Package ‘ACEt’

May 7, 2017

Type Package
Title Estimating Dynamic Heritability and Twin Model Comparison
Version 1.8.0
Date 2017-04-29
Author Liang He
Maintainer Liang He <lianghe@mit.edu>
Description Twin models that are able to estimate the dynamic behaviour of the variance components in the classical twin models with respect to age using B-splines and P-splines.
License GPL (>= 2)
Depends R (>= 3.1.1)
Imports Rcpp (>= 0.11.1), splines, MASS
Encoding UTF-8
LinkingTo BH, RcppArmadillo, Rcpp
Repository CRAN
Repository/R-Forge/Project acet
Repository/R-Forge/Revision 41
Repository/R-Forge/DateTimeStamp 2017-05-07 03:20:51
Date/Publication 2017-05-07 20:51:57 UTC
NeedsCompilation yes

R topics documented:

ACEt-package ................................................. 2
acetp_mcmc .................................................. 3
AtCtEt ....................................................... 4
AtCtEtp ..................................................... 6
data_ace .................................................... 8
plot_ace ..................................................... 8
test_acet ................................................... 9

Index 11
ACEt-package

Estimating Age Modification Effect on Genetic and Environmental Variance Components in Twin Models

Description

The package implements two novel twin models that are able to estimate dynamic behaviour of the variance components in the classical twin models with respect to age or other covariates such as calendar year.

Details

Package: ACEt
Type: Package
Version: 1.8.0
Date: 2017-04-29
License: GPL

Author(s)

Liang He
Maintainer: Liang He <lianghe@mit.edu>

References


Examples

# data(data_ace)

# AtCtEt(data_ace$mz, data_ace$dz, mod=c('d','d','c'), knot_a = 5, knot_c = 4)
# AtCtEtP(data_ace$mz, data_ace$dz, knot_a = 6, mod=c('d','l','c'))
**acetp_mcmc**  

*Compute CIs for the ACE(t)-p model*

---

**Description**

Compute the posterior mean and CIs for the ACE(t)-p model using the MCMC methods

**Usage**

`acetp_mcmc(acetp, iter_num = 10000, sd = 0.1, burnin = 1000)`

**Arguments**

- **acetp**: An object from the 'AtCtEtp' function.
- **iter_num**: The number of the iterations in the MCMC procedure.
- **sd**: The standard error of the normal proposal distribution in the MCMC algorithm. The default value is 0.1.
- **burnin**: The number of burn-in, which must be smaller than the number of iteration.

**Value**

- **beta_a_mc**: The estimates of the spline coefficients for the A component based on the posterior mean from the MCMC method.
- **beta_c_mc**: The estimates of the spline coefficients for the C component based on the posterior mean from the MCMC method.
- **beta_e_mc**: The estimates of the spline coefficients for the E component based on the posterior mean from the MCMC method.
- **cov_mc**: The posterior covariance matrix of the estimates of the spline coefficients.
- **knots_a**: A vector of the knot positions for the A component.
- **knots_c**: A vector of the knot positions for the C component.
- **knots_e**: A vector of the knot positions for the E component.

**Author(s)**

Liang He

**References**

Examples

```r
# data(data_ace)
# result <- AtCtEt(data_ace$mz, data_ace$dz, knot_a = 7, knot_c = 7)
# result_mc <- acetp_mcmc(result, iter_num=10000, burnin = 500)
```

---

**AtCtEt**

*Fitting the ACE(t) model*

---

**Description**

The ACE(t) model with the A, C and E variance components as functions with respect to age modelled by B-splines.

**Usage**

```r
AtCtEt(data_m, data_d, mod = c('d', 'd', 'd'), knot_a = 5, knot_c = 5, knot_e = 5,
loc = c('e', 'e', 'e'), boot=FALSE, num_b = 100, init = rep(0,3), robust = 0)
```

**Arguments**

- **data_m**
  - An $N_m \times 3$ data matrix for MZ twins. $N_m$ is the number of MZ twin pairs. The first two columns are centered trait values (i.e. the mean should be zero) and the third column is age (or other covariates).

- **data_d**
  - An $N_d \times 3$ data matrix for DZ twins. $N_d$ is the number of DZ twin pairs. The first two columns are centered trait values (i.e. the mean should be zero) and the third column is age (or other covariates).

- **mod**
  - A character vector of length 3. Each element specifies the function for the A, C or E component respectively. The A and C components can be 'd'(dynamic), 'c'(constant) or 'n'(NA). The E component can only be 'd' or 'c'. Thus, `model = c('d', 'c', 'c')` is corresponding to the classical ACE model.

- **knot_a**
  - The number of interior knots of the B-spline for the A component, which must be no less than 3. The default value is 5.

- **knot_c**
  - The number of interior knots of the B-spline for the C component, which must be no less than 3. The default value is 5.

- **knot_e**
  - The number of interior knots of the B-spline for the E component, which must be no less than 3. The default value is 5.

- **loc**
  - A 1x3 character vector indicating how to place knots for each component: evenly ("e") or quantile-based ("q"). The default value is "e".

- **boot**
  - A logical indicator of whether to use the bootstrap method to calculate the confidence interval. The default is FALSE.

- **num_b**
  - The number of replicates when the bootstrap method is used (i.e. `boot = TRUE`). The default value is 100.
init    A 3x1 vector of the initial values for the optimization. The default values are 1.
robust An integer indicating the number of different initial values that the function will randomly generate and try in the optimization. The default value is 0.

Details
If the variance is close to the boundary (0), it is better to use the bootstrap method to get the CIs. The optimization algorithm may sometimes end up with a local minimum. It is recommended to try different random initial values by setting 'robust'.

Value

n_beta_a The number of spline coefficients for the A component.
n_beta_c The number of spline coefficients for the C component.
n_beta_e The number of spline coefficients for the E component.
beta_a The estimated spline coefficients (if the model parameter is 'd') or variance (if the model parameter is 'c') of the A component.
beta_c The estimated spline coefficients (if the model parameter is 'd') or variance (if the model parameter is 'c') of the C component.
beta_e The estimated spline coefficients (if the model parameter is 'd') or variance (if the model parameter is 'c') of the E component.
hessian_ap The approximated Hessian matrix from the quasi-Newton algorithm.
hessian The Hessian matrix calculated analytically.
con An indicator of convergence of the optimization algorithm. An integer code 0 indicates successful completion. See 'optim' for more details.
lik The minus log-likelihood.
knots_a A vector of the knot positions for the A component.
knots_c A vector of the knot positions for the C component.
knots_e A vector of the knot positions for the E component.
boot A list containing pointwise CIs estimated from the bootstrap method when boot = TRUE

Author(s)
Liang He

References
Examples

```r
data(data_ace)

result <- atctetp(data_ace$mz, data_ace$dz, mod=c('d','d','c'))
```

---

**atctetp**

*Fitting the ACE(t)-p model*

Description

The ACE(t)-p model with the A, C and E variance components as functions with respect to age modelled by P-splines.

Usage

```r
atctetp(data_m, data_d, knot_a = 8, knot_c = 8, knot_e = 8, eps = 0.1, mod=c('d','d','d'), robust = 0)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>data_m</code></td>
<td>An $N_m \times 3$ data matrix for MZ twins. $N_m$ is the number of MZ twin pairs. The first two columns are centered trait values (i.e. the mean should be zero) and the third column is age (or other covariates).</td>
</tr>
<tr>
<td><code>data_d</code></td>
<td>An $N_d \times 3$ data matrix for DZ twins. $N_d$ is the number of DZ twin pairs. The first two columns are centered trait values (i.e. the mean should be zero) and the third column is age (or other covariates).</td>
</tr>
<tr>
<td><code>knot_a</code></td>
<td>The number of interior knots of the B-spline for the A component. The default value is 8.</td>
</tr>
<tr>
<td><code>knot_c</code></td>
<td>The number of interior knots of the B-spline for the C component. The default value is 8.</td>
</tr>
<tr>
<td><code>knot_e</code></td>
<td>The number of interior knots of the B-spline for the E component. The default value is 8.</td>
</tr>
<tr>
<td><code>eps</code></td>
<td>Tolerance for convergence of the EM algorithm iterations. The default value is 0.1.</td>
</tr>
<tr>
<td><code>mod</code></td>
<td>A character vector of length 3. Each element specifies the function for the A, C or E component respectively. The components can be 'd'(dynamic), 'c'(constant) or 'l'(linear). The default is c('d','d','d').</td>
</tr>
<tr>
<td><code>robust</code></td>
<td>An integer indicating the number of different initial values that the function will randomly generate and try in the optimization. The default value is 0.</td>
</tr>
</tbody>
</table>
Details

When the ‘mod’ argument for a component is ‘d’ (dynamic), the corresponding ‘beta’ is the spline coefficients. When the ‘mod’ argument for a component is ‘l’ (linear), the corresponding ‘beta’ is a vector of two values, the exponential of which (exp(beta)) are the variances at the minimum and maximum age (or other covariates) provided in the data. When the ‘mod’ argument for a component is ‘c’ (constant), the corresponding ‘beta’ has only one value and exp(beta) is the variance.

Value

- var_b_a: The estimated variance for the penalized coefficient for the A components.
- var_b_c: The estimated variance for the penalized coefficient for the C components.
- var_b_e: The estimated variance for the penalized coefficient for the E components.
- beta_a: The estimated spline coefficients of the A component. See ‘details’ for more information.
- beta_c: The estimated spline coefficients of the C component. See ‘details’ for more information.
- beta_e: The estimated spline coefficients of the E component. See ‘details’ for more information.
- con: An indicator of convergence of the optimization algorithm. An integer code 0 indicates successful completion. See ‘optim’ for more details.
- lik: The minus log marginal likelihood.
- knot_a: A vector of the knot positions for the A component.
- knot_c: A vector of the knot positions for the C component.
- knot_e: A vector of the knot positions for the E component.

Author(s)

Liang He

References


Examples

```r
# data(data_ace)

# result <- AtCtEtp(data_ace$zmz, data_ace$dz, knot_e = 7, knot_c = 5, mod=c('d', 'd', 'd'))
```
**data_ace**  
*Example data for the ACEt models*

**Description**
This is an example dataset consisting of traits and ages for MZ and DZ twins.

**Usage**
```
data(data_ace)
```

**Format**
The format is: *mz*: a matrix of simulated data for MZ twins. *dz*: a matrix of simulated data for DZ twins.

**Examples**
```
data(data_ace)
```

**plot_acet**  
*Plot variance curves for various ACE(t) or ACE(t)-p models with 95% CIs*

**Description**
Plot variance curves or a heritability curve (with 95% CIs) of the A, C and E components with respect to age modelled by B-splines or P-splines.

**Usage**
```
plot_acet(acet, boot = FALSE, heri = FALSE, xlab, ylab, main, col, legend = TRUE)
```

**Arguments**
- `acet` An object obtained from the B-splines or P-splines functions. For the P-splines functions, an object from the MCMC method must be used.
- `boot` A logical indicator of whether the confidence bands estimated from the bootstrap method are plotted. The default is FALSE. Only available for the AtCtEt model.
- `heri` A logical indicator of whether to plot the dynamic heritability curve. The default is FALSE.
- `xlab` The `xlab` argument in the plot function. The default is ’Age’.
- `ylab` The `ylab` argument in the plot function. The default is ’Variance’.
**Description**

Comparison of different ACE(t)-p models to test a linear or a constant variance component.

**Usage**

test_acetp(acetp, comp, sim = 100, robust = 0, pe = TRUE, verbose = TRUE)

**Arguments**

- **acetp**: An object from the AtCtEtp function.
- **comp**: The component for which linearity or constancy is tested. This component must be specified as splines or linear in the AtCtEtp function.
- **sim**: The number of the bootstrap resampling for approximating the null distribution when testing linearity.
- **robust**: An integer indicating the number of different initial values that the function will randomly generate and try in the optimization. The default value is 0.
pe        A logical argument indicating whether to use penalized spline model to test linearity. The default value is TRUE.
verbose   A logical argument indicating whether to print testing information and results. The default value is TRUE.

Details

When pe=TRUE, the linearity is tested under a p-spline framework in which an LRT is performed. Otherwise, a $\chi^2$ test is performed for linearity under a spline framework without penalty on smoothness.

Value

p         The p-value for the test.
llr       The LRT statistic for testing linearity.
llr_sim   The simulated null distribution of the LRT statistic for testing linearity.
chisq     The chisq statistic for testing a constant or linearity.

Author(s)

Liang He

References


Examples

# data(data_ace)

# result <- AtCtEtp(data_ace$mz, data_ace$dz, knot_e = 7, knot_c = 5, mod=c('d','d','l'))
# re <- test_acetp(result, comp='e')
Index

*Topic acetp_mcmc
  acetp_mcmc, 3
  AtCtEt, 4
  AtCtEtp, 6
  plot_acet, 8
  test_acetp, 9

*Topic atctet
  acetp_mcmc, 3
  AtCtEt, 4
  AtCtEtp, 6
  plot_acet, 8
  test_acetp, 9

*Topic atctetp
  acetp_mcmc, 3
  AtCtEt, 4
  AtCtEtp, 6
  plot_acet, 8
  test_acetp, 9

*Topic plot_acet
  acetp_mcmc, 3
  AtCtEt, 4
  AtCtEtp, 6
  plot_acet, 8
  test_acetp, 9

*Topic test_acetp
  acetp_mcmc, 3
  AtCtEt, 4
  AtCtEtp, 6
  plot_acet, 8
  test_acetp, 9

*Topic datasets
  data_ace, 8

*Topic package
  ACEt-package, 2
  ACEt-package, 2
  acetp_mcmc, 3
  AtCtEt, 4
  AtCtEtp, 6
  data_ace, 8
  plot_acet, 8
  test_acetp, 9