Package 'MATA'

R topics documented:

MATA	*Model-Averaged Tail Area Wald (MATA-Wald) Confidence Interval, Density, and Distribution*

Description

Functions for computing the Model-Averaged Tail Area Wald (MATA-Wald) confidence interval, density, and distribution. These are all constructed using single-model frequentist estimators and model weights. See details.

Usage

```
mata.wald(
  theta.hats,
  se.theta.hats,
  model.weights,
  mata.t,
  residual.dfs,
  alpha = 0.025
)

dmata.wald(
  theta,
  theta.hats,
  se.theta.hats,
  model.weights,
  mata.t,
  residual.dfs
)

pmata.wald(
  theta,
  theta.hats,
  se.theta.hats,
  model.weights,
  mata.t,
  residual.dfs
)
```

Arguments

theta.hats	A numeric vector containing the parameter estimates under each candidate model.
se.theta.hats	A numeric vector containing the estimated standard error of each value in theta.hats. If an element is zero, this corresponds to the parameter being fixed to the value give in theta.hats under a particular candidate model.
model.weights	A vector containing the weights for each candidate model (e.g. AIC weights or stacking weights). All model weights must be non-negative, and sum to one.
mata.t	Logical. TRUE for the normal linear model case, and FALSE otherwise. When TRUE, the argument residual.dfs must also be supplied.
residual.dfs	A vector containing the residual (error) degrees of freedom under each candidate model. This argument must be provided when mata.t = TRUE.
alpha	For mata.wald only, the desired lower and upper error rate. The value 0.025 corresponds to a 95% MATA-Wald confidence interval, and 0.05 to a 90% interval. Must be between 0 and 0.5. Default value is 0.025.
theta	For dmata.wald and pmata.wald only, a vector of theta values at which to evaluate the model-averaged confidence density or distribution function.

Details

`mata.wald` may be used to construct model-averaged confidence intervals, using the Model-Averaged Tail Area (MATA) construction (see Turek and Fletcher (2012) for details). The idea underlying this construction is similar to that of a model-averaged Bayesian credible interval. This function returns the lower and upper confidence limits of a MATA-Wald interval.

Closely related, the `dmata.wald` and `pmata.wald` functions evaluate the MATA-Wald confidence density and confidence distribution functions, which were developed by Fletcher et al. (2019).

Two usages are supported. For the normal linear model, or any other model where a t-based interval is appropriate (e.g., quasi-poisson), using option `mata.t = TRUE` corresponds to a MATA-Wald confidence interval, density, or distribution corresponding to the solutions of equations (2) and (3) of Turek and Fletcher (2012). The argument `residual.dfs` is required for this usage.

When the sampling distribution for the estimator is asymptotically normal (e.g. MLEs), possibly after a transformation, use option `mata.t = FALSE`. This corresponds to solutions to the equations in Section 3.2 of Turek and Fletcher (2012).

If the parameter is fixed to a certain value under one or more candidate models, this value should be provided in the `theta.hats` argument, along with a corresponding value of zero in `se.theta.hats`. For the model-averaged confidence density, this results in a point mass equal to the sum of the weights for these models.

Author(s)

Daniel Turek

References

Turek, D. and Fletcher, D. (2012). Model-Averaged Wald Confidence Intervals. Computational Statistics and Data Analysis, 56(9): 2809-2815.

Fletcher, D. (2018). Model Averaging. Berlin, Heidelberg: Springer Briefs in Statistics.

Fletcher, D., Dillingham, P. W., and Zeng, J. (2019). Model-averaged confidence distributions. Environmental and Ecological Statistics, 26: 367–384.

See Also

mata.wald, dmata.wald, pmata.wald

Examples

```
# Normal linear prediction:
# Generate single-model Wald and model-averaged MATA-Wald 95% confidence intervals
#
# Data 'y', covariates 'x1' and 'x2', all vectors of length 'n'.
# 'y' taken to have a normal distribution.
# 'x1' specifies treatment/group (factor).
# 'x2' a continuous covariate.
#
# Take the quantity of interest (theta) as the predicted response
# (expectation of y) when x1=1 (second group/treatment), and x2=15.
```

```
set.seed(0)
n = 20                                  # 'n' is assumed to be even
x1 = c(rep(0,n/2), rep(1,n/2))          # two groups: x1=0, and x1=1
x2 = rnorm(n, mean=10, sd=3)
y = rnorm(n, mean = 3*x1 + 0.1*x2)  # data generation

x1 = factor(x1)
m1 = glm(y ~ x1)                        # using 'glm' provides AIC values.
m2 = glm(y ~ x1 + x2)                   # using 'lm' doesn't.
aic = c(m1$aic, m2$aic)
delta.aic = aic - min(aic)
model.weights = exp(-0.5*delta.aic) / sum(exp(-0.5*delta.aic))
residual.dfs = c(m1$df.residual, m2$df.residual)

p1 = predict(m1, se=TRUE, newdata=list(x1=factor(1), x2=15))
p2 = predict(m2, se=TRUE, newdata=list(x1=factor(1), x2=15))
theta.hats = c(p1$fit, p2$fit)
se.theta.hats = c(p1$se.fit, p2$se.fit)

#  AIC model weights
model.weights

#  95% Wald confidence interval for theta (under Model 1)
theta.hats[1] + c(-1,1)*qt(0.975, residual.dfs[1])*se.theta.hats[1]

#  95% Wald confidence interval for theta (under Model 2)
theta.hats[2] + c(-1,1)*qt(0.975, residual.dfs[2])*se.theta.hats[2]

#  95% MATA-Wald confidence interval for theta:
mata.wald(theta.hats, se.theta.hats, model.weights, mata.t = TRUE, residual.dfs = residual.dfs)

#  Plot the model-averaged confidence density and distribution functions
#  on the interval [2, 7]
thetas <- seq(2, 7, by = 0.1)

dens <- dmata.wald(thetas, theta.hats, se.theta.hats, model.weights,
                  mata.t = TRUE, residual.dfs = residual.dfs)

dists <- pmata.wald(thetas, theta.hats, se.theta.hats, model.weights,
                   mata.t = TRUE, residual.dfs = residual.dfs)

par(mfrow = c(2,1))
plot(thetas, dens, type = 'l', main = 'Model-Averaged Confidence Density')
plot(thetas, dists, type = 'l', main = 'Model-Averaged Confidence Distribution')
```

Index