Description Functions to facilitate the estimation of Arfima-MLM models for repeated cross-sectional data and pooled cross-sectional time-series data (see Lebo and Weber 2015). The estimation procedure uses double filtering with Arfima methods to account for autocorrelation in repeated cross-sectional data followed by multilevel modeling (MLM) to estimate aggregate as well as individual-level parameters simultaneously.
ArfimaMLM-package

Arfima-MLM Estimation For Repeated Cross-Sectional Data And Pooled Cross-Sectional Time-Series Data

Description

This package provides functions to facilitate the estimation of Arfima-MLM models for repeated cross-sectional data and pooled cross-sectional time-series data (see Lebo and Weber 2015). The estimation procedure uses double filtering with Arfima methods to account for autocorrelation in longer repeated cross-sectional data followed by multilevel modeling (MLM) to estimate both aggregate- and individual-level parameters simultaneously.

Details

Package: ArfimaMLM
Type: Package
Version: 1.3
Date: 2015-01-20
License: GPL-2

The main function of the package is arfimaMLM, which implements Arfima and multilevel models on a repeated cross-sectional dataset as described by Lebo and Weber (forthcoming). Furthermore, the function arfimaOLS uses the same initial procedures but estimates a simple linear model instead of the multilevel model. The package also includes arfimaPrep, which prepares a dataset for subsequent analyses according to the Arfima-MLM framework without estimating the final model itself. fd is a wrapper function to estimate the fractional differencing parameter using hurstSpec of the fractal-package as well as procedures provided by the fracdiff-package (via ML, GPH, and Sperio) and to differentiate the series accordingly (mainly for internal use in arfimaMLM, arfimaOLS, and arfimaPrep).

Author(s)

Patrick Kraft, with contributions from Christopher Weber
Maintainer: Patrick Kraft <patrick.kraft@stonybrook.edu>

References


See Also

lme4, fracdiff, hurstSpec, arfimaMLM, arfimaOLS, arfimaPrep, fd
arfimaMLM

Description

Estimates Arfima-MLM model for repeated cross-sectional data or pooled cross-sectional time-series data. For the variables specified by the user, the function automatically implements the aggregation and fractional differencing of time/level variables as well as the necessary procedures to remove deterministic components from the dependent as well as the major independent variables.

Usage

arfimaMLM(formula, data, timevar

, d = "Hurst", arma = NULL

, ecmformula = NULL, decm = "Hurst"

, drop = 5, report.data = TRUE, ...)

Arguments

formula An object of the class “formula” that specifies the multilevel model to be estimated (see lmer for details): a two-sided linear formula object describing both the fixed-effects and fixed-effects part of the model, with the response on the left of a ~ operator and the terms, separated by + operators, on the right. Random-effects terms are distinguished by vertical bars (“|”) separating expressions for design matrices from grouping factors (i.e. the variable indicating the timepoints of the repeated cross-sectional design as well as potentially further clustering variables). See details below for further information about selecting variables for automatic aggregation, fractional differencing, and the removal of deterministic components.

data Data frame containing the original variables named in formula.

timevar Name of the variable indicating different timepoints in data.

d Call for a specific estimation method for the fractional differencing parameter in the fractal-package ("Hurst") or in the fracdiff-package ("ML", "GPH", or "Sperio"). Default estimation procedure is by estimating the Hurst exponent. If the user wants to specify the methods for each variable individually, d can be a list containing a call for every individual variable. Furthermore, the list can contain numeric values for differencing parameters which were estimated externally (e.g. 1 for simple differencing, also see example for further details). A variable will not be differenced if d is specified as 0.

arma List of variables for which AR and MA parameters are to be estimated (after fractional differencing) as well as a vector containing the respective orders of the model to fit. order[1] corresponds to the AR part and order[2] to the MA part, similar to the model specification in arima (just excluding the d parameter here). For variables specified in arma, the function will use the residuals of the ARMA model for the subsequent model estimation in order to remove their deterministic
components. All variables included in the arma list have to included in either varlist.fd, varlist.ydif, or varlist.xdif. It is also possible to keep some of the AR/MA parameters fixed at zero (e.g. if the model is only supposed to estimate AR[1] and AR[3] parameters, but not AR[2]). In order to specify such a model, replace the vector containing the orders of the model with a list containing two vectors indicating each individual AR or MA parameter to be estimated. Please see the examples for clarification.

ecmformula Specification of the cointegration regression to receive the residuals for the error correction mechanism (ecm) included in formula: linear formula object with the response on the left of a ~ operator and the independent variables, separated by + operators, on the right. Note that the variable names included here cannot be the original variable names included in data, but rather has to be extended by adding ".mean" to the original names, since the ecm is always based on the level/time aggregates (see example).

decm Call for estimation method for the fractional differencing parameter (see d for details). Can be either "Hurst", "ML", "GPH", or "Sperio". Default is "Hurst".

drop Number of time points from the beginning of the series dropped from analysis. Default is 5.

report.data Logical. arfimaMLM returns the transformed dataset used to estimate the final model as part of the results. Default is TRUE.

Further arguments passed to the estimation procedures used within the function (e.g. for lmer).

Details

• The original variable names of data used in formula can be extended by adding three different suffixes: .fd, .xdif, and .ydif. These suffixes select variables for transformations according to the Arfima-MLM framework before estimating the actual model.

The suffix .fd allows the user to select variables which are supposed to be transformed to a fractionally differenced level-variable (by aggregating individuals over each time point prior to fractionally differencing the series), or variables which are already included as a level-variable in the original dataset and are just supposed to be fractionally differenced before the multilevel model is estimated.

If the suffix .xdif is added to an independent variable, the variable is simply filtered through the timepoint averages:

\[ x_{\text{star}}[it] = x[it] - X[t] \]

If the suffix .ydif is added to the dependent variable (e.g. \( y[it] \)), the function will remove the temporal deterministic component from the individual level variable, such that it only consists of within-timepoint, as well as non-temporally autocorrelated between-timepoint variation:

\[ y_{\text{star}}[it] = y[it] - \left( Y[t] - \Delta[df]Y[t] \right) \]

• If formula contains ecm as one of the independent variables, and ecmformula is correctly specified, the function will include the lag of the fractionally differenced residuals of the cointegration regression as an error correction mechanism in the multilevel model. The ECM does not have to be estimated prior to calling the function.

• In order to prevent errors in the estimation procedure, none of the original variable names in data should include ".ydif", ".xdif" or ".fd".
Value

The function returns a list of the class `arfimaMLM` with the following items:

- **result**: Output of the multilevel model as specified in `formula`.
- **d**: Matrix of fractional differencing parameters estimated for the level variables (`.fd` and `.ydiff`) as well as the estimation method for each variable. Returns the specified value for `d` if it was specified in the initial call of the function.
- **arma**: List of `arima` results for each variable specified in the model call. Contains AR/MA estimates as well as the model residuals.
- **ecm**: Output of the cointegration regression (returned if `ecmformula` is specified). The lagged residuals of the cointegration regression are included in the multilevel model if `ecm` is included in `formula`.
- **data.mean**: Data frame of variable means declared in `formula` as `.fd`, `.xdf` or `.ydf` (as well as `.mean` in `ecmformula`) for each time point specified by the level variable in `timevar`.
- **data.fd**: Data frame of fractionally differenced level variables for each time point specified in `timevar`, which were declared as `.fd` or `.ydf` in `formula`. If `arma` was additionally specified for a variable, it contains the residuals of the ARMA model fitted after (fractionally) differencing.
- **data.merged**: Merged data frame used to estimate the multilevel model consisting of the original data, `data.mean`, `data.fd`, as well as the variables specified as `.xdf` and `.ydf` in `formula`.

Author(s)

Patrick Kraft

References


See Also

`lme4`, `fracdiff`, `hurstSpec`, `arfimaPrep`, `fd`, and `ArfimaMLM` for a package overview.

Examples

```r
require(fractal)
require(fracdiff)
require(lme4)

### set basic parameters for simulation
t = 100 # number of time points
n = 500 # number of observations within time point
N = t*n # total number of observations

### generate fractional ARIMA Time Series for y_t, x1_t, z1_t, z2_t
```
arfimaOLS

arfima-OLS for repeated cross-sectional data and/or pooled cross-sectional time-series data

set.seed(123)
y_t <- fracdiff.sim(t, d=0.4, mu=10)$series
x1_t <- fracdiff.sim(t, d=0.3, mu=5)$series
z1_t <- fracdiff.sim(t, d=0.1, mu=2)$series
z2_t <- fracdiff.sim(t, d=0.25, mu=3)$series

### simulate data
data <- NULL; data$time <- rep(seq(1:t),each=n)
data <- data.frame(data)
data$x1 <- rnorm(N,rep(x1_t,each=n),2)
data$x2 <- rnorm(N,0,40)
data$z1 <- rnorm(N,rep(z1_t,each=n),3)
data$z2 <- rep(z2_t,each=n)
b1 <- 0.2 + rep(rnorm(t,0,0.1),each=n)
data$y <- (b1$data$x1-0.05$data$x2+0.3*rep(z1_t,each=n)
          +0*data$z2+rnorm(N,rep(y_t,each=n),1))

### estimate models

# basic example
m1 <- arfimaMLM(y~x1 + x2 + z1 + z2 + H1 | time)

# model including error correction mechanism
# change estimation method for differencing parameter for all variables
m2 <- arfimaMLM(y~x1 + x2 + z1 + z2 + ecm | time)

# vary estimation method for differencing parameter between variables
# specify AR/MA models
m3 <- arfimaMLM(y~x1 + x2 + z1 + z2 + d=x1 | time)

# specify AR/MA models while holding AR[2] fixed for y
m4 <- arfimaMLM(y~x1 + x2 + z1 + z2 + d=x1 | time)

m1
summary(m2)
summary(m3$result)
m4$arma
arfimaOLS

Description

Estimates Arfima-OLS model for repeated cross-sectional data or pooled cross-sectional time-series data. For the variables specified by the user, the function automatically implements the aggregation and fractional differencing of time/level variables as well as the necessary procedures to remove deterministic components from the dependent as well as the major independent variables.

Usage

arfimaOLS(formula, data, timevar
         , d = "Hurst", arma = NULL
         , ecmformula = NULL, decm = "Hurst"
         , drop = 5, report.data = TRUE, ...)

Arguments

formula An object of the class "formula" that specifies the linear model to be estimated (see lm for details), typically with the response on the left of a ~ operator and the terms, separated by + operators, on the right. See details below for further information about selecting variables for automatic aggregation, fractional differencing, and the removal of deterministic components.

data Data frame containing the original variables named in formula.

timevar Name of the variable indicating different timepoints in data.

d Call for a specific estimation method for the fractional differencing parameter in the fractal-package ("Hurst") or in the fracdiff-package ("ML", "GPH", or "Sperio"). Default estimation procedure is by estimating the Hurst exponent. If the user wants to specify the methods for each variable individually, d can be a list containing a call for every individual variable. Furthermore, the list can contain numeric values for differencing parameters which were estimated externally (e.g. 1 for simple differencing, also see example for further details). A variable will not be differenced if d is specified as 0.

arma List of variables for which AR and MA parameters are to be estimated (after fractional differencing) as well as a vector containing the respective orders of the model to fit. order[1] corresponds to the AR part and order[2] to the MA part, similar to the model specification in arima (just excluding the d parameter here). For variables specified in arma, the function will use the residuals of the ARMA model for the subsequent model estimation in order to remove their deterministic components. All variables included in the arma list have to included in either varlist.fd, varlist.ydif, or varlist.xdif. It is also possible to keep some of the AR/MA parameters fixed at zero (e.g. if the model is only supposed to estimate AR[1] and AR[3] parameters, but not AR[2]). In order to specify such a model, replace the vector containing the orders of the model with a list containing two vectors indicating each individual AR or MA parameter to be estimated. Please see the examples for clarification.

ecmformula Specification of the cointegration regression to receive the residuals for the error correction mechanism (ecm) included in formula: linear formula object with the response on the left of a ~ operator and the independent variables, separated by + operators, on the right. Note that the variable names included here cannot...
be the original variable names included in data, but rather has to be extended by adding ".mean" to the original names, since the ecm is always based on the level/time aggregates (see example).

decm Call for estimation method for the fractional differencing parameter (see d for details). Can be either "Hurst", "ML", "GPH", or "Sperio". Default is "Hurst".

drop Number of time points from the beginning of the series dropped from analysis. Default is 5.

report.data Logical. arfimaOLS returns the transformed dataset used to estimate the final model as part of the results. Default is TRUE.

... Further arguments passed to the estimation procedures used within the function (e.g. for lm).

Details

- The original variable names of data used in formula can be extended by adding three different suffixes: .fd, .xdf, and .ydf. These suffixes select variables for transformations according to the Arfima-OLS framework before estimating the actual model. The suffix .fd allows the user to select variables which are supposed to be transformed to a fractionally differenced level-variable (by aggregating individuals over each time point prior to fractionally differenting the series), or variables which are already included as a level-variable in the original dataset and are just supposed to be fractionally differenced before the multilevel model is estimated.

If the suffix .xdf is added to an independent variable, the variable is simply filtered through the timepoint averages:

\[ x.star[it] = x[it] - X[t] \]

If the suffix .ydf is added to the dependent variable (e.g. y[it]), the function will remove the temporal deterministic component from the individual level variable, such that it only consists of within-timepoint, as well as non-temporally autocorrelated between-timepoint variation:

\[ y.star[it] = y[it] - (Y[t] - \Delta[df]Y[t]) \]

- If formula contains ecm as one of the independent variables, and ecmformula is correctly specified, the function will include the lag of the fractionally differenced residuals of the cointegration regression as an error correction mechanism in the multilevel model. The ECM does not have to be estimated prior to calling the function.

- In order to prevent errors in the estimation procedure, none of the original variable names in data should include " .ydf", " .xdf" or " .fd".

Value

The function returns a list of the class ‘arfimaOLS’ with the following items:

result Output of the linear model as specified in formula.

d Matrix of fractional differencing parameters estimated for the level variables ( .fd and .ydf) as well as the estimation method for each variable. Returns the specified value for d if it was specified in the initial call of the function.
arfimaOLS

- arma: List of arima results for each variable specified in the model call. Contains AR/MA estimates as well as the model residuals.
- ecm: Output of the cointegration regression (returned if ecmformula is specified). The lagged residuals of the cointegration regression are included in the multilevel model if ecm is included in formula.
- data.mean: Data frame of variable means declared in formula as .fd, .xdf or .ydf (as well as .mean in ecmformula) for each time point specified by the level variable in timevar.
- data.fd: Data frame of fractionally differenced level variables for each time point specified in timevar, which were declared as .fd or .ydf in formula. If arma was additionally specified for a variable, it contains the residuals of the ARMA model fitted after (fractionally) differencing.
- data.merged: Merged data frame used to estimate the OLS model consisting of the original data, data.mean, data.fd, as well as the variables specified as .xdf and .ydf in formula.

Author(s)

Patrick Kraft

References


See Also

lm, fracdiff, hurstSpec, arfimaPrep, fd, and ArfimaMLM for a package overview.

Examples

require(fractal)
require(fracdiff)

### set basic parameters for simulation
n = 100 # number of time points
N = 500 # number of observations within time point

### generate fractional ARIMA Time Series for y_t, x1_t, z1_t, z2_t
set.seed(123)
y_t <- fracdiff.sim(t, d=0.4, mu=10)$series
x1_t <- fracdiff.sim(t, d=0.3, mu=5)$series
z1_t <- fracdiff.sim(t, d=0.1, mu=2)$series
z2_t <- fracdiff.sim(t, d=0.25, mu=3)$series

### simulate data
data <- NULL; data$time <- rep(seq(1:t),each=n)
data <- data.frame(data)
data$x1 <- rnorm(N,rep(x1_t,each=n),2)
arfimaPrep

Preparing a dataset for subsequent analysis according to the Arfima-MLM/Arfima-OLS framework

Description

This function prepares a repeated cross-sectional dataset or pooled cross-sectional time-series data for subsequent analyses according to the Arfima-MLM/Arfima-OLS framework. This function is mainly used internally as part of arfimaMLM and arfimaOLS, but can also be used independently if the user prefers to separate the data preparation from the subsequent estimation of the multilevel (or simple linear) model. The function performs the aggregation and fractional differencing of time/level variables as well as the necessary procedures to remove deterministic components from the dependent as well as the major independent variables.
Usage

arfimaPrep(data, timevar
, varlist.mean, varlist.fd
, varlist.xdif, varlist.ydif
, d = "Hurst", arma = NULL
, ecmformula = NULL, decm = "Hurst"
, drop = 5, ...)

Arguments

data Data frame to be transformed by the function.
timevar Name of the variable indicating different timepoints in data.
varlist.mean Character vector of variable names in data that are averaged/aggregated over each timepoint specified by timevar. The variable list must include all variables listed in varlist.fd, varlist.xdif, varlist.ydif
varlist.fd Character vector of variable names in data that are fractionally differenced (after aggregating over each timepoint specified by timevar). The variable list must include all variables listed in varlist.ydif See details for further information.
varlist.xdif Character vector of variable names in data for which the within-timepoint deviation from the respective mean value is calculated (for each timepoint specified by timevar). See details for further information.
varlist.ydif Character vector of variable names in data for which the temporal deterministic component is removed by substracting the difference of the within-timepoint average and its stationary series free of autocorrelation (with each timepoint specified by timevar). See details for further information.
d Call for a specific estimation method for the fractional differencing parameter in the fractal-package ('"Hurst'') or in the fracdiff-package ('"ML'’, ‘"GPH'’, or ‘"Sperio'’). Default estimation procedure is by estimating the Hurst exponent. If the user wants to specify the methods for each variable individually, d can be a list containing a call for every individual variable. Furthermore, the list can contain numeric values for differencing parameters which were estimated externally (e.g. 1 for simple differencing, also see example for further details). A variable will not be differenced if d is specified as 0.
arma List of variables for which AR and MA parameters are to be estimated (after fractional differencing) as well as a vector containing the respective orders of the model to fit. order[1] corresponds to the AR part and order[2] to the MA part, similar to the model specification in arima (just excluding the d parameter here). For variables specified in arma, the function will use the residuals of the ARMA model for the subsequent model estimation in order to remove their deterministic components. All variables included in the arma list have to included in either varlist.fd, varlist.ydif, or varlist.xdif. It is also possible to keep some of the AR/MA parameters fixed at zero (e.g. if the model is only supposed to estimate AR[1] and AR[3] parameters, but not AR[2]). In order to specify such a model, replace the vector containing the orders of the model with a list containing two vectors indicating each individual AR or MA parameter to be estimated. Please see the examples for clarification.
ecmformula  Specification of the cointegration regression to receive the residuals for the error correction mechanism (ecm) to be included in the transformed dataset: linear formula object with the response on the left of a ~ operator and the independent variables, separated by + operators, on the right. Note that the variable names included here cannot be the original variable names included in data, but rather has to be extended by adding ".mean" to the original names, since the ecm is always based on the level/time aggregates (see example).

decm  Call for estimation method for the fractional differencing parameter (see d for details). Can be either "Hurst", "ML", "GPH", or "Sperio". Default is "Hurst".

drop  Number of time points from the beginning of the series dropped from analysis. Default is 5.

...  Further arguments passed to the estimation procedures used within the function.

Details

- The varlists varlist.fd, varlist.xdif, and varlist.ydif select variables from data for transformations according to the Arfima-MLM framework to prepare the estimation of the actual model.

  Adding variables in varlist.fd allows the user to select variables which are supposed to be transformed to a fractionally differenced level-variable (by aggregating individuals over each time point prior to fractionally differencing the series), or variables which are already included as a level-variable in the original dataset and are just supposed to be fractionally differenced before the multilevel model is estimated.

  For variables in varlist.xdif, the corresponding variables in data is simply filtered through the timepoint averages:

  \[ x.star[it] = x[it] - \bar{X}[t] \]

  For variables in varlist.ydif (e.g. \( y[it] \)), the function will remove the daily deterministic component from the individual level variable, such that it only consists of within-timepoint, as well as non-temporally autocorrelated between-timepoint variation:

  \[ y.star[it] = y[it] - (Y[t] - \Delta[df]Y[t]) \]

- In order to prevent errors in the estimation procedure, none of the original variable names in data should include ".fd", ".xdif", or ".ydif".

Value

The function returns a list of datasets and estimation results with the following items:

- data.mean  Data frame of variable means declared in varlist.mean, varlist.fd, varlist.xdif, or varlist.ydif for each time point specified by the level variable in timevar.
- data.fd  Data frame of fractionally differenced level variables for each time point specified in timevar, which were declared as .fd or .ydif in formula. If arma was additionally specified for a variable, it contains the residuals of the ARMA model fitted after (fractionally) differencing.
arfimaPrep

data_merged Merged data frame which can be subsequently used to estimate the multilevel model.
Consist of the original data, data.mean, data.fd, as well as the variables specified in varlist.xdif and .ydif

d Matrix of fractional differencing parameters estimated for the level variables (varlist.fd and varlist.ydif) as well as the estimation method for each variable. Returns the specified value for d if it was specified in the initial call of the function.

arma List of arima results for each variable specified in the model call. Contains AR/MA estimates as well as the model residuals.

ecm Output of the cointegration regression (returned if ecmformula is specified). The lagged residuals of the cointegration regression are included in data.fd and data.merged.

Author(s)
Patrick Kraft

References

See Also
fracdiff, hurstSpec, fd, and ArfimaMLM for a package overview.

Examples

```r
require(fractal)
require(fracdiff)

### set basic parameters for simulation
n = 500 # number of observations within time point
N = t*n # total number of observations

### generate fractional ARIMA Time Series for y_t, x1_t, z1_t, z2_t
set.seed(123)
y_t <- fracdiff.sim(t, d=0.4, mu=10)$series
x1_t <- fracdiff.sim(t, d=0.3, mu=5)$series
z1_t <- fracdiff.sim(t, d=0.1, mu=2)$series
z2_t <- fracdiff.sim(t, d=0.25, mu=3)$series

### simulate data
data <- NULL; data$time <- rep(seq(1:t),each=n)
data <- data.frame(data)
data$x1 <- rnorm(N,rep(x1_t,each=n),2)
data$x2 <- rnorm(N,0,40)
data$z1 <- rnorm(N,rep(z1_t,each=n),3)
data$z2 <- rep(z2_t,each=n)
```
fd <- Estimate fractional differencing parameter and/or differenciate series for a given d-value

### Description

This is a wrapper function for the `hurstSpec`-function as well as the `fracdiff`-package for internal use as part of the `arfimaMLM` function. The function estimates the fractional differencing param-
eter d of a series x and returns the fractionally differenced series. Alternatively, the differencing parameter can be estimated externally and included as a numeric argument in dval. In this case, the function returns the fractionally differenced series without estimating the differencing parameter itself.

**Usage**

```r
fd(x, dval = "Hurst", ...)

## S3 method for class 'character'
fd(x, dval = "Hurst", ...)

## S3 method for class 'numeric'
fd(x, dval, ...)
```

**Arguments**

- **x**: Numeric vector or time series to be fractionally differenced.
- **dval**: Call for a specific estimation method for the fractional differencing parameter in the `hurstSpec`-function ("Hurst") or the `fracdiff`-package ("ML", "GPH", or "Sperio"). Default is "Hurst". Alternatively, `dval` can be a numeric argument that will be directly passed to `diffseries` to calculate the fractionally differenced series according to the differencing parameter specified in `dval`. If a value of 0 is specified, the function will simply return the original series `x`.
- **...**: Further arguments passed to `hurstSpec`, `fracdiff`, `fdGPH`, `fdsperio`, or `diffseries`.

**Value**

- **series**: Fractionally differenced series.
- **estimator**: Method used to estimate differencing parameter ("Hurst", "ML", "GPH", or "Sperio"), prompts "external" if a differencing parameter was provided as numeric argument for `dval`.
- **value**: Estimated differencing parameter, returns the `dval` input if a numeric argument was provided for `dval`.

**Author(s)**

Patrick Kraft

**References**

See those in `hurstSpec`, `fracdiff`, `fdGPH`, `fdsperio`, and `diffseries`.

**See Also**

`hurstSpec`, `fracdiff`, `fdGPH`, `fdsperio`, `diffseries`
Examples

```r
require(fractald)
require(fracdiff)

set.seed(123)
series=fracdiff.sim(100, d=0.4, mu=10)$series

## S3 method for class 'character'
# estimates fractional differencing parameter d
# and differenciates series accordingly
series.fd1<-fd(series)
series.fd2<-fd(series, dval="ML")
series.fd3<-fd(series, dval="GPH")
series.fd4<-fd(series, dval="Sperio")

## S3 method for class 'numeric'
# differenciates series according to
# externally provided differencing parameter
series.fd5<-fd(series, dval=0.4)
```
Index

+Topic **manip**
  - arfimaMLM, 3
  - arfimaOLS, 6
  - arfimaPrep, 10
  - fd, 14
+Topic **models**
  - arfimaMLM, 3
  - arfimaOLS, 6
+Topic **multivariate**
  - arfimaMLM, 3
  - arfimaOLS, 6
+Topic **package**
  - ArfimaMLM-package, 2
+Topic **ts**
  - arfimaMLM, 3
  - arfimaOLS, 6
  - arfimaPrep, 10
  - fd, 14

ArfimaMLM, 5, 9, 13
ArfimaMLM (ArfimaMLM-package), 2
arfimaMLM, 2, 3, 10, 14
ArfimaMLM-package, 2
arfimaOLS, 2, 6, 10
arfimaPrep, 2, 5, 9, 10
arima, 3, 7, 11
diffseries, 15
fd, 2, 5, 9, 13, 14
fdGPH, 15
fdsSperio, 15
formula, 3, 7
fracdiff, 2, 3, 5, 7, 9, 11, 13–15
fractal, 3, 7, 11
hurstSpec, 2, 5, 9, 13–15
lm, 7, 9
lme4, 2, 5
lmer, 3

print.arfimaMLM (arfimaMLM), 3
print.arfimaOLS (arfimaOLS), 6
print.summary.arfimaMLM (arfimaMLM), 3
print.summary.arfimaOLS (arfimaOLS), 6
summary.arfimaMLM (arfimaMLM), 3
summary.arfimaOLS (arfimaOLS), 6