

# Package ‘kronos’

May 8, 2026

**Title** Microbiome Oriented Circadian Rhythm Analysis Toolkit

**Version** 1.0.0

**URL** <https://github.com/thomazbastiaanssen/kronos>

**Description** The goal of 'kronos' is to provide an easy-to-use framework to analyse circadian or otherwise rhythmic data using the familiar R linear modelling syntax, while taking care of the trigonometry under the hood.

**Depends** R (>= 3.5.0)

**Imports** stats, methods, ggplot2

**License** GPL (>= 3)

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Thomaz Bastiaanssen [aut, cre],  
Sarah-Jane Leigh [aut]

**Maintainer** Thomaz Bastiaanssen <thomazbastiaanssen@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-04-26 06:00:02 UTC

## Contents

bigdata . . . . .	2
bigmeta . . . . .	3
build_kronos_formula . . . . .	3
delistKronos_groupwise . . . . .	4
delistKronos_pairwise . . . . .	4
fit_cosinor_model . . . . .	5
fit_groupwise_model . . . . .	5
fw_kronos . . . . .	6
getKronos . . . . .	7
getKronos_fit . . . . .	8

getKronos_groupwise . . . . .	9
getKronos_input . . . . .	10
getKronos_pairwise . . . . .	11
getKronos_pairwise_p . . . . .	12
getKronos_params . . . . .	13
getKronos_trace . . . . .	14
get_cos_sine . . . . .	15
get_vars . . . . .	15
gg_kronos_acrogram . . . . .	16
gg_kronos_circle . . . . .	17
gg_kronos_sinusoid . . . . .	18
groupdata . . . . .	19
kronos . . . . .	19
kronosListToTable . . . . .	20
kronosOut-class . . . . .	21
kronos_anova . . . . .	22
kronos_predict . . . . .	22
onevariable . . . . .	23
pairwise_cosinor_model . . . . .	23
show,kronosOut-method . . . . .	24
twowaydata . . . . .	24

## Index 25

---

bigdata	<i>Snippet of example data to demonstrate the functionality of 'kronos' between and among three different groups</i>
---------	--

---

### Description

Snippet of example data to demonstrate the functionality of 'kronos' between and among three different groups

### Usage

```
bigdata
```

### Format

A long format data.frame object with three 113 columns, features, and 94 rows, samples.

### Source

Dummy data modified from untargeted hippocampal metabolomics over the day to demonstrate rhythmicity between multiple groups in a large dataset.

---

bigmeta	<i>Descriptive metadata for the 'bigdata' object, for the purpose of demonstration.</i>
---------	---

---

**Description**

Descriptive metadata for the 'bigdata' object, for the purpose of demonstration.

**Usage**

```
bigmeta
```

**Format**

A long format data.frame object with three 113 columns and 94 rows.

**Source**

Dummy metadata modified from untargeted hippocampal metabolomics over the day

---

build_kronos_formula	<i>Update 'kronos' formula in light of sine and cosine components</i>
----------------------	---

---

**Description**

Update 'kronos' formula.

**Usage**

```
build_kronos_formula(formula, time, verbose)
```

**Arguments**

formula	A formula. Use the time function to designate which variable represents time.
time	A string. Should be the column name containing the time values.
verbose	A boolean. Toggles whether to print diagnostic information while running. Useful for debugging errors on large data sets.

---

`delistKronos_groupwise`*Get Results from list of KronosOut Objects*

---

**Description**

These functions provides a unified wrapper to retrieve results from a list of kronosOut objects.

**Usage**

```
delistKronos_groupwise(kronos_list, padjust = TRUE)
```

**Arguments**

`kronos_list` a list of preferrably named kronosOut objects.  
`padjust` a boolean. Toggles FDR using Benjamini Hochbergs procedure.

**Value**

A table with circadian output stats per group per feature.

---

`delistKronos_pairwise` *Get Results from list of KronosOut Objects*

---

**Description**

These functions provides a unified wrapper to retrieve results from a list of kronosOut objects.

**Usage**

```
delistKronos_pairwise(kronos_list, padjust = TRUE)
```

**Arguments**

`kronos_list` a list of preferrably named kronosOut objects.  
`padjust` a boolean. Toggles FDR using Benjamini Hochbergs procedure.

**Value**

ANOVA-like adjusted p-values for how each factor interacts with time.

---

fit\_cosinor\_model      *Fit cosinor model*

---

### Description

Fit cosinor model for totality of data

### Usage

```
fit_cosinor_model(
  formula,
  data,
  time = NULL,
  verbose = verbose,
  for_pw = FALSE
)
```

### Arguments

formula	A formula. Use the <code>time</code> function to designate which variable represents time.
data	Input data
time	A string. Should be the column name containing the time values.
verbose	A boolean. Toggles whether to print diagnostic information while running. Useful for debugging errors on large datasets.
for_pw	A boolean. Toggles whether to perform pairwise ANOVAs as a TukeyHSD-like post-hoc.

---

fit\_groupwise\_model      *Fit cosinor model*

---

### Description

Fit cosinor model for one aspect of data. Called by main 'kronos' function.

### Usage

```
fit_groupwise_model(data, group, time, period, verbose)
```

### Arguments

data	input data
group	A character string. Signifies which group will be assessed.
time	A string. Should be the column name containing the time values.
period	A numeric. The length of a period, in the same format as the <code>time</code> parameter.
verbose	A boolean. Toggles whether to print diagnostic information while running. Useful for debugging errors on large data sets.

fw\_kronos

*Fit a cosinor model and extract relevant parameters on a feature table.***Description**

This wrapper applies `kronos()`, the main workhorse function in the 'kronos' package. It manages the individual functionalities of 'kronos', including rhythmicity analysis and differential rhythmicity.

**Usage**

```
fw_kronos(
  x,
  formula,
  metadata,
  time = NULL,
  period = 24,
  verbose = FALSE,
  pairwise = FALSE
)
```

**Arguments**

<code>x</code>	Input data. A table with rows being features and columns being samples
<code>formula</code>	A formula. Use the <code>time</code> function to designate which variable represents time. Leave the left-hand side of the formula empty as it will be sequentially replaced by every feature in the table.
<code>metadata</code>	A metadata table, with rows being samples and columns being metadata entries
<code>time</code>	A string. Should be the column name containing the time values.
<code>period</code>	A numeric. The length of a period, in the same format as the <code>time</code> parameter.
<code>verbose</code>	A boolean. Toggles whether to print diagnostic information while running. Useful for debugging errors on large datasets.
<code>pairwise</code>	A boolean. Toggles whether to perform pairwise ANOVAs as a TukeyHSD-like post-hoc.

**Value**

A list of `kronosOut` S4 objects containing coefficients and all operations.

**Examples**

```
#Load prepared data stored in 'kronos' library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
```

```
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

getKronos

*Get Results from KronosOut Object*

---

## Description

These functions provides a unified wrapper to retrieve results from a `kronosOut` object.

## Usage

```
getKronos(kronosOut, target)
```

## Arguments

<code>kronosOut</code>	a <code>kronosOut</code> output object.
<code>target</code>	the specific entry of the <code>kronosOut</code> object to be extracted.

## Value

A `data.frame` of results.

## Examples

```
#Load prepared data stored in 'kronos' library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
```

```

getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)

```

---

getKronos\_fit

*Get Results from KronosOut Object*


---

## Description

These functions provides a unified wrapper to retrieve results from a kronosOut object.

## Usage

```
getKronos_fit(kronosOut)
```

## Arguments

kronosOut      a kronosOut output object.

## Value

The model fit used.

## Examples

```

#Load prepared data stored in 'kronos' library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:

```

```
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

getKronos\_groupwise    *Get Results from KronosOut Object*

---

## Description

These functions provides a unified wrapper to retrieve results from a kronosOut object.

## Usage

```
getKronos_groupwise(kronosOut)
```

## Arguments

kronosOut        a kronosOut output object.

## Value

Rhythmicity parameters per group.

## Examples

```
#Load prepared data stored in 'kronos' library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
```

```

out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)

```

---

getKronos\_input

*Get Results from KronosOut Object*


---

### Description

These functions provides a unified wrapper to retrieve results from a kronosOut object.

### Usage

```
getKronos_input(kronosOut)
```

### Arguments

kronosOut      a kronosOut output object.

### Value

The data used as input for the model.

### Examples

```

#Load prepared data stored in 'kronos' library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

```

```
#Extracting data from the output object:  
kronosListToTable(out_list)
```

```
#Plotting:  
gg_kronos_acrogram(out_list)
```

---

getKronos\_pairwise      *Get Results from KronosOut Object*

---

## Description

These functions provides a unified wrapper to retrieve results from a kronosOut object.

## Usage

```
getKronos_pairwise(kronosOut)
```

## Arguments

kronosOut      a kronosOut output object.

## Value

Pairwise comparisons between groups.

## Examples

```
#Load prepared data stored in 'kronos' library  
data("kronos_demo")  
output <- kronos(formula = Variable_1 ~ time(Timepoint),  
data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)  
  
#Extracting data from the output object:  
getKronos_fit(output)  
getKronos_trace(output)  
getKronos_groupwise(output)  
  
#Plotting:  
gg_kronos_circle(output)  
gg_kronos_sinusoid(output)  
  
#For high-dimensional data, use fw_kronos:  
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),  
metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)  
  
#Extracting data from the output object:  
kronosListToTable(out_list)
```

```
#Plotting:
gg_kronos_acrogram(out_list)
```

---

getKronos\_pairwise\_p *Get Results from KronosOut Object*

---

### Description

These functions provides a unified wrapper to retrieve results from a kronosOut object.

### Usage

```
getKronos_pairwise_p(kronosOut)
```

### Arguments

kronosOut      a kronosOut output object.

### Value

ANOVA-like adjusted p-values for how each factor interacts with time.

### Examples

```
#Load prepared data stored in 'kronos' library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

getKronos_params	<i>Get Results from KronosOut Object</i>
------------------	--

---

### Description

These functions provides a unified wrapper to retrieve results from a kronosOut object.

### Usage

```
getKronos_params(kronosOut)
```

### Arguments

kronosOut      a kronosOut output object.

### Value

The names and values of additional circadian model parameters, mostly for plotting purposes.

### Examples

```
#Load prepared data stored in 'kronos' library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

`getKronos_trace`*Get Results from KronosOut Object*

---

**Description**

These functions provides a unified wrapper to retrieve results from a `kronosOut` object.

**Usage**

```
getKronos_trace(kronosOut)
```

**Arguments**

`kronosOut` a `kronosOut` output object.

**Value**

The traces per group for plotting.

**Examples**

```
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

get_cos_sine	<i>Get sine and cosine components</i>
--------------	---------------------------------------

---

**Description**

Based on 'cosinor' and 'limorhyde' packages

**Usage**

```
get_cos_sine(data, period, colnamePrefix = NULL)
```

**Arguments**

data	input data
period	A numeric. The length of a period, in the same format as the time parameter.
colnamePrefix	A character string. Typically the name of the "Time" variable.

---

get_vars	<i>Figure out what variable represents time. Called by main 'kronos' function.</i>
----------	--

---

**Description**

Extracts time from the formula and from the time argument. Also handles inconsistencies.

**Usage**

```
get_vars(formula, time, data, verbose = verbose)
```

**Arguments**

formula	A formula. Use the time function to designate which variable represents time.
time	A string. Should be the column name containing the time values.
data	input data
verbose	A boolean. Toggles whether to print diagnostic information while running. Useful for debugging errors on large data sets.

---

gg\_kronos\_acrogram     *A plotting method for acrophase circleplots using 'ggplot2'.*

---

### Description

Wrapper around 'ggplot2' to make circadian circleplots.

### Usage

```
gg_kronos_acrogram(kronosOutList)
```

### Arguments

kronosOutList     A list of KronosOut output objects from the main kronos function.

### Value

a 'ggplot2' compatible object.

### Examples

```
#Load prepared data stored in Kronos library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

gg\_kronos\_circle      *A plotting method for circadian plots using 'ggplot2'.*

---

## Description

Wrapper around 'ggplot2' to make circadian circleplots.

## Usage

```
gg_kronos_circle(kronosOut)
```

## Arguments

kronosOut      an output object from the main kronos function.

## Value

a 'ggplot2' compatible object.

## Examples

```
#Load prepared data stored in Kronos library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

gg\_kronos\_sinusoid     *A plotting method for circadian plots using 'ggplot2'.*

---

### Description

Wrapper around 'ggplot2' to make figures with a sinusoid trace.

### Usage

```
gg_kronos_sinusoid(kronosOut, fill = "unique_group")
```

### Arguments

kronosOut	an output object from the main kronos function.
fill	The name of the variable that should be used to mark different groups. In the case of a single group, leave empty.

### Value

a 'ggplot2' compatible object.

### Examples

```
#Load prepared data stored in Kronos library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

groupdata	<i>Snippet of example data to demonstrate the functionality of 'kronos' between and among three different groups</i>
-----------	--

---

**Description**

Snippet of example data to demonstrate the functionality of 'kronos' between and among three different groups

**Usage**

```
groupdata
```

**Format**

A long format data.frame object with three columns, and 94 rows, samples.

**Source**

Dummy data modified from PCR analysis of mouse ileum over the day to demonstrate rhythmicity between multiple groups.

---

kronos	<i>Fit a cosinor model and extract relevant parameters</i>
--------	--

---

**Description**

This is the main workhorse function in the 'kronos' package. It manages the individual functionalities of 'kronos', including rhythmicity analysis and differential rhythmicity.

**Usage**

```
kronos(  
  formula,  
  data,  
  time = NULL,  
  period = 24,  
  verbose = TRUE,  
  pairwise = TRUE  
)
```

**Arguments**

formula	A formula. Use the <code>time</code> function to designate which variable represents time.
data	input data
time	A string. Should be the column name containing the time values.
period	A numeric. The length of a period, in the same format as the <code>time</code> variable
verbose	A boolean. Toggles whether to print diagnostic information while running. Useful for debugging errors on large data sets.
pairwise	A boolean. Toggles whether to perform pairwise ANOVAs as a TukeyHSD-like post-hoc.

**Value**

A `kronosOut` S4 object containing coefficients and all operations.

**Examples**

```
#Load prepared data stored in Kronos library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

kronosListToTable	<i>Wrangle results from list of KronosOut Objects to publication ready table.</i>
-------------------	---

---

**Description**

These functions provides a unified wrapper to retrieve results from a list of kronosOut objects.

**Usage**

```
kronosListToTable(kronos_list, padjust = TRUE)
```

**Arguments**

`kronos_list` a list of preferably named kronosOut objects.  
`padjust` a boolean. Toggles FDR using Benjamini and Hochbergs procedure.

**Value**

A table with circadian output stats per group per feature.

**Examples**

```
#Load prepared data stored in 'kronos' library
data("kronos_demo")
output <- kronos(formula = Variable_1 ~ time(Timepoint),
  data = onevariable, period = 24, verbose = TRUE, pairwise = FALSE)

#Extracting data from the output object:
getKronos_fit(output)
getKronos_trace(output)
getKronos_groupwise(output)

#Plotting:
gg_kronos_circle(output)
gg_kronos_sinusoid(output)

#For high-dimensional data, use fw_kronos:
out_list = fw_kronos(x = bigdata[1:50,], formula = ~ Group + time(Timepoint),
  metadata = bigmeta, period = 24, verbose = FALSE, pairwise = TRUE)

#Extracting data from the output object:
kronosListToTable(out_list)

#Plotting:
gg_kronos_acrogram(out_list)
```

---

kronosOut-class	<i>An S4 class to contain all output parameters and plotting information from the 'kronos' main pipeline</i>
-----------------	--

---

**Description**

kronosOut is the main output container of the main 'kronos' functions.

**Slots**

input A data.frame with the data that was fed to the main workhorse function as 'x'  
 fit An lm fit for the entire model for the purpose of assessing differential rhythmicity.  
 to\_plot A data.frame with the traces required to plot individual sinusoid curves  
 ind\_fit A data.frame with the parameters from individual rhythmic model fits.  
 pairwise\_t A data.frame with the p.values for pairwise comparisons, if applicable.

---

kronos_anova	<i>Extract p-value from full fit</i>
--------------	--------------------------------------

---

**Description**

Compute p-values from full fit.

**Usage**

```
kronos_anova(fit, time)
```

**Arguments**

fit	A lm model fit.
time	A string. Should be the column name containing the time values.

---

kronos_predict	<i>Give tracing information for plotting purposes</i>
----------------	---

---

**Description**

Generate data needed to plot cosinor trace line.

**Usage**

```
kronos_predict(fit, period, time, factors, verbose = verbose)
```

**Arguments**

fit	A model fit
period	A numeric. The length of a period, in the same format as the time parameter.
time	A string. Should be the column name containing the time values.
factors	A vector. The names of the independent variables.
verbose	A boolean. Toggles whether to print diagnostic information while running. Useful for debugging errors on large data sets.

---

onevariable	<i>Snippet of example data to demonstrate the functionality of 'kronos' in the most simple scenario.</i>
-------------	--

---

**Description**

Snippet of example data to demonstrate the functionality of 'kronos' in the most simple scenario.

**Usage**

```
onevariable
```

**Format**

A long format data.frame object with three columns, and 31 rows, samples.

**Source**

Dummy data modified from metagenomic analysis of mouse caecal contents over the day to demonstrate rhythmicity within one group.

---

pairwise_cosinor_model	<i>Fit pairwise cosinor models as some sort of TukeyHSD.</i>
------------------------	--

---

**Description**

Fit cosinor model for subset of data. Called by main 'kronos' function.

**Usage**

```
pairwise_cosinor_model(data, formula, time, verbose)
```

**Arguments**

data	input data
formula	A formula. Use the time function to designate which variable represents time.
time	A string. Should be the column name containing the time values.
verbose	A boolean. Toggles whether to print diagnostic information while running. Useful for debugging errors on large data sets.

---

show, kronosOut-method *Show method for kronosOut object.*

---

### Description

method to print kronosOut object by calling show. Since kronosOut objects are typically unwieldy, also gives some tips on how to handle it.

### Usage

```
## S4 method for signature 'kronosOut'  
show(object)
```

### Arguments

object            An kronosOut object.

### Value

Does not return anything, for efficiency reasons. The obvious side effect is output to the terminal.

---

twowaydata            *Snippet of example data to demonstrate the functionality of 'kronos' in the two-factor design scenario.*

---

### Description

Snippet of example data to demonstrate the functionality of 'kronos' in the two-factor design scenario.

### Usage

```
twowaydata
```

### Format

A wide format data.frame object with nine columns, and 150 rows, samples.

### Source

Dummy data modified from metagenomic analysis of mouse caecal contents over the day to demonstrate rhythmicity within one group.

# Index

## \* datasets

- [bigdata](#), 2
- [bigmeta](#), 3
- [groupdata](#), 19
- [onevariable](#), 23
- [twowaydata](#), 24

- [bigdata](#), 2
- [bigmeta](#), 3
- [build\\_kronos\\_formula](#), 3

- [delistKronos\\_groupwise](#), 4
- [delistKronos\\_pairwise](#), 4

- [fit\\_cosinor\\_model](#), 5
- [fit\\_groupwise\\_model](#), 5
- [fw\\_kronos](#), 6

- [get\\_cos\\_sine](#), 15
- [get\\_vars](#), 15
- [getKronos](#), 7
- [getKronos\\_fit](#), 8
- [getKronos\\_groupwise](#), 9
- [getKronos\\_input](#), 10
- [getKronos\\_pairwise](#), 11
- [getKronos\\_pairwise\\_p](#), 12
- [getKronos\\_params](#), 13
- [getKronos\\_trace](#), 14
- [gg\\_kronos\\_acrogram](#), 16
- [gg\\_kronos\\_circle](#), 17
- [gg\\_kronos\\_sinusoid](#), 18
- [groupdata](#), 19

- [kronos](#), 19
- [kronos\\_anova](#), 22
- [kronos\\_predict](#), 22
- [kronosListToTable](#), 20
- [kronosOut-class](#), 21

- [onevariable](#), 23

- [pairwise\\_cosinor\\_model](#), 23

- [show, kronosOut-method](#), 24

- [twowaydata](#), 24