Package 'changepoint.np'

May 15, 2020

Type Package

Title Methods for Nonparametric Changepoint Detection

Version 1.0.2

Date 2020-05-15

Description Implements the multiple changepoint algorithm PELT with a nonparametric cost function based on the empirical distribution of the data. This package extends the changepoint package (see Killick, R and Eckley, I (2014) <doi:10.18637/jss.v058.i03>).

Depends changepoint

RdMacros Rdpack

Imports methods,zoo,Rdpack

License GPL

RoxygenNote 7.0.2

NeedsCompilation yes

Author Kaylea Haynes [aut],
Rebecca Killick [aut],
Paul Fearnhead [ths, ctb],
Idris Eckley [ths],
Daniel Grose [ctb, cre]

Maintainer Daniel Grose <changepoints@lancaster.ac.uk>

Repository CRAN

Date/Publication 2020-05-15 11:30:02 UTC

R topics documented:

---

cpt.np   *Identifying Changes using a Nonparametric Cost Function*

Description

Calculates the optimal positioning and number of changepoints for data given a user specified cost function and penalty.

Usage

```
cpt.np(
  data,
  penalty = "MBIC",
  pen.value = 0,
  method = "PELT",
  test.stat = "empirical_distribution",
  class = TRUE,
  minseglen = 1,
  nquantiles = 10
)
```

Arguments

data	A vector, ts object or matrix containing the data within which you wish to find a changepoint. If the data is a matrix, each row is considered as a separate dataset.
penalty	Choice of "None", "SIC", "BIC", "MBIC", AIC", "Hannan-Quinn", "Manual" and "CROPS" penalties. If Manual is specified, the manual penalty is contained in the pen.value parameter. If CROPS is specified, the penalty range is contained in the pen.value parameter; note this is a vector of length 2 which contains the minimum and maximum penalty value. Note CROPS can only be used if the method is "PELT". The predefined penalties listed DO count the changepoint as a parameter, postfix a 0 e.g."SIC0" to NOT count the changepoint as a parameter.
pen.value	The value of the penalty when using the Manual penalty option. A vector of length 2 (min,max) if using the CROPS penalty.
method	Currently the only method is "PELT".
test.stat	The assumed test statistic/distribution of the data. Currently only "empirical_distribution".
class	Logical. If TRUE then an object of class cpt is returned.
minseglen	Positive integer giving the minimum segment length (number of observations between changes), default is the minimum allowed by theory.
nquantiles	The number of quantiles to calculate when test.stat = "empirical_distribution".

Details

This function is used to find multiple changes in a data set using the changepoint algorithm PELT with a nonparametric cost function based on the empirical distribution. A changepoint is denoted as the first observation of the new segment.

Value

If class=TRUE then an object of S4 class "cpt" is returned. The slot cpts contains the changepoints that are returned. For class=FALSE the structure is as follows.

If data is a vector (single dataset) then a vector/list is returned depending on the value of method. If data is a matrix (multiple datasets) then a list is returned where each element in the list is either a vector or list depending on the value of method.

If method is PELT then a vector is returned containing the changepoint locations for the penalty supplied. If the penalty is CROPS then a list is returned with the elements:

cpt.out   A data frame containing the value of the penalty value where the number of segmentations changes, the number of segmentations and the value of the cost at that penalty value.

changepoints   The optimal changepoints for the different penalty values starting with the lowest penalty value.

Author(s)

Kaylea Haynes

References

Haynes K, Fearnhead P, Eckley IA (2017). "A computationally efficient nonparametric approach for changepoint detection." *Statistics and Computing*, **27**(5), 1293–1305. ISSN 1573-1375, doi: 10.1007/s1122201696875, https://doi.org/10.1007/s11222-016-9687-5.

Killick R, Fearnhead P, Eckley I (2012). "Optimal Detection of Changepoints With a Linear Computational Cost." *Journal of the American Statistical Association*, **107**, 1590-1598. doi: 10.1080/01621459.2012.737745.

Haynes K, A. Eckley I, Fearnhead P (2015). "Computationally Efficient Changepoint Detection for a Range of Penalties." *Journal of Computational and Graphical Statistics*, **26**, 1-28. doi: 10.1080/10618600.2015.1116445.

See Also

PELT in parametric settings: cpt.mean for changes in the mean, cpt.var for changes in the variance and cpt.meanvar for changes in the mean and variance.

Examples

```
#Example of a data set of length 1000 with changes in location
#(model 1 of Haynes, K et al. (2016)) with the empirical distribution cost function.

set.seed(12)

J <- function(x){
  (1+sign(x))/2
 }
```

```
n <- 1000
tau <- c(0.1,0.13,0.15,0.23,0.25,0.4,0.44,0.65,0.76,0.78,0.81)*n
h <- c(2.01, -2.51, 1.51, -2.01, 2.51, -2.11, 1.05, 2.16, -1.56, 2.56, -2.11)
sigma <- 0.5
t <- seq(0,1,length.out = n)
data <- array()
for (i in 1:n){
    data[i] <- sum(h*J(n*t[i] - tau)) + (sigma * rnorm(1))
}

out <- cpt.np(data, penalty = "SIC",method="PELT",test.stat="empirical_distribution",
              class=TRUE,minseglen=2, nquantiles =4*log(length(data)))
cpts(out)
#returns 100 130 150 230 250 400 440 650 760 780 810 as the changepoint locations.
plot(out)

#Example 2 uses the heart rate data .

data(HeartRate)
cptHeartRate <- cpt.np(HeartRate, penalty = "CROPS", pen.value = c(5,200),
                       method="PELT", test.stat="empirical_distribution",
                       class=TRUE,minseglen=2,
                       nquantiles =4*log(length(HeartRate)))
plot(cptHeartRate, diagnostic = TRUE)
plot(cptHeartRate, ncpts = 11)
```

HeartRate   *Recorded heart rate during a run*

Description

A dataset containing heart-rate recorded during a run.

Usage

```
data(HeartRate)
```

Format

A vector of recorded heart rates at points over time with 1160 data points.

Author(s)

Kaylea Haynes

References

Haynes K, Fearnhead P, Eckley IA (2017). "A computationally efficient nonparametric approach for changepoint detection." *Statistics and Computing*, **27**(5), 1293–1305. ISSN 1573-1375, doi: 10.1007/s1122201696875, https://doi.org/10.1007/s11222-016-9687-5.

Index