Package 'WeMix'

September 24, 2018

Version 2.1.0

Date 2018-09-17

Title Weighted Mixed-Effects Models, using Multilevel Pseudo Maximum Likelihood Estimation

Author Paul Bailey, Claire Kelley, Trang Nguyen, Huade Huo.

Maintainer Claire Kelley <ckelley@air.org>

Depends lme4, R (>= 3.3.0)

Imports numDeriv, statmod, Rmpfr, NPflow, Rcpp

Suggests testthat, knitr, rmarkdown, EdSurvey

Description Run mixed-effects models that include weights at every level. The WeMix package fits a Weighted Mixed model, also known as a multilevel, mixed, or hierarchical linear model (HLM). The weights could be inverse selection probabilities, such as those developed for an education survey where schools are sampled probabilistically, and then students inside of those schools are sampled probabilistically. Although mixed-effects models are already available in R, WeMix is unique in implementing methods for mixed models using weights at multiple levels. The model is fit using adaptive quadrature.

LinkingTo Rcpp, RcppArmadillo

License GPL-2

VignetteBuilder knitr

LazyData true

ByteCompile true

Note This publication was prepared for NCES under Contract No. ED-IES-12-D-0002 with American Institutes for Research. Mention of trade names, commercial products, or organizations does not imply endorsement by the U.S. government.

RoxygenNote 6.1.0

Encoding UTF-8

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-09-24 15:00:03 UTC

R topics documented:

`WeMix-package` *WeMix: Package to Estimate Weighted Mixed-Effects Models.*

Description

The WeMix package estimates mixed-effects models (also called multilevel models, mixed models, or hierarchical linear models) with survey weights. The likelihood function of such models with complex survey weights is not analytically calculable, so WeMix uses numerical integration (Gauss-Hermite and adaptive Gauss-Hermite quadrature) to estimate mixed-effects models with survey weights at all levels of the model.

Details

This method allows users to analyze data that may have unequal selection probability at both the individual and group levels. Note that lme4 is the preferred way to estimate such models when there are no survey weights or weights only at the lowest level, and our estimation starts with parameters estimated in lme4. WeMix is intended for use in cases where there are weights at all levels,and is only for use with fully nested data.

To start using WeMix, see the vignettes covering the mathematical background of mixed-effects model estimation and use the `mix` function to estimate models. Use `browseVignettes(package="WeMix")` to see the vignettes.

`calc_lin_lnl_quad_fast`

This function calcuates the liklihood of the model using integration by adaptive quadrature

---

Description

calculates the log-likelihood of an l level mixed model using adaptive quadrature. the genral model is y = Xb + ZU + e but that values of beta and U are not included in the call. Instead this information is contained in yhat which incorporates Xb at the top level and all relevant Zu information at lower levels.

Usage

```
calc_lin_lnl_quad_fast(y, yhat, level, Z, Qi, omega, W, C, qp, omegaFull,
  QiFull, ZFull, top = TRUE, atPoint = FALSE,
  integralMultiplierExponent = 0L, integralZColumn = 1L,
  verbose = TRUE, acc = 120L)
```

Arguments

y	a numeric vector, the response.
yhat	the current predicted values of the response
level	an integer that respresents the number of levels in the likelihood that is desired. In a two level model this function will be called with l=2 and it will recurse and call itself with l=1
Z	a list of the Z matricies where the index of Z indicates the level. of the Z matri. Z[[1]] is NULL because there is no individual lavel Zs.
Qi	the scaling factor for the adaptive quadratures (per group)
omega	a list of the b estimates for ech group
W	list of weight matricies. must have 'w' and 'index' columns
C	a list of Cholesky decompositions of the Sigma matricies. C[[1]] is simply the residual variance (a scalar) while C[[l]] for l > 1 is a matrix with the name number of rows and columns as the Z matrix for that level.
qp	Gaussian quadrature result from statmod::gauss.quad.
omegaFull	numeric, the b estimates for each group duplicated so each element has one row per observation (like Zfull)
QiFull	the scaling factors for adaptive quarature points duplicated so each element has one row per observation (like Zfull)
ZFull	Z expended such that each Z[[i]] matrix contains one row for each observation (ie each element has same number of rows as original data)
top	boolean set to TRUE to return a single scalar, otherwise returns a vector
atPoint	boolean, indicates likelihood should be calculated at single point at the top level and then integrated below that. This is useful for finding the maximum posterior (or likelihood) extimate for the random effects
integralMultiplierExponent	a single integer, the function evaluates the integral times the randome effect to this power when set to 0, it is just the log-likelihood when set to 1, this can be used to estimate the expected value.
integralZColumn	is the column index of Z to use integralMultiplierExponent on only one random effect at a time can be integrated over, and the integration happens at the top level.
verbose	boolean set to TRUE to get verbose output
acc	numeric, accuracy of the mpfr

Author(s)

Huade Huo, Paul Bailey, Claire Kelley

mix *Survey Weighted Mixed-Effects Models*

Description

Implements a survey weighted mixed-effects model using the provided formula.

Usage

```
mix(formula, data, weights, nQuad = 13L, run = TRUE, verbose = TRUE,
  acc0 = 120, keepAdapting = FALSE, start = NULL, fast = FALSE,
  family = NULL, center_group = NULL, center_grand = NULL)
```

Arguments

formula	a formula object in the style of lme4 that creates the model.
data	a data frame containing the raw data for the model.
weights	a character vector of weight variables found in data frame.
nQuad	an integer number of quadrature point to evaluate on. See notes for guidelines.
run	logical; TRUE runs the model why FALSE provides partial output for debugging or testing.
verbose	logical, default FALSE; set to TRUE to print results of intermediate steps
acc0	integer, the precision of mpfr, default 120
keepAdapting	logical, set to TRUE when the adaptive quadrature should adapt after every Newton step. Defaults to TRUE. FALSE should be used rarely for faster (but less accurate) results.
start	numeric vector representing the point at which the model should start optimization; takes the shape of c(coef, vars) from results (see help).
fast	logical; use c++ function for faster result. Defaults to TRUE.
family	the family; optionally used to specify generalized linear mixed models. Currently only binomial(link="logit") is supported.
center_group	a list where the name of each element is the name of the aggregation level, and the element is a formula of variable names to be group mean centered, for example to group mean center gender and age with in the group student: list("student"= ~gender+age), default value of NULL does not perform any group mean centering.
center_grand	a formula of variable names to be grand mean centered, for example to center the variable education by overall mean of education: ~education. Default is NULL which does no centering

Details

Uses adaptive quadrature following the method in Stata's GLAMMM. For additional details, see the vignette *Weighted Mixed Models* which provides extensive examples as well as a description of the mathematical basis of the estimation procedure. The main specification also shows comparisons to model specifications in other common software.

Notes:

- Standard errors of random effect variances are estimated by the Sandwich Method; see main vignette for details.
- To see the function that is maximized in the estimation of this model; see the section on "Model fitting" in the main vignette.
- When all weights above the individual level are 1, this is similar to a `lmer` and you should use `lme4` because it is much faster.
- Starting coefficients are not provided they are estimated using `lme4`.
- When the variance of a random effect is very low (<.1), we don't estimate it because very low variances create problems with numerical evaluation. In these cases, consider estimating without that RE.
- The model is estimated by maximum likelihood estimation, restricted maximum likelihood (REML) is not available.
- To choose number of quadrature points, a balance is needed between accuracy and speed-estimation time increases quadratically with the number of points chosen. In addition, an odd number of points is traditionally used. We recommend starting at 13 and increasing or decreasing as needed.

Value

object of class `WeMixResults`. This is a list with objects:

- lnlf - function, the likelihood function
- lnl - numeric, the logliklihood of the model
- coef - numeric vector, the estimated coefficients of the model
- vars- numeric vector, the variances
- call - the original call used
- levels - integer, the number of levels in the model
- ICC - numeric, the Intraclass Correlation Coefficient
- CMEAN - function the conditional mode function that can be called with par and omega to get the conditional mode of the likelihood function
- CMODE - function the conditional mean function that can be called with par and omega to get the conditional mean of the likelihood function
- Hessian - the second derivative of the likelihood function

Author(s)

Paul Bailey, Claire Kelley, and Trang Nguyen

Examples

```
## Not run:
library(WeMix)
library(lme4)

data(sleepstudy)
ss1 <- sleepstudy
doubles <- c(308, 309, 310) # subject with double obs
# Create weights
ss1$W1 <- ifelse(ss1$Subject %in% doubles, 2, 1)
ss1$W2 <- 1

# Run random-intercept 2-level model
mix1 <- mix(Reaction~ Days + (1|Subject),data=ss1, weights = c("W1","W2"),
            fast=TRUE, nQuad=13, verbose=FALSE)

# Run random-intercept 2-level model with group-mean centering
grp_centered <- mix(Reaction ~ Days + (1|Subject), data=ss1, weights = c("W1","W2"), nQuad=13,
          fast=TRUE, center_group = list("Subject" = ~Days),
          verbose=FALSE)

## End(Not run)
```

Index