Useful when the data does not fit into memory (for both frequentist and Bayesian regression), to make running time manageable (mainly for Bayesian regression), and to reduce the total running time because of reduced or less severe memory-spillover into the virtual memory. This is an implementation of Merge & Reduce for linear regression as described in Geppert, L.N., Ickstadt, K., Munteanu, A., & Sohler, C. (2020). 'Streaming statistical models via Merge & Reduce'. International Journal of Data Science and Analytics, 1-17, <doi:10.1007/s41060-020-00226-0>.

Depends R (>= 4.0.0), Rcpp (>= 1.0.5),

License GPL-2 | GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Suggests testthat (>= 2.3.2),

Imports data.table (>= 1.12.8),

Enhances rstan (>= 2.19.3),

NeedsCompilation no

Repository CRAN

Date/Publication 2020-09-22 08:20:02 UTC

R topics documented:

`exampleData` *Simulated Example Data*

Description

Simulated data set with 1500 observations for illustrational purposes.

Usage

```
exampleData
```

Format

A data frame with 1500 rows and 11 variables where V1-V10 are the predictors and V11 is the dependent variable.

`mrbayes` *Bayesian linear regression using Merge and Reduce*

Description

`mrbayes` is used to conduct Bayesian linear regression on very large data sets using Merge and Reduce as described in Geppert et al. (2020). Package `rstan` needs to be installed. When calling the function this is checked using `requireNamespace` as suggested by Hadley Wickham in "R packages" (section Dependencies, http://r-pkgs.had.co.nz/description.html, accessed 2020-07-31).

Usage

```
mrbayes(
  y,
  intercept = TRUE,
  fileMr = NULL,
  dataMr = NULL,
  obsPerBlock,
  dataStan = NULL,
  sep = "auto",
  dec = ".",
  header = TRUE,
  naStrings = "NA",
  colNames = NULL,
  naAction = na.fail,
  ...
)
```

Arguments

y	(character) Column name of the dependent variable.
intercept	(logical) Argument specifying whether the model should have an intercept term or not. Defaults to TRUE.
fileMr	(character) The name of a file, including the filepath, to be read in blockwise. Either fileMr or dataMr needs to be specified. When using this argument, the arguments sep, dec, header, naStrings, colNames (as in fread) are of relevance. Further options from fread are currently not supported. Also note that defaults might differ. In case the data to be read in has row names, note that these will be read in as regular column. This may need special treatment.
dataMr	(data.frame) The data to be used for the regression analysis. Either fileMr or dataMr needs to be specified. Note that the arguments sep, dec, header, naStrings, and colNames are ignored when dataMr is specified.
obsPerBlock	(numeric) Value specifying the number of observations in each block. This number has to be larger than the number of regression coefficients. Moreover, the recommended ratio of observations per regression coefficient is larger than 25 (Geppert et al., 2020). Note that the last block may contain less observations than specified depending on the sample size. If the number of observations in this last block is too small it is not included in the model and a warning is issued.
dataStan	(list) Optional argument. This argument is equivalent to the argument data in stan. If not specified the default dataStan, which makes use of all predictors is used. See section Details for the default dataStan and further notes on the syntax to be used when specifiying this argument.
sep	See documentation of fread. Default is "auto". Ignored when dataMr is specified.
dec	See documentation of fread. Default is ".". Ignored when dataMr is specified.
header	(logical) See documentation of fread. Defaults to TRUE. Ignored when dataMr is specified. If header is set to FALSE and no colNames are given, then column names default to "V" followed by the column number.
naStrings	(character) Optional argument. See argument na.strings of fread. Default is "NA". Ignored when dataMr is specified and optional when fileMr is used.
colNames	(character vector) Same as argument col.names of fread. Ignored when dataMr is specified and optional when fileMr is used.
naAction	(function) Action to be taken when missing values are present in the data. Currently only na.fail is supported.

`...` Further optional arguments to be passed on to `stan`, especially `pars` and arguments that control the behaviour of the sampling in `rstan` such as `chains`, `iter`, `warmup`, and `thin`. Please refer to `rstan`.

Value

Returns an object of class `"mrbayes"` which is a list containing the following components:

`level` Number of level of the final model in Merge and Reduce. This is equal to $\lceil \log_2 (\texttt{numberObs}/\texttt{obsPerBlock}) \rceil + 1$ and corresponds to the number of buckets in Figure 1 of Geppert et al. (2020).

`numberObs` The total number of observations.

`summaryStats` Summary statistics including the mean, median, quartiles, 2.5% and 97.5% quantiles of the posterior distributions for each regression coefficient and the error term's standard deviation sigma.

`diagnostics` Effective sample size (n_eff) and potential scale reduction factor on split chains (Rhat) calculated from the output of summary,stanfit-method. Note that, using Merge and Reduce, for each regression coefficient only one value is reported: For n_eff the minimum observed value on level 1 is reported and for Rhat the maximum observed value on level 1 is reported.

`modelCode` The model. Syntax as in argument `model_code` of stan.

`dataHead` First six rows of the data in the first block. This serves as a sanity check, especially when using the argument `fileMr`.

Details

Code of default `dataStan` makes use of all predictors:

```
dataStan = list(n = nrow(currentBlock),
d = (ncol(currentBlock) -1),
X = currentBlock[,-colNumY],
y = currentBlock[,colNumY])
```

where `currentBlock` is the current block of data to be evaluated, `n` the number of observations, `d` the number of variables (without intercept), `X` contains the predictors, and `y` the dependent variable. `colNumY` is the column number of the dependent variable that the function finds internally.

When specifying the argument `dataStan`, note two things:
1. Please use the syntax of the default `dataStan`, i.e. the object containing the data of the block to be evaluated is called `currentBlock`, the number of observations must be set to `n = nrow(currentBlock)`, `d` needs to be set to the number of variables without intercept, the dependent variable must be named `y`, and the independent variables must be named `X`.
2. The expressions within the list must be unevaluated: Therefore, use the function `quote`.

References

Geppert, L.N., Ickstadt, K., Munteanu, A., & Sohler, C. (2020). Streaming statistical models via Merge & Reduce. International Journal of Data Science and Analytics, 1-17,
doi: https://doi.org/10.1007/s41060-020-00226-0

Examples

```
# Package rstan needs to be installed for running this example.

if (requireNamespace("rstan", quietly = TRUE)) {
  n = 2000
  p = 4
  set.seed(34)
  x1 = rnorm(n, 10, 2)
  x2 = rnorm(n, 5, 3)
  x3 = rnorm(n, -2, 1)
  x4 = rnorm(n, 0, 5)
  y = 2.4 - 0.6 * x1 + 5.5 * x2 - 7.2 * x3 + 5.7 * x4 + rnorm(n)
  data = data.frame(x1, x2, x3, x4, y)

  normalmodell = '
  data {
  int<lower=0> n;
  int<lower=0> d;
  matrix[n,d] X; // predictor matrix
  vector[n] y; // outcome vector
  }
  parameters {
  real alpha; // intercept
  vector[d] beta; // coefficients for predictors
  real<lower=0> sigma; // error scale
  }
  model {
  y ~ normal(alpha + X * beta, sigma); // likelihood
  }
  '

  datas = list(n = nrow(data), d = ncol(data)-1,
               y = data[, dim(data)[2]], X = data[, 1:(dim(data)[2]-1)])
  fit0 = rstan::stan(model_code = normalmodell, data = datas, chains = 4, iter = 1000)
  fit1 = mrbayes(dataMr = data, obsPerBlock = 500, y = 'y')
}
```

`mrfrequentist` *Fitting frequentist linear models using Merge and Reduce*

Description

`mrfrequentist` is used to conduct frequentist linear regression on very large data sets using Merge and Reduce as described in Geppert et al. (2020).

Usage

```
mrfrequentist(
```

```
  formula,
  fileMr = NULL,
  dataMr = NULL,
  obsPerBlock,
  approach = c("1", "3"),
  sep = "auto",
  dec = ".",
  header = TRUE,
  naStrings = "NA",
  colNames = NULL,
  naAction = na.fail
)
```

Arguments

formula	(formula) See formula. Note that mrfrequentist currently supports numeric predictors only.
fileMr	(character) The name of a file, including the filepath, to be read in blockwise. Either fileMr or dataMr needs to be specified. When using this argument, the arguments sep, dec, header, naStrings, colNames (as in fread) are of relevance. Further options from fread are currently not supported. Also note that defaults might differ. In case the data to be read in has row names, note that these will be read in as regular column. This may need special treatment.
dataMr	(data.frame) The data to be used for the regression analysis. Either fileMr or dataMr needs to be specified. Note that the arguments sep, dec, header, naStrings, and colNames are ignored when dataMr is specified.
obsPerBlock	(numeric) Value specifying the number of observations in each block. This number has to be larger than the number of regression coefficients. Moreover, for approach 1 the recommended ratio of observations per regression coefficient is larger than 25 (Geppert et al., 2020). Note that the last block may contain less observations than specified depending on the sample size. If the number of observations in this last block is too small it is not included in the model and a warning is issued.
approach	(character) Approach specifying the merge technique. One of either "1" or "3". Approach "1" is based on a weighted mean procedure whereas approach "3" is an exact method based on blockwise calculations of X'X, y'X and y'y. See Geppert et al. (2020) for details on the approaches and section Details below for comments on approach "3".
sep	See documentation of fread. Default is "auto". Ignored when dataMr is specified.
dec	See documentation of fread. Default is ".". Ignored when dataMr is specified.
header	(logical) See documentation of fread. Defaults to TRUE. Ignored when dataMr is speci-

fied. If header is set to FALSE and no colNames are given, then column names default to "V" followed by the column number.

naStrings (character)
Optional argument. See argument na.strings of fread. Default is "NA". Ignored when dataMr is specified and optional when fileMr is used.

colNames (character vector)
Same as argument col.names of fread. Ignored when dataMr is specified and optional when fileMr is used.

naAction (function)
Action to be taken when missing values are present in the data. Currently only na.fail is supported.

Value

Returns an object of class "mrfrequentist" which is a list containing the following components **for both approaches "1" and "3":**

approach The approach used for merging the models. Either "1" or "3".

formula The model's formula.

level Number of level of the final model in Merge and Reduce. This is equal to $\lceil \log_2 (\texttt{numberObs}/\texttt{obsPerBlock}) \rceil + 1$ and corresponds to the number of buckets in Figure 1 of Geppert et al. (2020).

numberObs The total number of observations.

summaryStats Summary statistics reporting the estimated regression coefficients and their unbiased standard errors. Estimates are based on the merge technique as specified in the argument approach. For approach "1" the estimates of the standard errors are corrected dividing by $\sqrt{\lceil \texttt{numberObs / obsPerBlock} \rceil}$. For further details see Geppert et al. (2020). For approach "3" the unbiased estimates of the standard errors are given.

dataHead First six rows of the data in the first block. This serves as a sanity check, especially when using the argument fileMr.

terms Terms object.

Additionally for approach "3" only:

XTX The final model's crossprod(X,X).

yTX The final model's crossprod(y,X).

yTy The final model's crossprod(y,y).

Details

In approach "3" the estimated regression coefficients and their unbiased standard errors are calculated via qr decompositions on X'X (as in speedlm with argument method = "qr"). Moreover, the merge step uses the same idea of blockwise addition for X'X, y'y and y'X as speedglm's updating procedure updateWithMoreData. Conceptually though, Merge and Reduce is not an updating algorithm as it merges models based on a comparable amount of data along a tree structure to obtain a final model.

References

Geppert, L.N., Ickstadt, K., Munteanu, A., & Sohler, C. (2020). Streaming statistical models via Merge & Reduce. International Journal of Data Science and Analytics, 1-17, doi: https://doi.org/10.1007/s41060-020-00226-0

Examples

```
## run mrfrequentist() with dataMr
data(exampleData)
fit1 = mrfrequentist(dataMr = exampleData, approach = "1", obsPerBlock = 300,
formula = V11 ~ .)

## run mrfrequentist() with fileMr
filepath = system.file("extdata", "exampleFile.txt", package = "mrregression")
fit2 = mrfrequentist(fileMr = filepath, approach = "3", header = TRUE,
obsPerBlock = 100, formula = y ~ .)
```

`mrregression` *mrregression: Frequentist and Bayesian linear regression using Merge and Reduce.*

Description

Frequentist and Bayesian linear regression for large data sets. Useful when the data does not fit into memory (for both frequentist and Bayesian regression), to make running time manageable (mainly for Bayesian regression), and to reduce the total running time because of reduced or less severe memory-spillover into the virtual memory. The package contains the two main functions `mrfrequentist` and `mrbayes` as well as several S3 methods listed below. Note, that currently only numerical predictors are supported. Factor variables can be included in the model in dummy-coded form, e.g. using `model.matrix`. However, this may lead to highly variable or even unreliable estimates / posterior distributions if levels are not represented well in every single block. It is solely the user's responsibility to check that this is not the case!

Usage

```
## S3 method for class 'mrfrequentist'
coef(object, ...)

## S3 method for class 'mrfrequentist'
nobs(object, ...)

## S3 method for class 'mrfrequentist'
predict(object, data, ...)

## S3 method for class 'mrfrequentist'
summary(object, ...)

## S3 method for class 'summary.mrfrequentist'
```

```
print(x, ...)

## S3 method for class 'mrbayes'
nobs(object, ...)

## S3 method for class 'mrbayes'
summary(object, ...)

## S3 method for class 'summary.mrbayes'
print(x, ...)
```

Arguments

object	Object of class "mrfrequentist" or "mrbayes", respectively.
...	Currently only useful for method print.summary.mrfrequentist and approach "3". See arguments to function printCoefmat, especially digits and signif.stars.
data	A data.frame used to predict values of the dependent variable. Data has to contain all variables in the model, additional columns are ignored. Note that this is not an optional argument.
x	Object of class "summary.mrfrequentist" or "summary.mrbayes", respectively.

References

Geppert, L.N., Ickstadt, K., Munteanu, A., & Sohler, C. (2020). Streaming statistical models via Merge & Reduce. International Journal of Data Science and Analytics, 1-17, doi: https://doi.org/10.1007/s41060-020-00226-0

Index