Package: MisRepARMA (via r-universe)

June 7, 2026

Type Package

Title Misreported Time Series Analysis

Version 0.2.0

Date 2026-06-07

Encoding UTF-8

Maintainer David Moriña Soler <David.Morina@uab.cat>

Description Provides a simple and trustworthy methodology for the analysis of misreported continuous time series using either a frequentist (bootstrap-based EM algorithm) or a Bayesian (MCMC via JAGS) approach. The frequentist method is described in Morina et al. (2021) <doi:10.1038/s41598-021-02620-5>. The Bayesian extension fits the same ARMA model with misreporting structure using a full posterior distribution, providing credible intervals and DIC for model comparison, as described in Morina et al. (2024) <doi:10.1101/2024.02.26.24303373>.

Depends R (>= 3.5.0), mixtools, boot, tseries

Imports R2jags

Suggests coda

License GPL (>= 2)

NeedsCompilation no

Author David Moriña Soler [aut, cre] (ORCID: <https://orcid.org/0000-0001-5949-7443>), Amanda Fernández-Fontelo [aut], Alejandra Cabaña [aut], Pedro Puig [aut], Biel Abarca Galván [aut]

Repository https://cran.r-universe.dev

Date/Publication 2026-06-07 14:10:28 UTC

RemoteUrl https://github.com/cran/MisRepARMA

RemoteRef HEAD

RemoteSha b0751118d71dcf0fb99018a05204f145f62d9538

Contents

`MisRepARMA-package` *Misreported time series analysis*

Description

Provides a simple and trustworthy methodology for the analysis of misreported continuous time series. See Moriña, D, Fernández-Fontelo, A, Cabaña, A, Puig P. (2021) <https://arxiv.org/abs/2003.09202v2>.

Details

Package:	MisRepARMA
Type:	Package
Version:	0.2.0
Date:	2026-04-22
License:	GPL version 2 or newer
LazyLoad:	yes

The package implements function `fitMisRepARMA`, which is able to fit an ARMA time series model to misreported data, and the function `reconstruct` which is able to reconstruct the most likely real series.

Author(s)

David Moriña, Amanda Fernández-Fontelo, Alejandra Cabaña, Pedro Puig, Biel Abarca Galván

Mantainer: David Moriña Soler <dmorina@ub.edu>

References

Davison, A.C. and Hinkley, D.V. (1997) Bootstrap Methods and Their Application. Cambridge University Press.

Kunsch, H.R. (1989) The jackknife and the bootstrap for general stationary observations. Annals of Statistics, **17**, 1217–1241.

Moriña, D., Fernández-Fontelo, A., Cabaña, A., Puig, P. (2021): New statistical model for misreported data with application to current public health challenges. arXiv preprint (https://arxiv.org/pdf/2003.09202.pdf)

Politis, D.N. and Romano, J.P. (1994) The stationary bootstrap. Journal of the American Statistical Association, **89**, 1303–1313.

See Also

MisRepARMA-package, fitMisRepARMA, reconstruct

---

fitMisRepARMA *Fit ARMA model to misreported time series data*

Description

Fits an ARMA model to misreported (underreported or overreported) continuous time series data using either a frequentist bootstrap-based EM algorithm (method = "freq") or a Bayesian MCMC approach via JAGS (method = "bayes").

Usage

```
fitMisRepARMA(y, tol, B, p_AR, q_MA, covars = NULL, misReport = "U",
              method = "freq", n_chains = 3, n_burnin = NULL,
              n_thin = 1, w_prior_a = 1, w_prior_b = 1, ...)
```

Arguments

y	numeric vector or time series of observed (potentially misreported) data.
tol	tolerance to stop the iterative EM algorithm. A value of 1e-4 is recommended.
B	for method = "freq": number of bootstrap resamples. For method = "bayes": total MCMC iterations per chain (minimum 500 recommended; 2000 or more for reliable convergence).
p_AR	non-negative integer: order of the AR part of the latent ARMA process.
q_MA	non-negative integer: order of the MA part of the latent ARMA process.
covars	optional numeric matrix of covariates to account for trends or seasonal patterns. Default is NULL.
misReport	character string: "U" (default) for underreported data ($0 < q < 1$) or "O" for overreported data ($q > 1$).
method	character string: "freq" (default) for the frequentist bootstrap-based EM approach, or "bayes" for the Bayesian MCMC approach via JAGS.
n_chains	(Bayesian only) number of parallel MCMC chains. Default is 3.
n_burnin	(Bayesian only) burn-in iterations to discard. Default is NULL (20% of B).
n_thin	(Bayesian only) thinning interval. Default is 1.
w_prior_a	(Bayesian only) first shape parameter of the Beta prior for ω: $\omega \sim \text{Beta}(a, b)$. Default is 1 (uniform prior).
w_prior_b	(Bayesian only) second shape parameter of the Beta prior for ω. Default is 1 (uniform prior).
...	additional arguments passed to tsboot.

Details

The model assumes a latent ARMA(p, r) process X_t that is only partially observed through:

$$Y_t = \begin{cases} X_t & \text{with probability } 1-\omega \\ q \cdot X_t & \text{with probability } \omega \end{cases}$$

where q is the misreporting intensity and ω is its frequency.

The **frequentist** method estimates parameters via an iterative EM algorithm with bootstrap uncertainty quantification.

The **Bayesian** method samples from the full posterior via JAGS, initialising chains near the frequentist estimates. Convergence should be checked with the Gelman-Rubin statistic ($\hat{R} < 1.1$) accessible via `attr(fit, "jags")$BUGSoutput$summary`.

Value

An object of class `fitMisRepARMA` with elements:

`data`	the original observed series.
`t0`	point estimates (EM or posterior medians). Last element is AIC (frequentist) or DIC (Bayesian).
`t`	matrix of bootstrap replicates or MCMC posterior samples.

Attributes: `q` (misreporting intensity), `w` (frequency), `z` (misreporting indicator), `covars` (covariate fit), `x_rec` (Bayesian latent process), `DIC` (Bayesian), `jags` (full JAGS object, Bayesian only).

Author(s)

David Morina, Amanda Fernandez-Fontelo, Alejandra Cabana, Pedro Puig, Biel Abarca Galvan

References

Morina, D., Fernandez-Fontelo, A., Cabana, A., Puig, P. (2021): New statistical model for misreported data with application to current public health challenges. *Scientific Reports*, **11**, 23321. doi:10.1038/s41598021026205

See Also

`MisRepARMA-package`, `reconstruct`

Examples

```
set.seed(12345)
x   <- arima.sim(model = list(ar = 0.4), n = 100)
ind <- rbinom(100, 1, 0.6)
y   <- ifelse(ind == 0, x, x * 0.3)

mod_freq <- fitMisRepARMA(y, tol = 1e-4, B = 50, p_AR = 1, q_MA = 0,
                          misReport = "U", method = "freq")
summary(mod_freq)
```

```
reconstruct(mod_freq)

## Not run:
mod_bayes <- fitMisRepARMA(y, tol = 1e-4, B = 1000, p_AR = 1, q_MA = 0,
                           misReport = "U", method = "bayes",
                           n_chains = 3, n_burnin = 200)
summary(mod_bayes)

## End(Not run)
```

`reconstruct` *Reconstruct the most likely series*

Description

Reconstructs the most likely series.

Usage

```
reconstruct(object)
```

Arguments

`object`	object of class `fitMisRepARMA`.

Value

the function returns a vector of the same length of `data` containing the reconstruction of the most likely series.

Author(s)

David Moriña, Amanda Fernández-Fontelo, Alejandra Cabaña, Pedro Puig, Biel Abarca Galván

References

D. Moriña, A. Fernández-Fontelo, A. Cabaña, P. Puig (2021): New statistical model for misreported data with application to current public health challenges. arXiv preprint (https://arxiv.org/pdf/2003.09202.pdf)

Davison, A. C. and Hinkley, D. V. (1997) Bootstrap Methods and Their Applications. Cambridge University Press, Cambridge. ISBN 0-521-57391-2

See Also

`MisRepARMA-package`, `fitMisRepARMA`

Examples

```
### Example 1: frequentist reconstruction
set.seed(12345)
x   <- arima.sim(model = list(ar = 0.4), n = 100)
ind <- rbinom(100, 1, 0.6)
y   <- ifelse(ind == 0, x, x * 0.3)

fit  <- fitMisRepARMA(y, tol = 1e-6, B = 50, p_AR = 1, q_MA = 0,
                      covars = NULL, misReport = "U", method = "freq")
x_hat <- reconstruct(fit)
plot(y,     type = "l", col = "black", ylab = "Series",
     main = "Observed (black) vs reconstructed (red)")
lines(x_hat, col = "red", lty = 2)

## Not run:
### Example 2: Bayesian reconstruction (requires R2jags and JAGS)
fit_b  <- fitMisRepARMA(y, tol = 1e-6, B = 5000, p_AR = 1, q_MA = 0,
                        covars = NULL, misReport = "U", method = "bayes",
                        n_chains = 3, n_burnin = 1000)
x_hat_b <- reconstruct(fit_b)
lines(x_hat_b, col = "blue", lty = 3)
legend("topright", legend = c("Observed", "Freq.", "Bayes"),
       col = c("black","red","blue"), lty = c(1,2,3))

## End(Not run)
```

Index