

# Package: mixediffusion (via r-universe)

June 9, 2026

**Type** Package

**Title** Mixed-Effects Diffusion Models with General Drift

**Description** Provides tools for likelihood-based inference in one-dimensional stochastic differential equations with mixed effects using expectation–maximization (EM) algorithms. The package supports Wiener and Ornstein–Uhlenbeck diffusion processes with user-specified drift functions, allowing flexible parametric forms including polynomial, exponential, and trigonometric structures. Estimation is performed via Markov chain Monte Carlo EM.

**Version** 1.0.1

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Imports** dplyr, adaptMCMC

**Maintainer** Pedro Abraham Montoya Calzada  
<pedroabraham.montoya@gmail.com>

**Depends** R (>= 4.5)

**LazyData** true

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Pedro Abraham Montoya Calzada [aut, cre, cph] (ORCID:  
<<https://orcid.org/0009-0002-3497-210X>>), Rogelio Salinas  
Gutiérrez [aut, cph] (ORCID:  
<<https://orcid.org/0000-0002-1669-4460>>), Silvia  
Rodríguez-Narciso [aut, cph] (ORCID:  
<<https://orcid.org/0000-0001-5429-5914>>), Netzahualcōyotl  
Castañeda-Leyva [aut, cph] (ORCID:  
<<https://orcid.org/0000-0001-9414-3923>>)

**Repository** <https://abraham1011.r-universe.dev>

**Date/Publication** 2026-04-10 08:03:18 UTC

**RemoteUrl** <https://github.com/cran/mixeddiffusion>

**RemoteRef** HEAD

**RemoteSha** 7510f2904153f6f7789b953f5955430e8183b912

## Contents

datasim01 . . . . .	2
datasim02 . . . . .	3
datasim03 . . . . .	4
fit_ou . . . . .	5
fit_wiener . . . . .	7
plot_paths . . . . .	9

**Index** **10**

---

datasim01	<i>Simulated diffusion dataset 01</i>
-----------	---------------------------------------

---

### Description

A simulated dataset consisting of discretely observed trajectories generated from a one-dimensional Wiener diffusion model with random effects in the drift term.

### Usage

```
data(datasim01)
```

### Format

A data frame with observations from multiple experimental units and the following variables:

t Observation time.

unit Unit identifier.

Y Observed process value.

### Details

The data were generated from the stochastic differential equation

$$dY_k(t) = a_k t dt + \sigma dW_k(t),$$

where the unit-specific random effects follow a normal distribution

$$a_k \sim \mathcal{N}(10, 3),$$

and the diffusion variance is given by  $\sigma^2 = 500$ .

The process was simulated for  $K = 50$  units, each observed at  $n = 200$  equally spaced time points, with initial condition  $Y_{k0} = 1$ .

**Value**

A data frame with simulated data.

**Source**

Simulated data generated for illustrative purposes.

**References**

None.

**Examples**

```
data(datasim01)
plot_paths(df = datasim01)
```

---

datasim02

*Simulated diffusion dataset 02*

---

**Description**

A simulated dataset consisting of discretely observed trajectories generated from a one-dimensional Wiener diffusion model with random effects in the drift term.

**Usage**

```
data(datasim02)
```

**Format**

A data frame with observations from multiple experimental units and the following variables:

t Observation time.

unit Unit identifier.

Y Observed process value.

**Details**

The data were generated from the stochastic differential equation

$$dY_k(t) = a_k \sin(\pi t) dt + \sigma dW_k(t),$$

where the unit-specific random effects follow a normal distribution

$$a_k \sim \mathcal{N}(10, 3),$$

and the diffusion variance is given by  $\sigma^2 = 50$ .

The integrated drift used for simulation is given by

$$\mu_{\Delta}(a_k, t_i, t_{i-1}) = \frac{a_k}{\pi} \{\cos(\pi t_{i-1}) - \cos(\pi t_i)\}.$$

The process was simulated for  $K = 50$  units, each observed at  $n = 200$  equally spaced time points, with initial condition  $Y_{k0} = 1$ .

**Value**

A data frame with simulated data.

**Source**

Simulated data generated for illustrative purposes.

**References**

None.

**Examples**

```
data(datasim02)
plot_paths(df = datasim02)
```

---

datasim03

*Simulated diffusion dataset 03*

---

**Description**

A simulated dataset consisting of discretely observed trajectories generated from a one-dimensional Ornstein–Uhlenbeck diffusion model with a time-dependent mean function and random effects.

**Usage**

```
data(datasim03)
```

**Format**

A data frame with observations from multiple experimental units and the following variables:

t Observation time.

unit Unit identifier.

Y Observed process value.

**Details**

The data were generated from the stochastic differential equation

$$dY_k(t) = \lambda\{a_k t - Y_k(t)\} dt + \sigma dW_k(t),$$

where the unit-specific random effects follow a normal distribution

$$a_k \sim \mathcal{N}(10, 3),$$

the mean-reversion parameter is given by  $\lambda = 0.75$ , and the diffusion variance satisfies  $\sigma^2 = 500$ .

The process was simulated for  $K = 30$  units, each observed at  $n = 200$  equally spaced time points over the interval  $[0, 10]$ , with initial condition  $Y_k(0) = 0$ .

**Value**

A data frame with simulated data.

**Source**

Simulated data generated for illustrative purposes.

**References**

None.

**Examples**

```
data(datasim03)
plot_paths(df = datasim03)
```

---

fit\_ou

---

*Inference about the parameters of an Ornstein–Uhlenbeck process*


---

**Description**

Implements the EM algorithm to perform inference on the parameters of an Ornstein–Uhlenbeck process with mixed drift effects.

**Usage**

```
fit_ou(df,
  mu = "at^1",
  tol = 1e-4,
  max_iter = 100,
  theta = NULL,
  M = 100,
  verbose = TRUE,
  mu_cond = NULL,
  n_mcmc = 1000,
  burnin = 500)
```

**Arguments**

df	Data frame with the observed data. It must include the columns t (time), unit (unit identifier) and Y (the observed trajectory).
mu	Functional form of the drift. Supported drifts include "at^p", "exp(at)", "cos(at)" and "sin(at)". The default is "at^1".
tol	Convergence tolerance for the EM algorithm. The algorithm stops when the maximum absolute difference between the parameter estimates at two consecutive EM iterations is smaller than tol.
max_iter	Maximum number of EM iterations.
theta	Optional named vector of initial parameter values. If NULL, default initial values are used.
M	Number of Monte Carlo samples used to approximate the conditional expectations in the E-step.
verbose	Logical indicating whether to print EM iteration progress.
mu_cond	Optional user-supplied function defining the conditional mean of the process. If NULL, it is constructed automatically from mu.
n_mcmc	Number of MCMC iterations used in the E-step to sample the random effects.
burnin	Number of initial MCMC iterations discarded.

**Details**

The model is a one-dimensional Ornstein–Uhlenbeck diffusion defined by

$$dY_{kt} = \lambda\{\mu(t, a_k) - Y_{kt}\} dt + \sigma dW_{kt},$$

where  $\mu(t, a_k)$  is a user-specified drift function depending on a unit-specific random effect  $a_k \sim \mathcal{N}(\mu_a, \sigma_a^2)$ .

The parameter  $\lambda$  controls the strength of mean reversion toward the time-dependent mean.

For discretely observed trajectories, the conditional mean is given by

$$E\{Y_{kt_i} \mid Y_{kt_{i-1}}, a_k\} = Y_{kt_{i-1}} e^{-\lambda \Delta t_i} + \lambda \int_{t_{i-1}}^{t_i} e^{-\lambda(t_i-s)} \mu(s, a_k) ds.$$

The function mu\_cond represents this conditional mean. If not provided, it is constructed automatically from the drift specification supplied through mu using closed-form expressions.

**Value**

A named numeric vector containing the estimated model parameters:

mu_a	Estimated mean of the random effects distribution.
sigma2_a	Estimated variance of the random effects distribution.
sigma2	Estimated diffusion variance of the process.
lambda	Estimated mean reversion parameter.

**Examples**

```
library(mixeddiffusion)
data(datasim03)
plot_paths(df = datasim03)
fit <- fit_ou(df = datasim03, mu = "at^1",
             verbose = FALSE, max_iter = 1)
fit
```

fit\_wiener

*Inference about the parameters of a Wiener process***Description**

Implement the EM algorithm to perform inference on the parameters of a Wiener process with mixed drift effects.

**Usage**

```
fit_wiener(df,
           mu = "at^1",
           tol = 1e-4,
           max_iter = 100,
           theta = NULL,
           M = 100,
           verbose = TRUE,
           mu_dlt = NULL,
           n_mcmc = 1000,
           burnin = 500)
```

**Arguments**

df	Data frame with the observed data. It must include the columns t (time), unit (unit identifier) and Y (the observed trajectory).
mu	Functional form of the drift. Supported drifts include "at^p", "exp(at)", "cos(at)" and "sin(at)". The default is "at^1".
tol	Convergence tolerance for the EM algorithm. The algorithm stops when the maximum absolute difference between the parameter estimates at two consecutive EM iterations is smaller than tol.
max_iter	Maximum number of EM iterations.
theta	Optional named vector of initial parameter values. If NULL, default initial values are used.
M	Number of Monte Carlo samples used to approximate the conditional expectations in the E-step.
verbose	Logical indicating whether to print EM iteration progress.

mu_dlt	Optional user-supplied function defining the integrated drift term. If NULL, it is constructed automatically from mu.
n_mcmc	Number of MCMC iterations used in the E-step to sample the random effects.
burnin	Number of initial MCMC iterations discarded.

### Details

The model is a one-dimensional Wiener diffusion defined by

$$dY_{kt} = \mu(t, a_k)dt + \sigma dW_{kt}$$

,  
 where  $\mu(t, a_k)$  is a user-specified drift function depending on a unit-specific random effect  $a_k \sim \mathcal{N}(\mu_a, \sigma_a^2)$ .

For discretely observed trajectories, the mean of the increments is given by the integrated drift

$$\mu_{\Delta}(a_k, t_i, t_{i-1}) = \int_{t_{i-1}}^{t_i} \mu(s, a_k) ds.$$

The function `mu_dlt` represents this integrated drift. If not provided, it is constructed automatically from `mu` using closed-form expressions for common drift specifications.

### Value

A named numeric vector containing the estimated model parameters:

mu_a	Estimated mean of the random effects distribution.
sigma2_a	Estimated variance of the random effects distribution.
sigma2	Estimated diffusion variance of the Wiener process.

### Examples

```
library(mixeddiffusion)
data(datasim01)
plot_paths(df = datasim01)
fit <- fit_wiener(df = datasim01, mu = "at^1",
                 verbose = FALSE, max_iter = 1)
fit

# mu(ak,t) = ak*sin(pi*t)
data(datasim02)
plot_paths(df = datasim02)
mu_dlt_new <- function(ak,ti,ti_1){
  value <- -ak*(cos(pi*ti) - cos(pi*ti_1))
  return(value)
}

fit <- fit_wiener(df = datasim02, mu_dlt = mu_dlt_new,
                 verbose = FALSE, max_iter = 1)
fit
```

---

plot_paths	<i>Plot panel trajectories</i>
------------	--------------------------------

---

**Description**

Plots multiple trajectories from panel or longitudinal data grouped by unit.

**Usage**

```
plot_paths(df,  
  col = NULL,  
  lwd = 1,  
  xlab = "t",  
  ylab = "Y",  
  main = NULL,  
  ...)
```

**Arguments**

df	Data frame containing the observed data. It must include the columns <code>t</code> (time), <code>unit</code> (unit identifier) and <code>Y</code> (observed values).
col	Optional vector of colors, one per unit. If <code>NULL</code> , a grayscale palette is used.
lwd	Line width used for the trajectories.
xlab	Label for the x-axis.
ylab	Label for the y-axis.
main	Optional main title for the plot.
...	Additional graphical parameters passed to <code>plot</code> .

**Value**

A ggplot graph.

**Examples**

```
data(datasim02)  
plot_paths(datasim02)
```

# Index

## \* datasets

datasim01, [2](#)

datasim02, [3](#)

datasim03, [4](#)

[datasim01, 2](#)

[datasim02, 3](#)

[datasim03, 4](#)

[fit\\_ou, 5](#)

[fit\\_wiener, 7](#)

[plot\\_paths, 9](#)