

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

Johannes Ranke

Eurofins Regulatory AG
Weidenweg 15, CH-4310 Rheinfelden, Switzerland

and

University of Bremen

April 14, 2013

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

1	Introduction	1
2	Example	1
2.1	Data format	1
2.2	Model definition	2
2.3	Fitting the model	3
3	Acknowledgements	11

Key words: Kinetics, FOCUS, nonlinear optimisation

1 Introduction

Many approaches are possible regarding the evaluation of chemical degradation data. The **kinfit** package ([Ranke, 2012a](#)) in R ([R Development Core Team, 2012](#)) 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, 2011](#)) for simple data series for one parent compound in one compartment.

The **mkkin** package ([Ranke, 2012b](#)) 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 **mkkin** package

```
R> library("mkkin")
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(
+   name = rep("parent", 9),
+   time = c(0, 1, 3, 7, 14, 28, 63, 91, 119),
+   value = 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> SF0 <- mkmod(parent = list(type = "SF0"))
R> SFORB <- mkmod(parent = list(type = "SFORB"))
R> SF0_SF0 <- mkmod(
+   parent = list(type = "SF0", to = "m1", sink = TRUE),
+   m1 = list(type = "SF0"))
R> SFORB_SF0 <- mkmod(
+   parent = list(type = "SFORB", to = "m1", sink = TRUE),
+   m1 = list(type = "SF0"))

```

The model definitions given above define sets of linear first-order ordinary differential equations. In these cases, a coefficient matrix is also returned.

Other models that include time on the right-hand side of the differential equation are the first-order multi-compartment (FOMC) model and the Hockey-Stick (HS) model. At present, these models can only be used only for the parent compound.

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 then produces a fit using `modFit` from the same package. In cases of linear first-order differential equations, the solution used for calculating the cost function is based on the fundamental system of the coefficient matrix, as proposed by [Bates and Watts \(1988\)](#).

```
R> SFO.fit <- mkinfitt(SFO, FOCUS_2006_C)
```

```
Model cost at call 1 : 4718.953
Model cost at call 4 : 4718.953
Model cost at call 5 : 530.2647
Model cost at call 7 : 530.2647
Model cost at call 8 : 230.7217
Model cost at call 9 : 230.7217
Model cost at call 11 : 198.449
Model cost at call 12 : 198.449
Model cost at call 13 : 198.449
Model cost at call 14 : 196.6458
Model cost at call 15 : 196.6458
Model cost at call 16 : 196.6458
Model cost at call 17 : 196.5401
Model cost at call 18 : 196.5401
Model cost at call 19 : 196.5401
Model cost at call 20 : 196.5338
Model cost at call 21 : 196.5338
Model cost at call 22 : 196.5338
Model cost at call 23 : 196.5334
Model cost at call 24 : 196.5334
Model cost at call 25 : 196.5334
Model cost at call 26 : 196.5334
Model cost at call 28 : 196.5334
Model cost at call 29 : 196.5334
```

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

```
mkim version:      0.9.19
R version:         2.15.3
Date of fit:       Sun Apr 14 14:32:10 2013
Date of summary:   Sun Apr 14 14:32:10 2013
```

Equations:

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

Method used for solution of differential equation system:
analytical

Starting values for optimised parameters:

	initial	type	transformed
parent_0	100.0	state	100.000000
k_parent_sink	0.1	deparm	-2.302585

Fixed parameter values:
None

Optimised, transformed parameters:

	Estimate	Std. Error	Lower	Upper
parent_0	82.490	4.74	71.280	93.7000
k_parent_sink	-1.184	0.15	-1.539	-0.8294

Backtransformed parameters:

	Estimate	Lower	Upper
parent_0	82.4900	71.2800	93.7000
k_parent_sink	0.3061	0.2147	0.4363

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

Estimated formation fractions:

	ff
parent_sink	1

Parameter correlation:

	parent_0	k_parent_sink
parent_0	1.0000	0.5212
k_parent_sink	0.5212	1.0000

Data:

time	variable	observed	predicted	residual
0	parent	85.1	8.249e+01	2.608
1	parent	57.9	6.074e+01	-2.842
3	parent	29.9	3.293e+01	-3.034
7	parent	14.6	9.682e+00	4.918
14	parent	9.7	1.136e+00	8.564

28	parent	6.6	1.565e-02	6.584
63	parent	4.0	3.487e-07	4.000
91	parent	3.9	6.617e-11	3.900
119	parent	0.6	1.256e-14	0.600

```
R> SFORB.fit <- mkinfit(SFORB, FOCUS_2006_C)
```

```
Model cost at call 1 : 10077.99
Model cost at call 4 : 10077.99
Model cost at call 7 : 626.7873
Model cost at call 9 : 626.7873
Model cost at call 11 : 626.7873
Model cost at call 12 : 73.06755
Model cost at call 14 : 73.06754
Model cost at call 17 : 4.939855
Model cost at call 19 : 4.939855
Model cost at call 21 : 4.939854
Model cost at call 22 : 4.363721
Model cost at call 24 : 4.363721
Model cost at call 26 : 4.363721
Model cost at call 27 : 4.362718
Model cost at call 28 : 4.362718
Model cost at call 29 : 4.362718
Model cost at call 31 : 4.362718
Model cost at call 32 : 4.362714
Model cost at call 34 : 4.362714
Model cost at call 36 : 4.362714
Model cost at call 37 : 4.362714
```

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

```
mkim version:      0.9.19
R version:         2.15.3
Date of fit:       Sun Apr 14 14:32:11 2013
Date of summary:   Sun Apr 14 14:32:11 2013
```

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_bound
```

Method used for solution of differential equation system:
analytical

Starting values for optimised parameters:

	initial	type	transformed
parent_free_0	1e+02	state	100.000000
k_parent_free_sink	1e-01	deparm	-2.302585
k_parent_free_bound	1e-01	deparm	-2.302585
k_parent_bound_free	2e-02	deparm	-3.912023

Fixed parameter values:

	value	type
parent_bound	0	state

Optimised, transformed parameters:

	Estimate	Std. Error	Lower	Upper
parent_free_0	85.0000	0.89070	82.710	87.2900
k_parent_free_sink	-0.9288	0.03622	-1.022	-0.8357
k_parent_free_bound	-2.7870	0.11830	-3.091	-2.4830
k_parent_bound_free	-3.8750	0.18070	-4.339	-3.4100

Backtransformed parameters:

	Estimate	Lower	Upper
parent_free_0	85.00000	82.71000	87.29000
k_parent_free_sink	0.39500	0.35990	0.43360
k_parent_free_bound	0.06160	0.04544	0.08350
k_parent_bound_free	0.02076	0.01305	0.03304

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

Estimated formation fractions:

	ff
parent_free_sink	1

Estimated Eigenvalues of SFORB model(s):

	parent_b1	parent_b2
	0.45956	0.01785

Parameter correlation:

	parent_free_0	k_parent_free_sink	k_parent_free_bound	k_parent_bound_free
parent_free_0	1.00000	0.5217	0.1813	0.07644
k_parent_free_sink	0.52169	1.0000	0.6693	0.3062
k_parent_free_bound	0.18129	0.6693	1.0000	0.6756
k_parent_bound_free	0.07644	0.3062	0.6756	1.0000

Data:

time	variable	observed	predicted	residual
0	parent	85.1	85.003	0.09727
1	parent	57.9	58.039	-0.13912
3	parent	29.9	30.054	-0.15351

7	parent	14.6	13.866	0.73389
14	parent	9.7	9.787	-0.08657
28	parent	6.6	7.532	-0.93205
63	parent	4.0	4.033	-0.03269
91	parent	3.9	2.447	1.45347
119	parent	0.6	1.484	-0.88424

```
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 : 10642.61
Model cost at call 8 : 10642.61
Model cost at call 10 : 10642.61
Model cost at call 12 : 7148.119
Model cost at call 14 : 7148.118
Model cost at call 17 : 412.0354
Model cost at call 18 : 412.0353
Model cost at call 22 : 371.2203
Model cost at call 23 : 371.2203
Model cost at call 25 : 371.2203
Model cost at call 27 : 371.2134
Model cost at call 28 : 371.2134
Model cost at call 30 : 371.2134
Model cost at call 32 : 371.2134
```

```
R> summary(SFO_SFO.fit, data=FALSE)
```

```
mkin version: 0.9.19
R version: 2.15.3
Date of fit: Sun Apr 14 14:32:11 2013
Date of summary: Sun Apr 14 14:32:11 2013
```

Equations:

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

Method used for solution of differential equation system:
eigen

Starting values for optimised parameters:

	initial	type	transformed
parent_0	100.0	state	100.000000
k_parent_sink	0.1	deparm	-2.302585
k_parent_m1	0.1	deparm	-2.302585
k_m1_sink	0.1	deparm	-2.302585

Fixed parameter values:

	value	type
m1	0	state

Optimised, transformed parameters:

	Estimate	Std. Error	Lower	Upper
parent_0	99.600	1.61400	96.330	102.900
k_parent_sink	-3.038	0.07826	-3.197	-2.879
k_parent_m1	-2.980	0.04124	-3.064	-2.897
k_m1_sink	-5.248	0.13610	-5.523	-4.972

Backtransformed parameters:

	Estimate	Lower	Upper
parent_0	99.600000	96.330000	1.029e+02
k_parent_sink	0.047920	0.040890	5.616e-02
k_parent_m1	0.050780	0.046700	5.521e-02
k_m1_sink	0.005261	0.003992	6.933e-03

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	131.761	437.70

Estimated formation fractions:

	ff
parent_sink	0.4855
parent_m1	0.5145
m1_sink	1.0000

Parameter correlation:

	parent_0	k_parent_sink	k_parent_m1	k_m1_sink
parent_0	1.00000	0.60752	-0.06625	-0.1701
k_parent_sink	0.60752	1.00000	-0.08741	-0.6253
k_parent_m1	-0.06625	-0.08741	1.00000	0.4716
k_m1_sink	-0.17006	-0.62527	0.47164	1.0000

R> SFORB_SFO.fit <- mkinfit(SFORB_SFO, FOCUS_2006_D)

```
Model cost at call 1 : 19252.94
Model cost at call 3 : 19252.93
Model cost at call 6 : 19252.93
Model cost at call 9 : 18210.81
Model cost at call 10 : 18210.81
Model cost at call 14 : 18210.81
```

```

Model cost at call 16 : 1545.842
Model cost at call 17 : 1545.841
Model cost at call 24 : 949.7984
Model cost at call 25 : 949.7984
Model cost at call 30 : 949.7984
Model cost at call 31 : 564.269
Model cost at call 32 : 564.269
Model cost at call 36 : 564.269
Model cost at call 38 : 444.2523
Model cost at call 39 : 444.2523
Model cost at call 45 : 369.1132
Model cost at call 47 : 369.1132
Model cost at call 50 : 369.1132
Model cost at call 53 : 354.8494
Model cost at call 55 : 354.8494
Model cost at call 58 : 354.8493
Model cost at call 60 : 353.4189
Model cost at call 62 : 353.4189
Model cost at call 65 : 353.4189
Model cost at call 68 : 352.4084
Model cost at call 71 : 352.4084
Model cost at call 76 : 352.2537
Model cost at call 77 : 352.2537
Model cost at call 83 : 352.23
Model cost at call 86 : 352.23
Model cost at call 90 : 352.2092
Model cost at call 91 : 352.2092
Model cost at call 92 : 352.2092
Model cost at call 98 : 352.2058
Model cost at call 99 : 352.2058
Model cost at call 101 : 352.2058
Model cost at call 105 : 352.2058
Model cost at call 107 : 352.2058
Model cost at call 112 : 352.2049
Model cost at call 115 : 352.2049
Model cost at call 120 : 352.2049
Model cost at call 121 : 352.2049
Model cost at call 122 : 352.2049
Model cost at call 127 : 352.2049
Model cost at call 130 : 352.2049
Model cost at call 134 : 352.2048
Model cost at call 136 : 352.2048
Model cost at call 142 : 352.2048
Model cost at call 144 : 352.2048

```

```
R> summary(SFORB_SFO.fit, data=FALSE)
```

```

mkin version:    0.9.19
R version:       2.15.3

```

Date of fit: Sun Apr 14 14:32:14 2013

Date of summary: Sun Apr 14 14:32:14 2013

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_bound
[3] d_m1 = + k_parent_free_m1 * parent_free - k_m1_sink * m1
```

Method used for solution of differential equation system:

eigen

Starting values for optimised parameters:

	initial	type	transformed
parent_free_0	1e+02	state	100.000000
k_parent_free_sink	1e-01	deparm	-2.302585
k_parent_free_bound	1e-01	deparm	-2.302585
k_parent_bound_free	2e-02	deparm	-3.912023
k_parent_free_m1	1e-01	deparm	-2.302585
k_m1_sink	1e-01	deparm	-2.302585

Fixed parameter values:

	value	type
parent_bound	0	state
m1	0	state

Optimised, transformed parameters:

	Estimate	Std. Error	Lower	Upper
parent_free_0	101.1000	2.0200	96.970	105.200
k_parent_free_sink	-2.7480	0.4201	-3.601	-1.894
k_parent_free_bound	-1.7840	3.0620	-8.006	4.439
k_parent_bound_free	-0.6465	1.6320	-3.962	2.669
k_parent_free_m1	-2.7240	0.3875	-3.511	-1.936
k_m1_sink	-5.2570	0.1383	-5.538	-4.976

Backtransformed parameters:

	Estimate	Lower	Upper
parent_free_0	1.011e+02	9.697e+01	1.052e+02
k_parent_free_sink	6.407e-02	2.729e-02	1.505e-01
k_parent_free_bound	1.680e-01	3.333e-04	8.468e+01
k_parent_bound_free	5.239e-01	1.902e-02	1.443e+01
k_parent_free_m1	6.562e-02	2.986e-02	1.442e-01
k_m1_sink	5.213e-03	3.935e-03	6.905e-03

Residual standard error: 3.219 on 34 degrees of freedom

Chi2 error levels in percent:

	err.min	n.optim	df
All data	6.645	6	14
parent	7.207	5	4

```
m1          5.123          1 10
```

Estimated disappearance times:

```
          DT50   DT90
parent    6.805  24.05
m1       132.971 441.72
```

Estimated formation fractions:

```
          ff
parent_free_sink 0.494
parent_free_m1   0.506
m1_sink          1.000
```

Estimated Eigenvalues of SFORB model(s):

```
parent_b1 parent_b2
  0.7283   0.0933
```

Parameter correlation:

	parent_free_0	k_parent_free_sink	k_parent_free_bound	k_parent_bound_free	k_parent_free_m1	k_m1_sink
parent_free_0	1.0000	0.5432	0.34386	0.18007	-0.03946	-0.18007
k_parent_free_sink	0.5432	1.0000	0.94318	0.8179	0.9752	-0.20304
k_parent_free_bound	0.3439	0.9432	1.00000	0.95433	0.96053	-0.09286
k_parent_bound_free	0.1950	0.8179	0.95433	1.00000	0.96053	-0.08808
k_parent_free_m1	0.4401	0.9752	0.96053	0.96053	1.00000	-0.03946
k_m1_sink	-0.1801	-0.2030	-0.09286	-0.08808	-0.03946	1.00000

3 Acknowledgements

This package would not have been written without me being introduced to regulatory fate modelling of pesticides by Adrian Gurney during my time at Harlan Laboratories Ltd (formerly RCC Ltd). Parts of the package were written during my employment at Harlan.

References

- D. Bates and D. Watts. *Nonlinear regression and its applications*. Wiley-Interscience, 1988.
- 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.
- FOCUS Work Group on Degradation Kinetics. *Generic guidance for estimating persistence and degradation kinetics from environmental fate studies on pesticides in EU registration*, 1.0 edition, November 2011. URL <http://focus.jrc.ec.europa.eu/dk>.
- R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. URL <http://www.R-project.org>. ISBN 3-900051-07-0.
- Johannes Ranke. *kinfit: Routines for fitting simple kinetic models to chemical degradation data*, 2012a. URL <http://CRAN.R-project.org>.
- Johannes Ranke. *mkim: Routines for fitting kinetic models with one or more state variables to chemical degradation data*, 2012b. 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/>.