

mkin -  
Routines for fitting kinetic models with one or more  
state variables to chemical degradation data

**Johannes Ranke**  
Product Safety  
Harlan Laboratories Ltd.  
Zelgliweg 1, CH-4452 Itingen, Switzerland

June 9, 2010

**Abstract**

In the regulatory evaluation of chemical substances like plant protection products (pesticides), biocides and other chemicals, degradation data play an important role. For the evaluation of pesticide degradation experiments, detailed guidance has been developed, based on nonlinear optimisation. The R add-on package **mkin** implements fitting some of the models recommended in this guidance from within R and calculates some statistical measures for data series within one or more compartments, for parent and metabolites.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Example</b>	<b>1</b>
2.1	Data format . . . . .	1
2.2	Model definition . . . . .	2
2.3	Fitting the model . . . . .	3

**Key words:** Kinetics, FOCUS, nonlinear optimisation

## 1 Introduction

Many approaches are possible regarding the evaluation of chemical degradation data. The **kinfit** package ([Ranke, 2010a](#)) in R ([R Development Core Team, 2010](#)) implements the approach recommended in the kinetics report provided by the FORum for Co-ordination of pesticide fate models and their USe ([FOCUS Work Group on Degradation Kinetics, 2006](#)) for simple data series for one parent compound in one compartment.

The **mkim** package ([Ranke, 2010b](#)) extends this approach to data series with metabolites and more than one compartment and includes the possibility for back reactions.

## 2 Example

In the following, requirements for data formatting are explained. Then the procedure for fitting the four kinetic models recommended by the FOCUS group to an example dataset for parent only given in the FOCUS kinetics report is illustrated. The explanations are kept rather verbose in order to lower the barrier for R newcomers.

### 2.1 Data format

The following listing shows example dataset C from the FOCUS kinetics report as distributed with the **kinfit** package

```
R> library("mkim")
```

*This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.*

*See the LICENSE file distributed with the minpack.lm source code or <http://www.netlib.org/minpack/disclaimer> for the full license.*

```
R> FOCUS_2006_C
```

	name	time	value
1	parent	0	85.1
2	parent	1	57.9
3	parent	3	29.9
4	parent	7	14.6
5	parent	14	9.7
6	parent	28	6.6
7	parent	63	4.0
8	parent	91	3.9
9	parent	119	0.6

Note that the data needs to be in the format of a data frame containing a variable **name** specifying the observed variable, indicating the compound name and, if applicable, the compartment, a variable **time** containing sampling times, and a numeric variable **value** specifying the observed value of the variable. If a further variable **error** is present, this will be used to give different weights to the data points (the higher the error, the lower the weight, see the help page of the **modCost** function of the **FME** package ([Soetaert and Petzoldt, 2010](#))). Replicate measurements are not recorded in extra columns but simply appended, leading to multiple occurrences of the sampling times **time**.

Small to medium size dataset can be conveniently entered directly as R code as shown in the following listing

```
R> example_data <- data.frame(  
+   time = c(0, 1, 3, 7, 14, 28, 63, 91, 119),  
+   parent = c(85.1, 57.9, 29.9, 14.6, 9.7, 6.6, 4, 3.9, 0.6)  
+ )
```

## 2.2 Model definition

The next task is to define the model to be fitted to the data. In order to facilitate this task, a convenience function **mkmod** is available.

```
R> SFO <- mkmod(parent = list(type = "SFO"))  
R> SFORB <- mkmod(parent = list(type = "SFORB"))  
R> SFO_SFO <- mkmod(  
+   parent = list(type = "SFO", to = "m1", sink = TRUE),  
+   m1 = list(type = "SFO"))  
R> SFORB_SFO <- mkmod(  
+   parent = list(type = "SFORB", to = "m1", sink = TRUE),  
+   m1 = list(type = "SFO"))
```

## 2.3 Fitting the model

Then the model parameters should be fitted to the data. The function `mkinfitt` internally creates a cost function using `modCost` from the **FME** package and the produces a fit using `modFit` from the same package.

```
R> # Do not show significance stars as they interfere with vignette generation
R> options(show.signif.stars = FALSE)
R> SF0.fit <- mkinfitt(SF0, FOCUS_2006_C)
```

```
Model cost at call 1 : 4718.97
Model cost at call 4 : 4718.97
Model cost at call 5 : 572.411
Model cost at call 7 : 572.4109
Model cost at call 8 : 236.2073
Model cost at call 9 : 236.2073
Model cost at call 11 : 198.9360
Model cost at call 12 : 198.9360
Model cost at call 14 : 196.6777
Model cost at call 15 : 196.6777
Model cost at call 16 : 196.6777
Model cost at call 17 : 196.5422
Model cost at call 18 : 196.5422
Model cost at call 19 : 196.5422
Model cost at call 20 : 196.5341
Model cost at call 21 : 196.5341
Model cost at call 22 : 196.5341
Model cost at call 23 : 196.5336
Model cost at call 25 : 196.5336
Model cost at call 26 : 196.5336
Model cost at call 28 : 196.5336
Model cost at call 29 : 196.5336
Model cost at call 33 : 196.5336
```

```
R> summary(SF0.fit)
```

*Equations:*

```
[1] d_parent = - k_parent_sink * parent
```

*Starting values for optimised parameters:*

	initial	type	lower	upper
parent_0	100.0	state	0	Inf
k_parent_sink	0.1	deparm	0	Inf

*Fixed parameter values:*

None

*Optimised parameters:*

```
Estimate Std. Error t value Pr(>|t|)
```

parent_0	82.4919	4.7402	17.402	5.09e-07
k_parent_sink	0.3061	0.0459	6.668	0.000286

Residual standard error: 5.299 on 7 degrees of freedom

Chi2 error levels in percent:

	err.min	n.optim	df
All data	15.84	2	7
parent	15.84	2	7

Estimated disappearance times

	DT50	DT90
parent	2.265	7.523

Data:

time	variable	observed	predicted	residual
0	parent	85.1	82.491918964177	2.608
1	parent	57.9	60.742411633568	-2.842
3	parent	29.9	32.934530029534	-3.035
7	parent	14.6	9.682173180746	4.918
14	parent	9.7	1.136403227206	8.564
28	parent	6.6	0.015654900288	6.584
63	parent	4.0	0.000000361297	4.000
91	parent	3.9	-0.000000014466	3.900
119	parent	0.6	-0.000000001821	0.600

R> SFORB.fit <- mkinfit(SFORB, FOCUS\_2006\_C)

Model cost at call 1 : 7044.136  
 Model cost at call 4 : 7044.136  
 Model cost at call 7 : 3460.066  
 Model cost at call 9 : 3460.066  
 Model cost at call 11 : 3460.066  
 Model cost at call 13 : 309.8375  
 Model cost at call 15 : 309.8375  
 Model cost at call 17 : 309.8375  
 Model cost at call 18 : 27.19125  
 Model cost at call 20 : 27.19124  
 Model cost at call 23 : 4.43924  
 Model cost at call 25 : 4.43924  
 Model cost at call 28 : 4.362922  
 Model cost at call 31 : 4.362922  
 Model cost at call 33 : 4.362712  
 Model cost at call 38 : 4.36271  
 Model cost at call 43 : 4.36271  
 Model cost at call 47 : 4.36271  
 Model cost at call 49 : 4.36271

R> summary(SFORB.fit)

Equations:

```
[1] d_parent_free = - k_parent_free_sink * parent_free - k_parent_free_bound * parent_free
[2] d_parent_bound = + k_parent_free_bound * parent_free - k_parent_bound_free * parent_free
```

Starting values for optimised parameters:

	initial	type	lower	upper
parent_free_0	100.0	state	0	Inf
k_parent_free_sink	0.1	deparm	0	Inf
k_parent_free_bound	0.1	deparm	0	Inf
k_parent_bound_free	0.1	deparm	0	Inf

Fixed parameter values:

	value	type
parent_bound	0	state

Optimised parameters:

	Estimate	Std. Error	t value	Pr(> t )
parent_free_0	85.002720	0.890669	95.437	2.39e-09
k_parent_free_sink	0.395043	0.014308	27.610	1.17e-06
k_parent_free_bound	0.061599	0.007289	8.451	0.000381
k_parent_bound_free	0.020764	0.003752	5.533	0.002644

Residual standard error: 0.9341 on 5 degrees of freedom

Chi2 error levels in percent:

	err.min	n.optim	df
All data	2.662	4	5
parent	2.662	4	5

Estimated disappearance times

	DT50	DT90
parent	1.887	21.25

Data:

time	variable	observed	predicted	residual
0	parent	85.1	85.003	0.09728
1	parent	57.9	58.039	-0.13912
3	parent	29.9	30.054	-0.15358
7	parent	14.6	13.866	0.73388
14	parent	9.7	9.787	-0.08655
28	parent	6.6	7.532	-0.93201
63	parent	4.0	4.033	-0.03265
91	parent	3.9	2.446	1.45351
119	parent	0.6	1.484	-0.88421

```
R> SFO_SFO.fit <- mkinfit(SFO_SFO, FOCUS_2006_D)
```

Model cost at call 1 : 18994.29

Model cost at call 3 : 18994.29

```

Model cost at call 7 : 10641.39
Model cost at call 8 : 10641.39
Model cost at call 12 : 7145.459
Model cost at call 14 : 7145.458
Model cost at call 17 : 411.9739
Model cost at call 18 : 411.9738
Model cost at call 22 : 371.2194
Model cost at call 23 : 371.2194
Model cost at call 27 : 371.2127
Model cost at call 28 : 371.2127
Model cost at call 29 : 371.2127
Model cost at call 31 : 371.2127
Model cost at call 32 : 371.2127
Model cost at call 34 : 371.2127
Model cost at call 37 : 371.2127
Model cost at call 39 : 371.2127

```

```
R> summary(SFO_SFO.fit)
```

Equations:

```

[1] d_parent = - k_parent_sink * parent - k_parent_m1 * parent
[2] d_m1 = - k_m1_sink * m1 + k_parent_m1 * parent

```

Starting values for optimised parameters:

	initial	type	lower	upper
parent_0	100.0	state	0	Inf
k_parent_sink	0.1	deparm	0	Inf
k_m1_sink	0.1	deparm	0	Inf
k_parent_m1	0.1	deparm	0	Inf

Fixed parameter values:

	value	type
m1	0	state

Optimised parameters:

	Estimate	Std. Error	t value	Pr(> t )
parent_0	9.960e+01	1.614e+00	61.720	< 2e-16
k_parent_sink	4.792e-02	3.750e-03	12.778	6.10e-15
k_m1_sink	5.261e-03	7.159e-04	7.349	1.15e-08
k_parent_m1	5.078e-02	2.094e-03	24.248	< 2e-16

Residual standard error: 3.211 on 36 degrees of freedom

Chi2 error levels in percent:

	err.min	n.optim	df
All data	6.565	4	16
parent	6.827	3	6
m1	4.748	1	10

Estimated disappearance times

	DT50	DT90
parent	7.023	23.33
m1	131.760	437.70

Data:

time	variable	observed	predicted	residual
0	parent	99.46	99.5984955	-0.1384955
0	parent	102.04	99.5984955	2.4415045
1	parent	93.50	90.2378767	3.2621233
1	parent	92.50	90.2378767	2.2621233
3	parent	63.23	74.0731772	-10.8431772
3	parent	68.99	74.0731772	-5.0831772
7	parent	52.32	49.9120561	2.4079439
7	parent	55.13	49.9120561	5.2179439
14	parent	27.27	25.0125879	2.2574121
14	parent	26.64	25.0125879	1.6274121
21	parent	11.50	12.5346040	-1.0346040
21	parent	11.64	12.5346040	-0.8946040
35	parent	2.85	3.1478595	-0.2978595
35	parent	2.91	3.1478595	-0.2378595
50	parent	0.69	0.7162355	-0.0262355
50	parent	0.63	0.7162355	-0.0862355
75	parent	0.05	0.0607374	-0.0107374
75	parent	0.06	0.0607374	-0.0007374
100	parent	NA	0.0051506	NA
100	parent	NA	0.0051506	NA
120	parent	NA	0.0007155	NA
120	parent	NA	0.0007155	NA
0	m1	0.00	0.0000000	0.0000000
0	m1	0.00	0.0000000	0.0000000
1	m1	4.84	4.8029613	0.0370387
1	m1	5.64	4.8029613	0.8370387
3	m1	12.91	13.0240164	-0.1140164
3	m1	12.96	13.0240164	-0.0640164
7	m1	22.97	25.0447743	-2.0747743
7	m1	24.47	25.0447743	-0.5747743
14	m1	41.69	36.6900224	4.9999776
14	m1	33.21	36.6900224	-3.4800224
21	m1	44.37	41.6531097	2.7168903
21	m1	46.44	41.6531097	4.7868903
35	m1	41.22	43.3131256	-2.0931256
35	m1	37.95	43.3131256	-5.3631256
50	m1	41.19	41.2183070	-0.0283070
50	m1	40.01	41.2183070	-1.2083070
75	m1	40.09	36.4470161	3.6429839
75	m1	33.85	36.4470161	-2.5970161
100	m1	31.04	31.9815991	-0.9415991
100	m1	33.13	31.9815991	1.1484009



120	m1	25.15	28.7898086	-3.6398086
120	m1	33.31	28.7898086	4.5201914

## References

- FOCUS Work Group on Degradation Kinetics. *Guidance Document on Estimating Persistence and Degradation Kinetics from Environmental Fate Studies on Pesticides in EU Registration. Report of the FOCUS Work Group on Degradation Kinetics*, 2006. URL <http://focus.jrc.ec.europa.eu/dk>. EC Document Reference Sanco/10058/2005 version 2.0.
- R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2010. URL <http://www.R-project.org>. ISBN 3-900051-07-0.
- Johannes Ranke. *kinfit: Routines for fitting simple kinetic models to chemical degradation data*, 2010a. URL <http://CRAN.R-project.org>.
- Johannes Ranke. *mkln: Routines for fitting kinetic models with one or more state variables to chemical degradation data*, 2010b. URL <http://CRAN.R-project.org>.
- Karline Soetaert and Thomas Petzoldt. Inverse modelling, sensitivity and monte carlo analysis in R using package FME. *Journal of Statistical Software*, 33(3):1–28, 2010. URL <http://www.jstatsoft.org/v33/i03/>.