

# Package ‘UComp’

October 3, 2020

**Version** 2.1

**Title** Automatic Unobserved Components Models

**Description** Comprehensive analysis and forecasting  
of univariate time series using automatic  
unobserved components models and algorithms.

Harvey, AC (1989) <doi:10.1017/CBO9781107049994>.

Pedregal, DJ and Young PC (2002) <doi:10.1002/9780470996430>.

Durbin J and Koopman SJ (2012) <doi:10.1093/acprof:oso/9780199641178.001.0001>.

**Author** Diego J. Pedregal [aut, cre] (<<https://orcid.org/0000-0003-4958-0969>>)

**Depends** Rcpp (>= 1.0.3), R (>= 2.10)

**Imports** stats

**LinkingTo** Rcpp, RcppArmadillo

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Maintainer** Diego J. Pedregal <[Diego.Pedregal@uclm.es](mailto:Diego.Pedregal@uclm.es)>

**RoxygenNote** 7.1.1

**Date** 2020-10-05

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2020-10-03 14:40:02 UTC

## R topics documented:

airpas . . . . .	2
ch4 . . . . .	3
OECDgdp . . . . .	3
UC . . . . .	4
UCcomponents . . . . .	6
UCdisturb . . . . .	7
UCestim . . . . .	8

UCfilter . . . . .	9
UChp . . . . .	10
UCmodel . . . . .	10
UComp . . . . .	13
UCsetup . . . . .	14
UCsmooth . . . . .	16
UCvalidate . . . . .	17
USgdp . . . . .	17
<b>Index</b>	<b>19</b>

---

airpas	<i>Airpassengers in Spain</i>
--------	-------------------------------

---

### Description

Foreign arrivals by air in Spain in thousands of passengers (y). Easter and Business day dummy variables for airpas data (u).

### Usage

airpas

### Format

Time series objects.

Monthly data from January 1992 to December 2019

**airpas** air passengers data

**u** Easter and Business day dummy variables

### Source

**airpas**

### Examples

```
## Not run:
```

```
airpas
```

```
## End(Not run)
```

---

ch4	<i>Methane concentration at Cape Grim in Australia (ch4).</i>
-----	---

---

**Description**

Methane concentration at Cape Grim in Australia (ch4).

**Usage**

ch4

**Format**

Time series objects.

Monthly data from January 1992 to December 2019

**Source**

[CH4 data](#)

**Examples**

```
## Not run:
```

```
ch4
```

```
## End(Not run)
```

---

OECDgdp	<i>OECD GDP</i>
---------	-----------------

---

**Description**

Seasonally adjusted quarterly OECD real gross domestic product (OECDgdp).

**Usage**

OECDgdp

**Format**

Time series objects.

Quarterly data from 1962 to 2019

**Source**

[OECDgdp](#)

**Examples**

```
## Not run:
  OECDgdp

## End(Not run)
```

---

 UC
 

---



---

 UC
 

---

**Description**

Runs all relevant functions for UC modelling

**Usage**

```
UC(
  y,
  u = NULL,
  model = "?/none/?/?",
  h = NA,
  outlier = NA,
  tTest = FALSE,
  criterion = "aic",
  periods = NA,
  verbose = FALSE,
  stepwise = FALSE,
  p0 = NA,
  cLlik = TRUE,
  arma = TRUE
)
```

**Arguments**

- |       |  |
|-------|--|
| y     | a time series to forecast (it may be either a numerical vector or a time series object). This is the only input required. If a vector, the additional input <code>periods</code> should be supplied compulsorily (see below).  |
| u     | a matrix of input time series. If the output wanted to be forecast, matrix <code>u</code> should contain future values for inputs.   |
| model | the model to estimate. It is a single string indicating the type of model for each component. It allows two formats "trend/seasonal/irregular" or "trend/cycle/seasonal/irregular". The possibilities available for each component are: <ul style="list-style-type: none"> <li>• Trend: ? / none / rw / irw / llt / dt;</li> <li>• Seasonal: ? / none / equal / different;</li> <li>• Irregular: ? / none / arma(0, 0) / arma(p, q) - with p and q integer positive orders;</li> </ul> |

- Cycles: ? / none / combination of positive or negative numbers.  
Positive numbers fix the period of the cycle while negative values estimate the period taking as initial condition the absolute value of the period supplied. Several cycles with positive or negative values are possible and if a question mark is included, the model test for the existence of the cycles specified (check the examples below).

h	forecast horizon. If the model includes inputs h is not used, the length of u is used instead.
outlier	critical level of outlier tests. If NA it does not carry out any outlier detection (default). A negative value indicates critical minimum t test for one run of outlier detection after identification. A positive value indicates the critical minimum t test for outlier detection in any model during identification.
tTest	augmented Dickey Fuller test for unit roots (TRUE / FALSE). The number of models to search for is reduced, depending on the result of this test.
criterion	information criterion for identification ("aic", "bic" or "aicc").
periods	vector of fundamental period and harmonics.
verbose	intermediate results shown about progress of estimation (TRUE / FALSE).
stepwise	stepwise identification procedure (TRUE / FALSE).
p0	initial condition for parameter estimates.
cLlik	reserved input
arma	check for arma models for irregular components (TRUE / FALSE).

### Details

See help of UCmodel.

### Value

An object of class UComp. See UC. Standard methods applicable to UComp objects are print, summary, plot, fitted, residuals, logLik, AIC, BIC, coef, predict, tsdiag.

### Author(s)

Diego J. Pedregal

### See Also

[UC](#), [UCvalidate](#), [UCfilter](#), [UCsmooth](#), [UCdisturb](#), [UCcomponents](#), [UChp](#)

### Examples

```
y <- log(AirPassengers)
m1 <- UC(y)
m1 <- UC(y, model = "1lt/different/arma(0,0)")
```

---

UCcomponents

*UCcomponents*

---

### Description

Estimates components of UC models Standard methods applicable to UComp objects are print, summary, plot, fitted, residuals, logLik, AIC, BIC, coef, predict, tsdiag.

### Usage

```
UCcomponents(sys)
```

### Arguments

sys                    an object of type UComp created with UC

### Value

The same input object with the appropriate fields filled in, in particular:

comp                   Estimated components in matrix form

compV                   Estimated components variance in matrix form

### Author(s)

Diego J. Pedregal

### See Also

[UC](#), [UCmodel](#), [UCvalidate](#), [UCfilter](#), [UCsmooth](#), [UCdisturb](#), [UChp](#)

### Examples

```
m1 <- UC(log(AirPassengers))
m1 <- UCcomponents(m1)
```

---

`UCdisturb`*UCdisturb*

---

**Description**

Runs the Disturbance Smoother for UC models. Standard methods applicable to UComp objects are `print`, `summary`, `plot`, `fitted`, `residuals`, `logLik`, `AIC`, `BIC`, `coef`, `predict`, `tsdiag`.

**Usage**

```
UCdisturb(sys)
```

**Arguments**

`sys` an object of type UComp created with UC

**Value**

The same input object with the appropriate fields filled in, in particular:

<code>yFit</code>	Fitted values of output
<code>yFitV</code>	Variance of fitted values of output
<code>a</code>	State estimates
<code>P</code>	Variance of state estimates
<code>eta</code>	State perturbations estimates
<code>eps</code>	Observed perturbations estimates

**Author(s)**

Diego J. Pedregal

**See Also**

[UC](#), [UCmodel](#), [UCvalidate](#), [UCfilter](#), [UCsmooth](#), [UCcomponents](#), [UChp](#)

**Examples**

```
m1 <- UC(log(AirPassengers))
m1 <- UCdisturb(m1)
```

---

`UCestim`*UCestim*

---

**Description**

Estimates and forecasts UC models

**Usage**

```
UCestim(sys)
```

**Arguments**

`sys` an object of type `UComp` created with `UC`

**Details**

`UCestim` estimates and forecasts a time series using an UC model. Standard methods applicable to `UComp` objects are `print`, `summary`, `plot`, `fitted`, `residuals`, `logLik`, `AIC`, `BIC`, `coef`, `predict`, `tsdiag`.

**Value**

The same input object with the appropriate fields filled in, in particular:

<code>p</code>	Estimated parameters
<code>v</code>	Estimated innovations (white noise in correctly specified models)
<code>yFor</code>	Forecasted values of output
<code>yForV</code>	Variance of forecasted values of output
<code>criteria</code>	Value of criteria for estimated model

**Author(s)**

Diego J. Pedregal

**See Also**

[UC](#), [UCmodel](#), [UCvalidate](#), [UCfilter](#), [UCsmooth](#), [UCdisturb](#), [UCcomponents](#), [UChp](#)

**Examples**

```
m1 <- UCsetup(log(AirPassengers))
m1 <- UCestim(m1)
```

---

`UCfilter`*UCfilter*

---

**Description**

Runs the Kalman Filter for UC models. Standard methods applicable to UComp objects are `print`, `summary`, `plot`, `fitted`, `residuals`, `logLik`, `AIC`, `BIC`, `coef`, `predict`, `tsdiag`.

**Usage**

```
UCfilter(sys)
```

**Arguments**

`sys` an object of type UComp created with UC

**Value**

The same input object with the appropriate fields filled in, in particular:

<code>yFit</code>	Fitted values of output
<code>yFitV</code>	Variance of fitted values of output
<code>a</code>	State estimates
<code>P</code>	Variance of state estimates

**Author(s)**

Diego J. Pedregal

**See Also**

[UC](#), [UCmodel](#), [UCvalidate](#), [UCsmooth](#), [UCdisturb](#), [UCcomponents](#), [UChp](#)

**Examples**

```
m1 <- UC(log(AirPassengers))
m1 <- UCfilter(m1)
```

UChp

*UChp*

---

**Description**

Hodrick-Prescott filter estimation

**Usage**

```
UChp(y, lambda = 1600)
```

**Arguments**

y	A time series object
lambda	Smoothing constant (default: 1600)

**Value**

The cycle estimation

**Author(s)**

Diego J. Pedregal

**See Also**

[UC](#), [UCmodel](#), [UCvalidate](#), [UCfilter](#), [UCsmooth](#), [UCcomponents](#), [UCdisturb](#)

**Examples**

```
cycle <- UChp(USgdp)
plot(cycle)
```

---

UCmodel*UCmodel*

---

**Description**

Estimates and forecasts UC general univariate models

**Usage**

```
UCmodel(
  y,
  u = NULL,
  model = "?/none/?/?",
  h = NA,
  outlier = NA,
  tTest = FALSE,
  criterion = "aic",
  periods = NA,
  verbose = FALSE,
  stepwise = FALSE,
  p0 = NA,
  cLlik = TRUE,
  arma = TRUE
)
```

**Arguments**

y	a time series to forecast (it may be either a numerical vector or a time series object). This is the only input required. If a vector, the additional input periods should be supplied compulsorily (see below).
u	a matrix of input time series. If the output wanted to be forecast, matrix u should contain future values for inputs.
model	the model to estimate. It is a single string indicating the type of model for each component. It allows two formats "trend/seasonal/irregular" or "trend/cycle/seasonal/irregular". The possibilities available for each component are: <ul style="list-style-type: none"> <li>• Trend: ? / none / rw / irw / llt / dt;</li> <li>• Seasonal: ? / none / equal / different;</li> <li>• Irregular: ? / none / arma(0, 0) / arma(p, q) - with p and q integer positive orders;</li> <li>• Cycles: ? / none / combination of positive or negative numbers. Positive numbers fix the period of the cycle while negative values estimate the period taking as initial condition the absolute value of the period supplied. Several cycles with positive or negative values are possible and if a question mark is included, the model test for the existence of the cycles specified (check the examples below).</li> </ul>
h	forecast horizon. If the model includes inputs h is not used, the length of u is used instead.
outlier	critical level of outlier tests. If NA it does not carry out any outlier detection (default). A negative value indicates critical minimum t test for one run of outlier detection after identification. A positive value indicates the critical minimum t test for outlier detection in any model during identification.
tTest	augmented Dickey Fuller test for unit roots (TRUE / FALSE). The number of models to search for is reduced, depending on the result of this test.
criterion	information criterion for identification ("aic", "bic" or "aicc").

periods	vector of fundamental period and harmonics.
verbose	intermediate results shown about progress of estimation (TRUE / FALSE).
stepwise	stepwise identification procedure (TRUE / FALSE).
$p_0$	initial condition for parameter estimates.
cLlik	reserved input
arma	check for arma models for irregular components (TRUE / FALSE).

### Details

UCmodel is a function for modelling and forecasting univariate time series according to Unobserved Components models (UC). It sets up the model with a number of control variables that govern the way the rest of functions in the package will work. It also estimates the model parameters by Maximum Likelihood and forecasts the data. Standard methods applicable to UComp objects are print, summary, plot, fitted, residuals, logLik, AIC, BIC, coef, predict, tsdiag.

### Value

An object of class UComp. It is a list with fields including all the inputs and the fields listed below as outputs. All the functions in this package fill in part of the fields of any UComp object as specified in what follows (function UC fills in all of them at once):

After running UCmodel or UCestim:

p	Estimated parameters
v	Estimated innovations (white noise in correctly specified models)
yFor	Forecasted values of output
yForV	Variance of forecasted values of output
criteria	Value of criteria for estimated model

After running UCvalidate:

table	Estimation and validation table
-------	---------------------------------

After running UCcomponents:

comp	Estimated components in matrix form
compV	Estimated components variance in matrix form

After running UCfilter, UCsmooth or UCdisturb:

yFit	Fitted values of output
yFitV	Variance of fitted values of output
a	State estimates
P	Variance of state estimates
aFor	Forecasts of states
PFor	Forecasts of states variances

After running UCdisturb:

eta	State perturbations estimates
eps	Observed perturbations estimates

**Author(s)**

Diego J. Pedregal

**See Also**

[UC](#), [UCvalidate](#), [UCfilter](#), [UCsmooth](#), [UCdisturb](#), [UCcomponents](#), [UChp](#)

**Examples**

```
y <- log(AirPassengers)
m1 <- UCmodel(y)
m1 <- UCmodel(y, model = "11t/equal/arma(0,0)")
```

---

UComp

*UComp*

---

**Description**

A package for fast automatic identification of Unobserved Components models

**Details**

UComp is a package for time series modelling and forecasting of Unobserved Components models inspired on the structural family due to A.C. Harvey (Basic Structural Model: BSM), enhanced with automatic identification tools by Diego J. Pedregal. The package is designed for automatic identification among a wide range of possible models for trends, cycles, seasonal and irregular components. The model may include exogenous variables. ARMA irregular components and automatic detection of outliers are also possible.

**References**

- Harvey AC (1989). *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge University Press.
- de Jong, P. & Penzer, J. (1998). Diagnosing Shocks in Time Series, *Journal of the American Statistical Association*, 93, 442, 796-806.
- Pedregal, D. J., & Young, P. C. (2002). Statistical approaches to modelling and forecasting time series. In M. Clements, & D. Hendry (Eds.), *Companion to economic forecasting* (pp. 69–104). Oxford: Blackwell Publishers.
- Durbin J, Koopman SJ (2012). *Time Series Analysis by State Space Methods*. 38. Oxford University Press.
- Proietti T. and Luati A. (2013). Maximum likelihood estimation of time series models: the Kalman filter and beyond, in *Handbook of research methods and applications in empirical macroeconomics*, ed. Nigar Hashimzade and Michael Thornton, E. Elgar, UK.

**Maintainer**

Diego J. Pedregal

**Author(s)**

Diego J. Pedregal

UCsetup

*UCsetup***Description**

Sets up UC general univariate models

**Usage**

```
UCsetup(
  y,
  u = NULL,
  model = "?/none/?/?",
  h = NA,
  outlier = NA,
  tTest = FALSE,
  criterion = "aic",
  periods = NA,
  verbose = FALSE,
  stepwise = FALSE,
  p0 = NA,
  cLlik = TRUE,
  arma = TRUE
)
```

**Arguments**

- |       |  |
|-------|--|
| y     | a time series to forecast (it may be either a numerical vector or a time series object). This is the only input required. If a vector, the additional input <code>periods</code> should be supplied compulsorily (see below).  |
| u     | a matrix of input time series. If the output wanted to be forecast, matrix <code>u</code> should contain future values for inputs.   |
| model | the model to estimate. It is a single string indicating the type of model for each component. It allows two formats "trend/seasonal/irregular" or "trend/cycle/seasonal/irregular". The possibilities available for each component are: <ul style="list-style-type: none"> <li>• Trend: ? / none / rw / irw / llt / dt;</li> <li>• Seasonal: ? / none / equal / different;</li> <li>• Irregular: ? / none / arma(0, 0) / arma(p, q) - with p and q integer positive orders;</li> </ul> |

- Cycles: ? / none / combination of positive or negative numbers.  
Positive numbers fix the period of the cycle while negative values estimate the period taking as initial condition the absolute value of the period supplied. Several cycles with positive or negative values are possible and if a question mark is included, the model test for the existence of the cycles specified (check the examples below).

h	forecast horizon. If the model includes inputs h is not used, the length of u is used instead.
outlier	critical level of outlier tests. If NA it does not carry out any outlier detection (default). A negative value indicates critical minimum t test for one run of outlier detection after identification. A positive value indicates the critical minimum t test for outlier detection in any model during identification.
tTest	augmented Dickey Fuller test for unit roots (TRUE / FALSE). The number of models to search for is reduced, depending on the result of this test.
criterion	information criterion for identification ("aic", "bic" or "aicc").
periods	vector of fundamental period and harmonics.
verbose	intermediate results shown about progress of estimation (TRUE / FALSE).
stepwise	stepwise identification procedure (TRUE / FALSE).
p0	initial condition for parameter estimates.
cLlik	reserved input
arma	check for arma models for irregular components (TRUE / FALSE).

**Details**

See help of UC.

**Value**

An object of class UComp. See UCmodel. Standard methods applicable to UComp objects are print, summary, plot, fitted, residuals, logLik, AIC, BIC, coef, predict, tsdiag.

**Author(s)**

Diego J. Pedregal

**See Also**

[UC](#), [UCmodel](#), [UCvalidate](#), [UCfilter](#), [UCsmooth](#), [UCdisturb](#), [UCcomponents](#), [UChp](#)

**Examples**

```
y <- log(AirPassengers)
m1 <- UCsetup(y)
m1 <- UCsetup(y, model = "11t/equal/arma(0,0)")
m1 <- UCsetup(y, outlier = 4)
```

---

UCsmooth

*UCsmooth*

---

### Description

Runs the Fixed Interval Smoother for UC models. Standard methods applicable to UComp objects are print, summary, plot, fitted, residuals, logLik, AIC, BIC, coef, predict, tdiag.

### Usage

```
UCsmooth(sys)
```

### Arguments

sys            an object of type UComp created with UC

### Value

The same input object with the appropriate fields filled in, in particular:

yFit	Fitted values of output
yFitV	Variance of fitted values of output
a	State estimates
P	Variance of state estimates

### Author(s)

Diego J. Pedregal

### See Also

[UC](#), [UCmodel](#), [UCvalidate](#), [UCfilter](#), [UCdisturb](#), [UCcomponents](#), [UChp](#)

### Examples

```
m1 <- UC(log(AirPassengers))
m1 <- UCsmooth(m1)
```

---

UCvalidate

*UCvalidate*

---

**Description**

Shows a table of estimation and diagnostics results for UC models. Equivalent to print or summary.

**Usage**

```
UCvalidate(sys, printScreen = TRUE)
```

**Arguments**

sys                    an object of type UComp created with UC  
printScreen          print to screen or just return output table

**Value**

The same input object with the appropriate fields filled in, in particular:

table                Estimation and validation table

**Author(s)**

Diego J. Pedregal

**See Also**

[UC](#), [UCmodel](#), [UCfilter](#), [UCsmooth](#), [UCdisturb](#), [UCcomponents](#), [UChp](#)

**Examples**

```
m1 <- UC(log(AirPassengers))  
m1 <- UCvalidate(m1)
```

---

USgdp

*US GDP*

---

**Description**

Seasonally adjusted quarterly US real gross domestic product (USgdp).

**Usage**

```
USgdp
```

**Format**

Time series objects.  
Quarterly data from 1962 to 2019

**Source**

USgdp

**Examples**

```
## Not run:  
USgdp  
  
## End(Not run)
```

# Index

## \* datasets

- airpas, 2
- ch4, 3
- OECDgdp, 3
- USgdp, 17

airpas, 2

ch4, 3

OECDgdp, 3

UC, 4, 5–10, 13, 15–17

UCcomponents, 5, 6, 7–10, 13, 15–17

UCdisturb, 5, 6, 7, 8–10, 13, 15–17

UCestim, 8

UCfilter, 5–8, 9, 10, 13, 15–17

UChp, 5–9, 10, 13, 15–17

UCmodel, 6–10, 10, 15–17

UComp, 13

UCsetup, 14

UCsmooth, 5–10, 13, 15, 16, 17

UCvalidate, 5–10, 13, 15, 16, 17

USgdp, 17