Package 'spINAR'

April 8, 2024

Type Package

Title (Semi)Parametric Estimation and Bootstrapping of INAR Models

Version 0.2.0

Date 2024-04-08

Maintainer Maxime Faymonville <faymonville@statistik.tu-dortmund.de>

Description Semiparametric and parametric estimation of INAR models including a finite sample refinement (Faymonville et al. (2022) <doi:10.1007/s10260-022-00655-0>) for the semiparametric setting introduced in Drost et al. (2009) <doi:10.1111/j.1467-9868.2008.00687.x>, different procedures to bootstrap INAR data (Jentsch, C. and Weiß, C.H. (2017) <doi:10.3150/18-BEJ1057>) and flexible simulation of INAR data.

License GPL (>= 3)

Encoding UTF-8

Depends R (>= 3.6.0)

Imports checkmate (>= 1.8.5), progress, stats

RoxygenNote 7.2.3

URL https://github.com/MFaymon/spINAR

BugReports https://github.com/MFaymon/spINAR/issues

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Author Maxime Faymonville [aut, cre] (<https://orcid.org/0000-0001-5565-0711>),
Javiera Riffo [aut] (<https://orcid.org/0000-0002-5372-7196>),
Jonas Rieger [aut] (<https://orcid.org/0000-0002-0007-4478>),
Carsten Jentsch [aut] (<https://orcid.org/0000-0001-7824-1697>),
Christian H. Weiß [ctb] (<https://orcid.org/0000-0001-8739-6631>)

Repository CRAN

Date/Publication 2024-04-08 14:00:02 UTC

R topics documented:

`spINAR-package` *(Semi)parametric estimation and bootstrapping of INAR models*

Description

Semiparametric and parametric estimation of INAR models including a finite sample refinement for the semiparametric setting, different procedures to bootstrap INAR data and flexible simulation of INAR data.

Semiparametric INAR Model

The package provides a flexible simulation of INAR data by inserting a user-defined pmf argument in the `spinar_sim` function. Using `spinar_est`, it allows for semiparametric estimation of the INAR model along Drost et al. (2009) and additionally, it includes a small sample refinement `spinar_penal` (Faymonville et al., 2022) together with a validation of the upcoming penalization parameters (`spinar_penal_val`). Furthermore, it contains a semiparametric INAR bootstrap procedure implemented in `spinar_boot` (Jentsch and Weiß, 2017).

Parametric INAR Model

In addition to the semiparametric model, the package also allows for parametric simulation (`spinar_sim`), parametric estimation (`spinar_est_param`) and parametric bootstrapping (`spinar_boot`) of INAR data.

Author(s)

Maintainer: Maxime Faymonville <faymonville@statistik.tu-dortmund.de> (ORCID)

Authors:

- Javiera Riffo <javiera.riffo@tu-dortmund.de> (ORCID)
- Jonas Rieger <rieger@statistik.tu-dortmund.de> (ORCID)
- Carsten Jentsch <jentsch@statistik.tu-dortmund.de> (ORCID)

Other contributors:

- Christian H. Weiß <weissc@hsu-hh.de> (ORCID) [contributor]

References

Faymonville, M., Jentsch, C., Weiß, C.H. and Aleksandrov, B. (2022). "Semiparametric Estimation of INAR Models using Roughness Penalization". Statistical Methods & Applications. doi:10.1007/s10260022006550.

Jentsch, C. and Weiß, C. H. (2017), "Bootstrapping INAR Models". Bernoulli 25(3), pp. 2359–2408. doi:10.3150/18BEJ1057.

Drost, F., Van den Akker, R. and Werker, B. (2009), "Efficient estimation of auto-regression parameters and innovation distributions for semiparametric integer-valued AR(p) models". Journal of the Royal Statistical Society. Series B 71(2), pp. 467–485. doi:10.1111/j.14679868.2008.00687.x.

See Also

Useful links:

- https://github.com/MFaymon/spINAR
- Report bugs at https://github.com/MFaymon/spINAR/issues

`spinar_boot` *(Semi)parametric INAR bootstrap procedure*

Description

INAR bootstrap procedures for the semiparametric and the parametric INAR setting, where the latter allows for moment- and maximum likelihood-based estimation and Poisson, geometrically and negative binomially distributed innovations.

Usage

```
spinar_boot(
  x,
  p,
  B,
  setting,
  type = "mom",
  distr = "poi",
  M = 100,
  level = 0.05,
  progress = TRUE
)
```

Arguments

`x`	`[integer]` vector with integer observations.
`p`	`[integer(1)]` order of the INAR model, where `p` $\in \{1, 2\}$.

B [integer(1)]
number of bootstrap repetitions.

setting [string(1)]
estimation setting ∈ {"sp", "p"}, where "sp" defines a semiparametric setting and "p" a parametric setting.

type [string(1)]
type of estimation ∈ {"mom", "ml"}, where "mom" (default) performs moment-based estimation and "ml" maximum likelihood-based estimation.

distr [string(1)]
parametric family of innovation distribution ∈ {"poi", "geo", "nb"}, where "poi" (default) denotes Poi(lambda), "geo" Geo(prob) and "nb" NB(r, prob) distributions.

M [integer(1)]
upper limit for the innovations.

level [numeric(1)]
level for the bootstrap confidence intervals (percentile interval and Hall's percentile interval (bootstrap-t-interval without studentization)).

progress [logical(1)]
Should a nice progress bar be shown? Turning it off, could lead to significantly faster calculation. Default is TRUE.

Value

[named list] with entries

x_star [matrix] of bootstrap observations with length(x) rows and B columns.

parameters_star [matrix] of bootstrap estimated parameters with B rows. If setting = "sp", each row contains the estimated coefficients $\texttt{alpha}_1, ..., \texttt{alpha}_p$ and the estimated entries of the pmf $\texttt{pmf}_0, \texttt{pmf}_1$, ... where $\texttt{pmf}_i$ represents the probability of an innovation being equal to i. If setting = "p", each row contains the estimated coefficients $\texttt{alpha}_1, ..., \texttt{alpha}_p$ and the estimated parameter(s) of the innovation distribution.

bs_ci_percentile [named matrix] with the lower and upper bounds of the bootstrap percentile confidence intervals for each parameter in parameters_star.

bs_ci_hall [named matrix] with the lower and upper bounds of Hall's bootstrap percentile confidence intervals for each parameter in parameters_star.

Examples

```
# generate data
dat1 <- spinar_sim(n = 200, p = 1, alpha = 0.5,
                   pmf = c(0.3, 0.3, 0.2, 0.1, 0.1))
dat2 <- spinar_sim(n = 200, p = 2, alpha = c(0.2, 0.3),
                   pmf = dgeom(0:60, 0.5))


# semiparametric INAR(1) bootstrap
spinar_boot(x = dat1, p = 1, B = 50, setting = "sp")
```

```
# parametric Geo-INAR(2) bootstrap using moment-based estimation
spinar_boot(x = dat2, p = 2, B = 50, setting = "p", type = "mom", distr = "geo")
```

---

spinar_est *Semiparametric estimation of INAR models*

Description

Semiparametric estimation of the autoregressive parameters and the innovation distribution of INAR(p) models, $p \in \{1, 2\}$. The estimation is conducted by maximizing the conditional likelihood of the model.

Usage

```
spinar_est(x, p)
```

Arguments

x	[integer] vector with integer observations.
p	[integer(1)] order of the INAR model, where $p \in \{1, 2\}$.

Value

Vector containing the estimated coefficients $alpha_1, ..., alpha_p$ and the estimated entries of the pmf pmf_0, pmf_1,... where pmf_i represents the probability of an innovation being equal to i.

Examples

```
# generate data
dat1 <- spinar_sim(n = 200, p = 1, alpha = 0.5,
                   pmf = c(0.3, 0.3, 0.2, 0.1, 0.1))
dat2 <- spinar_sim(n = 200, p = 2, alpha = c(0.2, 0.3),
                   pmf = c(0.25, 0.2, 0.15, 0.1, 0.1, 0.1, 0.1))


# semiparametric estimation of INAR(1) model
spinar_est(x = dat1, p = 1)
# semiparametric estimation of INAR(2) model
spinar_est(x = dat2, p = 2)
```

`spinar_est_param` *Parametric estimation of INAR models*

Description

Parametric estimation of the autoregressive parameters and the innovation distribution of INAR(p) models, $\mathsf{p} \in \{1, 2\}$, with Poisson, geometrically or negative binomially distributed innovations. The estimation can either be moment- or maximum likelihood-based.

Usage

```
spinar_est_param(x, p, type, distr)
```

Arguments

`x`	[`integer`] vector with integer observations.
`p`	[`integer(1)`] order of the INAR model, where $\mathsf{p} \in \{1, 2\}$.
`type`	[`string(1)`] type of estimation $\in$ {`"mom"`, `"ml"`}, where `"mom"` performs moment-based estimation and `"ml"` maximum likelihood-based estimation.
`distr`	[`string(1)`] parametric family of innovation distribution $\in$ {`'poi'`, `'geo'`, `'nb'`}, where `"poi"` denotes Poi(`lambda`), `"geo"` Geo(`prob`) and `"nb"` NB(`r`, `prob`) distributions.

Value

Named vector containing the estimated coefficients $\mathtt{alpha}_1, ..., \mathtt{alpha}_p$ and the estimated parameter(s) of the innovation distribution.

Examples

```
# generate data
# Poi-INAR(1) data
dat1 <- spinar_sim(n = 200, p = 1, alpha = 0.5, pmf = dpois(0:20, 1))
# Geo-INAR(2) data
dat2 <- spinar_sim(n = 200, p = 2, alpha = c(0.2, 0.3),
                   pmf = dgeom(0:60, 0.5))
# NB-INAR(1) data
dat3 <- spinar_sim(n = 200, p = 1, alpha = 0.5, pmf = dnbinom(0:40, 2, 2/3))

# moment-based parametric estimation of Poi-INAR(1) model
spinar_est_param(x = dat1, p = 1, type = "mom", distr = "poi")
# moment-based parametric estimation of Geo-INAR(2) model
spinar_est_param(x = dat2, p = 2, type = "mom", distr = "geo")
```

```
# maximum likelihood-based parametric estimation of NB-INAR(1) model
spinar_est_param(x = dat3, p = 1, type = "ml", distr = "nb")
```

`spinar_penal` *Penalized semiparametric estimation of INAR models*

Description

Semiparametric penalized estimation of the autoregressive parameters and the innovation distribution of INAR(p) models, $\mathtt{p} \in \{1, 2\}$. The estimation is conducted by maximizing the penalized conditional likelihood of the model. If both penalization parameters are set to zero, the function coincides to the spinar_est function of this package.

Usage

```
spinar_penal(x, p, penal1 = 0, penal2 = 0)
```

Arguments

`x`	[integer] vector with integer observations.
`p`	[integer(1)] order of the INAR model, where $\mathtt{p} \in \{1, 2\}$.
`penal1`	L_1 penalization parameter (default value zero results in no L_1 penalization)
`penal2`	L_2 penalization parameter (default value zero results in no L_2 penalization)

Value

Vector containing the penalized estimated coefficients $\mathtt{alpha}_1, ..., \mathtt{alpha}_p$ and the penalized estimated entries of the pmf $\mathtt{pmf}_0, \mathtt{pmf}_1$,... where $\mathtt{pmf}_i$ represents the probability of an innovation being equal to i.

Examples

```
# generate data
dat1 <- spinar_sim(n = 50, p = 1, alpha = 0.5,
                   pmf = c(0.3, 0.25, 0.2, 0.15, 0.1))

# penalized semiparametric estimation
spinar_penal(x = dat1, p = 1, penal1 = 0, penal2 = 0.1)
```

spinar_penal_val *Validated penalized semiparametric estimation of INAR models*

Description

Semiparametric penalized estimation of the autoregressive parameters and the innovation distribution of INAR(p) models, $p \in \{1, 2\}$. The estimation is conducted by maximizing the penalized conditional likelihood of the model. Included is a possible validation of one or both penalization parameters. If no validation is wanted, the function coincides to the spinar_penal function of this package.

Usage

```
spinar_penal_val(
  x,
  p,
  validation,
  penal1 = NA,
  penal2 = NA,
  over = NA,
  folds = 10,
  init1 = 1,
  init2 = 1,
  progress = TRUE
)
```

Arguments

x	[integer] vector with integer observations.
p	[integer(1)] order of the INAR model, where $p \in \{1, 2\}$.
validation	[logical(1)] indicates whether validation is wanted.
penal1	[numeric(1)] L_1 penalization parameter. It will be ignored if validation = TRUE and over $\in \{"both", "L_1"\}$. It is mandatory if validation = FALSE.
penal2	[numeric(1)] L_2 penalization parameter. It will be ignored if validation = TRUE and over $\in \{"both", "L_2"\}$. It is mandatory if validation = FALSE.
over	[string(1)] validation over "both" penalization parameters or only over "L_1" or "L_2". It is mandatory if validation = TRUE, otherwise it will be ignored.
folds	[integer(1)] number of folds for (cross) validation.

init1	[numeric(1)] initial value for penal1 in validation. Default value is init1 = 1.
init2	[numeric(1)] initial value for penal2 in validation. Default value is init2 = 1
progress	[logical(1)] Should a nice progress bar be shown? Turning it off, could lead to significantly faster calculation. Default is TRUE.

Value

If validation = FALSE, the function returns a vector containing the penalized estimated coefficients $\texttt{alpha}_1, ..., \texttt{alpha}_p$ and the penalized estimated entries of the pmf $\texttt{pmf}_0, \texttt{pmf}_1$... where $\texttt{pmf}_i$ represents the probability of an innovation being equal to i.

If validation = TRUE, the function returns a named list, where the first entry contains the penalized estimated coefficients $\texttt{alpha}_1, ..., \texttt{alpha}_p$ and the penalized estimated entries of the pmf $\texttt{pmf}_0, \texttt{pmf}_1$,... where $\texttt{pmf}_i$ represents the probability of an innovation being equal to i. The second (and if over = both also the third entry) contain(s) the validated penalization parameter(s).

Examples

```
# generate data
dat1 <- spinar_sim(n = 50, p = 1, alpha = 0.5,
                   pmf = c(0.3, 0.3, 0.2, 0.1, 0.1))


# penalized semiparametric estimation with validation over L1
spinar_penal_val(x = dat1, p = 1, validation = TRUE, penal2 = 0.1,
                 over = "L1")
# penalized semiparametric estimation with validation over both L1 and L2
spinar_penal_val(x = dat1, p = 1, validation = TRUE, over = "both")
```

---

spinar_sim *Simulation of (semi)parametric integer autoregressive (INAR) models*

Description

Generating INAR(p) observations, where $\texttt{p} \in \{1, 2\}$. It allows for general pmfs which can be generated parametrically or "manually" (semiparametrically).

Usage

```
spinar_sim(n, p, alpha, pmf, prerun = 500)
```

Arguments

n	[integer(1)] number of observations.
p	[integer(1)] lag of the INAR(p) model, where p $\in \{1, 2\}$.
alpha	[integer(p)] vector of INAR coefficients $\texttt{alpha}_1, ..., \texttt{alpha}_p$.
pmf	[numeric] vector of probability mass function $\texttt{pmf}_0, ..., \texttt{pmf}_k$ where $\texttt{pmf}_i$ represents the probability of an innovation being equal to i.
prerun	[integer(1)] number of observations which are generated additionally and then omitted (to ensure stationarity).

Value

Vector with n INAR(p) observations.

Examples

```
# generate (semiparametrically) 100 INAR(1) observations with
# alpha_1 = 0.5 and a manually set pmf
spinar_sim(n = 100, p = 1, alpha = 0.5, pmf = c(0.3, 0.3, 0.2, 0.1, 0.1))

# generate 100 obervations of an INAR(2) model with
# alpha_1 = 0.2, alpha_2 = 0.3 and Poi(1)-innovations
spinar_sim(n = 100, p = 2, alpha = c(0.2, 0.3), pmf = dpois(0:20,1))
```

Index