

# Package ‘grangersearch’

May 8, 2026

**Title** Granger Causality Testing for Time Series

**Version** 0.1.0

**Description** Performs Granger causality tests on pairs of time series to determine causal relationships. Uses Vector Autoregressive (VAR) models to test whether one time series helps predict another beyond what the series' own past values provide. Returns structured results including p-values, test statistics, and causality conclusions for both directions.

**License** MIT + file LICENSE

**Depends** R (>= 4.1.0)

**URL** <https://github.com/nkorf/grangersearch>

**BugReports** <https://github.com/nkorf/grangersearch/issues>

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.3

**Imports** vars, stats, rlang (>= 1.0.0), tibble, generics

**Suggests** testthat (>= 3.0.0), knitr, rmarkdown

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Nikolaos Korfiatis [aut, cre] (ORCID:  
<<https://orcid.org/0000-0001-6377-4837>>)

**Maintainer** Nikolaos Korfiatis <nkorf@ionio.gr>

**Repository** CRAN

**Date/Publication** 2026-01-12 19:00:02 UTC

## Contents

example_causality . . . . .	2
glance.granger_result . . . . .	3

granger_causality_test . . . . .	4
granger_lag_select . . . . .	6
granger_search . . . . .	7
plot.granger_lag_select . . . . .	9
print.granger_result . . . . .	10
print.granger_search_result . . . . .	11
summary.granger_result . . . . .	11
tidy.granger_result . . . . .	12
<b>Index</b>	<b>13</b>

---

example_causality	<i>Example Time Series Data with Known Causal Relationship</i>
-------------------	--

---

## Description

A dataset containing two time series where `cause_x` Granger-causes `effect_y`. This data is useful for demonstrating and testing the Granger causality test.

## Usage

```
example_causality
```

## Format

A data frame with 200 rows and 3 variables:

**time** Integer. Time index from 1 to 200.

**cause\_x** Numeric. The "cause" time series, a random walk.

**effect\_y** Numeric. The "effect" time series, which depends on lagged values of `cause_x`.

## Details

The data was generated with the following process:

- `cause_x` is a random walk:  $X_t = X_{t-1} + \epsilon_t$
- `effect_y` depends on lagged `cause_x`:  $Y_t = 0.7 \cdot X_{t-1} + \nu_t$

where  $\epsilon_t \sim N(0, 1)$  and  $\nu_t \sim N(0, 0.5^2)$ .

When tested, `cause_x` should Granger-cause `effect_y`, but not vice versa.

## Source

Simulated data generated with seed 42.

## Examples

```
data(example_causality)

# Test for Granger causality
result <- granger_causality_test(
  example_causality,
  cause_x,
  effect_y
)
print(result)
```

---

glance.granger\_result *Glance at a granger\_result Object*

---

## Description

Returns a tibble with a single row containing model-level summary statistics. Compatible with the broom package conventions.

## Usage

```
## S3 method for class 'granger_result'
glance(x, ...)
```

## Arguments

**x** A granger\_result object.  
**...** Additional arguments (ignored).

## Value

A tibble with one row and columns:

**nobs** Integer. Number of observations.

**lag** Integer. VAR lag order used.

**alpha** Numeric. Significance level used.

**test** Character. Test type used.

**bidirectional** Logical. TRUE if causality detected in both directions.

**x\_causes\_y** Logical. TRUE if x Granger-causes y.

**y\_causes\_x** Logical. TRUE if y Granger-causes x.

**Examples**

```
set.seed(123)
x <- cumsum(rnorm(100))
y <- c(0, x[1:99]) + rnorm(100, sd = 0.5)
result <- granger_causality_test(x = x, y = y)
glance(result)
```

---

granger\_causality\_test

*Perform a Granger Causality Test on Two Time Series*


---

**Description**

Tests whether one time series Granger-causes another and vice versa. A variable X is said to Granger-cause Y if past values of X help predict Y beyond what past values of Y alone provide.

**Usage**

```
granger_causality_test(.data = NULL, x, y, lag = 1, alpha = 0.05, test = "F")
```

**Arguments**

<code>.data</code>	A data frame, tibble, or NULL. If provided, x and y are evaluated as column names within this data frame (tidyverse-style).
<code>x</code>	Either a numeric vector/time series, or (if <code>.data</code> is provided) an unquoted column name.
<code>y</code>	Either a numeric vector/time series of the same length as x, or (if <code>.data</code> is provided) an unquoted column name.
<code>lag</code>	Integer. The lag order for the VAR model. Default is 1.
<code>alpha</code>	Numeric. Significance level for the causality test (between 0 and 1). Default is 0.05.
<code>test</code>	Character. Type of test to perform. Currently only "F" (F-test) is supported. Default is "F".

**Details**

The Granger causality test is based on the idea that if X causes Y, then past values of X should contain information that helps predict Y above and beyond the information contained in past values of Y alone.

This function fits Vector Autoregressive (VAR) models using the `vars` package and performs F-tests to compare restricted and unrestricted models. The test is performed in both directions to detect unidirectional or bidirectional causality.

Note that Granger causality is a statistical concept based on prediction and temporal precedence. It does not necessarily imply true causal mechanisms.

**Value**

An object of class `granger_result` containing:

**x\_causes\_y** Logical. TRUE if X Granger-causes Y at the specified alpha level.

**y\_causes\_x** Logical. TRUE if Y Granger-causes X at the specified alpha level.

**p\_value\_xy** Numeric. P-value for the test of X causing Y.

**p\_value\_yx** Numeric. P-value for the test of Y causing X.

**test\_statistic\_xy** Numeric. Test statistic for X causing Y.

**test\_statistic\_yx** Numeric. Test statistic for Y causing X.

**lag** Integer. The lag order used.

**alpha** Numeric. The significance level used.

**test** Character. The test type used.

**n** Integer. Number of observations.

**x\_name** Character. Name of the X variable.

**y\_name** Character. Name of the Y variable.

**call** The matched call.

**Tidyverse Compatibility**

This function supports tidyverse-style syntax:

- Pipe-friendly: use with `%>%` or `|>`
- NSE column selection: pass unquoted column names when using a data frame
- Use `tidy.granger_result()` to get a tibble of results
- Use `glance.granger_result()` for model-level summary

**References**

Granger, C. W. J. (1969). Investigating Causal Relations by Econometric Models and Cross-spectral Methods. *Econometrica*, 37(3), 424-438.

**See Also**

[VAR](#) for the underlying VAR model, [causality](#) for an alternative implementation, `tidy.granger_result()` for tidying results.

**Examples**

```
# Vector-based usage
set.seed(123)
n <- 100
x <- cumsum(rnorm(n))
y <- c(0, x[1:(n-1)]) + rnorm(n, sd = 0.5)

result <- granger_causality_test(x = x, y = y)
print(result)
```

```

# Tidyverse-style with data frame
library(tibble)
df <- tibble(
  price = cumsum(rnorm(100)),
  volume = c(0, cumsum(rnorm(99)))
)

# Using pipe and column names
df |> granger_causality_test(price, volume)

# Get tidy results as tibble
result |> tidy()

# Different lag order
df |> granger_causality_test(price, volume, lag = 2)

```

---

granger\_lag\_select      *Lag Selection Analysis for Granger Causality*

---

## Description

Analyzes how Granger causality test results change across different lag orders. Returns detailed results for all lag-pair combinations, useful for optimal lag selection and visualization.

## Usage

```
granger_lag_select(.data, ..., lag = 1:4, alpha = 0.05, test = "F")
```

## Arguments

.data	A data frame or tibble containing the time series variables.
...	<tidy-select> Columns to include. If empty, all numeric columns are used.
lag	Integer vector. The lag orders to test. Default is 1:4.
alpha	Numeric. Significance level. Default is 0.05.
test	Character. Test type. Default is "F".

## Details

Unlike [granger\\_search\(\)](#) which returns only the best lag for each pair, this function returns results for all lag values tested. This is useful for:

- Visualizing how p-values change with lag order
- Selecting the optimal lag for each relationship
- Understanding the temporal dynamics of causality

**Value**

A tibble with one row per (cause, effect, lag) combination:

**cause** Character. The potential cause variable.

**effect** Character. The potential effect variable.

**lag** Integer. The lag order tested.

**statistic** Numeric. The F-test statistic.

**p.value** Numeric. The p-value.

**significant** Logical. Whether significant at alpha.

**See Also**

[granger\\_search\(\)](#) for getting best results across lags, [plot.granger\\_lag\\_select\(\)](#) for built-in visualization.

**Examples**

```
set.seed(123)
n <- 100
df <- data.frame(
  A = cumsum(rnorm(n)),
  B = cumsum(rnorm(n))
)
df$B <- c(0, 0.7 * df$A[1:(n-1)]) + rnorm(n, sd = 0.5)

# Get results for lags 1 through 5
lag_results <- granger_lag_select(df, lag = 1:5)

# Can be used with ggplot2 for visualization
# library(ggplot2)
# ggplot(lag_results, aes(x = lag, y = p.value, color = paste(cause, "->", effect))) +
#   geom_line() + geom_point() +
#   geom_hline(yintercept = 0.05, linetype = "dashed") +
#   labs(title = "P-values by Lag Order", color = "Direction")
```

**Description**

Performs Granger causality tests on all pairwise combinations of variables in a dataset. This is the core "search" functionality of the package, enabling discovery of causal relationships among multiple time series.

**Usage**

```
granger_search(
  .data,
  ...,
  lag = 1,
  alpha = 0.05,
  test = "F",
  include_insignificant = FALSE
)
```

**Arguments**

<code>.data</code>	A data frame or tibble containing the time series variables.
<code>...</code>	<code>&lt;tidy-select&gt;</code> Columns to include in the analysis. If empty, all numeric columns are used.
<code>lag</code>	Integer or integer vector. The lag order(s) for VAR models. If a vector (e.g., 1:4), tests are performed at each lag and the best (lowest p-value) result is returned for each pair. Default is 1.
<code>alpha</code>	Numeric. Significance level for hypothesis testing. Default is 0.05.
<code>test</code>	Character. Test type, currently only "F" supported. Default is "F".
<code>include_insignificant</code>	Logical. If FALSE (default), only return significant causal relationships. If TRUE, return all pairwise results.

**Details**

This function tests all  $n(n - 1)$  directed pairs for  $n$  variables. For each pair (X, Y), it tests whether X Granger-causes Y.

When multiple lags are specified (e.g., `lag = 1:4`), the function tests each pair at every lag and returns the result with the lowest p-value. This is useful for discovering the optimal lag structure.

The function is useful for exploratory analysis when you have multiple time series and want to discover which variables have predictive relationships.

**Value**

A tibble with one row per directed pair tested, containing:

**cause** Character. The potential cause variable name.

**effect** Character. The potential effect variable name.

**statistic** Numeric. The F-test statistic.

**p.value** Numeric. The p-value of the test.

**significant** Logical. Whether the result is significant at alpha.

**lag** Integer. The lag order used (best lag if multiple were tested).

**Multiple Testing**

When testing many pairs (and especially many lags), consider adjusting for multiple comparisons. The returned p-values are unadjusted. You can apply corrections such as Bonferroni or Benjamini-Hochberg using `stats::p.adjust()`.

**See Also**

`granger_causality_test()` for testing a single pair.

**Examples**

```
# Create dataset with multiple time series
set.seed(123)
n <- 100
df <- data.frame(
  A = cumsum(rnorm(n)),
  B = cumsum(rnorm(n)),
  C = cumsum(rnorm(n))
)
# B is caused by lagged A
df$B <- c(0, 0.7 * df$A[1:(n-1)]) + rnorm(n, sd = 0.5)

# Search for all causal relationships
granger_search(df)

# Include all results, not just significant ones
granger_search(df, include_insignificant = TRUE)

# Select specific columns
granger_search(df, A, B)

# Search across multiple lags (returns best lag for each pair)
granger_search(df, lag = 1:4)

# Search with specific lag
granger_search(df, lag = 2)
```

---

```
plot.granger_lag_select
```

*Plot Lag Selection Results*

---

**Description**

Creates a visualization of p-values across different lag orders for Granger causality tests.

**Usage**

```
## S3 method for class 'granger_lag_select'
plot(x, ...)
```

**Arguments**

`x` A `granger_lag_select` object from `granger_lag_select()`.  
`...` Additional arguments (ignored).

**Details**

This function creates a line plot showing how p-values change across different lag orders for each directed pair. A horizontal dashed line indicates the significance threshold (alpha).

For more customized plots, use the data directly with `ggplot2`.

**Value**

A base R plot (invisibly returns the input).

**Examples**

```
set.seed(123)
df <- data.frame(A = cumsum(rnorm(100)), B = cumsum(rnorm(100)))
df$B <- c(0, 0.7 * df$A[1:99]) + rnorm(100, sd = 0.5)

lag_results <- granger_lag_select(df, lag = 1:5)
plot(lag_results)
```

---

`print.granger_result` *Print Method for granger\_result Objects*

---

**Description**

Print Method for `granger_result` Objects

**Usage**

```
## S3 method for class 'granger_result'
print(x, ...)
```

**Arguments**

`x` A `granger_result` object.  
`...` Additional arguments (ignored).

**Value**

Invisibly returns the input object.

---

```
print.granger_search_result
```

*Print Method for granger\_search\_result Objects*

---

**Description**

Print Method for granger\_search\_result Objects

**Usage**

```
## S3 method for class 'granger_search_result'  
print(x, ...)
```

**Arguments**

x	A granger_search_result object.
...	Additional arguments (ignored).

**Value**

Invisibly returns the input object.

---

```
summary.granger_result
```

*Summary Method for granger\_result Objects*

---

**Description**

Summary Method for granger\_result Objects

**Usage**

```
## S3 method for class 'granger_result'  
summary(object, ...)
```

**Arguments**

object	A granger_result object.
...	Additional arguments (ignored).

**Value**

Invisibly returns the object.

---

tidy.granger\_result *Tidy a granger\_result Object*

---

### Description

Returns a tibble with one row per direction tested, containing test results. Compatible with the broom package conventions.

### Usage

```
## S3 method for class 'granger_result'  
tidy(x, ...)
```

### Arguments

`x` A granger\_result object.  
`...` Additional arguments (ignored).

### Value

A tibble with columns:

**direction** Character. The causal direction tested (e.g., "x -> y").

**cause** Character. The name of the potential cause variable.

**effect** Character. The name of the potential effect variable.

**statistic** Numeric. The F-test statistic.

**p.value** Numeric. The p-value of the test.

**significant** Logical. Whether the result is significant at the alpha level.

### Examples

```
set.seed(123)  
x <- cumsum(rnorm(100))  
y <- c(0, x[1:99]) + rnorm(100, sd = 0.5)  
result <- granger_causality_test(x = x, y = y)  
tidy(result)
```

# Index

## \* datasets

example\_causality, 2

causality, 5

example\_causality, 2

glance.granger\_result, 3

glance.granger\_result(), 5

granger\_causality\_test, 4

granger\_causality\_test(), 9

granger\_lag\_select, 6

granger\_lag\_select(), 10

granger\_search, 7

granger\_search(), 6, 7

plot.granger\_lag\_select, 9

plot.granger\_lag\_select(), 7

print.granger\_result, 10

print.granger\_search\_result, 11

stats::p.adjust(), 9

summary.granger\_result, 11

tidy.granger\_result, 12

tidy.granger\_result(), 5

VAR, 5